IJCAT - International Journal of Computing and Technology, Volume 1, Issue 10, November 2014
ISSN : 2348 - 6090
www.IJCAT.org

Object Recognition for Video Retrieval

[1] Ch.S.S. Sravya, [2] D. Mounisha, [3] B.Giridhar, [4] Bhavani Shankar Panda

[1, 2] B.Tech Student

[3, 4] Asst.prof in CSE dept

Abstract - Video indexing and retrieval have a wide spectrum of promising applications, motivating the interest of researchers Worldwide. This paper offers a tutorial and an overview of the landscape of general strategies in visual content-based video indexing and retrieval, focusing on methods for video structure analysis, including shot boundary detection, key frame extraction and scene segmentation, extraction of features including static key frame features, object features and motion features, video data mining, video annotation, video retrieval including query interfaces, similarity measure and relevance feedback, and video browsing. Finally, we analyze future research directions.

Keywords **- Feature extraction, video annotation, video browsing, video retrieval, video structure analysis.**

1. Introduction

MULTIMEDIA information indexing and retrieval are required to describe, store, and organize multimedia information and to assist people in finding multimedia resources conveniently and quickly. Dynamic video is an important form of multimedia information. Videos have the following characteristics:

1) much richer content than individual images; 2) huge amount of rawdata; and 3) very little prior structure. These characteristics make the indexing and retrieval of videos quite difficult.

In the past, video databases have been relatively small, and indexing and retrieval have been based on keywords annotated manually. More recently, these databases have become much larger and content-based indexing and retrieval are required, based on the automatic analysis of videos with the minimum of human participation.

Content-based video indexing and retrieval have a wide range of applications such as quick browsing of video folders, analysis of visual electronic commerce (such as analysis of interest trends of users' selections and orderings, analysis of correlations between advertisements and their effects), remote instruction, digital museums, news event analysis [96], intelligent management of web videos (useful video search and harmful video tracing), and video surveillance.

It is the broad range of applications that motivates the interests of researchers worldwide. The following two examples of research activity are particularly noteworthy. 1) Since 2001, the National Institute of Standards and Technology has been sponsoring the annual Text Retrieval Conference (TREC) Video Retrieval Evaluation (TRECVid) to promote progress in video analysis and retrieval. Since 2003, TRECVid has been independent of TREC. TRECVid provides a large-scale test collection of videos, and dozens of participants apply their content-based video retrieval algorithms to the collection . 2) The goal of video standards is to ensure compatibility between description interfaces for video contents in order to facilitate the development of fast and accurate video retrieval algorithms.

The main standards for videos are the moving picture experts group (MPEG) and the TV-Anytime Standard [254]. There exist many investigations that adopt the MPEG-7 to extract features to classify video contents or to describe video objects in the compressed domain.

A video may have an auditory channel as well as a visual channel. The available information from videos includes the following [66], [67]: 1) video metadata, which are tagged texts embedded in videos, usually including title, summary, date, actors, producer, broadcast duration, file size, video format, copyright, etc.; 2) audio information from the auditory channel; 3) transcripts: Speech transcripts can be obtained by speech recognition and caption texts can be read using optical character recognition techniques; 4) visual information contained in the images themselves from the visual channel. If the video is included in aweb page, there are usually web page texts associated with the video. In this paper, we focus on the visual contents of videos and give a survey on visual content-based video indexing and retrieval.

The importance and popularity of video indexing and retrieval have led to several survey papers, which are listed in Table I, together with the publication years and topics.

In general, each paper covers only a subset of the topics in video indexing and retrieval. For example, Smeaton *et al*. [263] give a good review of video shot boundary detection during seven years of the TRECVid activity. Snoek and Worring [262] present a detailed reviewof concept-based video retrieval. They emphasize semantic concept detection, video search using semantic concepts, and the evaluation of algorithms using the TRECVid databases. Ren *et al*. [278] review the state of the art of spatiotemporal semantic information-based video retrieval. Schoeffmann *et al*. [261] give a good review of interfaces and applications of video browsing systems.

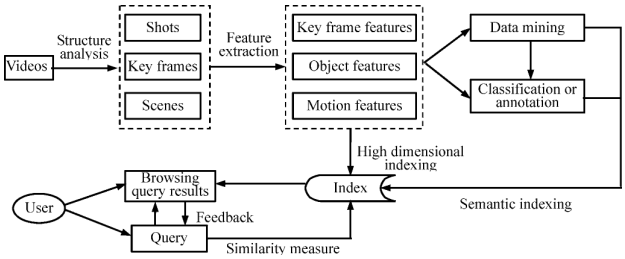

Fig. 1. Generic framework for visual content-based video indexing and retrieval.

Unlike previous reviews, we give a more general overview on the overall process of a video indexing and retrieval framework which is outlined in Fig. 1. The framework includes the following: 1) structure analysis: to detect shot boundaries, extract key frames, and segment scenes; 2) feature extraction from segmented video units (shots or scenes): These features include static features in key frames, object features, motion features, etc.; 3) video data mining using the extracted features; 4) video annotation: using extracted features and mined knowledge to build a semantic video index. The semantic index together with the high-dimensional index of video feature vectors constitutes the total index for video sequences that are stored in the database; 5) query: the video database is searched for the desired videos using the index and the video similarity measures; 6) video browsing and feedback: The videos found in response to a query are returned to the user to browse in the form of a video summary, and subsequent search results are optimized through relevance feedback. In this paper, we review recent developments and analyze future open directions in visual content-based video indexing and retrieval. The main contributions of this survey are as follows.

1) Video indexing and retrieval components are discussed in a clearly organized hierarchical manner, and interlinks between these components are shown.

2) To examine the state of the art, each task involved in visual content-based video indexing and retrieval is divided into subprocesses and various categories of approaches to the subprocesses are discussed. The merits and limitations of the different approaches are summarized. For the tasks for which there exist surveys, we focus on reviewing recent papers as a supplement to the previous surveys. For the tasks that have not yet been specially surveyed, detailed reviews are given.

3) We discuss in detail future directions in visual contentbased video indexing and retrieval.

The aforesaid contributions clearly distinguish our survey from the existing surveys on video indexing and retrieval. To our knowledge, our survey is the broadest.

The remainder of this paper is organized as follows: Section II briefly reviews the work related to video structure analysis. Section III addresses feature extraction. Section IV discusses video data mining, classification, and annotation. Section V describes the approaches for video query and retrieval. Section VI presents video summarization for browsing. Section VII analyzes possible directions for future research. Section VIII summarizes this paper.

2. Video Structure Analysis

Generally, videos are structured according to a descending hierarchy of video clips, scenes, shots, and frames.Video structure analysis aims at segmenting a video into a number of structural elements that have semantic contents, including shot boundary detection, key frame extraction, and scene segmentation.

A. Shot Boundary Detection

A shot is a consecutive sequence of frames captured by a camera action that takes place between start and stop operations, which mark the shot boundaries [10]. There are strong content correlations between frames in a shot. Therefore, shots are considered to be the fundamental units to organize the contents of video sequences and the primitives for higher level semantic annotation and retrieval tasks. Generally, shot boundaries are classified as cut in which the transition between successive shots is abrupt and gradual transitions which include dissolve, fade in, fade out, wipe, etc., stretching over a number of frames. Cut detection is easier than gradual transition detection.

The research on shot boundary detection has a long history, and there exist specific surveys on video shot boundary detection. For completeness, we only briefly introduce the basic categories of methods for shot boundary detection and their merits and limitations, and review some recent papers as a supplement to [16] and [263]. Methods for shot boundary detection usually first extract visual features from each frame, then measure similarities between frames using the extracted features,

and, finally, detect shot boundaries between frames that are dissimilar. In the following, we discuss the main three steps in shot boundary detection: feature extraction, similarity measurement [113], and detection. The features used for shot boundary detection include color histogram [87] or block color histogram, edge change ratio, motion vectors [85], [163], together with more novel features such as scale invariant feature transform [83], corner points [82], information saliency map [77], etc. Color histograms are robust to small camera motion, but they are not able to differentiate the shots within the same scene, and they are sensitive to large camera motions. Edge features are more invariant to illumination changes and motion than color histograms, and motion features can effectively handle the influence of object and camera motion. However, edge features and motion features as well as more complicated features cannot in general outperform the simple color histograms [16].

To measure similarity between frames using the extracted features is the second step required for shot boundary detection. Current similarity metrics for extracted feature vectors include the 1-norm cosine dissimilarity, the Euclidean distance, the histogram intersection, and the chi-squared similarity, as well as some novel similarity measures such as the earth mover's distance [87] and mutual information. The similarity measures include pair-wise similarity measures that measure the similarities between consecutive frames and window similarity measures that measure similarities between frames within a window. Windowbased similarity measures incorporate contextual information to reduce the influence of local noises or disturbances, but they need more computation than the pair-wise similarity measures. Using the measured similarities between frames, shot boundaries can be detected. Current shot boundary detection approaches can be classified into threshold-based and statistical learning-based.

1) Threshold-Based Approach: The threshold-based approach

detects shot boundaries by comparing the measured pair-wise similarities between frames with a predefined threshold [47], : When a similarity is less than the threshold, a boundary is detected. The threshold can be global, adaptive, or global and adaptive combined. 1) The global threshold-based algorithms use the same threshold, which is generally set empirically, over the whole video, as in [180]. The major limitation of the global threshold-based algorithms is that local content variations are not effectively incorporated into the estimation of the global threshold, therefore influencing the boundary detection accuracy. 2) The adaptive threshold-based algorithms compute the threshold locally within a sliding window. Detection performance is often improved when an adaptive threshold is used instead of a global threshold [65].

However, estimation of the adaptive threshold is more difficult than estimation of the global threshold and users are required to be more familiar with characteristics of videos in order to choose parameters such as the size of the sliding window. 3) Global and adaptive combined algorithms adjust local thresholds, taking into account the values of the global thresholds. Quenot *et al.* [264] define the thresholds for cut transition detection, dissolve transition detection, and flash detection as the functions of two global thresholds that are obtained from a tradeoff between recall and precision. Although this algorithm only needs to tune two global thresholds, the values of the functions are changed locally. The limitation of this algorithm is that the functional relations between the two global thresholds and the locally adaptive thresholds are not easy to determine.

2) Statistical Learning-Based Approach*:* The statistical learning-based approach regards shot boundary detection as a classification task in which frames are classified as shot change or no shot change depending on the features that they contain.

Supervised learning and unsupervised learning are both used.

a) Supervised learning-based classifiers: The most commonly used supervised classifiers for shot boundary detection are the support vector machine (SVM) and Adaboost.

1) *SVM* [11], [21]: Chavez *et al.* [84] use the SVM as a two-class classifier to separate cuts from noncuts. A kernel function is used to map the features into a highdimensional space in order to overcome the influence of changes in illumination and fast movement of objects.

Zhao *et al.* [61] exploit two SVM classifiers, in a sliding window, to detect cuts and gradual transitions, respectively. Ling *et al.* [58] first extract several features from each frame, and then use the SVM to classify the frames using these features into three categories: cut, gradual transition, and others. Yuan *et al.* [16] and Liu *et al.* [72] combine the threshold-based method with an SVM-based classifier.

First, the candidate boundaries are selected using the threshold-based method, and then the SVM classifier is used to verify the boundaries. The SVM-based algorithms are widely used for shot boundary detection [265] because of their following merits.

a) They can fully utilize the training information and maintain good generalization.

b) They can deal efficiently with a large number of features by the use of kernel functions.

c) Many good SVM codes are readily available.

2) Adaboost: Herout *et al.* [63] make cut detection a pattern recognition task to which the Adaboost algorithm is applied. Zhao and Cai [85] apply the Adaboost algorithm to shot boundary detection in the compressed domain. The color and motion features are roughly classified first using a fuzzy classifier, and then each frame is classified as a cut, gradual, or no change frame using the Adaboost classifier. The main merit of the Adaboost boundary classifiers is that a large number of features can be handled:

These classifiers select a part of features for boundary classification.

3) Others: Other supervised learning algorithms have been employed for shot boundary detection. For instance, Cooper *et al.* [191] use the binary *k* nearest-neighbor (kNN) classifier, where the similarities between frames within the particular temporal interval are used as its input. Boreczky and Wilcox [121] apply hidden Markov (HMM) models with separate states to model shot cuts, fades, dissolves, pans, and zooms.

The merits of the aforementioned supervised-learning approaches are that there is no need to set the thresholds used in the threshold-based approaches, and different types of features can be combined to improve the detection accuracy. The limitation is their heavy reliance on a well-chosen training set containing both positive and negative examples.

b) Unsupervised learning-based algorithms: The unsupervised learning-based shot boundary detection algorithms are classified into frame similarity-based and frame-based. The frame similarity-based algorithms cluster the measurements of similarity between pairs of frames into two clusters: the cluster with lower values of the similarities corresponds to shot boundaries and the cluster with higher values of the similarities corresponds to nonboundaries. Clustering algorithms such as K-means and fuzzy K-means [64] have been used. The framebased algorithms treat each shot as a cluster of frames that have similar visual content. Chang *et al.* [83] use clustering ensembles to group different frames into their corresponding shots. Lu *et al.* [12] use K-means clustering, and Damnjanovic *et al.* [57] use spectral clustering to cluster frames to detect the different shots. The merit of clustering-based approaches is that the training dataset is not needed. Their limitations are that temporal sequence progression information is not preserved, and they are inefficient in recognizing the different types of gradual transition.

Shot boundary detection approaches can be classified into uncompressed domain-based and compressed domain-based. To avoid time-consuming video decompression, the features available in the compressed domain such as discrete cosine transform coefficients,DCimage andMBtypes, and motion vectors can be directly employed for shot boundary detection [40], [60], [85].

However, the compressed domain-based approach is highly dependent on the compression standards, and it is less accurate than the uncompressed domain-based approach.

Recently, the detection of gradual transitions has received more attention. Ngo [41] detects dissolves based on multiresolution analysis. Yoo *et al.* [131] detect gradual transitions according to the variance distribution curve of edge information in frame sequences.

B. Key Frame Extraction

There are great redundancies among the frames in the same shot; therefore, certain frames that best reflect the shot contents are selected as key frames [15], [39], [170], [193] to succinctly represent the shot. The extracted key frames should contain as much salient content of the shot as possible and avoid as much redundancy as possible. The features used for key frame extraction include colors (particularly the color histogram), edges, shapes, optical flow, MPEG-7 motion descriptors such as temporal motion intensity and spatial distribution of motion activity [206], MPEG discrete cosine coefficient and motion vectors [202], camera activity, and features derived from image variations caused by camera motion [161], [208]. Referring to [39], current approaches to extract key frames are classified into six categories: sequential comparison-based, global comparison-based, reference frame-based, clusteringbased, curve simplification-based, and object/event-based.

1) Sequential Comparison Between Frames*:* In these algorithms, frames subsequent to a previously extracted key frame are sequentially compared with the key frame until a frame which is very different from the key frame is obtained. This frame is selected as the next key frame. For instance, Zhang *et al.* [209] used the color histogram difference between the current frame and the previous key frame to extract key frames.

Zhang *et al.* [210] use the accumulated energy function computed from image-block displacements across two successive frames to measure the distance between frames to extract key frames. The merits of the sequential comparison-based algorithms include their simplicity, intuitiveness, low computational complexity, and adaptation of the number of key frames to the length of the shot. The limitations of these algorithms include the following. 1) The key frames represent local properties of the shot rather than the global properties. b) The irregular distribution and uncontrolled number of key frames make these algorithms unsuitable for applications that need an even distribution or a fixed number of key frames. c) Redundancy can occur when there are contents appearing repeatedly in the same shot.

2) Global Comparison Between Frames: The algorithms based on global differences between frames in a shot distribute key frames by minimizing a predefined objective function that depends on the application. In general, the objective function has one of the following four forms [39].

1) Even temporal variance: These algorithms select key frames in a shot such that the shot segments, each of which is represented by a key frame, have equal temporal variance. The objective function can be chosen as the sum of differences between temporal variances of all the segments. The temporal variance in a segment can be approximated by the cumulative change of contents across consecutive frames in the segment [208] or by the difference between the first and last frames in the segment. For instance, Divakaran *et al.* [211] obtain key frames by dividing the shot into segments with equal cumulative motion activity using the MPEG-7 motion activity descriptor, and then, the frame located at the halfway point of each segment is selected as a key frame.

2) Maximum coverage: These algorithms extract key frames by maximizing their representation coverage, which is the number of frames that the key frames can represent [39].

If the number of key frames is not fixed, then these algorithms minimize the number of key frames subject to a predefined fidelity criterion; alternatively, if the number of key frames is fixed, the algorithms maximize the number of frames that the key frames can represent [212], [213]. For instance, Chang *et al*. [214] specify the coverage of a key frame as the number of the frames that are visually similar to the key frame. A greedy algorithm is used iteratively to find key frames.

3) Minimum correlation: These algorithms extract key frames to minimize the sum of correlations between key frames (especially successive key frames), making key frames as uncorrelated with each other as possible. For instance, Porter *et al.* [215] represent frames in a shot and their correlations using a directed weighted graph. The shortest path in the graph is found and the vertices in the shortest path which corresponds to minimum correlation between frames designate the key frames.

4) Minimum reconstruction error*:* These algorithms extract key frames to minimize the sum of the differences between each frame and its corresponding predicted frame reconstructed from the set of key frames using interpolation. These algorithms are useful for certain applications, such as animation. Lee and Kim [216] use an iterative procedure to select a predetermined number of key frames, in order to reduce the shot reconstruction error as much as possible. Liu *et al.* [217] propose a key frame selection algorithm based on the extent to which key frames record the motion during the shot. In the algorithm, an inertiabased frame interpolation algorithm is used to interpolate frames.

The merits of the aforesaid global comparison-based algorithms include the following. 1) The key frames reflect the global characteristics of the shot. 2) The number of key frames is controllable. 3) The set of key frames is more concise and less redundant than that produced by the sequential comparison-based algorithms. The limitation of the global comparison-based algorithms is that they are more computationally expensive than the sequential comparison-based algorithms.

3) Reference Frame: These algorithms generate a reference frame and then extract key frames by comparing the frames in the shot with the reference frame. For instance, Ferman and Tekalp [204] construct an alpha-trimmed average histogram describing the color distribution of the frames in a shot. Then, the distance between the histogram of each frame in the shot and the alpha-trimmed average histogram is calculated. Key frames are located using the distribution of the distance curve. Sun *et al*. [205] construct a maximum occurrence frame for a shot.

Then, a weighted distance is calculated between each frame in the shot and the constructed frame. Key frames are extracted at the peaks of the distance curve. The merit of the reference frame-based algorithms is that they are easy to understand and implement. The limitation of these algorithms is that they depend on the reference frame: If the reference frame does not adequately represent the shot, some salient contents in the shot may be missing from the key frames.

4) Clustering: These algorithms cluster frames and then choose frames closest to the cluster centers as the key

frames. Girgensohn and Boreczky [199] select key frames using the complete link method of hierarchical agglomerative clustering in the color feature space. Yu *et al*. [200] extract key frames using the fuzzy K-means clustering in the color feature subspace. Gibson *et al*. [201] use Gaussian mixture models (GMM) in the eigenspace of the image, in which the number of GMM components is the required number of clusters. The merits of the clustering-based algorithms are that they can use generic clustering algorithms, and the global characteristics of a video can be reflected in the extracted key frames. The limitations of these algorithms are as follows: First, they are dependent on the clustering results, but successful acquisition of semantic meaningful clusters is very difficult, especially for large data, and second, the sequential nature of the video cannot be naturally utilized: Usually, clumsy tricks are used to ensure that adjacent frames are likely to be assigned to the same cluster.

5) Curve Simplification: These algorithms represent each frame in a shot as a point in the feature space. The points are linked in the sequential order to form a trajectory curve and then searched to find a set of points which best represent the shape of the curve. Calic and Izquierdo [218] generate the frame difference metrics by analyzing statistics of the macroblock features extracted from the MPEG compressed stream.

The key frame extraction method is implemented using difference metrics curve simplification by the discrete contour evolution algorithm. The merit of the curve simplification-based algorithms is that the sequential information is kept during the key frame extraction. Their limitation is that optimization of the best representation of the curve has a high computational complexity.

6) Objects/Events: These algorithms [192] jointly consider key frame extraction and object/event detection in order to ensure that the extracted key frames contain information about objects or events. Calic and Thomas [196] use the positions of regions obtained using frame segmentation to extract key frames where objects merge. Kim and Hwang [197] use shape features to extract key frames that can represent changes of human gestures.

Liu and Fan [194] select initial key frames based on the color histogram and use the selected key frames to estimate a GMMfor object segmentation. The segmentation results and the trained GMM are further used to refine the initial key frames. Song and Fan [195] propose a joint key frame extraction and object segmentation method by constructing a unified feature space for both processes, where key frame extraction is formulated as a feature selection process for object segmentation in the context of GMM-based video modeling. Liu *et al*. [203] propose a triangle model of perceived motion energy for motion patterns in videos. The frames at the turning points of the motion acceleration and motion deceleration are selected as key frames. Han and Kweon [220] extract key frames by the maximum curvature of camera motion at each temporal scale. The key frames provide temporal interest points for classification of video events. The merit of the object/event-based algorithms is that the extracted key frames are semantically important, reflecting objects or the motion patterns of objects. The limitation of these algorithms is that object/event detection strongly relies on heuristic rules specified according to the application. As a result, these algorithms are efficient only when the experimental settings are carefully chosen.

Because of the subjectivity of the key frame definition, there is no uniform evaluation method for key frame extraction. In general, the error rate and the video compression ratio are used as measures to evaluate the result of key frame extraction. Key frames giving low error rates and high compression rates are preferred. In general, a low error rate is associated with a low compression rate. The error rate depends on the parameters in the key frame extraction algorithms. Examples of these parameters are the thresholds in sequential comparison-based, global comparison-based, reference frame-based, and clustering-based algorithms, as well as the parameters to fit the curve in the curve simplification-based algorithms. Users choose the parameters according to the error rate that can be tolerated.

C. Scene Segmentation

Scene segmentation is also known as story unit segmentation.

In general, a scene is a group of contiguous shots that are coherent with a certain subject or theme. Scenes have higher level semantics than shots. Scenes are identified or segmented out by grouping successive shots with similar content into a meaningful semantic unit. The grouping may be based on information from texts, images, or the audio track in the video.
According to shot representation, scene segmentation approaches can be classified into three categories: key framebased, audio and visual information integration-based, and background-based.

1) Key Frame-Based Approach: This approach [145] represents each video shot by a set of key frames from which features are extracted. Temporally close shots with similar features are grouped into a scene. For instance, Hanjalic *et al*. [140] compute similarities between shots using block matching of the key frames. Similar shots are

linked, and scenes are segmented by connecting the overlapping links. Ngo *et al*. [144] extract and analyze the motion trajectories encoded in the temporal slices of image volumes. A motion-based key frame selection strategy is, thus, used to compactly represent shot contents. Scene changes are detected by measuring the similarity of the key frames in the neighboring shots. The limitation of the key frame-based approach is that key frames cannot effectively represent the dynamic contents of shots, as shots within a scene are generally correlated by dynamic contents within the scene rather than by key frame-based similarities between shots.

2) Audio and Vision Integration-Based Approach: This approach selects a shot boundary where the visual and audio contents change simultaneously as a scene boundary. For instance,
Sundaram and Chang [147] detect audio scenes and video scenes separately. A time-constrained nearest neighbor algorithm is used to determine the correspondences between these two sets of scenes. The limitation of the audio and visual integration based approach is that it is difficult to determine the relation between audio segments and visual shots.

3) Background-Based Approach: This approach segments scenes under the assumption that shots belonging to the same scene often have similar backgrounds. For instance, Chen *et al.* [139] use a mosaic technique to reconstruct the background of each video frame. Then, the color and texture distributions of all the background images in a shot are estimated to determine the shot similarity and the rules of filmmaking are used to guide the shot grouping process. The limitation of the background based approach is the assumption that shots in the same scene have similar backgrounds: sometimes the backgrounds in shots in a scene are different.

According to the processing method, current scene segmentation approaches can be divided into four categories: merging based, splitting-based, statistical model-based, and shot boundary classification-based.
a) Merging-based approach: This approach gradually merges similar shots to form a scene in a bottom-up style. Rasheed and Shah [133] propose a two-pass scene segmentation algorithm. In the first pass, over segmentation of scenes is carried out using backward shot coherence. In the second pass, the over segmented scenes are identified using motion analysis and then merged. Zhao *et al*. [134] propose a best first model merging algorithm for scene segmentation. The algorithm takes each shot as a hidden state and loops upon the boundaries between consecutive shots by a left-right HMM.

b) Splitting-based approach: This approach splits the whole video into separate coherent scenes using a top-down style. For instance, Rasheed and Shah [136] construct a shot similarity graph for a video and partition the graph using normalized cuts. The subgraphs represent individual scenes in the video. Tavanapong and Zhou [138] introduce a scene definition for narrative films and present a technique to cluster relevant shots into a scene using this definition.

c) Statistical model-based approach: This approach constructs statistical models of shots to segment scenes. Zhai and Shah [132] use the stochastic Monte Carlo sampling to simulate the generation of scenes. The scene boundaries are updated by diffusing, merging, and splitting the scene boundaries estimated in the previous step. Tan and Lu [137] use the GMM to cluster video shots into scenes according to the features of individual shots. Each scene is modeled with a Gaussian density.

Gu *et al*. [149] define a unified energy minimization framework in which the global content constraint between individual shots and the local temporal constraint between adjacent shots are both represented. A boundary voting procedure decides the optimal scene boundaries.

d) Shot boundary classification-based approach*:* In this approach, features of shot boundaries are extracted and then used to classify shot boundaries into scene boundaries and nonscene boundaries. Goela *et al*. [148] present a genre-independent method to detect scene boundaries in broadcast videos. In their method, scene segmentation is based on a classification with the two classes of “scene change” and “nonscene change.” An SVMis used to classify the shot boundaries. Hand-labeled video scene boundaries from a variety of broadcast genres are used to generate positive and negative training samples for the SVM. The common point in the merging-based, splitting-based, and statistical model-based approaches is that the similarities between different shots are used to combine similar shots into scenes. This is simple and intuitive. However, in these approaches, shots are usually represented by a set of selected key frames, which often fail to represent the dynamic contents of the shots. As a result, two shots are regarded as similar, if their key frames are in the same environment rather than if they are visually similar. The shot boundary classificationbased approach takes advantage of the local information about shot boundaries. This ensures that algorithms with low computational complexities are easy to obtain. However, lack of global information about shots inevitably reduces the accuracy of scene segmentation. It is noted that most current approaches for scene segmentation exploit the characteristics of specific video domains such as movies, TVs, and news broadcasts [150], [152], [153], for

example, using the production rules by which movies or TV shows are composed. The accuracy of scene segmentation is improved, but it is necessary to construct *a priori* model for each application.

3. Feature Extraction

To extract features according to video structural analysis results is the base of video indexing and retrieval.We focus on the visual features suitable for video indexing and retrieval. These mainly include features of key frames, objects, and motions. Auditory features and text features are not covered.

A. Static Features of Key Frames

The key frames of a video reflect the characteristics of the video to some extent. Traditional image retrieval techniques can be applied to key frames to achieve video retrieval. The static key frame features useful for video indexing and retrieval are mainly classified as color-based, texture-based, and shape-based.

1) Color-Based Features: Color-based features include color histograms, color moments, color correlograms, a mixture of Gaussian models, etc. The exaction of color-based features depends on color spaces such as RGB, HSV, YCbCr and normalized r-g, YUV, and HVC. The choice of color space depends on the applications. Color features can be extracted from the entire image or from image blocks into which the entire image is partitioned. Color-based features are the most effective image features for video indexing and retrieval. In particular, color histogram and color moments are simple but efficient descriptors.

Amir *et al.* [222] compute color histogram and color moments for video retrieval and concept detection. Yan and Hauptmann [229] first split the image into 5×5 blocks to capture local color information. Then in each block, color histogram and color moments are extracted for video retrieval. Adcock *et al.* [226] use color correlograms to implement a video search engine. The merits of color-based features are that they reflect human visual perception, they are easy to extract, and their extraction has low computational complexity. The limitation of color-based features is that they do not directly describe texture, shape, etc., and are, thus, ineffective for the applications in which texture or shape is important.

2) Texture-Based Features: Texture-based features are object surface-owned intrinsic visual features that are independent of color or intensity and reflect homogenous phenomena in images. They contain crucial information about the organization of object surfaces, as well as their correlations with the surrounding environment. Texture features in common use include Tamura features, simultaneous autoregressive models, orientation features, wavelet transformation-based texture features, co-occurrence matrices, etc. Amir *et al.* [222] use concurrence texture and Tamura features including coarseness, contrast and directionality for the TRECVid-2003 video retrieval task. Hauptmann *et al.* [223] use Gabor wavelet filters to capture texture information for a video search engine. They design 12 oriented energy filters. The mean and variance of the filtered outputs are concatenated into a texture feature vector.

Hauptmann *et al.* [228] divide the image into 5×5 blocks and compute texture features using Gabor-wavelet filters in each block. The merit of texture-based features is that they can be effectively applied to applications in which texture information is salient in videos. However, these features are unavailable in nontexture video images.

3) Shape-Based Features: Shape-based features that describe object shapes in the image can be extracted from object contours or regions. A common approach is to detect edges in images and then describe the distribution of the edges using a histogram. Hauptmann *et al.* [223] use the edge histogram descriptor (EHD) to capture the spatial distribution of edges for the video search task in TRECVid-2005. The EHD is computed by counting the number of pixels that contribute to the edge according to their quantized directions. To capture local shape features, Foley *et al.* [224] and Cooke *et al.* [225] first divide the image into 4×4 blocks and then extract a edge histogram for each block. Shape-based features are effective for applications in which shape information is salient in videos. However, they are much more difficult to extract than color- or texture-based features.

B. Object Features

Object features include the dominant color, texture,, size, etc., of the image regions corresponding to the objects. These features can be used to retrieve videos likely to contain similar objects [17]. Faces are useful objects in many video retrieval systems. For example, Sivic *et al.* [18] construct a person retrieval system that is able to retrieve a ranked list of shots containing a particular person, given a query face in a shot. Le *et al.* [19] propose a method to retrieve faces in broadcast news videos by integrating temporal information into facial intensity information. Texts in a video are extracted as one type of object to help understand video contents. Li and Doermann [20] implement
text-based video indexing and retrieval by expanding the semantics of a query and using the Glimpse matching

method to perform approximate matching instead of exact matching. The limitation of object-based features is that identification of objects in videos is difficult and time-consuming. Current algorithms focus on identifying specific types of objects, such as faces, rather than various objects in various scenes.

C. Motion Features

Motion is the essential characteristic distinguishing dynamic videos from still images. Motion information represents the visual content with temporal variation. Motion features are closer to semantic concepts than static key frame features and object features. Video motion includes background motion caused by camera motion and foreground motion caused by moving objects. Thus, motion-based features for video retrieval can be divided into two categories: camera-based and object-based. For camera-based features, different camera motions, such as "zooming in or out," "panning left or right," and "tilting up or down," are estimated and used for video indexing. Video retrieval using only camera-based features has the limitation that they cannot describe motions of key objects.

Object-based motion features have attracted much more interest in recent work. Object-based motion features can be further classified into statistics-based, trajectory-based, and objects' spatial relationships-based.

1) Statistics-Based: Statistical features of the motions of points in frames in a video are extracted to model the distribution of global or local motions in the video. For instance, Fablet *et al.* [233] use causal Gibbs models to represent the spatiotemporal distribution of appropriate local motion-related measurements computed after compensating for the estimated dominant image motions in the original sequence. Then, a general statistical framework is developed for video indexing and retrieval. Ma and Zhang [234] transform the motion vector field to a number of directional slices according to the energy of the motion. These slices yield a set of moments that form a multidimensional vector called motion texture. The motion texture is used for motion-based shot retrieval. The merit of statisticsbased features is that their extraction has low computational complexity. The limitation of these features is they cannot represent object actions accurately and cannot characterize the relations between objects.

2) Trajectory-Based: Trajectory-based features [22] are extracted by modeling the motion trajectories of objects in videos. Chang *et al.* [236] propose an online video retrieval system supporting automatic object-based indexing and spatiotemporal queries. The system includes algorithms for automated video object segmentation and tracking. Bashir *et al.* [237] present a motion trajectory-based compact indexing and efficient retrieval mechanism for video sequences. Trajectories are represented by temporal orderings of subtrajectories. The subtrajectories are then represented by their principal component analysis coefficients.

Chen and Chang [238] use wavelet decomposition to segment each trajectory and produce an index based on velocity features. Jung *et al.* [25] base their motion model on polynomial curve fitting. The motion model is used as an indexing key to access individual objects. Su *et al.* [26] construct motion flows from motion vectors embedded in MPEG bitstreams to generate continual motion information in the form of a trajectory. Given a trajectory, the system retrieves a set of trajectories that are similar to it. Hsieh *et al.* [27] divide trajectories into several small segments, and each segment is described by a semantic symbol.

A distance measure combining an edit distance and a visual distance is exploited to match trajectories for video retrieval. The merit of trajectory-based features is that they can describe object actions. The limitation of these features is that their extraction depends on correct object segmentation and tracking and automatic recording of trajectories, all of which are still very challenging tasks.

3) Objects' Relationship-Based: These features describe spatial relationships between objects. Bimbo *et al.* [235] describe relationships between objects using a symbolic representation scheme which is applied to video retrieval. Yajima *et al.* [24] query the movements of multiple moving objects and specify the spatiotemporal relationships between objects by expressing each object's trace on a timeline. The merit of objects' relationship-based features is that they can intuitively represent relationships between multiple objects in the temporal domain. The limitation of these features is that it is difficult to label each object and its position.

4. Video Data Mining, Classification, And Annotation

Video data mining, classification, and annotation rely heavily on video structure analysis and the extracted video features. There are no boundaries between video data mining, video classification, and video annotation. In particular, the concepts of video classification and annotation are very similar. In this section, we review the basic concepts and approaches for video data mining, classification, and annotation. The annotation is the basis for the detection of video's semantic concepts and the construction of semantic indices for videos.

A. Video Data Mining

The task of video data mining is, using the extracted features, to find structural patterns of video contents, behavior patterns of moving objects, content characteristics of a scene, event patterns [230], [232] and their associations, and other video semantic knowledge [45], in order to achieve video intelligent applications, such as video retrieval [118]. The choice of a strategy for video data mining depends on the application. Current strategies include the following.

1) Object Mining: Object mining is the grouping of different instances of the same object that appears in different parts in a video. It is very hard because the appearance of an object can change a great deal from one instance to another. Sivic and Zisserman [86] use a spatial neighborhood technique to cluster the features in the spatial domain of the frames. These clusters are used to mine frequently appearing objects in key frames. Anjulan and Canagarajah [81] extract stable tracks from shots. These stable tracks are combined into meaningful object clusters, which are used to mine similar objects. Quack *et al*. [28] present a method for mining frequently occurring objects and scenes from videos. Object candidates are detected by finding recurring spatial arrangements of affine covariant regions.

2) Special Pattern Detection: Special pattern detection applies to actions or events for which there are *a priori* models, such as human actions, sporting events [127], traffic events, or crime patterns. Laptev *et al.* [124] propose an appearance-based method that recognizes eight human actions in movies, e.g., answer phone, get out of a car, handshake, hug person, kiss. They extract local space-time features in space-time pyramids, build a spatial-temporal bag-of-features, and employ multichannel nonlinear SVMs for recognition. Ke *et al*. [125] propose a template-based method that recognizes human actions, such as picking up a dropped object or waving in a crowd. They oversegment the video to obtain spatial-temporal patches, and combine shape and optical flow cues to match testing patches and templates. Liu *et al*. [126] detect events in a football match, including penalty kicks, free kicks near the penalty box, and corner kicks in football games. Li and Porikli [128] detect six traffic patterns using a Gaussian mixture HMMframework, and Xie *et al.* [129] extract traffic jam events by analyzing the road background features. Nath [130] detects crime patterns using a clustering algorithm.

3) Pattern Discovery*:* Pattern discovery is the automatic discovery of unknown patterns in videos using unsupervised or semisupervised learning. The discovery of unknown patterns is useful to explore newdata in a video set or to initializemodels for further applications. Unknown patterns are typically found by clustering various feature vectors in the videos. The discovered patterns have the following applications: 1) detecting unusual events [230] that are often defined by their dissimilarity to discovered patterns; 2) associating clusters or patterns with words for video retrieval, etc; 3) building supervised classifiers based on the mined clusters for video classification or annotation, etc. Burl [105] describes an algorithm for mining motion trajectories to detect trigger events, determine typical or anomalous patterns of activities, classify activities into named categories, cluster activities, determine interactions between entities, etc.

Hamid *et al*. [2] use *n*-grams and suffix trees to mine motion patterns by analyzing event subsequences over multiple temporal scales. The mined motion patterns are used to detect unusual events. Turaga *et al*. [1] use a generative model to capture and represent a diverse class of activities, and build affine and view invariance of the activity into the distance metric for clustering.

The clusters correspond to semantically meaningful activities. Cutler and Davis [14] compute an object's self-similarity as it evolves in time, and apply time–frequency analysis to detect and characterize the periodic motion. The periodicity is analyzed using the 2-D lattice structures inherent in similarity matrices.

4) Video Association Mining: Video association mining is mainly used to discover inherent relations between different events or the most frequent association patterns for different objects, such as the simultaneous occurrence of two objects, frequency of shot switches, and association between video types [118]. Video association mining also includes the deduction of interassociations between semantic concepts in the same shot from existing annotations or the inference of a semantic concept for the current shot from detection results of neighboring shots, etc. Pan and Faloutsos [102] propose an algorithm to find correlations between different events in news programs, such as those between "earthquake" and "volcano" or "tourism" and "wine." Zhu *et al*. [100] propose explicit definitions and evaluationmeasures for video associations by integrating distinct feature of the video data. Their algorithm introduces multilevel sequential association mining to explore associations between audio and visual cues, classifies the associations by assigning each of them a class label, and uses their appearances in the video to construct video indices.

Yan *et al.* [13] describe various multiconcept relational learning algorithms based on a unified probabilistic graphical model representation and use graphical models to mine the relationship between video concepts.

Liu *et al.* [231] use association-mining techniques to discover interconcept associations in the detected concepts, and mine intershot temporal dependence, in order to improve the accuracy of semantic concept detection.

5) Tendency Mining*:* Tendency mining is the detection and analysis of trends of certain events by tracking current events [118]. Xie *et al.* [103] propose a news video mining method, which involves two visualization graphs: the time-tendency graph and the time-space distribution graph. The time tendency graph records the tendencies of events, while the time-space distribution graph records the spatial-temporal relations between various events. Oh and Bandi [104] mine the tendency of a traffic jam by analyzing the spatial-temporal relations between objects in videos.

6) Preference Mining*:* For news videos, movies, etc., the user's preferences can be mined [118]. For instance, Kules *et al.* [101] propose a personalized multimedia news portal to provide a personalized news service by mining the user's preferences.

B. Video Classification

The task of video classification [106], [245] is to find rules or knowledge from videos using extracted features or mined results and then assign the videos into predefined categories. Video classification is an important way of increasing the efficiency of video retrieval. The semantic gap between extracted formative information, such as shape, color, and texture, and an observer's interpretation of this information, makes content-based video classification very difficult.

Video content includes semantic content and editing effects. Referring to [23], semantic content classification can be performed on three levels: video genres, video events, and objects in the video, where genres have rougher and wider detection range; and events and objects have thinner and limited detection range. In the following, we discuss edit effect classification, genre classification, event classification, and object classification, respectively.

1) Edit Effect Classification: Editing effects depend on the ways for editing videos, such as camera motion and the composition of scenes and shots. Editing effects themselves are not a part of video content, but they influence the understanding of video content; therefore, they may be used in video semantic classification.

For instance, Ekin *et al.* [165] classify shots of soccer videos into long, in-field medium, close-up, and out-of-field views using cinematic features and further detect events such as play, break, and replay. Xu *et al.* [246] use the domain-specific feature of grass-area-ratio to classify frames of soccer videos into global, zoom-in, and close-up views and obtain play/break statuses of games from the sequences of labeled frames. Tan *et al.* [247] estimate camera motion using data from the MPEG stream, and further classify basketball shots into wide-angle and close-up views and detect events such as fast breaks, shots at Nthe basket, etc.

2) Video Genre Classification*:* Video genre classification is the classification of videos into different genres such as "movie," "news," "sports," and "cartoon." Approaches to classify video genres can be classified into statistic-based, rule- or knowledgebased, and machine learning-based [23].

a) Statistic-based approach*:* This approach classifies videos by statistically modeling various video genres. Fisher *et al.* [89] classify videos as news, car race, tennis, animated cartoon, and commercials. First, video syntactic properties such as color statistics, cuts, camera motion, and object motion are analyzed. Second, these properties are used to derive more abstract film style attributes such as camera panning and zooming, speech, and music. Finally, these detected style attributes are mapped into film genres. Based on characteristics of films, Rasheed *et al.* [123] only use four visual features, namely average shot length, color variance, motion content, and lighting key, to classify films into comedies, actions, dramas, or horror films. The classification is achieved using mean shift clustering. Some methods only utilize dynamic features to classify video genres. Roach *et al.* [122] propose a cartoon video classification method that uses motion features of foreground objects to distinguish between cartoons and noncartoons. Roach *et al.* [108] classify videos based on the dynamic content of short video sequences, where foreground object motion and background camera motion are extracted from videos. The classified videos include sports, cartoons, and news.

b) Rule- or knowledge-based approach*:* This approach applies heuristic rules from domain knowledge to low-level features to classify videos. Chen and Wong [109] develop a knowledge-based video classification method, in which the relevant knowledge is coded in the form of generative rules with confidences to form a rule-base. The Clip language is used to compile a video content classification system using the rulebase. Zhou *et al.* [110] propose a supervised rule-based video classification system, in which higher semantics are derived from a joint use of low-level features along with classification rules that are derived through a supervised learning process. Snoek *et al.* [93] propose a video classification and indexing

method, combining video creation knowledge to extract semantic concepts from videos by exploring different paths through three consecutive analysis steps: the multimodel video content analysis step, the video style analysis step, and the context analysis step. Zhou *et al*. [107] propose a rule-based video classification system that applies video content analysis, feature extraction and clustering techniques to the semantic clustering of videos. Experiments on basketball videos are reported.

c) Machine learning-based approach: This approach uses labeled samples with low-level features to train a classifier or a set of classifiers for videos. Mittal and Cheong [112] use the Bayesian network to classify videos. The association between a continuous and nonparametric descriptor space and the classes is learned and the minimum Bayes error classifier is deduced. Qi *et al*. [97] propose a video classification framework using SVMs-based active learning. The results of clustering all the videos in the dataset are used as the input to the framework. The accuracy of the classifiers is improved gradually during the active-learning process. Fan *et al*. [98] use multiple levels of concepts of video contents to achieve hierarchical semantic classification of videos to enable highly efficient access to video contents. Truong *et al*. [90] classify videos into the genres of cartoons, commercials, music, news, and sports. The features used include the average shot length, the percentage of each type of transition, etc. The C4.5 decision tree is used to build the classifier for genre labeling. Yuan *et al*. [240] present an automatic video genre classification method based on a hierarchical ontology of video genres. A series of SVM classifiers united in a binary-tree form assign each video to its genre. Wu *et al*. [154] propose an online video semantic classification framework, in which local and global sets of optimized classification models are online trained by sufficiently exploiting both local and global statistic characteristics of videos. Yuan *et al*. [155] learn concepts from a large-scale imbalanced dataset using support cluster machines.

From the aforesaid video genres classification approaches, the following conclusions can be drawn [23]. 1) These approaches either use static features only, dynamic features only, or combine them both. 2) All the approaches preferably employ global statistical low-level features. This is because such features are robust to video diversity, making them appropriate for video genre classification. Many algorithms attempt to add some semantic features on the basis of these low-level features. 3) Prior domain knowledge is widely used in video genres classification. To use knowledge or rules can improve the classification efficiency for special domains, but the corresponding algorithms cannot be generalized to videos from other domains.

3) Event Classification: An event can be defined as any human-visible occurrence that has significance to represent video contents. Each video can consist of a number of events, and each event can consist of a number of subevents. To determine the classes of events in a video is an important component of content-based video classification [3], and it is connected with event detection in video data mining. There is a great deal of published work on event classification. Yu *et al.* [115] detect and track balls in broadcast soccer videos and extract ball trajectories, which are used to detect events such as hand ball and ball possession by a team. Chang *et al*. [111] detect and classify highlights in baseball game videos using HMMmodels that are learned from special shots identified as highlights. Duan *et al.* [116] propose a visual feature representation model for sports videos. This model is combined with supervised learning to perform a top-down semantic shot classification. These semantic shot classes are further used as a midlevel representation for high-level semantic analysis. Xu *et al*. [94] present an HMM-based framework for video semantic analysis. Semantics in different granularities are mapped to a hierarchical model in which a complex analysis problem is decomposed into subproblems. The framework is applied to basketball event detection. Osadchy and Keren [119] offer a natural extension of the "antiface" method to event detection, in both the gray-level and feature domains. Xie *et al*. [151] employ HMM and dynamic programming to detect the sports video concepts of "play," "no play," etc. Pan *et al.* [114] extract visual features and then use an HMM to detect slow-motion replays in sports videos.

From the aforesaid event classification algorithms, the following conclusions can be drawn [23]. 1) In contrast with genre classification, event classification needs more complex feature extraction. 2) Complicated motion measures are often attached to event classifiers. Some event classification methods employ only dynamic features, involving the accurate tracking of moving objects or rough region-based motion measures, and then classify the object motions in order to recognize motion events.

4) Object Classification: Video object classification which is connected with object detection in video data mining is conceptually the lowest grade of video classification. The most common detected and classified object is the face [120]. Object detection often requires the extraction of structural features of objects and classification of these features. Prior knowledge such as an object appearance model is often incorporated into the process of object feature extraction and classification. Hong *et al.* [92] propose an object-based algorithm to classify video shots. The objects in shots are represented using features of color, texture, and trajectory. A neural network is used to cluster correlative shots, and each

cluster is mapped to one of 12 categories. A shot is classified by finding the best matching cluster. Dimitrova *et al.* [91] propose a method to classify four types of TV programs. Faces and texts are detected and tracked, and the number of faces and texts is used to label each frame of a video segment. An HMM is trained for each type using the frame labels as the observation symbols. The limitation of object classification for video indexing is that it is not generic; video object classification only works in specific environments.

C. Video Annotation

Video annotation [4], [117], [241] is the allocation of video shots or video segments to different predefined semantic concepts, such as person, car, sky, people walking. Video annotation is similar to video classification, except for two differences [239]: 1) Video classification has a different category/concept ontology compared with video annotation, although some of the concepts could be applied to both; and 2) video classification applies to complete videos, while video annotation applies to video shots or video segments. Video annotation and video classification share similar methodologies: First, low-level features are extracted, and then certain classifiers are trained and employed to map the features to the concept/category labels. Corresponding to the fact that a video may be annotated with multiple concepts, the approaches for video annotation can be classified as isolated concept-based annotation, context-based annotation, and integrated-based annotatin [244].

1) Isolated Concept-Based Annotation*:* This annotation method trains a statistical detector for each of the concepts in a visual lexicon, and the isolated binary classifiers are used individually and independently to detect multiple semantic concepts—correlations between the concepts are not considered. Feng *et al.* [8] use the multiple-Bernoulli distribution to model image and video annotation. The multiple-Bernoulli model explicitly focuses on the presence or absence of words in the annotation, based on the assumption that each word in an annotation is independent of the other words. Naphade and Smith [69] investigate the efficiencies of a large variety of classifiers, including GMM, HMM, kNN, and Adaboost, for each concept. Song *et al.* [9] introduce active learning together with semisupervised learning to perform semantic video annotation. In this method, a number of two-class classifiers are used to carry out the classification with multiple classes. Duan *et al.* [116] employ supervised learning algorithms based on the construction of effective midlevel representations to perform video semantic shot classification for sports videos. Shen *et al.* [73] propose a cross-training strategy to stack concept detectors into a single discriminative classifier and to handle the classification errors that occur when the classes overlap in the feature space. The limitation of isolated concept-based annotation is that the associations between the different concepts are not modeled.

2) Context-Based Annotation*:* To use contexts for different concepts [71] can improve concept detection performance. The task of context-based annotation is to refine the detection results of the individual binary classifiers or infer higher level concepts from detected lower level concepts using a context based concept fusion strategy. For instance, Wu *et al*. [248] use an ontology-based learning method to detect video concepts. An ontology hierarchy is used to improve the detection accuracy of the individual binary classifiers. Smith and Naphade [249] construct model vectors based on the detection scores of individual classifiers to mine the unknown or indirect correlations between specific concepts and then train an SVM to refine the individual detection results. Jiang *et al*. [250] propose an active-learning method to annotate videos. In the method, users annotate a few concepts for a number of videos, and the manual annotations are then used to infer and improve detections of other concepts. Bertini *et al*. [251] propose an algorithm that uses pictorially enriched ontologies that are created by an unsupervised clustering method to perform automatic soccer video annotation. Occurrences of events or entities are automatically associated with higher level concepts, by checking their proximity to visual concepts that are hierarchically linked to higher level semantics. Fan *et al*. [32], [253] propose a hierarchical boosting scheme, which incorporates concept ontology and multitask learning, to train a hierarchical video classifier that exploits the strong correlations between video concepts. The limitation of context-based annotation is that the improvement of contextual correlations to individual detections is not always stable because the detection errors of the individual classifiers can propagate to the fusion step, and partitioning of the training samples into two parts for individual detections and conceptual fusion, respectively, causes that there are no sufficient samples for the conceptual fusion because of usual complexity of the correlations between the concepts.

3) Integration-Based Annotation*:* This annotation method simultaneously models both the individual concepts and their correlations: The learning and optimization are done simultaneously. The entire set of samples is used simultaneously to model the individual concepts and their correlations. Qi *et al*. [244] propose a correlative multilabel algorithm, which constructs a new feature vector that captures both the characteristics of concepts and the correlations between concepts. The limitation of the integration-based annotation is its high computational complexity.

The learning of a robust and effective detector for each concept requires a sufficiently large number of accurately labeled training samples, and the number required increases exponentially with the feature dimension. Recently, some approaches have been proposed to incorporate unlabeled data into the supervised learning process in order to reduce the labeling burden. Such approaches can be classified into semisupervised-based and active-learning-based.

a) Semisupervised learning*:* This approach uses unlabeled samples to augment the information in the available labeled examples. Yan and Naphade [74], [146] present semisupervised cross feature learning for cotraining-based video concept detection and investigate different labeling strategies in cotraining involving unlabeled data and a small number of labelled videos.
b) Active learning*:* Active learning is also an effective way to handle the lack of labeled samples. Song *et al*. [6] propose an active-learning algorithm for video annotation based on multiple complementary predictors and incremental model adaptation. Furthermore, Song *et al*. [7] propose a video annotation framework based on an active learning and semisupervised ensemble method, which is specially designed for personal video databases.

5. Query and Retrieval

Once video indices are obtained, content-based video retrieval [5] can be performed. On receiving a query, a similarity measure method is used, based on the indices, to search for the candidate videos in accordance with the query. The retrieval results are optimized by relevance feedback, etc. In the following, we review query types, similarity matching, and relevance feedback.

A. Query Types

Nonsemantic-based video query types include query by example, query by sketch, and query by objects. Semantic-based video query types include query by keywords and query by natural language.

B. Similarity Measure

Video similarity measures play an important role in contentbased video retrieval. Methods to measure video similarities can be classified into feature matching, text matching, ontologybased matching, and combination-based matching. The choice of method depends on the query type.

6. Video Summarization and Browsing

Video summarization [39], [156], [157], [181] removes the redundant data in videos and makes an abstract representation or summary of the contents, which is exhibited to users in a readable fashion to facilitate browsing. Video summarization complements video retrieval [183], by making browsing of retrieved videos faster, especially when the total size of the retrieved videos is large: The user can browse through the abstract representations to locate the desired videos. A detailed review on video browsing interfaces and applications can be found in [261].

There are two basic strategies for video summarization.

1) Static video abstracts*:* each of which consists of a collection of key frames extracted from the source video.
2) Dynamic video skims: each of which consists of a collection of video segments (and corresponding audio segments) that are extracted from the original video and then concatenated to form a video clip which is much shorter than the original video.

These two strategies can be combined to form hierarchical video summarizations. In the following, the different methods for video summarization are briefly reviewed. As video summarization is a research topic which is as large as video retrieval, we focus on reviewing papers published in the last four years, as a supplement to previous surveys [39], [181] on video summarization.

7. Future Developments

Although a large amount of work has been done in visual content-based video indexing and retrieval, many issues are still open and deserve further research, especially in the following areas.

1) Motion Feature Analysis. The effective use of motion information is essential for content-based video retrieval. To distinguish between background motion and foreground motion, detect moving objects and events, combine static features and motion features, and construct motion-based indices are all important research areas.

2) Hierarchical Analysis of Video Contents. One video may
contain different meanings at different semantic levels. Hierarchical organization of video concepts is required for semanticbased video indexing and retrieval. Hierarchical analysis requires the decomposition of high-level semantic concepts into a series of low-level basic semantic concepts

and their constraints. Low-level basic semantic concepts can be directly associated with low-level features, and high-level semantic concepts can be deduced from low-level basic semantic concepts by statistical analysis. In addition, building hierarchical semantic relations between scenes, shots, and key frames, on the basis of video structural analysis; establishing links between classifications with the three different levels: genres, event and object; and hierarchically organizing and visualizing retrieval results are all interesting research issues.

3) Hierarchical Video Indices. Corresponding to hierarchical video analysis, hierarchical video indices can be utilized in video indexing. The lowest layer in the hierarchy is the index storemodel corresponding to the high-dimensional feature index structure. The highest layer is the semantic index model describing the semantic concepts and their correlations in the videos to be retrieved. The middle layer is the index context model that links the semantic concept model and the store model. Dynamic, online, and adaptive updating of the hierarchical index model, handling of temporal sequence features of videos during index construction and updating, dynamic measure of video similarity based on statistic feature selection, and fast video search using hierarchical indices are all interesting research questions.

4) Fusion of Multimodels. The semantic content of a video is usually an integrated expression of multiple models. Fusion of information from multiple models can be useful in contentbased video retrieval [38], [95]. Description of temporal relations between different kinds of information from multiple models, dynamic weighting of features of different models, fusion of information from multiple models that express the same theme, and fusion of multiple model information in multiple levels are all difficult issues in the fusion analysis of integrated models.

8. Conclusion

We have presented a review on recent developments in visual content-based video indexing and retrieval. The state of the art of existing approaches in each major issue has been described with the focus on the following tasks: video structure analysis including shot boundary detection, key frame extraction and scene segmentation, extraction of features of static key frames, objects and motions, video data mining, video classification and annotation, video search including interface, similarity measure and relevance feedback, and video summarization and browsing. At the end of this survey, we have discussed future directions such as affective computing-based video retrieval and distributed network video retrieval.

References

[1] P. Turaga, A. Veeraraghavan, and R. Chellappa, "From videos to verbs:
Mining videos for activities using a cascade of dynamical systems," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, Jun. 2007, pp. 1–8.

[2] R.Hamid, S.Maddi,A. Bobick, and M. Essa, "Structure from statistics—
Unsupervised activity analysis using suffix trees," in *Proc. IEEE Int. Conf*
Comput. Vis., Oct., 2007, pp. 1–8.

[3] G. Lavee, E. Rivlin, and M. Rudzsky, "Understanding video events: A
survey of methods for automatic interpretation of semantic occurrences
in video," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 39, no. 5,
pp. 489–504, Sep. 2009.

[4] J. Tang, X. S. Hua, M. Wang, Z. Gu, G. J. Qi, and X. Wu, "Correlative
linear neighborhood propagation for video annotation," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 39, no. 2, pp. 409–416, Apr. 2009.

[5] X. Chen, C. Zhang, S. C. Chen, and S. Rubin, "A human-centered multiple instance learning framework for semantic video retrieval," *IEEE*
Trans. Syst, Man, Cybern., C: Appl. Rev., vol. 39, no. 2, pp. 228–233,
Mar. 2009.

[6] Y. Song, X.-S. Hua, L. Dai, and M.Wang, "Semi-automatic video annotation based on active learning with multiple complementary predictors," in *Proc. ACM Int. Workshop Multimedia Inf. Retrieval*, Singapore, 2005, pp. 97–104.

[7] Y. Song, X.-S. Hua, G.-J. Qi, L.-R. Dai, M. Wang, and H.-J. Zhang,
"Efficient semantic annotation method for indexing large personal video
database," in *Proc. ACM Int. Workshop Multimedia Inf. Retrieval*, Santa
Barbara, CA, 2006, pp. 289–296.

[8] S. L. Feng, R. Manmatha, andV. Lavrenko, "Multiple Bernoulli relevance models for image and video annotation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun./Jul. 2004, vol. 2, pp. 1002–1009.

[9] Y. Song, G.-J. Qi, X.-S. Hua, L.-R. Dai, and R.-H.Wang, "Video annotation by active learning and semi-supervised ensembling," in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jul. 2006, pp. 933–936.

[10] C. H Yeo, Y. W. Zhu, Q. B. Sun, and S. F Chang, "A Framework for
sub-window shot detection," in *Proc. Int. Multimedia Modelling Conf.*, Jan. 2005, pp. 84–91.

[11] G. Camara-Chavez, F. Precioso, M. Cord, S. Phillip-Foliguet, and A. de
A. Araujo, "Shot boundary detection by a hierarchical supervised approach,"
in *Proc. Int. Conf. Syst., Signals Image Process.*, Jun. 2007, pp. 197–200.

[12] H. Lu, Y.-P. Tan, X. Xue, and L. Wu, "Shot boundary detection using
unsupervised clustering and hypothesis testing," in *Proc. Int. Conf. Commun. Circuits Syst.*, Jun. 2004, vol. 2, pp. 932–936.

[13] R.Yan, M.-Y.Chen, andA.G.Hauptmann, "Mining relationship between
video concepts using probabilistic graphical model," in *Proc. IEEE Int.
Conf. Multimedia Expo.*, Jul. 2006, pp. 301–304.

[14] R. Cutler and L. S. Davis, "Robust real-time periodic motion detection,
analysis, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 781–796, Aug. 2000.

[15] K. W. Sze, K. M. Lam, and G. P. Qiu, "A new key frame representation
for video segment retrieval," *IEEE Trans. Circuits Syst. Video Technol.*,
vol. 15, no. 9, pp. 1148–1155, Sep. 2005.