PERGAMON

Pattern Recognition 35 (2002) 945–965

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video

Sameer Antani[a], Rangachar Kasturi[a,*], Ramesh Jain[b]

[a]Department of Computer Science and Engineering, The Pennsylvania State University, University Park, PA 16802, USA
[b]Praja, Inc., 10455-B Pacific Center Court, San Diego, CA 92121, USA

Received 5 June 2000; received in revised form 16 February 2001; accepted 19 March 2001

Abstract

The need for content-based access to image and video information from media archives has captured the attention of researchers in recent years. Research efforts have led to the development of methods that provide access to image and video data. These methods have their roots in pattern recognition. The methods are used to determine the similarity in the visual information content extracted from low level features. These features are then clustered for generation of database indices. This paper presents a comprehensive survey on the use of these pattern recognition methods which enable image and video retrieval by content. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Content-based retrieval; Pattern recognition; Image databases; Video databases

1. Introduction

There has been a growing demand for image and video data in applications due to the significant improvement in the processing technology, network subsystems and availability of large storage systems. This demand for visual data has spurred a significant interest in the research community to develop methods to archive, query and retrieve this data based on their content. Such systems, called content based image (and video) retrieval (CBIR) systems or visual information retrieval systems (VIRS) [1], employ many pattern recognition methods developed over the years. Here the term *pattern recognition methods* refer to their applicability in feature extraction, feature clustering, generation of database indices, and determining similarity in content of the query and database elements. Other surveys on techniques for content based retrieval of image and video have been presented in Refs. [2–6]. A discussion on issues relevant to visual information retrieval and capturing the semantics present in images and video are presented in Ref. [7]. Applications where CBIR systems can be effectively utilized ranging from scientific, medical imaging to geographic information systems, and video-on-demand systems among numerous others. Commercial CBIR systems have been developed by Virage Inc.,[1] MediaSite Inc.[2] and IBM. In this paper we present significant contributions published in the literature that use pattern recognition methods for providing content based access to visual information. In view of the large number of methods that can perform specific tasks in the content based access to video problem, several researchers have presented critical evaluations of the state of art in content based retrieval methods for image and video. An evaluation of various image features and their performance in image retrieval has been done by Di Lecce and Guerriero [8]. Mandal et al. [9] present a critical evaluation

* Corresponding author. Tel.: +1-814-863-4254; fax: +1-814-865-3176.
E-mail address: kasturi@cse.psu.edu (R. Kasturi).

[1] Virage: http://www.virage.com.

[2] MediaSite: http://www.mediasite.com.

PII: S0031-3203(01)00086-3

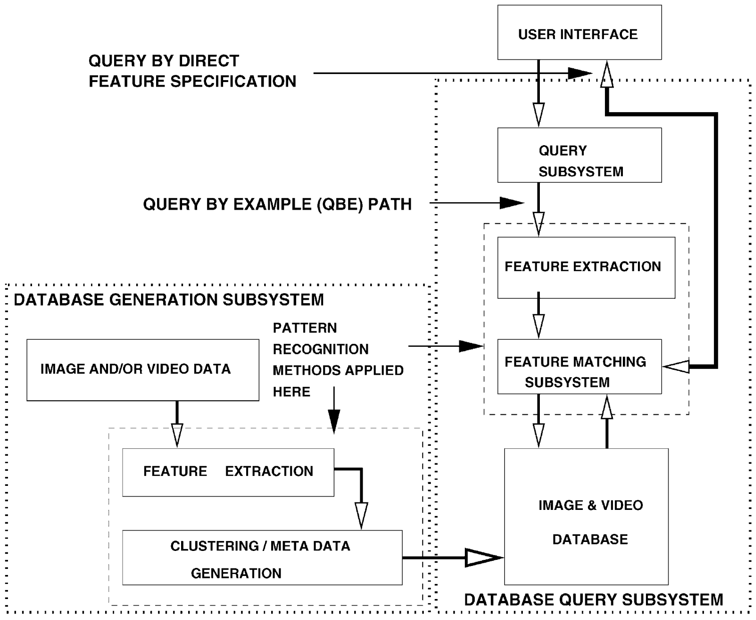
USER INTERFACE
QUERY BY DIRECT FEATURE SPECIFICATION
QUERY SUBSYSTEM
QUERY BY EXAMPLE (QBE) PATH
FEATURE EXTRACTION
DATABASE GENERATION SUBSYSTEM
PATTERN RECOGNITION METHODS APPLIED HERE
IMAGE AND/OR VIDEO DATA
FEATURE MATCHING SUBSYSTEM
FEATURE EXTRACTION
IMAGE & VIDEO DATABASE
CLUSTERING / META DATA GENERATION
DATABASE QUERY SUBSYSTEM

Fig. 1. Typical present day image and video database system.

of image and video indexing techniques that operate in the compressed domain. Gargi et al. [10] characterize the performance of video shot change detection methods that use color histogram based techniques as well as motion based methods in the MPEG[3] compressed and uncompressed domain.

Research in retrieval of non-geometric pictorial information began in the early 1980s. A relational database system that used pattern recognition and image manipulation was developed for retrieving LANDSAT images [11]. This system also introduced the preliminary concepts of query-by-pictorial-example (QPE), which has since matured to be better known as query-by-example (QBE). Since then, the research has come a long way in developing various methods that retrieve visual information by its content, which depend on an integrated feature extraction/object recognition subsystem. Features such as color, shape and texture are used by the pattern recognition subsystem to select a good match in response to the user query. Present day systems can at best define the visual content with partial semantics. In essence these are forms of quantifiable measurements of features extracted from image and video data. Humans, on the other hand, attach semantic meaning to visual content. While the methods for extracting/matching images and video remain primarily statistical in nature [12], CBIR systems attach the annotations to the extracted features for pseudo-semantic retrieval.

Importance of pattern recognition methods in image and video databases: Image and video databases are different in many aspects from the traditional databases containing text information. Image and video data can be perceived differently making it very difficult to maintain complete content information of images or video. The queries would tend to be complex and better expressed in natural language form. The focus for the computer vision and pattern recognition community is also to develop methods which can convert the query specification into meaningful set of query features. A typical present day image and video database system block diagram is shown in Fig. 1. The figure shows that the database system has two main points of interface, the database generation subsystem and the database query subsystem. Pattern recognition methods are applied in both these subsystems. The inputs to the database generation subsystem are image and video data. The necessary features are extracted from this data and then appropriately clustered to generate indices to be used by the database query subsystem. The database query subsystem matches a user query using appropriate features to the indexed elements of the database.

Although many methods have been developed for querying images that make use of exotic features extracted from the visual information, successful methods need to be fine tuned for each application. In traditional object recognition the expected geometry, shape, texture, and color of the object are known beforehand. But,

[3] MPEG: Motion Picture Experts Group—http://www.mpeg.org.

with a *real* image and video database system, the data is sourced from diverse environments. This leads us to face the reality that a fixed and a priori decided set of features will not work well. Also, typical users of such systems may not be able to comprehend the complex methods used to implement the visual information retrieval. Hence a more intuitive interface needs to be given to the users, so that the query can then be mapped to appropriate parameters for the retrieval subsystem. All of the above lead to the need for a middle layer between the user query and the retrieval subsystem that will translate the query into the use of an appropriate feature to query the image. This feature should be easily computable in real time and also be able to capture the essence of the human query.

The remainder of the paper is organized as follows. Section 2 describes some of the successful image/video database systems briefly. Section 3 discusses some methods and issues related to similarity matching. We have separated the discussion on the multimedia database systems into image database and video database systems. Section 4 describes the features used and pattern recognition methods employed for image databases. Section 5, on the other hand, describes the classification methods used for features extracted from video data. We present conclusions in Section 6.

2. Image and video database systems: examples

Several successful multimedia systems which have been developed in recent years are described in the literature. In this section we shall highlight some of the mature systems, the features they use and the primary classification method(s) utilized by them. A book by Gong [13] discusses some of the image database systems. Other systems are mentioned due to their contributions to the development of methods for content based image and video retrieval in Sections 4 and 5.

Content based retrieval engine (*CORE*): In Ref. [14], the authors describe and discuss the requirements of multimedia information systems. A multimedia information system is composed of multimedia objects, defined as a six tuple, for which it must offer creation, management, retrieval, processing, presentation and usability functions. The authors admit that a multimedia object is a complex component that is difficult to describe. It is audio-visual in nature and hence can have multiple interpretations. Its description is context sensitive and hierarchical relationships can be formed between these objects. Content based retrieval is classified into four distinct categories; visual retrieval, similarity retrieval, fuzzy retrieval and text retrieval. Each of these are addressed in the CORE architecture. It supports color based and word/phonetics based retrieval. It also supports fuzzy matching along with a feedback loop for additional reliability. Two real world systems have been developed on the CORE concept; Computer Aided Facial Image Inference and Retrieval (CAFIIR) and System for Trademark Archival and Retrieval (STAR) [15].

WebSeek: WebSeek[4] is a prototype image and video search engine which collects images and videos from the Web and catalogs them. It also provides tools for searching and browsing [16] using various content based retrieval techniques that incorporate the use of color, texture and other properties. The queries can be a combination of text based searches along with image searches. Relevance feedback mechanisms are used to enhance performance.

VideoQ: VideoQ is a web enabled content based video search system [17]. VideoQ[5] expands the traditional search methods (e.g., keywords and subject navigation) with a novel search technique that allows users to search compressed video based on a rich set of visual features and spatio-temporal relationships. Visual features include color, texture, shape, and motion. Spatio-temporal video object query extends the principle of query by sketch in image databases to a temporal sketch that defines the object motion path.

Blobworld: Bolbworld[6] is a system for content based image retrieval by finding coherent regions that may correspond to objects [18]. It is claimed that current image retrieval systems do not perform well in two key areas—finding images based on regions of interest (objects) and presenting results that are difficult to comprehend since the system is presented to the user as a black box. The Blobworld system performs search on parts of an image which is treated as a group of blobs roughly representing homogeneous color or texture regions. Each pixel in the image is described using a vector of 8 entries, the color in $L^*a^*b^*$ space, the texture properties of contrast, anisotropy and polarity and the position (x, y). The image is treated as mixture of Gaussians and is segmented using the Expectation-Minimization algorithm. The user can query Blobworld using simple queries of finding images similar to a selected blob or creating complex queries using other blobs.

The multimedia analysis and retrieval system (*MARS*): The multimedia analysis and retrieval system (MARS) [19] allows image queries using color, texture, shape and layout features. 2D color histograms in the HSV color space are used as the color feature, texture is represented using the coarseness, contrast and directionality (CCD) values. Shape based queries are matched using modified fourier descriptors (MFD). The layout queries uses a combination of the above features with respect to the image. The retrieval methods are Fuzzy Boolean Retrieval, where the distance between

[4] WebSeek: http://www.ctr.columbia.edu/webseek/.

[5] VideoQ: http://www.ctr.columbia.edu/VideoQ.

[6] BlobWorld: http://elib.cs.berkeley.edu/photos/blobworld.

the query image and the database image is the degree of membership to the fuzzy set of images that match the specified feature, and Probabilistic Boolean Retrieval, which determines the measure of probability that the image matches the user's information need. Recent updates to the system include query formulation and expansion along with relevance feedback.

PicToSeek: The PicToSeek[7] system [20] uses the color variation, saturation, transition strength, background and grayness as the indexing features to provide content based access to images. The system is based on the philosophy that the color based image indexing systems published in the literature do not take into account the camera position, geometry and illumination. The color transition strength is the number of hue changes in the image. The HSV color system is selected since it has the value parameter which represents the image intensity and which can be extended to provide intensity invariant matching. A reflection model is constructed for each object in the image in the sensor space and the photometric color invariant features are computed that are used for illumination invariant image retrieval. In their experiments, histogram cross-correlation provided the best results. Sub-image matching is provided through the use color-invariant active contours (snakes). The system also uses a color invariant gradient scheme for retrieval of multi-valued images. The PicToSeek system provides a combination of color, texture and shape features for image retrieval.

Content-based image retrieval in digital libraries (*C-BIRD*): The C-BIRD[8] system presents an illumination invariant approach to retrieval of color images through the color channel normalization step [21]. The color information is reduced from 3 planes by projecting them onto a single plane connecting the unity values on the R, G and B color axes. While usage of the chromaticity plane provides illumination invariance, it fails to provide chrominance changes in the illumination. This is achieved by applying L_2 normalization on the R, G and B vectors formed by taking the value of each of these components at each pixel location. Histogram intersection is used for image matching. The histogram size is reduced by using the discrete cosine transform. Object search is facilitated through feature localization and a three step matching algorithm which operates on image locales that are formed of dominant color regions. The texture in these regions is determined by computing the edge separation and directionality, while the shape is represented using the generalized Hough transform.

Query-by-image-content (*QBIC*): The query-by-image-(and video)-content (QBIC) system developed at IBM Almaden Research Center is described as a set of technologies and associated software that allows a user to search, browse and retrieve image, graphic and video data from large on-line collections [22]. The system allows image and video databases to be queried using visual features such as color, layout and texture that are matched against a meaningfully clustered database of precomputed features. A special feature in QBIC is the use of gray level images where the spatial distribution of gray level texture and edge information is used. The feature vector thus generated has a very high dimension (sometimes over 100). QBIC allows QBE and query-by-user-sketch type queries. Video retrieval is supported by generation of video storyboards which consist of representative frames selected from subsequences separated by significant changes such as scene cuts or gradual transitions. Technologies from QBIC are combined with results from speech recognition research to form an integrated system called CueVideo [23].

Informedia digital video library: The Informedia Digital Video Library Project[9] at Carnegie Mellon University is a full fledged digital video library under development [24]. The developers of the system project that a typical digital library user's interests will lie in short video clips and content-specific segments *skims*. The library developers have designed methods to create a short synopsis of each video. Language understanding is applied to the audio track to extract meaningful keywords. Each video in the database is then represented as a group of representative frames extracted from the video at points of significant activity. This activity may be abrupt scene breaks, some form of rapid camera movement, gradual changes from one scene to another, and points in the video where the keywords appear. Caption text[10] is also extracted from these frames which add to the set of indices for the video.

Other systems: Chabot[11] is a picture retrieval system which uses a relational database management system for storing and managing the images and their associated textual data [25]. To implement retrieval from the current collection of images, Chabot integrates the use of stored text and other data types with content-based analysis of the images to perform "concept queries". The project has a mix of annotated and un-annotated images, and aims to develop methods for retrieving them based on color, texture and shape. The Manchester multimedia system [26] uses geometric features such as area, perimeter, length, bounding box, longest chord, etc., of the image objects as features. VisualGREP [27] is a system for comparing and retrieving video sequences. The features used by this system are the color atmosphere represented as color coherence vector (CCV), the motion intensity represented by the edge change ratio (ECR), and presence of frontal

[7] PicToSeek: http://www.wins.uva.nl/research/isis/zomax/.

[8] C-BIRD: http://jupiter.cs.sfu.ca/cbird/.

[9] Informedia: http://www.informedia.cs.cmu.edu.

[10] Text added in video post-production by graphic editing machines.

[11] Chabot : http://www.cs.berkeley.edu/ ginger/chabot.html.

face in the image. Cox et al. describe the *PicHunter* system in Ref. [28] that is based on a Bayesian framework which is used to capture the relevance feedback to improve search results.

3. Similarity measures

This section describes the similarity measures used for matching visual information and the approaches taken to improve similarity results. The similarity is determined as a distance between some extracted feature or a vector that combines these. The most common scheme adopted is the use of histograms which are deemed similar if their distance is less than or equal to a preset distance threshold. Some of the common histogram comparison tests are described here. In other cases, the features are clustered based on their equivalency which is defined as the distance between these features in some multi-dimensional space. The most commonly used distance measures are the Euclidean distance, the Chamfer distance, the Hausdorff distance, the Mahalanobis distance, etc. The clustering methods commonly used are k-means clustering, fuzzy k-means, minimum spanning tree (MST), etc. Details on these can be found in a pattern classification text such as Ref. [29]. A review of statistical pattern recognition and clustering and classification techniques is covered in Ref. [30].

If H_{curr} and H_{prev} represent the histograms having n bins and i is the index into the histograms then the difference between these histograms is given by D as shown in Eq. (1). Histogram intersection is defined as in Eq. (2) where (when normalized) a value close to 1.0 represents similarity. The Yakimovsky test was proposed to detect the presence of an edge at the boundary of two regions. The expression for Yakimovsky Likelihood Ratio is given by Eq. (3), σ_1^2 and σ_2^2 are the individual variances of the histograms while σ_0^2 is the variance of the histogram generated from the pooled data. The numbers m and n are the number of elements in the histogram. A low value of y indicates a high similarity:

$$D = \sum_{i=0}^{n} |H_{curr}(i) - H_{prev}(i)|, \tag{1}$$

$$D_{int} = \frac{\sum_{i=1}^{n} \min(H_{curr}(i), H_{prev}(i))}{\sum_{i=1}^{n} H_{prev}(i)}, \tag{2}$$

$$y = \frac{(\sigma_0^2)^{m+n}}{(\sigma_1^2)^m (\sigma_2^2)^n}. \tag{3}$$

The χ^2 test for comparing two histograms as proposed by Nagasaka and Tanaka [31] is given by Eq. (4), where low value for χ^2 indicates a good match. The Kolmogorov–Smirnov test, given by Eq. (5), is based on the cumulative distribution of the two sets of data. Let $CH_{prev}(j)$ and $CH_{curr}(j)$ represent the cumulated number of entities up to the jth bin of the histograms of the previous and current frame given by H_{prev} and H_{curr}. A low value of D results for a good match. The Kuiper test, defined in Eq. (6), is similar to the Kolmogorov–Smirnov test, but is more sensitive to the tails of the distributions:

$$\chi^2 = \sum_{i=1}^{n} \frac{(H_{curr}(i) - H_{prev}(i))^2}{(H_{curr}(i) + H_{prev}(i))^2}, \tag{4}$$

$$D = \max_j |CH_{prev}(j) - CH_{curr}(j)|, \tag{5}$$

$$D = \max_j [CH_{prev}(j) - CH_{curr}(j)] + \max_j [CH_{curr}(j) - CH_{prev}(j)], \tag{6}$$

$$d = (I - M)^{\mathrm{T}} \mathscr{A} (I - M). \tag{7}$$

The perceptual similarity metric [32], defined in Eq. (7), is similar to the quadratic distance function. Here I and M are histograms with n bins. Matrix $\mathscr{A} = [a_{ij}]$ contains similarity weighting coefficients between colors corresponding to bins i and j. Other metrics seen in the literature are the Kantorovich metric (also referred to as the Hutchison metric) [33], the Choquet integral distance [34], the Minkowski distance metric given by Eq. (8), three special cases of the L_M metric (L_1, L_2 and the L_∞), and the Canberra distance metric shown in Eq. (9). The Czekanowski coefficient, defined in Eq. (10), is applicable only to vectors with non-negative components. In Eqs. (8)–(10), p is the dimension of the vector $\vec{x}_i$ and x_i^k is the kth element of $\vec{x}_i$ [35]:

$$d_M(i,j) = \left(\sum_{k=1}^{p} |(x_i^k - x_j^k)|^M \right)^{1/M}, \tag{8}$$

$$d_c(i,j) = \sum_{i=1}^{p} \frac{|x_i^k - x_j^k|}{|x_i^k + x_j^k|}, \tag{9}$$

$$d_z(i,j) = 1 - \frac{2 \sum_{k=1}^{p} \min(x_i^k, x_j^k)}{\sum_{k=1}^{p} (x_i^k + x_j^k)}. \tag{10}$$

Brunelli and Mich [36] present a study on the effectiveness of the use of histograms in image retrieval. Stricker [37] has done a study on the discrimination power of the histogram based color indexing techniques. In his work he makes the observation that histogram based techniques would work effectively only if the histograms are sparse. He also determines the lower and upper bounds on the number of histograms that fit in the defined color space and suggests ways to determine the threshold based on these bounds.

3.1. Improving similarity results

Content based searches can be classified into three types, the *target search* where a specific image is sought, the *category search* where one or more images from a category are sought, and the *open-ended* browsing in which the user seeks an image by specifying visually important properties [38]. Histograms and other features can be used for target searches while open-ended browsing can be achieved by allowing the user to select a visual item and finding other similar visual information. Relevance feedback has been proposed as a method for improving category based searches [39–42]. At each step the user marks the search results as relevant or irrelevant to the specified query. The image database learns from the user response and categorizes the images based on user input. Meilhac and Nastar [38] present the use of relevance feedback in category searches. The authors use Parzen window estimator for density estimation, the Bayes Inference rule to compute the probability of relevance versus the probability on non-relevance in the database. The strategy has been used in SurfImage [12] which is an interactive content-based image retrieval system.

Minka and Picard [43] observe that it is often difficult to select the right set of features to extract to make the image database robust. They use multiple feature models in a system that *learns* from the user queries and responses about the type of feature that would best serve the user. The problems of using multiple features is reduced by a multi-stage clustering and weight generation process. The stages closest to the user are trained the fastest and the adaptations are propagated to earlier stages improving overall performance with each use. Classification based approach using the Fisher Linear Discriminant and the Karhunen–Loeve transform is described in Ref. [44].

3.2. Determining visual semantics

Santini and Jain [45] present a study on some of the problems found in attempting to extract meaning from images in a database. They argue that the image semantics is a poorly defined entity. It is more fruitful to derive a correlation between the intended user semantics and the simple perceptual clues that can be extracted. It is contended that the user should have a global view of the placement of images in the database and also be able to manipulate its environment in a simple and intuitive way. They point out that specifying “more texture” and “less color” is not intuitive to the user in the effect it will have on the database response to the query. This places every image in the context of other images that, given the current similarity criterion, are similar to it. Two interfaces are proposed for meaningful results to be possible; viz., the *direct manipulation* interface and the *visual concepts* interface.

Direct manipulation allows the manipulation of the image locations in various clusters, thereby forming new semantics and allowing different similarity interpretations. Such an interface will require three kinds of spaces to be defined; viz., the feature space, the query space and the display space. When specifying visual concepts, the user could select equivalent images from the database and place them in a bucket thus creating new semantics. The equivalency of the images in the bucket is decided by the user on a concept meaningful to the current application. But this equivalency could be in a high dimensional feature space and mapping this to a display space could be a challenging task. For such a specification to be possible the database must allow arbitrary similarity measures to be accommodated and also be able to make optimal use of the semantics induced by the use of such visual concepts.

3.3. Similarity measures

Similarity measures for selecting images based on characteristics of human similarity measurement are studied in Ref. [46]. The authors apply results from human psychological studies to develop a metric for qualifying image similarity. The basis for their work is the need to find a similarity measure relatively independent of the feature space used. They state that if S_A and S_B are two stimuli, represented as vectors in a space of suitable dimension, then the similarity between the two can be measured by a psychological distance function $d(S_A, S_B)$. They introduce the difference between *perceived dissimilarity* d and *judged dissimilarity* δ. The two are related by a monotonically decreasing function g given by $\delta(S_A,S_B)=g[d(S_A,S_B)]$. The requirements of the distance function are constancy of self-similarity, minimality, and symmetrical distance between the stimuli.

The authors lay the foundation for set-theoretic and fuzzy set-theoretic similarity measures. They also introduce three operators which are similar in nature to the *and*, *or* and *not* operators used in relational databases but are able to powerfully express complex similarity queries. It is shown that the assumption made by most similarity based methods that the feature space is Euclidean is incorrect. They analyze some similarity measures proposed in the psychological literature to model human similarity perception, and show that all of them challenge the Euclidean assumption in non-trivial ways. The suggestion is that similarity criteria must work not only for images *very* similar to the query, as would be the case for matching, but for all the images *reasonably* similar to the query. The global characteristics of the distance measure used are much more important in this case.

[12] SurfImage Demo: http://www-rocq.inria.fr/cgi-bin/imedia/surfimage.cgi.

The propositions used to assess the similarity are predicates. For textures, a wavelet transform of the image is taken, and the energy (sum of the squares of the coefficients) in each sub-band is computed and included in the feature vector. To apply the similarity theory, a limited number of predicate-like features such as *luminosity*, *scale*, *verticality* and *horizontality* are computed. The distance between two images can be defined according to the Fuzzy Tversky model:

$$\begin{aligned} S(\phi,\psi) = {} & \sum_{i=1}^{p} \max\{\mu_i(\phi),\mu_i(\psi)\} \\ & - \alpha \sum_{i=1}^{p} \max\{\mu_i(\phi)-\mu_i(\psi),0\} \\ & + \beta \sum_{i=1}^{p} \max\{\mu_i(\phi)-\mu_i(\psi),0\}, \end{aligned} \tag{11}$$

where p is the number of predicates, μ is the truth value vector formed on predicates applied to images ϕ and ψ, and α and β are constants which decide the importance of the results.

4. Pattern recognition methods for image databases

The methods described in the literature have been found to use three types of features extracted from the image. These features are color based, shape based and texture based. Some systems use a combination of features to index the image database.

4.1. Color based features

Color has been the most widely used feature in CBIR systems. It is a strong cue for retrieval of images and also is computationally least intensive. Color indexing methods have been studied using many color spaces, viz., RGB, YUV, HSV, $L^*u^*v^*$, $L^*a^*b^*$, the opponent color space and the Munsell space. The effectiveness of using color is that it is an identifying feature that is local to the image and largely independent of view and resolution. The major obstacles that should be overcome to find a good *image* match are variation in viewpoint, occlusion, and varying image resolution. Swain and Ballard [47] use histogram intersection to match the query image and the database image. Histogram intersection is robust against these problems and they describe a preprocessing method to overcome change in lighting conditions.

In the PICASSO system [48] the image is hierarchically organized into non-overlapping blocks. The clustering of these blocks is then done on the basis of color energy in the $L^*u^*v^*$ color space. Another approach similar to the Blobworld project is to segment the image into regions by segmenting it into non-overlapping blocks of a fixed size and merging them on overall similarity [49]. A histogram is then formed for each such region and a family of histograms describes the image. Image matching is done in two ways—chromatic matching is done by histogram intersection; and geometric matching is done by comparing the areas occupied by the two regions. A similarity score is the weighted average of the chromatic and geometric matching. Other color based features derived from block based segmentation of the image are the mean, standard deviation, RMS, skew and kurtosis of the image pixel $I(p)$ in a window [50]. These are represented using a five-dimensional feature vector. The image similarity is then a weighted Euclidean distance between the corresponding feature vectors. The similarity score using these statistical feature vectors are given by

$$S_{stat}(A,B) = \sqrt{\sum_{f=1}^{F} w_f(df(f,A)-df(f,B))^2}. \tag{12}$$

In Eq. (12), $df(f,I)$ denotes the value of distribution of feature f in image I. F denotes the number of features used to describe the color distribution and w_f is the weight of the fth feature vector which is larger for features of smaller variance. Lin et al. [51] use 2D-pseudo-hidden Markov model (2D-PHMM) for color based CBIR. The reason for using 2D-PHMM is its flexibility in image specification and partial image matching. 2 1-D HMMs are used to represent the rows and the columns of the quantized color images. Müller et al. [52] present an improvement by combining shape and color into a single statistical model using HMMs for rotation invariant image retrieval.

Pass et al. [53] have developed color coherence histograms to overcome the matching problems with standard color histograms. Color coherence vectors (CCV) include spatial information along with the color density information. Each histogram bin j is replaced with a tuple (α_j,β_j), where α is the number of pixels of the color in bin j that are coherent, or belong to a contiguous region of largely that color. The β number indicates the number of incoherent pixels of that color. They also propose the use of joint histograms for determining image similarity. A joint histogram is defined as a multi-dimensional histogram where each entry counts the number of pixels in the image that describe some particular set of features such as color, edge density, texture, gradient magnitude and rank. Huang et al. [54] propose the use of color correlograms over CCVs. A color correlogram expresses the spatial correlation of pairs of colors with distance. It is defined as

$$\gamma^{(k)}_{c_i,c_j}(\mathscr{I}) \stackrel{\triangle}{=} \Pr_{\substack{p_1\in\mathscr{I}_{c_i}\\ p_2\in\mathscr{I}}} [p_2\in\mathscr{I}_{c_j} \,|\, |p_1-p_2|=k]. \tag{13}$$

Eq. (13) gives the probability that, in the image $\mathscr{I}$, the color of the pixel p_2 at distance k from the pixel p_1 with color c_i will be c_j. This includes the information of the spatial correlation between colors in an image, thus making it robust against change of viewpoint, zoom-in, zoom-out, etc.

Tao and Grosky [55] propose using Delauney triangulation based approach to classifying image regions based on color. Cinque et al. [56] use spatial-chromatic histograms for determining image similarity. Each entry in this histogram consists of a tuple with three entries, the probability of the color, the baricenter of the pixels with the same color which is akin to the centroid, and the standard deviation in the spread of the pixels of the same color in the image. The image matching is done by separately considering the color and spatial information about the pixels in a weighted function. Chahir and Chen [57] segment the image to determine color-homogeneous objects and the spatial relationship between these regions for image retrieval. Smith and Li [58] use composite region templates (CRTs) to identify and characterize the scenes in the images by spatial relationships of the regions contained in the image. The image library pools the CRTs from each image and creates classes based on the frequency of the labels. Bayesian inference is used to determine similarity. Brambilla et al. [59] use multi-resolution wavelet transforms in the modified $L^*u^*v^*$ space to obtain image signatures for use in content based image retrieval applications. The multi-resolution wavelet decomposition is performed using the Haar wavelet transform by filtering consecutively along horizontal and vertical directions. Image similarity is determined through a supervised learning approach.

Vailaya et al. [60] propose classification of images into various classes which the can be easily identified. They use the Bayes decision theory to classify vacation photographs images into indoor versus outdoor and scenery versus building images. The class-conditional probability density functions are estimated under a vector quantization (VQ) framework. MDL-type principle is used to determine the optimal size of the vector codebook. Androutsos et al. [61] drive home the importance of being able to specify colors which should be excluded from the image retrieval query. Most color based image retrieval systems allow users to specify colors in the target image, but do not handle specifications for color exclusion. The authors index images on color vectors in the RGB color space defining regions in the image. The similarity metric is based on the angular distance between the query vector and the target vector and is given by

$$\beta(x_i,x_j) = \exp\left(-\alpha\left(1 - \left[1 - \frac{2}{\pi}\cos^{-1}\left(\frac{x_i \cdot x_j}{|x_i|\,|x_j|}\right)\right] \times \left[1 - \frac{|x_i - x_j|}{\sqrt{3.255^2}}\right]\right)\right). \tag{14}$$

In Eq. (14), x_i and x_j are the three-dimensional color vectors being compared, α is a design parameter and $2/\pi$ and $\sqrt{3.255^2}$ are normalizing factors. A vector defining the minimum such distances for each query color is formed. This vector then defines the multi-dimensional query space. From the target images selected by this process those images which have minimum distance to the excluded colors are then removed:

$$D^M_{q,i} = |\tilde{f}_q - \tilde{f}_i| = \sum_{R,G,B} |\mu_q - \mu_i|, \tag{15}$$

$$D^E_{q,i} = \sqrt{(\tilde{f}_q - \tilde{f}_i)^2} = \sqrt{\sum_{R,G,B} (\mu_q - \mu_i)^2}. \tag{16}$$

While histogram and other matching methods are very effective for color based matching, color cluster provides a healthy alternative to it. In Refs. [62–64], several approaches to color based image retrieval are discussed. The distance method uses a feature vector which is formed of the mean values obtained from the 1-D histograms (normalized for the number of pixels) of each color component R, G and B given by $\tilde{f} = (\mu_R, \mu_G, \mu_B)$, where μ_i is the mean of the distribution of color component i. The distance between the query image q with feature vector $\tilde{f}_q$ and the database image i with feature vector $\tilde{f}_i$ is computed using the Manhattan distance measure, given by $D^M_{q,i}$, and the Euclidean distance measure, given by $D^E_{q,i}$, as shown in Eqs. (15) and (16), respectively. If all the colors of the images in the database could be perceptually approximated, the reference color method can be used. In this case the feature vector is given by $\tilde{f} = (\lambda_1, \lambda_2, \ldots, \lambda_n)$, where λ_i is the relative pixel frequency of the reference color i. A weighted Euclidean distance measure is used to compute the similarity between the query image and the database image. The drawback of the reference color table based method is that one has to know all the colors of all the images in the database. This makes modifications to the database difficult and is also infeasible for real world images and large archives. In Ref. [63], the authors present a color clustering based approach, in the $L^*u^*v^*$ space, to determination of image similarity. Such approaches are good for images with few dominant colors. Two classifiers are discussed for image matching, the minimum distance classifier adjusted for pixel population and a classifier based on the Markov random field process. The latter uses the spatial correlation property of image pixels. This method was found to be an improvement over methods proposed earlier and does very well on color images without requiring a priori information about the images. Kankanhalli et al. [65] present a method for combing color and spatial clustering for image retrieval. Some other methods for clustering color images can be found in Refs. [66,67]. A comparison and study of color clustering on image retrieval is presented in Refs. [68,69].

Hierarchical color clustering has been proposed as a more robust approach to indexing and retrieval of images [70,71]. Several problems have been identified with traditional histogram based similarity measures. The computational complexity is directly related to the histogram resolution. Color quantization reduces this complexity, but also causes loss of information and suffers from increased sensitivity to quantization boundaries. Wang and Kuo [70] develop a hierarchical feature representation and comparison scheme for image retrieval. The similarity between images is defined as the distance between the features at the kth resolution.

4.2. Shape based features

Different approaches are taken for matching shapes by the CBIR systems. Some researchers have projected their use as a matching tool in QBE type queries. Others have projected its use for query-by-user-sketch type queries. The argument for the latter being that in a user sketch the human perception of image similarity is inherent and the image matching sub-system does not need to develop models of human measures of similarity. One approach adopts the use of deformable image templates to match user sketches to the database images [72–74]. Since the user sketch may not be an exact match of the shape in the database, the method elastically deforms the user template to match the image contours. An image for which the template has to undergo minimal deformation, or, loses minimum energy, is considered as the best match. A low match means that the template is lying in areas where the image gradient is 0. By maximizing the matching function and minimizing the elastic deformation energy, a match can be found. The distance between two images (or image regions) is given by

$$\mathscr{D}(T,I,u) = \int\int_S (Tu_1(x_1,x_2),u_2(x_1,x_2)) - I(x_1,x_2))^2 \,\mathrm{d}(x_1,x_2), \quad (17)$$

$$\mathscr{J}(f) = \int\int_S ((f_{x_1x_1})^2 + 2(f_{x_1x_2})^2 + (f_{x_2x_2})^2)\,\mathrm{d}(x_1,x_2), \quad (18)$$

$$\mathscr{F} = \mu(\mathscr{J}(u_1) + \mathscr{J}(u_2)) + \mathscr{D}(T,I,u). \quad (19)$$

In Eq. (17), x_1 and x_2 are coordinate of some point on the grid on surface S, $u=(u_1,u_2)$ defines the deforming function causing the template and the target image to match, resulting in new coordinates given by $u_1(x_1,x_2)$ and $u_2(x_1,x_2)$ for template T and image I. The total amount of bending of the surface defined by $(x_1,x_2,f(x_1,x_2))$ is measured as in Eq. (18). The deformation energy is thus $\mathscr{J}(u_1)+\mathscr{J}(u_2)$. The balance between the amount of warp and the energy associated with it is given by Eq. (19), where μ controls the stiffness of the template. Adoram and Lew [75] use gradient vector flow (GVF) based active contours (snakes) to retrieve objects by shape. They note that deformable templates are highly dependent on their initialization and are unable to handle concavities. The authors present results by combining GVF snakes with invariant moments. Günsel and Tekalp [76] define a shape similarity based directly on the elements of the mismatch matrix derived from the eigenshape decomposition. A proximity matrix is formed using the eigenshape representation objects. The distance between the eigenvectors of the query and target object proximity matrices forms the mismatch matrix. The elements of the mismatch matrix indicate the matched feature points. These are then used to determine the similarity between the shapes.

A different approach to CBIR based on shape has been through use of implicit polynomials for effective representation of geometric shape structures [77]. Implicit polynomials are robust, stable and exhibit invariant properties. The method is based on fixing a polynomial to a curve patch. A vector consisting of the parameters of this curve is used to match the image to the query. A typical database would contain the boundary curve vectors at various resolutions to make the matching robust. Alferez and Wang [78] present a method to index shapes which is invariant to affine transformations, rigid-body motion, perspective transforms and change in illumination. They use a parameterized spline and wavelets to describe the objects. Petrakis and Milios [79] use a dynamic programming based approach for matching shapes at various levels of shape resolution. Mokhtarian and Abbasi [80] apply the curvature scale space based matching for retrieval of shapes under affine transform.

Sharvit et al. [81] propose the use of shock structures to describe shapes. The describe the symmetry-based representation as one which retains the advantages of the local, edge-based correlation approaches, as well as of global deformable models. It is termed as an intermediate representation. Two benefits of this approach have been outlined; the computation of similarity between shapes and the hierarchical symmetries captured in a graph structure. Rui et al. [82] propose the use of multiple matching methods to make the retrieval robust. They define the requirements of the parameter as invariance and compact form of representation. The authors define a modified Fourier descriptor (MFD) which is an interpolated form of the low frequency coefficients of the Fourier descriptor normalized to unit arc-length. They also calculate the orientation of the major axis. The matching of the images is then done using the Euclidean distance, MFD matching, Chamfer distance and Hausdorff distance. Although these matching tools have been used in this system, they can also be used to match shapes which have been specified using other appropriate descriptors. Jain and Vailaya present a study of shape based retrieval methods with respect to trademark image databases [83]. An invariant-based shape retrieval approach has been presented by Kliot and Rivlin [84]. Semi-local multi-valued invariant signatures are used to describe the images. Such representation when used with containment trees, a data structure introduced by the authors, allows for matching shapes which have undergone a change in viewpoint, or are under partial occlusion. It also allows retrieval by sketch. The invariant shape reparameterization is done by applying various transforms (translation, rotation, scale) to the curve signature.

Translation, rotation and scale invariance, which is imperative for shape based retrieval, can also be achieved through the use of Fourier–Mellin descriptors. Derrode et al. [85] base their system on these and describe them to be stable under small shape distortions and numerical approximations. The analytical Fourier–Mellin transform (AFMT) is used to extract the Fourier–Mellin descriptors. For an irradiance function $f(r,\theta)$ representing a gray level image over $\Re^2$ and the origin of the polar coordinates located over the image centroid, the AFMT of f is given in Eq. (20). For all k in Z, v in $\Re$, and $\sigma > 0$, f is assumed to be square summable under the measure $\mathrm{d}r\,\mathrm{d}\theta/r$:

$$M_f^\sigma(k,v) = \frac{1}{2\pi}\int_0^1\int_0^{2\pi} f(r,\theta)r^{\sigma-\mathrm{i}v}\mathrm{e}^{-\mathrm{i}k\theta}\frac{\mathrm{d}r}{r}\,\mathrm{d}\theta. \qquad (20)$$

4.3. Texture based features

The visual characteristics of homogeneous regions of real-world images are often identified as texture. These regions may contain unique visual patterns or spatial arrangements of pixels which gray level or color in a region alone may not sufficiently describe. Typically, textures have been found to have strong statistical and/or structural properties. The textures have been expressed using several methods. One system uses the quadrature mirror filter (QMF) representation of the textures on a quad-tree segmentation of the image [86]. Fisher's discriminant analysis is the used to determine a good discriminant function for the texture features. The Mahalanobis distance is used to classify the features. An image can be described by means of different orders of statistics of the gray values of the pixels inside a neighborhood [87]. The features extracted from the image histogram, called the first order features, are mean, standard deviation, third moment and entropy. The second order features are homogeneity, contrast, entropy, correlation, directionality and uniformity of the gray level pixels. Also included is the use of several other third order statistics from run-length matrices. A vector composed of these features is then classified based on the Euclidean distance.

The Gabor filter and wavelets can also be used to generate the feature vector for the texture [88]. The assumption is that the texture regions are locally homogeneous. The feature vector is then constructed from multiple scales and orientations comprising of the means and standard deviations at each. Other work done by Puzicha et al. [89] uses Gabor filters to generate the image coefficients. The probability distribution function of the coefficients is used in several distance measures to determine image similarity. The distances used are the Kolmogorov–Smirnov distance, the χ^2-statistic, the Jeffery divergence, weighted mean and variance among others. Mandal et al. [90] propose a fast wavelet histogram technique to index texture images. The proposed technique reduces the computation time compared to other wavelet histogram techniques. A rotation, translation and scale invariant approach to content based retrieval of images is presented by Milanese and Cherbuliez [91]. The Fourier power spectra coefficients transformed to logarithmic–polar coordinate system are used as the image signature for image matching. The image matching is done using the Euclidean distance measure. Strobel et al. [92] have developed a modified maximum a posteriori (MMAP) algorithm to discriminate between hard to separate textures. The authors use the normalized first order features, two circular Moran autocorrelation features and Pentland's fractal dimension to which local mean and variance has been added. The Euclidean distance metric is used to determine the match. To decorrelate the features the singular value decomposition (SVD) algorithm is used.

In the Texture Photobook due to Pentand et al. [93] the authors use the model developed by Liu and Picard [94] based on the Wold decomposition for regular stationary stochastic processes in 2D images. This model generates parameters that are close if the images are similar. If the image is treated as a homogeneous 2D discrete random field, then the 2D Wold like decomposition is a sum of three mutually orthogonal components which qualitatively appear as periodicity, directionality and randomness, respectively. The Photobook consist of three stages. The first stage determines if there is a strong periodic structure. The second stage of processing occurs for periodic images on the peaks of their Fourier transform magnitudes. The third stage of processing is applied when the image is not highly structural. The complexity component is modeled by use of a multi-scale simultaneous autoregressive (SAR) model. The SAR parameters of different textures are compared using the Mahalanobis distance measure.

4.4. Combination of features

Several approaches take advantage of the different features by applying them together. The features are often combined into a single feature vector. The distances between the images is then determined by the distance between the feature vectors.

Zhong and Jain [95] present a method for combining color, texture and shape for retrieval of objects from an image database without preindexing the database. Image matching is done in the discrete cosine transform (DCT) domain (assuming that the stored images are in the JPEG[13] format). Color and texture are used to prune the database and then deformable template matching is performed for identifying images that have the specified shape. Mojsilovic et al. [96] propose a method which combines the color and texture information for matching and retrieval of images. The authors extract various features such as the overall color, directionality, regularity, purity, etc., and build a rule based grammar for identifying each image:

$$w_i = \left(0.5 + \frac{0.5\mathrm{ff}}{\max\{\mathrm{ff}\}}\right) \log\left(\frac{N}{n}\right), \tag{21}$$

$$S(Q,D) = \frac{\sum_{k=1}^{t} \min\{w_{Qk}, w_{Ik}\}}{\sum_{k=1} t w_{Qk}}. \tag{22}$$

The PicToSeek system developed by Gevers and Smeulders uses a combination of color, texture and shape features for image retrieval. The images are described by a set of features and the weights associated with them. The weights are computed as the product of the features' frequency times the inverse collection frequency factor, given by Eq. (21), where feature frequency is given by ff, N is the number of images in the database, and n denotes the number of images to which a feature value is assigned. The image similarity is determined by Eq. (22), where Q is the query image vector of features as associated weights and D is the database image vector; each having t features.

Scarloff et al. [97] propose combining visual and textual cues for retrieval of images from the World Wide Web. Words surrounding an image in a HTML document or those included in the title, etc., may serve to describe the image. This information is associated with the image through the use of the latent semantic indexing technique. Normalized color histogram in the quantized $L^*u^*v^*$ space and texture orientation distribution using steerable pyramids form cues that are used to visually describe each image. The aggregate visual vector is composed of 12 vectors that are computed by applying color and texture features from on the entire image and five sub-image regions. It is assumed that the vectors are independent and have a Gaussian distribution. SVD is applied for these vectors over a randomly chosen subset of the image database and the average and covariance values for each subvector is computed. The dimensionality of the vector components is reduced by using principal component analysis. The final global similarity measure is a linear combination of the text and visual cues in normalized Minkowski distance. Relevance feedback is used to weight queries. Berman and Shapiro describe a flexible image database system (FIDS) [98] with an aim to reduce the computation time required to calculate the distance measures between images in a large database. FIDS presents the user with multiple measures for determining similarity and presents a method for combining these measures. The triangle inequality is used to minimize the index searches which are initially performed using keys which describe certain image features. The distance measures in FIDS compare the color, local binary partition texture measure, edges, wavelets, etc.

5. Pattern recognition methods for video databases

Pattern recognition methods are applied at several stages in the generation of video indices and in video retrieval. Video data has the feature of time in addition to the spatial information contained in each frame. Typically, a video sequence is made up of several subsequences, which are uniform in overall content between their start and end points marked by *cuts* or *shot changes*. A *scene change* is defined as a point when the scene content changes, e.g. indoor to outdoor scene. Some times these end points may also be gradual changes, called by the generic name *gradual transitions*. Gradual transitions are slow changes from the scene in one subsequence to the next that are further classified as *blends*, *wipes*, *dissolves*, etc. Thus, a video database can be indexed at the cut (or transition) points. Some methods cluster *similar* frames while others take the approach of finding peaks in the difference plot. This process of finding scene changes in a video sequence is also called *indexing*. Some methods go beyond this step by determining the camera motion parameters and classifying it as *zooms*, *pans*, etc. It has been observed that if the subsequences in the video clips can be identified and the scene change points marked, then mosaicing the data within a subsequence can provide a necessary representation for efficient retrieval. A video storyboard can be created by selecting a representative (key) frame from each shot. Visualization of the pictorial content and generation of story boards has been presented by Yeung and Yeo [99]. They define video visualization as a joint process of video analysis and subsequent generation of representative visual presentation for viewing and understanding purposes.

5.1. Video shot detection techniques

The simplest, and probably the most noisy, method for detecting scene changes is performing a pixel level differencing between two succeeding video frames. If the percentage change in the pixels is greater than a preset threshold, then a cut is declared. Another simple

[13] JPEG: Joint Photographic Experts Group—http://www.jpeg.org.

method is to perform the likelihood ratio test [3], given by Eq. (23). In this, each frame is divided into k equally sized blocks. Then, the mean (μ) and variance (σ^2) for each block is calculated. If the likelihood λ is greater than a threshold T, then the count of the blocks changed is incremented. The process is repeated for all the blocks. If the number of blocks changed is greater than another threshold T_B, then a scene break is declared. This method is a little more robust than the pixel differencing method but is insensitive to two blocks being different yet having the same density functions:

$$\lambda = \frac{[((\sigma_i^2 + \sigma_{i+1}^2)/2) + ((\mu_i - \mu_{i+1})/2)^2]^2}{\sigma_i^2 * \sigma_{i+1}^2}. \tag{23}$$

Many methods have been developed that use the color and/or intensity content of the video frame to classify scene changes [100]. Smith and Kanade [24] describe the use of comparative histogram difference measure for marking scene breaks. The peaks in the difference plot are detected using a threshold. Zabih et al. [101] detect cuts, fades and dissolves by counting the new and old edge pixels. Hampapur et al. [102] use chromatic scaling to determine the locations of cuts, dissolves, fades, etc. Kim et al. [103] present an approach to scene change detection by applying morphological operations on gray scale and binarized forms of the frames in a video sequence.

DCT coefficients from compressed video frames are also used to determine dissimilarity between frames [104–110]. One method applies standard statistical measures such as the Yakimovsky test, χ^2 test, and the Kolmogorov–Smirnov test to the histograms of average block intensities to detect shot changes. Another uses the error power of the frame as a feature to determine differences with other frames. The ratio of intra-coded macroblocks to inter-coded macroblocks and the ratio of backward prediction motion vectors to forward prediction motion vectors can also be used to detect shot changes in P- and B- frames. An intensity difference plot can also be used to detect gradual transitions. For these the ideal curve is parabolic. A dip in curve marks the center of dissolve. A curve is expected because the variance of frame intensity gradually increases, then stabilizes and gradually decreases into the new shot.

Hanjalic and Zhang [111] choose to minimize the average detection error probability criterion for detecting shot changes in video. The problem is reduced to deciding between two hypotheses; *No shot boundary* or *shot boundary*. They present a statistical framework for detecting shots and a variety of gradual transitions. The a priori information to be used in the error probability function is derived from statistical studies of shot lengths of a large number of motion pictures. These are fit to the Poisson function that is controlled by a parameter which is varied for different video types. Other shot detection methods are found in Refs. [112–116]. A comparison of various shot detection metrics is presented in [10,117–119].

5.2. Motion based video classification

The methods described in this section detect motion from compressed and uncompressed video. The motion information is used to detect shot changes (cuts), detect activity in the video or detect camera motion parameters (such as zooms, pans, etc.). Video sequences can be temporarily segmented into shots using local and global motion features. These features can then be used for determining shot similarity [120].

Akutsu et al. [121] have developed a motion vector based video indexing method. Motion vectors refer to the vector generated by a feature moving from one location in a frame to another location in a succeeding frame. Cut detection is based on the average inter-frame correlation coefficient based on the motion vectors. Two methods are proposed to determine the coefficient. The first is the inter-frame similarity based on motion. The other value is the ratio of velocity to the motion distance in each frame which represents motion smoothness. Camera operations are detected by applying motion analysis to the motion vectors using the Hough transform. A spatial line in the image space is represented as a point in the Hough space. Conversely, a sinusoid in the Hough space represents a group of lines in the image space intersecting at the same point. In this case, the lines intersect at the convergence/divergence point of the motion vectors. Thus, if the Hough transform of the motion vectors is least squares fit to a sinusoid, the point of divergence/convergence of vectors indicates the type of camera motion. Another method for detecting video zooms through camera motion analysis is presented by Fischer et al. [122]. Other block based motion estimation techniques for detecting cuts that operate on uncompressed video can be found in Refs. [123,124]. In the Informedia Project [24] trackable features of the object are used to form flow vectors. The mean length, mean phase and phase variance of the flow vectors determine if the scene is a static scene, pan or a zoom. MPEG motion vectors are also used to detect the type of camera motion [106,125]. MPEG motion flow extraction is simple and fast compared to optical flow and is suited for dense motion flows since each motion vector is for a 16 × 16 sub-image. Fatemi et al. [126] detect cuts and gradual transitions by dividing the video sequence into (overlapping) subsequence of 3 consecutive frames with a fourth predicted frame.

Naphade et al. [127] use the histogram difference metric (HDM) and the spatial difference metric (SDM) as features in k-means clustering. They contend that a shot change occurs when both these values are reasonably large. Günsel and Tekalp [128] use the sum of the bin-to-bin differences in the color histograms of the current and the next frame and the difference between the current frame histogram and the mean of the histograms of previous k frames to determine shot changes and select keyframes. If either component for a given frame is greater than the threshold, then it is labeled as a keyframe. A shot change is determined if the first component is greater than the threshold. The threshold is determined automatically by treating the shot change detection as a two class problem originally used to binarize images. Deng and Manjunath [129] describe the video object based recognition system NeTra-V. The system operates on MPEG compressed videos and performs region segmentation based on color and texture properties on I-frames. These regions are then tracked across the frames and a low level region representation is generated for video sequences:

$$M_{p,q}[I](m) = \sum_{k,l} I(k,l)\delta(m + qk - pl). \tag{24}$$

Coudert et al. [130] present a method for fast analysis of video sequences using the Mojette transform given by Eq. (24), where $I(k,l)$ denotes the image pixel, δ is the Dirac function and m is the Mojette index. The direction of the projection is determined by integers p and q given by $\tan(\varphi) = -p/q$. In essence $M(m)$ is the summation of intensity pixels along the line defined by $m - qk + pl = 0$. The transform is a discrete version of the Radon transform. Motion is estimated by the projected motion in the transform domain. The analysis is performed on the 1D representation of the image. Dağtaş et al. [131] present motion based video retrieval methods based on the trajectory and trail models. In the case of the trajectory model, the describe methods to retrieve objects based on the location, scale and temporal characteristics of trajectory motion. For retrieval of video clips independent of the temporal characteristics of the object motion, the trail based model is proposed. The approach uses the Euclidean distance measure and the Fourier transform for convolution.

Wu et al. [132] present an algorithm for detecting wipes in a video sequence. Wipes are used frequently in television news programs to partition stories. Detecting such graphic effects is useful for abstracting video. The method sums up the average pixel difference in DC images extracted from MPEG compressed video in the direction orthogonal to the wipe. A wipe creates a plateau in the difference plot which can be detected using standard deviation. Corridoni and Del Bimbo [133] present methods for indexing movie content by detecting special effects such as fades, wipes, dissolves and mattes. Fades and dissolves are modeled as the change in pixel intensity over the duration of the fade or dissolve. Wipes are detected by applying cut detection techniques to a series of frames and analyzing the difference graph. Mattes are treated as a special case of dissolves. A comparison of several methods for detection of special effects is covered in Ref. [134].

5.3. Video abstraction

Once the shot and scene changes in a video are detected, the camera and object motion characterized, it is necessary to select keyframes from the scenes or shots to generate a video storyboard. Hence, the research community has turned its attention to the problem of selection of keyframes. It is important to note that selecting a fixed frame from every shot is not a complete solution to the problem of video classification or abstraction. It is necessary to select keyframes from the salient shot changes only. A comprehensive study on existing methods for determining keyframes and an optimal clustering based approach is presented in Ref. [135].

Yeung et al. [136] use time-constrained clustering and scene transition graph (STG) to group similar shots together provide the basis for story analysis. Video is segmented into shots which are then clustered based on temporal closeness and visual similarity. The STGs are then used for forming the story units in videos. Rui et al. [137] take a more generalized approach to clustering similar shots. The approach is to determine visual similarity between shots and the temporal difference between shots. The overall similarity between shots is a weighted sum of the shot color similarity and the shot activity similarity.

Lienhart et al. [138] use video abstraction methods for classifying movies. Subsequent to shot segmentation, the movie subsequences are clustered together based on color similarity. Scene information is extracted based on audio signals and associated dialog. Special events such as explosions, close-up shots of actors, gunfire, etc., and credit titles are extracted for indexing the movie. Corridoni and Del Bimbo [133] also form a structured representation of movies. Their approach detects shot changes and gradual transitions such as fades, dissolves, wipes and mattes. Methods to separate commercials from television broadcasts have been presented in Refs. [139–141].

Zhuang et al. [142] use unsupervised clustering to cluster the first frame of every shot. The keyframes within each cluster provide a notion of similarity while providing indices into the video. Hanjalic et al. [143] break video sequences into logically similar segments based on visual similarity between the shots. The similarity is determined between shots by registering the similarity between MPEG-DC frames using the $L^*u^*v^*$ color space. In related work [135] present a method for determining the optimal number of clusters for a video. Each frame in the video is described by a D-dimensional feature vector. The value of D is dependent on the sequence length. The video is initially clustered into N clusters whose value is determined from the lengths of the video sequences in the database. In order to find an optimal number of clusters, the authors perform cluster validity analysis and present a cluster separation measure given by Eq. (25):

$$\rho(n) = \frac{1}{n}\sum_{i=1}^{n} \max_{1 \leqslant j \leqslant n \wedge i \neq j} \left(\frac{\xi_i + \xi_j}{\mu_{ij}} \right), \quad n \geqslant 2, \tag{25}$$

where,

$$\xi_i = \left\{ \frac{1}{E_i} \sum_{k=1}^{E_i} |\vec{\phi}(k \mid k \in i) - \vec{\phi}(c_i)|^{\eta_i} \right\}^{1/\eta_1},$$

$$\mu_{ij} = \left\{ \sum_{v=1}^{D} |\varphi_v(c_i) - \varphi_v(c_j)|^{\eta_2} \right\}^{1/\eta_2}.$$

Here E_i and ξ_i are number of elements and the dispersion of cluster i having centroid c_i, respectively. μ_{ij} is the Minkowski metric characterizing centroids of clusters i and j. $1, \ldots, N$ clusters are under consideration. $\vec{\phi}$ is the D-dimensional feature vector containing features φ_v. Different metrics are obtained for parameters η_1 and η_2. The suitable cluster structure for a video is determined by analyzing the normalized curve of $\rho(n)$. The keyframes are then selected by choosing the frames closest to the cluster centroid.

Chen et al. [144] describe ViBE, a video indexing and browsing environment. Video indexing is performed using generalized traces. A generalized trace is a high dimensional feature vector containing histogram intersection for each of the YUV color components with respect to the previous frame, their standard deviation for the current frame, and statistics from the MPEG stream about the number of intracoded, forward predicted and bidirectionally predicted macroblocks. This feature vector is used in a binary regression tree to determine shot boundaries. Each shot is then represented using a shot tree formed using the Ward's clustering algorithm also known as the minimum variance method. A similarity pyramid of shots is formed to aid browsing. Bouthemy et al. [145] describe the DiVAN project, which aims at building a distributed architecture for the managing and providing access to large television archives and has a part of it devoted to automatic segmentation of video. The video sequence is segmented into shots and gradual transitions along with camera motion analysis. Dominant colors are used to index keyframes. Chang et al. [146] present a keyframe selection algorithm based on the semi-Hausdorff Distance which is defined in Eq. (26), where glb represents the greatest lower bound, ε is the error bound and $A, B \subset \chi$ are two point sets in metric space (χ, d) where d is a predefined distance function. The authors also present keyframe extraction algorithms based on the keyframe selection criterion and methods to structure video for efficient browsing:

$$d_{SH}(A,B) = glb\{\varepsilon \mid A \subset U(B,\varepsilon)\}, \tag{26}$$

where

$$U(B,\varepsilon) = \bigcup_{x \in B} B_d(x,\varepsilon),$$

$$B_d(x,\varepsilon) = \{y \mid d(x,y) \leqslant \varepsilon\}.$$

Sahouria and Zakhor [147] present methods to describe videos using principal component analysis and generate high level scene description without shot segmentation. They also use HMMs to analyze the motion vectors extracted from MPEG P-frames to analyze sport sequences. Vasconcelos and Lippman [148] develop statistical models for shot duration and shot activity in video. These models are described as the first steps towards the semantic characterization of video. The authors develop Bayesian procedures for the tasks of video segmentation and semantic characterization since the knowledge of the video is available as prior knowledge from the shot activity.

5.4. *Video retrieval*

Video shot segmentation and abstraction are two parts of a larger problem of content based video retrieval. Once the video has been segmented into shots which may be represented by keyframes, scene transition graphs or multi-level representations, it is necessary to provide methods for similarity based retrieval. Jain et al. [149] present a method for retrieval of video clips which may be longer than a single shot. The methods allow retrieval based on keyframe similarity, akin to the image database approach. A video clip is then generated as a result by merging shot boundaries of those shots whose keyframes are highly similar to the query specification. In another approach subsampled versions of video shots are compared to a query clip. Image database matching techniques are extended for content based video retrieval in Ref. [150]. Fuzzy classification and relevance feedback is used for retrieval. A multi-dimensional feature vector is formed from color and motion segmentation of each video frame that is used to construct a vector and extract key shots and keyframes.

5.5. *Performance evaluation of video shot detection methods*

When a number of methods exist for performing a certain task, it is necessary to determine which method performs well on a generalized data set. Towards this, we have conducted a thorough evaluation of video shot segmentation methods that use color, motion and compression parameters [10,151]. In this paper we present the some results from the evaluation, the reader is encouraged to refer to our original publications wherein

Table 1
Cut detection performance: recall at precision = 95%

Color space	Histogram differencing method									
	B2B1D (%)	B2B2D (%)	B2B3D (%)	CHI1D (%)	CHI2D (%)	CHI3D (%)	INT1D (%)	INT2D (%)	INT3D (%)	AVG (%)
RGB	60		68	45		45	60		65	10
HSV	65	63	68	63	53	54	61	61	63	15
YIQ	68	37	62	62	23	49	64	50	60	14
LAB	70	51	65	59	42	50	64	57	63	15
LUV	68	37	64	59	29	47	67	49	67	14
MTM	69	65	68	59	54	55	67	63	70	13
XYZ	60	65	64	47	57	35	68	68	57	8
OPP	65	34	57	60	34	45	64	50	67	14
YYY	55			46			45			6

Table 2
Cut detection performance of MPEG algorithms

Algorithm	Detects	MDs	FAs	Recall (%)	Precision (%)
MPEG-A	932	27	14820	97	6
MPEG-B	473	486	3059	49	13
MPEG-C	286	673	795	30	26
MPEG-D	754	205	105	79	88
MPEG-E	862	97	4904	90	15
MPEG-F	792	167	663	83	54

detail on the methods and the evaluation has been provided. Other comparisons can be found in Refs. [8,9]. Four histogram comparison measures, Bin-to-bin difference (B2B), χ^2 test (CHI), histogram intersection (INT), and difference of average colors (AVG), were evaluated on eight color spaces, viz., RGB, HSV, YIQ, XYZ, $L^*a^*b^*$, $L^*u^*v^*$, Munsell and Opponent color spaces. Each of the histogram comparison measures were tested in 1, 2 and 3 dimensions. Six methods that use MPEG compression parameters were also evaluated [105,110,152–154,109] These have been labelled as *MPEG-A* through *MPEG-F*, respectively. Finally three algorithms that use block matching on uncompressed video data were evaluated [121,124,155], labeled as *Algorithm Block-A* through *Algorithm Block-C*.

As shown in Table 1 among the differencing methods, histogram intersection is the best. The CHI method did significantly poorly and is thus not suitable for coarse histogram differencing for shot-change detection. Amongst the bin-to-bin histogram comparison methods using 3 1D is less computationally intensive and hence more attractive. Amongst the MPEG algorithms (see Table 2), MPEG-D is clearly the best method for cut detection with both high precision and recall. Several other comparisons can be found in our work [10].

6. Summary and conclusions

This paper has listed the *state-of-the-art* in methods developed for providing content-based access to visual information. The visual information may be in the form of images or video stored in archives. These methods rely on pattern recognition methods for feature extraction, clustering, and matching the features. Pattern recognition thus plays a significant role in content based recognition and has applications in more than one sub-system in the database. Yet, very little work has been done on addressing the issue of human perception of visual data content. The approaches taken by the computer vision, image analysis and pattern recognition community have been *bottom up*. It is not only necessary to develop better pattern recognition methods to capture the visually important features from the image, but also to develop them such that they are simple, efficient and easily mapped to human queries. As presented above, efforts are being made to extend queries beyond use of simple color, shape, texture, and/or motion based features. Specification of queries using spatial and temporal fuzzy relationships between objects and sub-image blocks is being explored. Another aspect in content-based retrieval is the development of useful and efficient indexing structures. Commonly, the image databases cluster the features to enhance the similarity search. Similarity searches often are generalized k-nearest neighbor searches where performance and accuracy can be easily traded. Many approaches rely on clustering of indices which are then stored in a commercial database management systems. Other approaches used modified forms of other tree structures. Robust structures allowing the use of features and fuzzy queries need to be developed and a quantitative and qualitative performance analysis published. The Pattern Recognition community with its rich tradition of making fundamental contributions to diverse applications thus has a great opportunity to make break-through advances in content based image and video information retrieval systems.

References

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

[2] Y. Rui, T.S. Huang, S.F. Change, Image retrieval: current techniques, promising directions, and open issues, J. Visual Commun. Image Representation 10 (1) (1999) 39–62.

[3] G. Ahanger, T.D.C. Little, A survey of technologies for parsing and indexing digital video, J. Visual Commun. Image Representation 7 (1) (1996) 28–43.

[4] R. Brunelli, O. Mich, C.M. Modena, A survey on the automatic indexing of video data, J. Visual Commun. Image Representation 10 (2) (1999) 78–112.

[5] S. Antani, R. Kasturi, R. Jain, Pattern recognition methods in image and video databases: past, present and future, Joint IAPR International Workshops SSPR and SPR. Lecture Notes in Computer Science, Vol. 1451, 1998, pp. 31–58.

[6] F. Idris, S. Panchanathan, Review of image and video indexing techniques, J. Visual Commun. Image Representation 8 (2) (1997) 146–166.

[7] C. Colombo, A. Del Bimbo, P. Pala, Semantics in visual information retrieval, IEEE Multimedia 6 (3) (1999) 38–53.

[8] A. Di Lecce, V. Guerriero, An evaluation of the effectiveness of image features for image retrieval, J. Visual Commun. Image Representation 10 (4) (1999) 351–362.

[9] M.K. Mandal, F. Idris, S. Panchanathan, A critical evaluation of image and video indexing techniques in the compressed domain, Image Vision Comput. 17 (7) (1999) 513–529.

[10] U. Gargi, R. Kasturi, S.H. Strayer, Performance characterization of video-shot-change detection methods, IEEE Trans. Circuits Systems Video Technol. 10 (1) (2000) 1–13.

[11] N.-S. Chang, K.-S. Fu, Query-by-pictorial-example, IEEE Trans. Software Eng. 6 (6) (1980) 519–524.

[12] A. Gupta, S. Santini, R. Jain, In search of information in visual media, Commun. ACM 40 (12) (1997) 34–42.

[13] Y. Gong, Intelligent Image Databases—Towards Advanced Image Retrieval, Kluwer Academic Publishers, Boston, 1998.

[14] J.K. Wu, B.M. Mehtre, C.P. Lam, A. Desai Narasimhalu, J.J. Gao, Core: a content-based retrieval engine for multimedia information systems, Multimedia Systems 3 (1) (1995) 25–41.

[15] J.K. Wu, P. Lam, B.M. Mehtre, Y.J. Gao, A.D. Narasimhalu, Content based retrieval for trademark registration, Multimedia Tools Appl. 3 (3) (1996) 245–267.

[16] S.-F. Chang, J.R. Smith, M. Beigi, A. Benitez, Visual information retrieval from large distributed online repositories, Commun. ACM 40 (12) (1997) 62–71.

[17] D. Zhong, S.-F. Chang, An integrated approach for content-based video object segmentation and retrieval, IEEE Trans. Circuits Systems Video Technol. 9 (8) (1999) 1259–1268.

[18] C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein, J. Malik. Blobworld, A system for region-based image indexing and retrieval, Third International Conference on Visual Information and Information Systems (VISUAL'99), Appears in Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 509–516.

[19] K. Porkaew, M. Ortega, S. Mehrotra, Query reformulation for content based multimedia retrieval in MARS, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 747–751.

[20] T. Gevers, A.W.M. Smeulders, Pictoseek: combining color and shape invariant features for image retrieval, IEEE Trans. Image Process. 9 (1) (2000) 102–119.

[21] Z.-N. Li, O.R. Zaïane, Z. Tauber, Illumination invariance and object model in content-based image and video retrieval, J. Visual Commun. Image Representation 10 (3) (1999) 219–244.

[22] W. Niblack, X. Zhu, J.L. Hafner, T. Breuel, et al., Updates to the QBIC system, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases VI, Vol. SPIE 3312, 1997, pp. 150–161.

[23] D. Poncelon, S. Srinivasan, A. Amir, D. Petkovic, D. Diklic, Key to effective video retrieval: effective cataloging and browsing, ACM International Conference on Multimedia, 1998, pp. 99–107.

[24] M.A. Smith, T. Kanade, Video skimming for quick browsing based on audio and image characterization, Technical Report CMU-CS-95-186 Carnegie Mellon University, 1995.

[25] V. Ogle, M. Stonebraker, Chabot: retrieval from a relational database of images, IEEE Comput. 28 (9) (1995) 40–48.

[26] C. Goble, M. O'Docherty, P. Crowther, M. Ireton, J. Oakley, C. Xydeas, The Manchester multimedia information system, Proceedings of the E. D. B. T.'92 Conference on Advances in Database Technology, Vol. 580, 1994, pp. 39–55.

[27] R. Lienhart, W. Effelsberg, R. Jain, VisualGREP: a systematic method to compare and retrieve video sequences, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Database VI, Vol. SPIE 3312, 1997, pp. 271–282.

[28] I.J. Cox, M.L. Miller, T.P. Minka, T.V. Papathomas, P.N. Yianilos, The bayesian image retrieval system, pichunter: theory, implementation and psychophysical experiments, IEEE Trans. Image Process. 9 (1) (2000) 20–37.

[29] P.E. Duda, R.O. Hart, D.G. Stork, Pattern Classification, 2nd Edition, Wiley, Inc., New York, NY, 2000.

[30] A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: a review, IEEE Trans. Pattern Anal. Mach. Intell. 22 (1) (2000) 4–37.

[31] A. Nagasaka, Y. Tanaka, Automatic video indexing and full-video search for object appearances, Proceedings of IFIP 2nd Working Conference on Visual Database Systems, 1992, pp. 113–127.

[32] J. Hafner, H.S. Sawhney, W. Equitz, M. Flickner, W. Niblack, Efficient color histogram indexing for quadratic form distance functions, IEEE Trans. Pattern Anal. Mach. Intell. 17 (7) (1995) 729–736.

[33] K. Chen, S. Demko, R. Xie, Similarity-based retrieval of images using color histograms, Proceedings of

IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases VII, Vol. SPIE 3656, 1999.

[34] M. Popescu, P. Gader, Image content retrieval from image databases using feature integration by Choquet integral, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases VII, Vol. SPIE 3656, 1999.

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

[36] R. Brunelli, O. Mich, On the use of histograms for image retrieval, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 143–147.

[37] M. Stricker, Bounds of the discrimination power of color indexing techniques, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases II, Vol. SPIE 2185, 1994, pp. 15–24.

[38] C. Meilhac, C. Nastar, Relevance feedback and category search in image databases, IEEE International Conference on Multimedia Computing Systems, Vol. 1, 1999, pp. 512–517.

[39] G. Ciocca, R. Schettini, Using relevance feedback mechanism to improve content-based image retrieval, Third International Conference on Visual Information and Information Systems (VISUAL'99), Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 107–114.

[40] E. Di Sciascio, G. Mingolla, M. Mongiello, Content-based image retrieval over the Web using query by sketch and relevance feedback, Third International Conference on Visual Information and Information Systems (VISUAL'99), Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 123–130.

[41] J. Peng, B. Bhanu, S. Qing, Probabilistic feature relevance learning for content-based image retrieval, Comput. Vision Image Understanding 75 (1–2) (1999) 150–164.

[42] D. Squire, W. Müller, H. Müller, Relevance feedback and term weighting schemes for content-based image retrieval, Third International Conference on Visual Information and Information Systems (VISUAL'99), Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 549–556.

[43] T.P. Minka, R.W. Picard, Interactive learning with a society of models, Pattern Recognition 30 (4) (1997) 565–581.

[44] M.S. Lew, D.P. Huijsmans, D. Denteneer, Content based image retrieval: KLT, projections, or templates, Proceedings of the First International Workshop on Image Databases and Multi-Media Search, 1996, pp. 27–34.

[45] S. Santini, R. Jain, Beyond query by example, ACM International Conference on Multimedia, 1998, pp. 345–350.

[46] S. Santini, R. Jain, Similarity measures, IEEE Trans. Pattern Anal. Mach. Intell. 21 (9) (1999) 871–883.

[47] M.J. Swain, D.H. Ballard, Color indexing, Int. J. Comput. Vision 7 (1) (1991) 11–32.

[48] A. Del Bimbo, M. Mugnaini, P. Pala, F. Turco, Visual querying by color perceptive regions, Pattern Recognition 31 (9) (1998) 1241–1253.

[49] C. Colombo, A. Del Bimbo, Color-induced image representation and retrieval, Pattern Recognition 32 (10) (1999) 1685–1696.

[50] A. Soffer, Image categorization using N × M-grams, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases V, Vol. SPIE 3022, 1997, pp. 121–132.

[51] H.C. Lin, L.L. Wang, S.N. Yang, Color image retrieval based on hidden markov-models, IEEE Trans. Image Process. 6 (2) (1997) 332–339.

[52] S. Müller, S. Eickeler, G. Rigoll, Multimedia database retrieval using hand-drawn sketches, International Conference on Document Analysis and Recognition, 1999, pp. 289–292.

[53] G. Pass, R. Zabih, Comparing images using joint histograms, ACM J. Multimedia Systems 7 (2) (1999) 119–128.

[54] J. Huang, S. Ravi Kumar, M. Mitra, W.-J. Zhu, R. Zabih, Spatial color indexing and applications, Int. J. Comput. Vision 35 (3) (1999) 245–268.

[55] Y. Tao, W.I. Grosky, Spatial color indexing: a novel approach for content-based image retrieval, IEEE International Conference on Multimedia Computing Systems, Vol. 1, 1999, pp. 530–535.

[56] L. Cinque, S. Levialdi, K.A. Olsen, A. Pellicanò, Color-based image retrieval using spatial-chromatic histograms, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 969–973.

[57] Y. Chahir, L. Chen, Efficient content-based image retrieval based on color homogenous objects segmentation and their spatial relationship characterization, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 969–973.

[58] J.R. Smith, C.-S. Li, Image classification and querying using composite region templates, Comput. Vision Image Understanding 75 (1–2) (1999) 165–174.

[59] C. Brambilla, A. Della Ventura, I. Gagliardi, R. Schetini, Multiresolution wavelet transform and supervised learning for content-based image retrieval, IEEE International Conference on Multimedia Computing Systems, Vol. 1, 1999, pp. 183–188.

[60] A. Vailaya, A. Jain, H.-J. Zhang, On image classification: city images vs. landscapes, J. Visual Commun. Image Representation 31 (12) (1998) 1921–1935.

[61] D. Androutsos, K.N. Plataniotis, A.N. Venetsanopoulos, A novel vector-based approach to color image retrieval using a vector angular-based distance metric, Comput. Vision Image Understanding 75 (1–2) (1999) 46–58.

[62] G.P. Babu, B.M. Mehtre, M.S. Kankanhalli, Color indexing for efficient image retrieval, Multimedia Tools Appl. 1 (4) (1995) 327–348.

[63] M.S. Kankanhalli, B.M. Mehtre, J.K. Wu, Cluster based color matching for image retrieval, Pattern Recognition 29 (4) (1996) 701–708.

[64] B.M. Mehtre, M.S. Kankanhalli, A.D. Narasimhalu, G.C. Man, Color matching for image retrieval, Pattern Recognition Lett. 16 (3) (1995) 325–331.

[65] M.S. Kankanhalli, B.M. Mehtre, H.Y. Huang, Color and spatial feature for content-based image retrieval, Pattern Recognition Lett. 20 (1) (1999) 109–118.

[66] S. Krishnamachari, M. Abdel-Mottaleb, A scalable algorithm for image retrieval by color, Proceedings of

the International Conference on Image Processing, 1998, pp. 119–122.

[67] K. Hirata, S. Mukherjea, W.-S. Li, Y. Hara, Integrating image matching and classification for multimedia retrieval, IEEE International Conference on Multimedia Computing Systems, Vol. 1, 1999, pp. 257–263.

[68] P. Scheunders, A comparison of clustering algorithms applied to color image quantization, Pattern Recognition Lett. 18 (11–13) (1997) 1379–1384.

[69] J. Wang, W.-J., Yang, R. Acharya, Color clustering techniques for color-content-based image retrieval from image databases, IEEE International Conference on Multimedia Computing Systems, 1997, pp. 442–449.

[70] J. Wang, W.-J. Yang, R. Acharya, Efficient access to and retrieval from a shape image database, Workshop on Content Based Access to Image and Video Libraries, 1998, pp. 63–67.

[71] U. Gargi, R. Kasturi, Image database querying using a multi-scale localized color-representation, Workshop on Content Based Access to Image and Video Libraries, 1999, pp. 28–32.

[72] A. Del Bimbo, P. Pala, Effective image retrieval using deformable templates, Proceedings of the International Conference on Pattern Recognition, 1996, pp. 120–124.

[73] S. Scarloff, Deformable prototypes for encoding shape categories in image databases, Pattern Recognition 30 (4) (1997) 627–641.

[74] P. Pala, S. Santini, Image retrieval by shape and texture, Pattern Recognition 32 (3) (1999) 517–527.

[75] M. Adoram, M.S. Lew, IRUS: image retrieval using shape, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 597–602.

[76] B. Günsel, A. Murat Tekalp, Shape similarity matching for query-by-example, Pattern Recognition 31 (7) (1998) 931–944.

[77] Z. Lei, T. Tasdizen, D. Cooper, Object signature curve and invariant shape patches for geometric indexing into pictorial databases, Proceedings of IS&T/SPIE Conference on Multimedia Storage and Archiving Systems II, Vol. SPIE 3229, 1997, pp. 232–243.

[78] R. Alferez, Y.-F., Wang, Image indexing and retrieval using image-derived, geometrically and illumination invariant features, IEEE International Conference on Multimedia Computing Systems, Vol. 1, 1999, pp. 177–182.

[79] E.G.M. Petrakis, E. Milios, Efficient retrieval by shape content, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 616–621.

[80] F. Mokhtarian, S. Abbasi, Retrieval of similar shapes under affine transform, Third International Conference on Visual Information and Information Systems (VISUAL'99), Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 566–574.

[81] D. Sharvit, J. Chan, H. Tek, B.B. Kimia, Symmetry-based indexing of image databases, J. Visual Commun. Image Representation 9 (4) (1998) 366–380.

[82] Y. Rui, T.S. Huang, S. Mehrotra, M. Ortega, Automatic matching tool selection using relevance feedback in MARS, Second International Conference on Visual Information Systems (VISUAL'97), 1997, pp. 109–116.

[83] A.K. Jain, A. Vailaya, Shape-based retrieval: a case study with trademark image databases, Pattern Recognition 31 (9) (1998) 1369–1390.

[84] M. Kliot, E. Rivlin, Invariant-based shape retrieval in pictorial databases, Comput. Vision Image Understanding 71 (2) (1998) 182–197.

[85] S. Derrode, M. Daoudi, F. Ghorbel, Invariant content-based image retrieval using a complete set of Fourier–Mellin descriptors, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 877–881.

[86] J.R. Smith, S.-F. Chang, Quad-tree segmentation for texture-based image query, ACM International Conference on Multimedia 1994, pp. 279–286.

[87] M. Borchani, G. Stammon, Use of texture features for image classification and retrieval, Proceedings of IS&T/SPIE Conference on Multimedia Storage and Archiving Systems II, Vol. SPIE 3229, 1997, pp. 401–406.

[88] B.S. Manjunath, W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 837–842.

[89] J. Puzicha, T. Hofmann, J.M. Buhmann, Non-parametric similarity measures for unsupervised texture segmentation and image retrieval, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1997, pp. 267–272.

[90] M.K. Mandal, T. Aboulnasr, S. Panchanathan, Fast wavelet histogram techniques for image indexing, Comput. Vision Image Understanding 75 (1–2) (1999) 99–110.

[91] R. Milanese, M. Cherbuliez, A rotation, translation, and scale invariant approach to content-based image retrieval, J. Visual Commun. Image Representation 10 (2) (1999) 186–196.

[92] N. Strobel, C.S. Li, V. Castelli, MMAP: modified maximum a posteriori algorithm for image segmentation in large image/video databases, Proceedings of IEEE International Conference on Image Processing, 1997, pp. 196–199.

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

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

[95] Y. Zhong, A. Jain, Object localization using color, texture and shape, Pattern Recognition 33 (4) (2000) 671–684.

[96] A. Mojsilovic, J. Kovacevic, J. Hu, R.J. Safranek, S.K. Ganpathy, Matching and retrieval based on vocabulary and grammar of color patterns, IEEE Int. Conf. Multimedia Computing Systems, Vol. 1, 1999, pp. 189–194.

[97] S. Scarloff, M. La Cascia, S. Sethi, L. Taycher, Unifying textual and visual cues for content-based image retrieval on the World Wide Web, Comput. Vision Image Understanding 75 (1–2) (1999) 86–98.

[98] A.P. Berman, L.G. Shapiro, A flexible image database system for content-based retrieval, Comput. Vision Image Understanding 75 (1–2) (1999) 175–195.

[99] M.M. Yeung, B.-L. Yeo, Video visualization for compact presentation and fast browsing of pictorial content, IEEE Trans. Circuits Systems Video Technol. 7 (5) (1997) 771–785.

[100] Y. Gong, An accurate and robust method for detecting video shot boundaries, IEEE International Conference on Multimedia Computing Systems, Vol. 1, 1999, pp. 850–854.

[101] R. Zabih, R. Miller, K. Mai, A feature-based algorithm for detecting and classifying scene breaks, ACM International Conference on Multimedia, 1995, pp. 189–200.

[102] A. Hampapur, R. Jain, T. Weymouth, Production model based digital video segmentation, J. Multimedia Tools Appl. 1 (1) (1995) 9–46.

[103] W.M. Kim, S.M.-H. Song, H. Kim, C. Song, B.W. Kwoi, S.G. Kim, A morphological approach to scene change detection and digital video storage and retrieval, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases VII, Vol. SPIE 3656, 1999, pp. 733–743.

[104] F. Arman, A. Hsu, M.-Y. Chiu, Feature management for large video databases, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases I, Vol. SPIE 1908, 1993, pp. 2–12.

[105] H.J. Zhang et al., Video parsing using compressed data, SPIE Symposium on Electronic Imaging Science and Technology: Image and Video Processing II, 1994, pp. 142–149.

[106] N.V. Patel, I.K. Sethi, Video shot detection and characterization for video databases, (Special issue on multimedia), Pattern Recognition 30 (1997) 583–592.

[107] H.C. Liu, G.L. Zick, Scene decomposition of MPEG compressed video, SPIE/IS&T Symposium on Electronic Imaging Science and Technology: Digital Video Compression: Algorithms and Technologies, Vol. 2419, 1995, pp. 26–37.

[108] B. Yeo, B. Liu, Rapid scene analysis on compressed video, IEEE Trans. Circuits Systems Video Technol. 5 (6) (1995) 533–544.

[109] K. Shen, E.J. Delp, A fast algorithm for video parsing using MPEG compressed sequences, IEEE International Conference on Image Processing, October 1995, pp. 252–255.

[110] J. Meng, Y. Juan, S.F. Chang, Scene change detection in a MPEG compressed video sequence, SPIE/IS&T Symposium on Electronic Imaging Science and Technology: Digital video Compression: Algorithms and Technologies, Vol. 2419, 1995.

[111] A. Hanjalic, H. Zhang, Optimal shot boundary detection based on robust statistical methods, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 710–714.

[112] K. Tse, J. Wei, S. Panchanathan, A scene change detection algorithm for MPEG compressed video sequences, Canadian Conference on Electrical and Computer Engineering (CCECE'95), Vol. 2, 1995, pp. 827–830.

[113] V. Kobla, D.S. Doermann, K.-I. Lin, C. Falutsos, Compressed domain video indexing techniques using DCT and motion vector information in MPEG video, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases V, Vol. SPIE 3022, 1997, pp. 200–211.

[114] Y. Yusoff, J. Kittler, W. Christmas, Combining multiple experts for classifying shot changes in video sequences, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 700–704.

[115] H.H. Yu, W. Wolf, A hierarchical multiresolution video shot transition detection scheme, Comput. Vision Image Understanding 75 (1/2) (1999) 196–213.

[116] M.K. Mandal, S. Panchanathan, Video segmentation in the wavelet domain, Proceedings of IS&T/SPIE Conference on Multimedia Storage and Archiving Systems III, Vol. SPIE 3527, 1998, pp. 262–270.

[117] R.M. Ford, A quantitative comparison of shot boundary detection metrics, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases VII, Vol. SPIE 3656, 1999, pp. 666–676.

[118] R. Lienhart, Comparison of autmatic shot boundary detection, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases VII, Vol. SPIE 3656, 1999, pp. 290–301.

[119] J.S. Boreczky, L.A. Rowe, Comparison of video shot boundary detection techniques, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases IV, Vol. SPIE 2670, 1996, pp. 170–179.

[120] R. Fablet, P. Bouthemy, Motion-based feature extraction and ascent hierarchical classification for video indexing and retrieval, Third International Conference on Visual Information and Information Systems (VISUAL'99), Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 221–228.

[121] A. Akutsu et al., Video indexing using motion vectors, Proceedings of SPIE Visual Communications and Image Processing, Vol. 1818, 1992, pp. 1522–1530.

[122] S. Fischer, I. Rimac, R. Steinmetz, Automatic recognition of camera zooms, Third International Conference on Visual Information and Information Systems (VISUAL'99), Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 253–260.

[123] M. Cherfaoui, C. Bertin, Temporal segmentation of videos: a new approach, SPIE/IS&T Symposium on Electronic Imaging Science and Technology: Digital Video Compression: Algorithms and Technologies, Vol. 2419, 1995.

[124] B. Shahraray, Scene change detection and content-based sampling of video sequences, SPIE/IS&T Symposium on Electronic Imaging Science and Technology: Digital Video Compression: Algorithms and Technologies, Vol. 2419, 1995, pp. 2–13.

[125] E. Ardizzone, M. La Cascia, A. Avanzato, A. Bruna, Video indexing using MPEG motion compensation vectors, IEEE International Conference on Multimedia Computing Systems, Vol. 2, 1999, pp. 725–729.

[126] O. Fatemi, S. Zhang, S. Panchanathan, Optical flow based model for scene cut detection, Canadian Conference on Electrical and Computer Engineering, Vol. 1, 1996, pp. 470–473.

[127] M.R. Naphade, R. Mehrotra, A.M. Ferman, J. Warnick, T.S. Huang, A.M. Tekalp, A high-performance shot boundary detection algorithm using multiple cues,

Proceedings of IEEE International Conference on Image Processing, 1998, pp. 884–887.

[128] B. Günsel, A. Murat Tekalp, Content-based video abstraction, Proceedings of IEEE International Conference on Image Processing, 1998, pp. 128–132.

[129] Y. Deng, B.S. Manjunath, NeTra-V: toward an object-based video representation, IEEE Trans. Circuits Systems Video Technol. 8 (5) (1998) 616–627.

[130] F. Coudert, J. Benois-Pineau, P.-Y. Le Lann, D. Barba, Binkey: a system for video content analysis "on the fly", IEEE International Conference on Multimedia Computing Systems, Vol. 1, 1999, pp. 679–684.

[131] S. Dağtaş, W. Al-Khatib, A. Ghafoor, R.L. Kashyap, Models for motion-based video indexing and retrieval, IEEE Trans. Image Process. 9 (1) (2000) 88–101.

[132] M. Wu, W. Wolf, B. Liu, An algorithm for wipe detection, Proceedings of IEEE International Conference on Image Processing, 1998, pp. 893–897.

[133] J.M. Corridoni, A. Del Bimbo, Structured representation and automatic indexing of movie information content, Pattern Recognition 31 (2) (1998) 2027–2045.

[134] V. Kobla, D. DeMenthon, D. Doermann, Special effect edit detection using video trails: a comparison with existing techniques, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases VII, Vol. SPIE 3656, 1999, pp. 302–313.

[135] A. Hanjalic, H. Zhang, An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis, IEEE Trans. Circuits Systems Video Technol. 9 (8) (1999) 1280–1289.

[136] M. Yeung, B.-L. Yeo, B. Liu, Segmentation of video by clustering and graph analysis, J. Visual Commun. Image Representation 71 (1) (1998) 94–109.

[137] Y. Rui, T.S. Huang, S. Mehrotra, Eploring video structure beyond shots, IEEE International Conference on Multimedia Computing Systems, 1998, pp. 237–240.

[138] R. Lienhart, S. Pfeiffer, W. Effelsberg, Video abstracting, Commun. ACM 40 (12) (1997) 54–62.

[139] C. Colombo, A. Del Bimbo, P. Pala, Retrieval of commercials by video semantics, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1998, pp. 572–577.

[140] R. Lienhart, C. Kuhmünch, W. Effelsberg, On the detecttion and recognition of television commercials, IEEE International Conference on Multimedia Computing Systems, 1997, pp. 509–516.

[141] J.M. Sánchez, X. Binefa, J. Vitrià, P. Radeva, Local color analysis for scene break detection applied to TV commercials recognition, Third International Conference on Visual Information and Information Systems (VISUAL'99), Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 237–244.

[142] Y. Zhuang, Y. Rui, T.S. Huang, S. Mehrotra, Adaptive key frame extraction using unsupervised clustering, Proceedings of IEEE International Conference on Image Processing, 1998, pp. 866–870.

[143] A. Hanjalic, R.L. Lagendijk, J. Biemond, Automatically segmenting movies into logical story units, Third International Conference on Visual Information and Information Systems (VISUAL'99), Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 229–236.

[144] J.-Y. Chen, C. Taskiran, A. Albiol, C.A. Bouman, E.J. Delp, ViBE: a video indexing and browsing environment, Proceedings of IS&T/SPIE Conference on Multimedia Storage and Archiving Systems IV, Vol. SPIE 3846, 1999, pp. 148–164.

[145] P. Bouthemy, C. Garcia, R. Ronfard, G. Tziritas, E. Veneau, D. Zugaj, Scene segmentation and image feature extraction for video indexing and retrieval, Third International Conference on Visual Information and Information Systems (VISUAL'99), Lecture Notes in Computer Science, Vol. 1614, 1999, pp. 245–252.

[146] H.S. Chang, S. Sull, S.U. Lee, Efficient video indexing scheme for content-based retrieval, IEEE Trans. Circuits Systems Video Technol. 9 (8) (1999) 1269–1279.

[147] E. Sahouria, A. Zakhor, Content analysis of video using principal components, IEEE Trans. Circuits Systems Video Technol. 9 (8) (1999) 1290–1298.

[148] N. Vasconcelos, A. Lippman, Statistical models of video structure for content analysis and characterization, IEEE Trans. Image Process. 9 (1) (2000) 3–19.

[149] A. Jain, A. Vailaya, W. Xiong, Query by video clip, Proceedings of International Conference on Pattern Recognition, 1998, pp. 909–911.

[150] Y.S. Avrithis, A.D. Doulamis, N.D. Doulamis, S.D. Kollias, A stochastic framework for optimal key frame extraction from MPEG video databases, Comput. Vision Image Understanding 75 (1/2) (1999) 3–24.

[151] U. Gargi, R. Kasturi, S. Antani, Performance characterization and comparison of video indexing algorithms, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1998, pp. 559–565.

[152] H.C. Liu, G.L. Zick, Automated determination of scene changes in MPEG compressed video, ISCAS—IEEE International Symposium on Circuits and Systems, 1995.

[153] B.-L. Yeo, B. Liu, A unified approach to temporal segmentation of motion JPEG and MPEG compressed video, IEEE International Conference on Multimedia Computing Systems, 1995, pp. 81–89.

[154] I.K. Sethi, N.V. Patel, A statistical approach to scene change detection, Proceedings of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases III, Vol. SPIE 2420, 1995, pp. 329–338.

[155] R. Kasturi, R.C. Jain, Computer Vision: Principles, IEEE Computer Society Press, Silver Spring, MD, 1991, pp. 469–480.

About the Author—SAMEER ANTANI has been a Ph.D. candidate with the Dept. of Computer Science and Engineering at The Pennsylvania State University since 1995. He received his Master of Engineering degree in 1998. He obtained his B.E. (Computer) from the University of Pune, India, in 1994. His research has been in the area of Automated Content Analysis of Visual Information. His interests are in Computer Vision, Content Analysis of Visual and Multimedia Information, Document Image Analysis. Human Computer Interaction, Image Processing and Computer Graphics. Sameer has been a student member of the IEEE for the past 10 years.

About the Author—RANGACHAR KASTURI is a Professor of Computer Science and Engineering and Electrical Engineering at the Pennsylvania State University. He received his degrees in electrical engineering from Texas Tech University (Ph.D., 1982 and M.S., 1980) and from Bangalore University, India (B.E. 1968). He worked as a research and development engineer in India during 1968–78. He served as a Visiting Scholar at the University of Michigan during 1989–90 and as a Fulbright Lecturer at the Indian Institute of Science, Bangalore during 1999.

Dr. Kasturi is an author of the textbook, Machine Vision, McGraw-Hill, 1995. He has also authored/edited the books Computer Vision: Principles and Applications, Document Image Analysis. Image Analysis Applications, and Graphics Recognition: Methods and Applications. He has directed many projects in document image analysis, image sequence analysis, and video indexing. His earlier work in image restoration resulted in the design of many optimal filters for recovering images corrupted by signal-dependent film-grain noise.

Dr. Kasturi is serving as the Vice President, Publications of the IEEE Computer Society. He is also serving as the First Vice President of the International Association for Pattern Recognition (IAPR). He was the Editor-in-Chief of the IEEE Transactions of Pattern Analysis and Machine Intelligence during 1995–98 and the Editor-in-chief of Machine Vision and Applications journal during 1993–94. He served in the IEEE Computer Society's Distinguished Visitor Program during 1987–90.

Dr. Kasturi is a General Chair of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2001 and the General Chair: Technical Program for the International Conference on Pattern Recognition (ICPR), 2002. He was a General Chair for the International Conference on Document Analysis and Recognition (ICDAR), 1999.

Dr. Kasturi is a Fellow of the IEEE and a Fellow of IAPR. He received the Penn State Engineering Society Outstanding Research Award in 1996 and was elected to the Texas Tech Electrical Engineering Academy in 1997. He is a Golden Core Member of the IEEE Computer Society and has received its Meritorious Service Award and Certificates of Appreciation.

About the Author—RAMESH JAIN, Ph.D., is both an entrepreneur and a research scientist. He was the founding chairman of Virage (NASDAQ: VRGE), a San Mateo-based company developing systems for media management solutions and visual information management.

Dr. Jain was also the chairman of Imageware Inc., an Ann Arbor, Michigan-based company dedicated to linking virtual and physical worlds by providing solutions for surface modeling, reverse engineering, rapid prototyping and inspection.

Dr. Jain is a Professor Emeritus of Electrical and Computer Engineering, and Computer Science and Engineering at University of California, San Diego. He is a world-renowned pioneer in multimedia information systems, image databases, machine vision, and intelligent systems. While professor of engineering and computer science at the University of Michigan, Ann Arbor and the University of California, San Diego, he founded and directed artificial intelligence and multimedia information systems labs. He is also the founding editor of IEEE MultiMedia magazine. He has also been affiliated with Stanford University. IBM Almaden Research Labs, General Motors Research Labs, Wayne State University, University of Texas at Austin, University of Hamburg, Germany, and Indian Institute of Technology, Kharagpur, India.

Ramesh is a Fellow of IEEE. AAAI, and Society of Photo-Optical Instrumentation Engineers, and member of ACM, Pattern Recognition Society, Cognitive Science Society, Optical Society of America and Society of Manufacturing Engineers. He has been involved in organization of several professional conferences and workshops, and served on editorial boards of many journals. He was the founding Editor-in-Chief of IEEE Multimedia, and is on the editorial boards of Machine Vision and Applications, Pattern Recognition, and Image and Vision Computing. He received his Ph.D. from IIT, Kharagpur in 1975 and his B.E. from Nagpur University in 1969.