Multi-View Video Summarization

Yanwei Fu, Yanwen Guo, Yanshu Zhu, Feng Liu, Chuanming Song, and Zhi-Hua Zhou, *Senior Member, IEEE*

***Abstract*—Previous video summarization studies focused on monocular videos, and the results would not be good if they were applied to multi-view videos directly, due to problems such as the redundancy in multiple views. In this paper, we present a method for summarizing multi-view videos. We construct a spatio-temporal shot graph and formulate the summarization problem as a graph labeling task. The spatio-temporal shot graph is derived from a hypergraph, which encodes the correlations with different attributes among multi-view video shots in hyperedges. We then partition the shot graph and identify clusters of event-centered shots with similar contents via random walks. The summarization result is generated through solving a multi-objective optimization problem based on shot importance evaluated using a Gaussian entropy fusion scheme. Different summarization objectives, such as minimum summary length and maximum information coverage, can be accomplished in the framework. Moreover, multi-level summarization can be achieved easily by configuring the optimization parameters. We also propose the multi-view storyboard and event board for presenting multi-view summaries. The storyboard naturally reflects correlations among multi-view summarized shots that describe the same important event. The event-board serially assembles event-centered multi-view shots in temporal order. Single video summary which facilitates quick browsing of the summarized multi-view video can be easily generated based on the event board representation.**

***Index Terms*—Multi-objective optimization, multi-view video, random walks, spatio-temporal graph, video summarization.**

I. Introduction

WITH the rapid development of computation, communication, and storage infrastructures, multi-view video systems that simultaneously capture a group of videos and record the video content of the occurrence of events with considerable overlapping field of views (FOVs) across multiple cameras have become more and more popular. In contrast to the rapid development of video collection and storage techniques, consuming these multi-view videos still remains a problem. For instance, watching a large number of videos to grasp important information quickly is a big challenge.

Video summarization, as an important video content service, produces a condensed and succinct representation of video content, which facilitates the browsing, retrieval, and storage of the original videos. There has been a rich literature on summarizing a long video into a concise representation, such as a key-frame sequence [1]–[6] and a video skim [7]–[20]. These existing methods provide effective solutions to summarization. However, they focus on monocular videos. Multi-view video summarization has been rarely addressed, though multi-view videos are widely used in surveillance systems equipped in offices, banks, factories, and crossroads of cities for private and public securities. For the all-weather, day, and night multi-view surveillance systems, video data recorded increases dramatically every day. In addition to surveillance, multi-view videos are also popular in sports broadcast. For example, in the soccer match, the cameramen usually replay the goals recorded by different cameras distributed in the football stadium. Multi-view video summarization refers to the problem of summarizing multi-view videos into informative video summaries, usually presented as dynamic video shots coming from multi-views, by considering content correlations within each view and among multiple views. The multi-view summaries will provide salient events with more rich information than less salient ones. This will allow the user to grasp the important information from multiple perspectives of the multi-view videos without watching the whole of them. Multi-view summarization will also benefit the storage, analysis, and management of multi-view video content.

Applying the existing monocular video summarization methods to each component of a multi-view video group could lead to a redundant summarization result as each component has overlapping information with the others. To generate a concise multi-view video summary, information correlations as well as discrepancies among multi-view videos should be taken into account. It is also not good to directly apply previous methods to the video sequence formed by simply combining the multi-view videos. Furthermore, since multi-view videos often suffer from different lighting conditions in distinctive views, it is nontrivial to evaluate the importance of shots in each view video and to merge each component into an integral video summary in a robust way, especially when the multi-view videos are captured nonsynchronously. It is thus important to have effective multi-view summarization techniques.

In this paper, we present a method for the summarization of multi-view videos. We first parse the video from each view into shots. Content correlations among multi-view shots are important to produce an informative and compact summary. We use a hypergraph to model such correlations, in which each kind of hyperedge characterizes a kind of correlation among shots. By converting the hypergraph into a spatio-temporal shot graph,

Manuscript received August 28, 2009; revised December 21, 2009 and March 26, 2010; accepted May 18, 2010. Date of publication June 07, 2010; date of current version October 15, 2010. This work was supported in part by the National Science Foundation of China under Grants 60703084, 60723003, and 60721002, the National Fundamental Research Program of China (2010CB327903), the Jiangsu Science Foundation (BK2009081), and the Jiangsu 333 High-Level Talent Cultivation Program. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Zhu Liu.

Y. Fu, Y. Zhu, C. Song, and Z.-H. Zhou are with the National Key Lab for Novel Software Technology, Nanjing University, Nanjing 210093, China (e-mail: ztwztq2006@gmail.com; yszhu@cs.hku.hk; chmsong@graphics.nju.edu.cn; zhouzh@nju.edu.cn).

Y. Guo is with the National Key Lab for Novel Software Technology, Nanjing University, Nanjing 210093, China, and also with the Jiangyin Information Technology Research Institute of Nanjing University (e-mail: ywguo@nju.edu.cn).

F. Liu is with the Department of Computer Sciences, University of Wisconsin-Madison, Madison, WI 53562 USA (e-mail: fliu@cs.wisc.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2010.2052025

the edge weights can qualitatively measure similarities among shots. We associate the value of a graph node with shot importance computed by a Gaussian entropy fusion scheme. Such a scheme can calculate the importance of shots in the presence of brightness difference and conspicuous noises, by emphasizing useful information and precluding redundancy among video features. With the graph representation, the final summary is generated through the event clustering based on random walks and a multi-objective optimization process.

To the best of our knowledge, this paper presents the first multi-view video summarization method. It has the following features.

- A spatio-temporal shot graph is used for the representation of multi-view videos. Such a representation makes the multi-view summarization problem tractable in the light of graph theory. The shot graph is derived from a hypergraph which embeds different correlations among video shots within each view as well as across multiple views.
- Random walks are used to cluster the event-centered shot clusters, and the final summary is generated by multi-objective optimization. The multi-objective optimization can be flexibly configured to meet different summarization requirements. Additionally, multi-level summaries can be achieved easily through setting different parameters. In contrast, most previous methods can only summarize the videos from a specific perspective on the summaries.
- The multi-view video storyboard and the event-board are presented for representing multi-view video summary. The storyboard naturally reflects correlations among multi-view summarized shots that describe the same important event. The event-board serially assembles event-centered multi-view shots in temporal order. With the event-board, a single video summary that facilitates quick browsing of the summarized video can be easily generated.

The rest of this paper is organized as follows. We briefly review previous work in Section II. In Section III, we present a high-level overview of our method. The two key components of our method, spatio-temporal shot graph construction and multi-view summarization, are presented in Sections IV and V, respectively. We evaluate our method in Section VI and conclude the paper in Section VII.

II. Related Work

This paper is made possible by many inspirations from previous work on video summarization. A comprehensive review of the state-of-the-art video summarization methods can be found in [21]. In general, two basic forms of video summaries exist, i.e., the static key frames and dynamic video skim. The former consists of a collection of salient images fetched from the original video sequence, while the latter is composed of the most representative video segments extracted from the video source.

Key frame extraction should take into account the underlying dynamics of video content. DeMenthon *et al.* [1] regarded video sequence as a curve in high-dimensional space. The curve is recursively simplified with a tree structure representation. The frames corresponding to junctions between curve segments at different tree levels are viewed as key frames. Hanjalic *et al.* [3] divided video sequence into clusters and selected optimal ones using an unsupervised procedure for cluster-validity analysis. The centroids of clusters are chosen as key frames. Li *et al.* [4] formulated key frame extraction as a rate-distortion Min-max optimization problem. The optimal solution is solved by dynamic programming. Besides, Orriols *et al.* [5] addressed summarization under a Bayesian framework. An EM algorithm with a generative model is developed to generate representative frames. Note that key frames can be transformed into skim by joining up the segments that enclose them, and vice versa.

In contrast to key frames, an advantage of video skim is that signals in other modalities such as audio information can be included. Furthermore, skim preserves the time-evolving nature of the original video, making it more interesting and impressive. Video saliency is necessary for summarization to produce the representative skim. For static image, Ma *et al.* [22] calculated visual feature contrast as saliency. A normalized saliency value for each pixel is computed. To evaluate saliency of video sequence, multi-modal features such as motion vector and audio frequency should be considered [11], [16], [19]. Ma *et al.* [11] presented a generic framework of user attention model through multiple sensory perceptions. Visual and aural attentions are fused into an attention curve, based on which key frames and video skims are extracted around the crests. Recently, You *et al.* [19] also introduced a method for human perception analysis by combining motion, contrast, special scenes, and statistical rhythm cues. They constructed a perception curve for labeling three-level summary, namely, keywords, key frames, and video skim.

Various mechanisms have been used to generate video skim. Nam *et al.* [12] proposed to adaptively sample the video with visual activity-based sampling rate. Semantically meaningful summaries are achieved through an event-oriented abstraction. By measuring shots' visual complexity and analyzing speech data, Sundaram *et al.* [17] generated audio-visual skims with constrained utility maximization that maximizes information content and coherence. Since summarization can be viewed as a dimension reduction problem, Gong and Liu proposed to summarize video by using singular value decomposition (SVD) [9]. The SVD properties they derived help to output the skim with user-specified length. Gong's another method [8] produces video summary by minimizing visual content redundancy of the input video. Previous viewers' browsing log will assist in future viewers. Yu *et al.*'s method [20] learns user understanding of video content. A ShotRank is constructed to measure importance of video shot. The top ranking shots are chosen as video skim.

Some techniques for generating video skims are domain-dependent. For example, Babaguchi [7] presented an approach for abstracting soccer game videos by highlights. Using event-based indexing, an abstracted video clip is automatically created based on impact factors of events. Soccer events can be detected by using temporal logic models [23] or goalmouth detection [24]. Much attention has been paid to rush video summarization [25]–[27]. Rush videos often contain redundant and repetitive contents, by exploring which a concise summary can be generated. The methods in [15] and [18] focus on summarizing music videos via the analysis of audio, visual, and text. The summary is generated based on the alignment of boundaries of the chorus, shot class, and repeated lyrics of the music video.

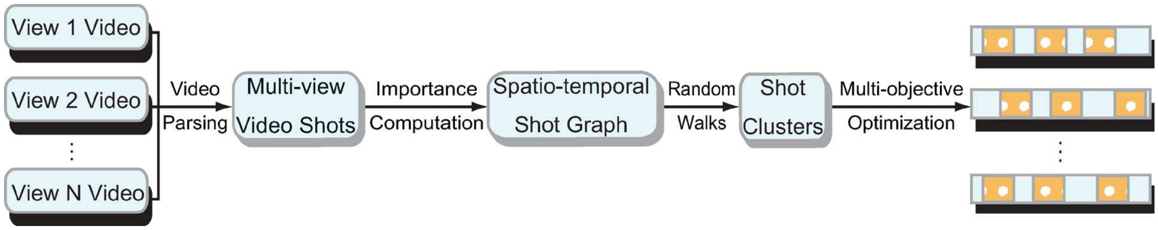


Fig. 1. Overview of our multi-view video summarization method.

Besides, automatic music summarization has been considered in [28].

Graph model has also been used for video summarization. Lu *et al.* [10] developed a graph optimization method that computes optimal video skim in each scene via dynamic programming. Ngo *et al.* [13] used temporal graph analysis to effectively capsulate information for video structure and highlight. Through modeling the video evolution by temporal graph, their method can automatically detect scene changes and generate summaries. Lee *et al.* [29] presented a scenario-based dynamic video abstraction method using graph matching. Multi-level scenarios generated by a graph-based video segmentation and a hierarchical segment are used to segment a video into shots. Dynamic video abstractions are accomplished by accessing the hierarchy level-by-level. Another graph-based video summarization method is given by Peng and Ngo [14]. Highlighted events can be detected by a graph clustering algorithm, incorporating an effective similarity metric of video clips. Comparing with their methods, we focus on multi-view videos. Due to content correlations among multi-views, the spatio-temporal shot graph we constructed has more complicated node connections, making summarization challenging.

The above methods provide many effective solutions to mono-view video summarization. However, to the best of our knowledge, few methods are dedicated to multi-view video summarization. Multi-view video coding (MVC) algorithms [30]–[32] also deal with the multi-view videos. Using techniques such as motion estimation, disparity estimation, and so on, MVC removes information redundancy in spatial and temporal domains. The video content is however unchanged. Therefore, MVC could not remove redundancy at the semantic level. In contrast, our multi-view video summarization method makes an effort to pave the way for this, by exploring the content correlations among multi-view video shots and selecting those most representative shots for summary.

III. Overview

We construct a spatio-temporal shot graph to represent the multi-view videos. Multi-view summarization is achieved through event-centered shot clustering via random walks and multi-objective optimization. Spatio-temporal shot graph construction and the multi-view summarization are the two key components. The overview of our method is shown in Fig. 1.

To construct the shot graph, we first parse the input multi-view videos into content-consistent video shots. Dynamic and important static shots are reserved as a result. The preserved shots are used as graph nodes and the corresponding shot importance values are used as node values. For evaluating the importance, a Gaussian entropy fusion model is developed to fuse together a set of intrinsic video features. The multi-view shots usually have diverse correlations with different attributes, such as temporal adjacency and content similarity. We use a hypergraph to systematically characterize the correlations among shots. A hypergraph is a graph in which an edge, usually named as a hyperedge, can link a subset of nodes. Each kind of correlation among multi-view shots is thus represented with a kind of hyperedge in the hypergraph. The hypergraph is further converted into a spatio-temporal shot graph where correlations of shots in each view and across multi-views are mapped to edge weights.

To implement multi-view summarization on the spatio-temporal graph, we employ random walks to cluster those event-centered similar shots. Using them as the anchor points, final summarized multi-view shots are generated by a multi-objective optimization model that supports different user requirements as well as multi-level summarization.

We use the multi-view video storyboard and the event-board to represent the multi-view summaries. The multi-view storyboard demonstrates the event-centered summarized shots in a multi-view manner as shown in Fig. 5. In contrast, the event-board shown in Fig. 6 assembles those summarized shots along the timeline.

IV. Spatio-Temporal Shot Graph

It is difficult to directly generate summarization, especially the video skims from multi-view videos. A common idea is to first parse the videos into shots. In this way, video summarization is transformed into a problem of selecting a set of representative shots. Obviously, the selected shots should favor interesting events. Meanwhile, these shots should be nontrivial. To achieve this, content correlations as well as disparities among shots are taken into account. In previous methods for mono-view video summarization, each shot only correlates with its similar shots along the temporal axis. The correlations are simple, and easily modeled. However, for the multi-view videos, each shot correlates closely with not only the temporally adjacent shots in its own view but also the spatially neighboring shots in other views. Relationships among shots increase exponentially relative to the mono-view video, and the correlations are thus very complicated. To better explore such correlations, we consider them with different attributes, for instance, temporal adjacency, content similarity, and high-level semantic correlation separately. A hypergraph is initially introduced to systematically model the correlations in which each graph node denotes a shot resulting from video parsing, while each type of hyperedge characterizes the relationship among shots. We then transform the hypergraph into a weighted spatio-temporal shot graph. The weights on graph edges thus qualitatively evaluate correlations among multi-view shots.

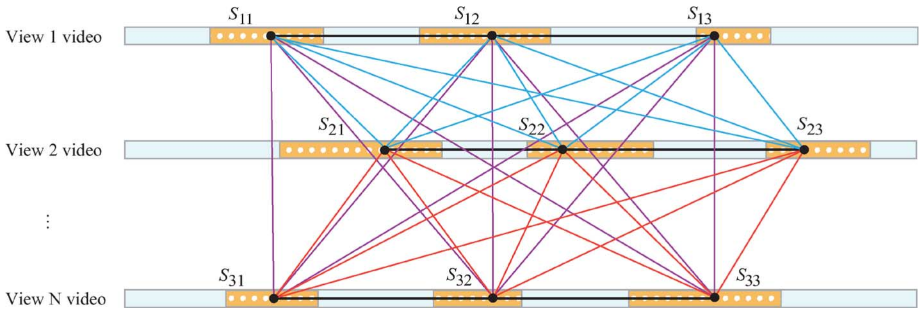


Fig. 2. Spatio-temporal shot graph $G(V, E, W)$. Each node in G represents a shot, and its value is the shot importance. Each edge connects a pair of nodes (shots) with correlation which is evaluated by shots' similarity. Without losing generality, only three shots in each view are given for illustration. To expose clearly the graph, the edges in graph are with several colors, and each shot is represented as an orange segment.

A. *Graph Construction*

We first parse the multi-view videos into shots. Various algorithms have been proposed for shot detection [33]–[37]. In [34], Ngo *et al.* proposed an approach through the analysis of slices extracted by partitioning video and collecting temporal signature. It has proven effective in detecting camera breaks such as cuts, wipes, and dissolves. Xiang *et al.* [36] used a cumulative multi-event histogram over time to represent video content. An online segmentation algorithm named forward-backward relevance is developed to detect breaks in video content. For multi-view videos, especially those surveillance videos, the cameras remain nearly stable, and the videos recorded only contain the same scene in most cases. The shot mainly contains those temporally contiguous frames which share the same semantic concept with relatively higher probability. To detect the shots, we basically adopt the algorithm proposed in [36], and further discard those shots with lower activities. In particular, for every shot detected, we first compute the differential image sequence of adjacent frames. Each image can then be converted into a binary image by comparing the absolute value of each pixel against a threshold. We compute for each shot a normalized activity value through counting the total number of its nonzero pixels and dividing it by the product of frame number and frame resolution. We sort the activity values of all shots and select the activity threshold interactively.

Parsing the multi-view videos into shots allows us to seek solution of summarization in a more compact shot space. Actually, each shot correlates with the similar shots in its own view as well as the ones in other views. This characteristic makes the weighted graph a suitable representation of multi-view videos, by viewing shots as nodes and converting the correlations between shots into edge weights. We extend the graph model for mono-view video summarization [10], [13], [14] and segmentation [38] to a spatio-temporal shot graph. Connectivity of the graph we constructed is inherently complicated due to the spatio-temporal correlations among multi-view shots.

The multi-view videos are treated as a weighted undirected shot graph $G(V, E, W)$ as illustrated in Fig. 2. Each node in V represents a shot resulting from video parsing. Its value is the importance of shot calculated by the Gaussian entropy fusion model. The edge set E connects every pair of nodes if they are closely correlated. The edge weight W measures node similarity by taking into account their correlations in terms of different attributes. We model such correlations among shots with a hypergraph in which each type of hyperedge denotes a kind of correlation. By converting the hypergraph into the spaito-temporal graph, the edge weights quantitatively evaluate correlations among shots. Note that the shot graph is called a spatio-temporal graph in the sense that it embeds the scene information coming from different spatial views. The "spatio-temporal" here differs from its traditional definition on the monocular video sequence.

By representing the multi-view videos as the spatio-temporal shot graph, correlations among shots are naturally and intuitively reflected in the graph. Moreover, the graph nodes carry shot importance, which is necessary to create a concise and representative summary. We describe the Gaussian entropy fusion model and hypergraph in Sections IV-B and IV-C separately.

B. *Shot Importance Computation*

By representing multi-view videos with graph, multi-view video summarization is converted into a task of selecting the most representative video shots. The selection of representative shots often varies with different people. In this sense, detecting representative shots generally involves understanding video content based on human perception and is very difficult. To make it computationally tractable, we instead quantitatively evaluate the shot importance by considering low-level image features as well as high-level semantics. We introduce a Gaussian entropy fusion model to fuse a set of low-level features such as color histogram and wavelet coefficients, and compute an importance score. For high-level semantics, we mainly consider human faces now. Moreover, we take into account the interesting events for specific types of videos, since video summarization is often domain-specific.

1) Shot Importance by Low-Level Features: We develop a Gaussian entropy fusion model to measure shot information by integrating low-level features. In contrast, previous mono-view video summarization methods generally combine features with linear or nonlinear fusion schemes. Such schemes would not necessarily lead to the optimal performance for our multi-view videos when the videos are contaminated by noises. This is especially true for multi-view surveillance videos which often suffer from different lighting conditions across multiple views. Under such circumstance, we should robustly and fairly evaluate the importance of the shots that may capture the same interesting event in multiple views under different illuminations. To account for this, we need to emphasize the portion of shot-related useful information in multi-view videos, and depress the influence of noises simultaneously. Based upon such observation, we first extract from the videos a set of intrinsic low-level features which are often correlated with each other.

We now mainly take into account the visual features. They are color histogram feature, edge histogram feature, and wavelet feature [9], [39], [40]. The features in other modalities, such as textual and aural features used in previous video analysis methods [11], [19], however, can also be integrated into our method. Without losing generality, for shot S with n frames, suppose that overall M feature vector sets are extracted. We expand each feature F_i into a one-column vector. Two arbitrary features F_i and F_j may have different dimensions. We denote the feature sets by $\{F_i\}_{i=1}^{M}$.

The feature sets contain shot-related useful information. Besides, they are often contaminated by noises. The Gaussian entropy fusion model aims at emphasizing the useful information of feature sets and simultaneously minimizing noise influence. We can relatively safely assume that different features have uncorrelated noise characteristics. The interaction of feature sets is shot-related information expressed as

$$I(S) \approx \sum_{i=1}^{M} I(F_i) - I\left(\overset{M}{\underset{i=1}{\cup}} F_i\right). \tag{1}$$

In the above formula, importance of shot S is measured by adding up information amount of the individual features and subtracting information amount of their union. Since noises contained in different features are uncorrelated, the above formula weakens noise influence and the useful information is emphasized. According to information theory, a measure of the amount of information is entropy. We then add up the entropy values of all feature sets and subtract the entropy of their union from the sum

$$H(S) = \sum_{i=1}^{M} H(F_i) - H(F_1, F_2, \ldots, F_M) \tag{2}$$

where $F_i = (f_{i,1}, f_{i,2}, \ldots, f_{i,n})^T$. f_{ij} is the ith feature set for the jth frame of shot S. $H(\cdot)$ denotes entropy of the feature.

To estimate the probability of $p(F_i)$ and $p(F_1, F_2, \ldots, F_M)$, a common idea is to approximate them with the Gaussian distribution

$$p(F_i) \sim \mathcal{N}(\mathbf{0}, \Sigma^i) \tag{3}$$

$$p(F_1, F_2, \ldots, F_M) \sim \mathcal{N}(\mathbf{0}, \Sigma) \tag{4}$$

where Σ^i is the covariance matrix of $\{f_{i,j}\}_{j=1}^{n}$ $(i = 1, \ldots, M)$, and Σ is the one of $\{f_{i,j}\}_{i=1}^{M}$. $F_1, F_2, \ldots, F_M$ are normalized by

$$f_{i,j}^{*} = \frac{f_{i,j} - \frac{1}{n}\sum\limits_{j=1}^{n} f_{i,j}}{\sqrt{\frac{1}{n}\sum\limits_{j=1}^{n}\left(f_{i,j} - \frac{1}{n}\sum\limits_{j=1}^{n} f_{i,j}\right)^2}}. \tag{5}$$

By virtual of nonlinear time series analysis [41], the Gaussian entropy of shot S is finally expressed as

$$H(S) = \frac{1}{2}\sum_{j=1}^{n} \log_2(\Sigma_{jj}) - \frac{1}{2}\log_2 |\Sigma| \tag{6}$$

where Σ_{jj} is the jth element in the diagonal of matrix Σ. $|\Sigma| = \prod_{j=1}^{n} \lambda_j$. λ_j is eigenvalue of Σ.

The entropy H is a measure of information encoded by the shot S. We take it as the importance. An additional advantage of the Gaussian entropy fusion scheme is that it works well as long as the union of feature vector groups covers most useful information of multi-view videos. Therefore, instead of using all the feature sets, it would be sufficient if some well-defined feature sets are available.

2) Shot Importance by High-Level Semantics: Humans are usually important content in video sequence. We employ the Viola-Jones face detector [42] to detect faces in each frame. In addition, video summarization is often domain-specific. Definition of shot importance may vary according to different video genres. For instance, in a baseball game video, the shots that contain "home run", "catch", and "hit" usually catch much user attention. Many methods have been suggested to detect interesting events for specific type videos, such as abnormal detection [43] in surveillance video, excitement and interestingness detection in sports video [7], [23], [24], brilliant music detection [28], and so on. A detailed description of these methods is beyond the scope of this paper. However, for specific type of multi-view videos, interesting event detection can be integrated into our method. For those shots that contain faces in most frames or interest events, the importance scores are set to 1.

C. Correlations Among Shots by Hypergraph

Basically, three kinds of relationships among multi-view shots are considered:

- *Temporal adjacency*. Two shots are likely to describe the same event if one shot is temporally adjacent to the other.
- *Visual similarity*. Two shots are related to each other if they are visually similar.
- *Semantic correlation*. Two shots may correlate with each other due to the same event or semantic object such as a face occurs in both shots.

Temporal adjacency implies that adjacent video shots may share the same semantic concepts with relatively higher probability. For two shots S_i and S_j, the temporal similarity is defined as

$$W_t(S_i, S_j) = \frac{1}{\alpha_1 + \alpha_2 \cdot d + \alpha_3 \cdot d^2} \tag{7}$$

where $d = |t_i - t_j|$ computes the temporal distance. t_i and t_j are the time stamp of their middle frames. d is further integrated into a light attenuation function, in which the parameters α_1, α_2, and α_3 control the temporal similarity. We use the following ways to set their values. Given the training shots parsed from a mono-view video, we first compute temporal similarities of each shot pair given initial values. Then we modify the values until the temporal similarities computed are in accordance with our observation. Through experiments, α_1, α_2, and α_3 are set as 1, 0.01, and 0.0001, respectively. Different settings of the three parameters will have the same effect on summarization if the values are given with regard to an invariant relative magnitude. We use similar ways to set other parameters in similarity computation.

Visual similarity is computed by

$$W_v(S_i, S_j) = e^{-k*VisSim(S_i,S_j)} \tag{8}$$

where k is a control parameter set to 0.1. For computational efficiency, we select three frames, namely, the first, middle, and last, from S_i and S_j separately and calculate the visual similarity according to their color histogram H_C and edge histogram H_E [40] distances

$$VisSim(S_i, S_j) = w \cdot |H_C(S_i) - H_C(S_j)| + |H_E(S_i) - H_E(S_j)|. \tag{9}$$

The use of edge histogram weakens the influence of lighting difference across multi-view shots. w here is a weight that is empirically set to 0.5. For specific domains, W_v could be modified to accommodate more complex texture information, as well as motion features.

Semantic correlation of video shots is often related to the occurrences of specific events. Besides, it varies with different video genres. For instance, for surveillance videos, there is a definite correlation between two shots if the same human face is detected in both shots. However, for football game videos, there exists a strong correlation among the shots that record all the goals in a match. Since a comprehensive study of semantic correlation is beyond the scope of this paper, in our current implementation, we allow the user to interactively specify semantically correlated video shots. Semantic correlation value W_s of two correlated shots S_i and S_j is set to 1.

To measure the total similarity W of shots S_i and S_j, a straightforward way is to fuse together the above similarity values with certain weights and to construct the spatio-temporal graph directly. Such a scheme, however, may destroy the original relationship once the fusion weights could not be set properly. For instance, two shots with large temporal distance visually resemble each other, imaging a people who repeats his actions at a 24-h interval in the same scene. A strong correlation between the two shots should exist. Nevertheless, improper weights will make W too small and negligible. A natural way to remedy the flaw occurring above is to represent the correlations among multi-view shots as a hypergraph. A hypergraph is a graph in which an edge can connect more than two nodes. This edge is named as a hyperedge [44] which often links a subset of nodes. Obviously, in this sense, an ordinary graph is a special kind of hypergraph.

In our hypergraph, the nodes just represent video shots. To construct the hyperedges, we build for each relationship, i.e., temporal adjacency, visual similarity as well as semantic correlation, an ordinary graph and apply graph clustering algorithm to the nodes. All the nodes in a cluster are then linked by a hyperedge in the hypergraph. Note that two cluster may overlap as for each relationship the clustering algorithm is performed. The weight on the hyperedge is the average of relation values of all pairs of nodes in the same cluster.

Generally, there are two methods to transform a hypergraph into a general graph. One is directly using the hypergraph partition algorithm such as normalized hypergraph cut [45]. The other seeks solution through clique expansion or star expansion [46]. We employ clique expansion to convert the hypergraph into the spatio-temporal graph. By clique expansion, each hyperedge is expanded into a clique, in which the weight on each pair of nodes is taken as the weight on the hyperedge. On the spatial temporal shot graph, edge weight W is the sum of edge weights derived from those cliques to which the edge belongs. In addition, to further simplify the graph, the edge weight W is set to zero if it is smaller than a predefined threshold.

V. Multi-View Summarization

The spatio-temporal shot graph is a suitable representation of multi-view video structure, since it carries shot information and meanwhile reflects intuitively correlations among shots. Due to the correlations among multi-view shots, the shot graph has complicated connectivity. This makes the summarization task challenging. We must generate the most representative graph nodes (shots) by taking into consideration the connections as well as users' requirements. Our basic observation is that, with the shot graph, the multi-view video summarization can be formulated as a graph labeling problem. We accomplish this in two steps. We first cluster those event-centered similar shots, and pick out the candidates for summarization by random walks. Final summary is generated by a multi-objective optimization process that is specifically devised for accommodating different user requirements.

A. Shot Clustering by Random Walks

To cluster similar shots, we first sample a small number of important shots and then cluster the shots of the same events by random walks. We adopt random walks in this step rather than other graph partition algorithms such as graph cut because of the following reasons.

On one hand, random walks hs proven to be effective in handling large and complex graphs, even in the presence of conspicuous noises. It is thus suitable for our clustering task which needs to partition the spatio-temporal shot graph with complicated node connections. Graph cut, however, is prone to the small cut and noise influence [47].

On the other hand, our graph partition is a K-way segmentation problem given sampled shots indicating seeds for candidate clusters. Random walks algorithm works well for such problem. The random walker starts from each unsampled node (shot) and determines for it the most preferable sampled shot's cluster. The final clusters thus obtained are actually event-centered. In general, many events can be represented as object activities and interactions, showing different motion patterns [48]. For the event captured by multi-view shots, similarities among shots in terms of visual, temporal, as well as semantic correlations should be large. In addition, each event may have at least a central shot which has a high shot importance. We can take it as one of the best views recording this event. The random walks-based shot clustering fulfills these requirements in that we select the shots with higher importance as seeded nodes. Such shots just can be viewed as the centers of events. Furthermore, the weight on graph is defined in form of shots' similarities which makes clustering event relevant shots possible. Notice that the property of our event-centered clustering also facilitates video retrieval by allowing the user to specify their interested shots as seeds. The final clusters containing seeds are thus the retrieval results.

Although a detailed description of random walks theory is beyond the scope of this paper, it essentially works as follows.

First, we partition the node set V into seeded nodes V_S and unseeded nodes V_U, satisfying that the value of each seed in V_S exceeds an entropy threshold.

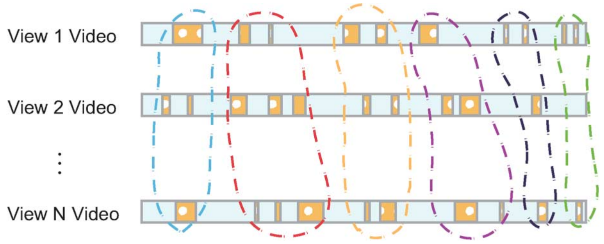


Fig. 3. Graph partition by random walks. Shot clusters generated by random walks are enclosed by dashed circles.

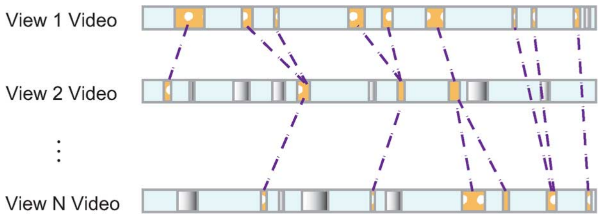


Fig. 4. Final video summary resulting from optimization. Dashed lines connect the shots that are reserved in the same shot cluster.

We then define the combinatorial Laplacian matrix for graph as follows:

$$L_{ij} = \begin{cases} \sum_j W(S_i, S_j), & \text{if } i = j \\ -W(S_i, S_j), & \text{if } S_i, S_j \text{ are linked by edge} \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

L is an $N_n \times N_n$ dimensional sparse, symmetric, and positive definite matrix, where N_n is the number of nodes in graph.

We further decompose L into blocks corresponding to nodes in V_S and V_U separately as

$$L = \begin{bmatrix} L_S & B \\ B^T & L_U \end{bmatrix}. \quad (11)$$

For each unseeded node, the final determination to which seeded cluster it belongs to is made by solving

$$L_U X_U = -B^T X_S \quad (12)$$

where X_U represents the probabilities that unseeded nodes belong to seeded nodes' clusters. X_S denotes the matrix that marks the cluster category of seeded nodes. We use a Conjugate Gradient algorithm [49] to solve the linear formulation of random walks, which runs very fast.

In the end, to favor important events with long duration, we filter out trivial shot clusters with low entropy values. Furthermore, the two clusters whose similarity exceeds a given threshold are merged together. The remainder shot clusters are used as candidates for summarization in multi-objective optimization (Fig. 3).

B. Multi-Objective Optimization

Users normally have various requirements over summarization, according to different kinds of application scenarios. In general, a good summary should achieve the following goals simultaneously. 1) Minimize shot number. The retrieval application of summary requires that a small number of shots should be generated. 2) Minimize summary length. The minimum length of summary would be of great help to video storage. 3) Maximize information coverage. To achieve enough information coverage, the sum of resulting shots' entropy value in each cluster must exceed a certain threshold. 4) Maximize shot correlation. It would be much better if shots in every resulting cluster strongly correlate with each other. This yields the most representative shots for the interesting event.

To meet the above requirements, we design a multi-objective optimization model to generate final summary. The optimization follows the complexity incompatibility principle [50]. We formulate the summarization as a graph labeling problem. For the shot cluster C_S with n_s shots, the decision whether or not the shots should be in the summary is denoted by $x = (x_1, x_2, \ldots, x_{n_s}), \forall x \in X$. X is the 0/1 solution space in which $x_i = 1$ stands for reserved shot and 0 stands for unreserved one (Fig. 4).

The multi-objective optimization function is given by

$$\max\{-f_1(x), -f_2(x), f_3(x), f_4(x)\} \; s.t. \begin{cases} g(x) \le D_{max} \\ h(x) \ge R_{min} \end{cases} \quad (13)$$

where $f_1(x) = \sum_{i=1}^{n_s} x_i$, $f_2(x) = \sum_{i=1}^{n_s} D_i \cdot x_i, D_i > 0$, $f_3(x) = \sum_{i=1}^{n_s} R_i \cdot x_i$, and $f_4(x) = (1/2) \cdot \sum_{i,j=1, i \neq j}^{n_s} W(S_i, S_j) \cdot x_i \cdot x_j$.

f_1, f_2, f_3, and f_4 denote the total shot number, summary length, information coverage, and shot correlation within cluster, respectively. D_i and R_i are length and importance of shot i separately. $g(x)$ and $h(x)$ are defined in forms of fuzzy set [51]

$$g(x) = \mu\left(f_2(x)\right), \quad h(x) = \mu\left(f_3(x)\right)$$

with $\mu(f_i(x)) = [f_i(x) - \inf f_i(x)]/[\sup f_i(x) - \inf f_i(x)]$.

D_{max} is the maximum allocated length of one cluster. R_{min} is the minimum information entropy of C_S. They are defined as

$$D_{max} = \lambda_1 \cdot D, \quad R_{min} = \lambda_2 \cdot R$$

where D and R are the total length of shots in C_S and the sum of importance values, respectively. λ_1 and λ_2 are the parameters that control summary granularity. The two constraints mean that the total length of shots in C_S after optimization should be less than D_{max}, whereas the entropy should be greater than R_{min}. We will show in experiments, by flexibly configuring λ_1 and λ_2, multi-level summarization can be easily achieved.

We further define the minimum function

$$u\left(F(x)\right) = \min_{1 \le i \le 4} \left\{\eta_i \mu\left(f_i(x)\right)\right\} \quad (14)$$

in which $F(x) = (\mu(f_1(x)), \mu(f_2(x)), \mu(f_3(x)), \mu(f_4(x)))^T$. $\eta_{i,i=1,\ldots,4}$ are coefficients that control the weights of objective functions satisfying $\sum_{i=1}^{4} \eta_i = 1$ and $\eta_i \ge 0$. They can be configured according to different user requirements.

By employing the Max-Min method, the multi-objective optimization is transformed into the following 0-1 mixed integer programming problem:

$$x^* = \operatorname*{argmax}_{x \in X} u\left(F(x)\right) \; s.t. \; A \cdot F \le \begin{pmatrix} D_{max} \\ -R_{min} \\ -u(F) \end{pmatrix} \quad (15)$$

TABLE I
DETAILS OF MULTI-VIEW VIDEOS AND SUMMARIES

Multi-view Videos	No. of Views	Video Length (Mins.)	Levels of Summary	Level	Summary Length (Mins.)	λ_2 Info. Reserved (%)
office1	4	11:16/8:43/11:22/14:58	1	Level 1	1:53	70
campus	4	15:19/13:51/12:30/15:03	1	Level 1	4:02	60
office lobby	3	08:14/08:14/08:14	2	Level 1	2:56	60
				Level 2	5:14	70
road	3	5:11/8:49/8:46	2	Level 1	2:21	60
				Level 2	4:28	70
badminton	3	5:07/5:00/5:00	3	Level 1	0:50	60
				Level 2	1:08	65
				Level 3	2:08	70

with $A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -1 & -1 & -1 & -1 \end{pmatrix}$. x^* is the final optimization result to be solved.

This integer programming is a typical knapsack problem in combinatorial optimization. We use a pseudo-polynomial time dynamic programming algorithm [52] to solve it. The algorithm runs fast for all of our experiments.

VI. EXPERIMENTS

We conducted experiments on several multi-view videos, including typical indoor and outdoor environments. The office1, campus, office lobby, and road videos are typical surveillance videos, since surveillance videos are one of the most important multi-view video types. Some multi-view videos are semi-synchronous or nonsynchronous. Most multi-view videos are captured by three or four ordinary cameras with overall 360 degree coverage of the scene. To further verify our method, we also deliberately shoot an outdoor scene by four cameras with only 180 degree coverage. Note that all of the videos are captured using the web cameras or handheld ordinary video cameras by nonspecialists, making some of them unstable and obscure. Moreover, some videos have quite different brightness across multi-views. These issues pose great challenges to the multi-view video summarization.

Table I shows the information on experimental data. All experimental results were collected in a PC equipped with P4 3.0-GHZ CPU and 1 GB of memory. The multi-view videos as well as summaries can be found in the demo page http://cs.nju.edu.cn/ywguo/summarization.html.

Note that we sacrifice the visual quality of original multi-view videos to meet the space limitation of online storage by compressing them with high compression ratios.

Display of multi-view summary. We employ here the *multi-view storyboard* to represent the multi-view video summary, as illustrated in Fig. 5. The storyboard naturally reflects spatial and temporal information of the resulting shots as well as their correlations, allowing the user to walk through and analyze the summarized shots in a natural and intuitive way. In the storyboard, each shot in summary is represented by its middle frame. By clicking on the yellow block highlighted with corresponding shot number, the user can browse the summarized video shot. Dashed lines connect those shots of the same scene-event derived from random walks clustering and multi-objective optimization. By means of the multi-view storyboard, we further introduce an *event-board* to display the multi-view summary as illustrated in Fig. 6. The summarized shots are assembled along the timeline across multi-views. Each shot is represented with a box and the number in box illustrates the view to which the shot belongs. Dashed blue boxes represent those events that are recorded by more than one shot or different views. By clicking on the boxes, the shots can be displayed. Obviously, through the event-board, we can easily generate *a single video summary* that includes all the summarized shots. We show some examples of the single video summary in our demo page. One of its advantages over storyboard is that it allows the rapid browse of summarized result. If the user needs to browse the summary within limited time, the single summary would be a good choice.

A distinct characteristic of the multi-view videos is that the events are captured with overlapping across multiple views. To generate a compact yet highly informative summary, it is usually important to summarize a certain event only in the most informative view, and to avoid repetitive summary. This is especially true if the user only hopes to obtain a short length video summary. Our method realizes this. One example is shown in the summary of multi-view office1 videos. In the 24th shot, the girl who opened the door and went to her cube is only reserved in the second view, although she appeared in four views simultaneously. The man who opened the door in the 24th shot and left the room in the 25th shot is only reserved in the second view. In this sense, our method can be applied to the selection of optimal views. In addition, the method supports summarizing the same event using temporally successive multi-view shots. The event is recorded by the shots describing it with the best views in its duration.

On the other hand, it is also reasonable to produce a multi-view summary for the same event. For example, for a traffic accident, all videos in multi-views are often crucial in responsibility identification and verification. Our method handles this case successfully. In the multi-view office1 videos, three guys intruded the views and left the room. This action is reserved simultaneously in the 22nd shot of the second view and 40th shot of the fourth view. Other typical examples are the 28th and 35th shots, 30th and 46th shots, and 38th and 49th shots. Such summaries are attributed to two points. First, the shot importance computation algorithm fairly computes the importance of multi-view shots, even in the presence of brightness difference and noises. Second, the summarization method makes the most of correlations among multi-view shots.

Multi-level summarization can be conveniently achieved by our method. We only need to configure the two parameters λ_1 and λ_2 in multi-objective optimization. As aforementioned, λ_1

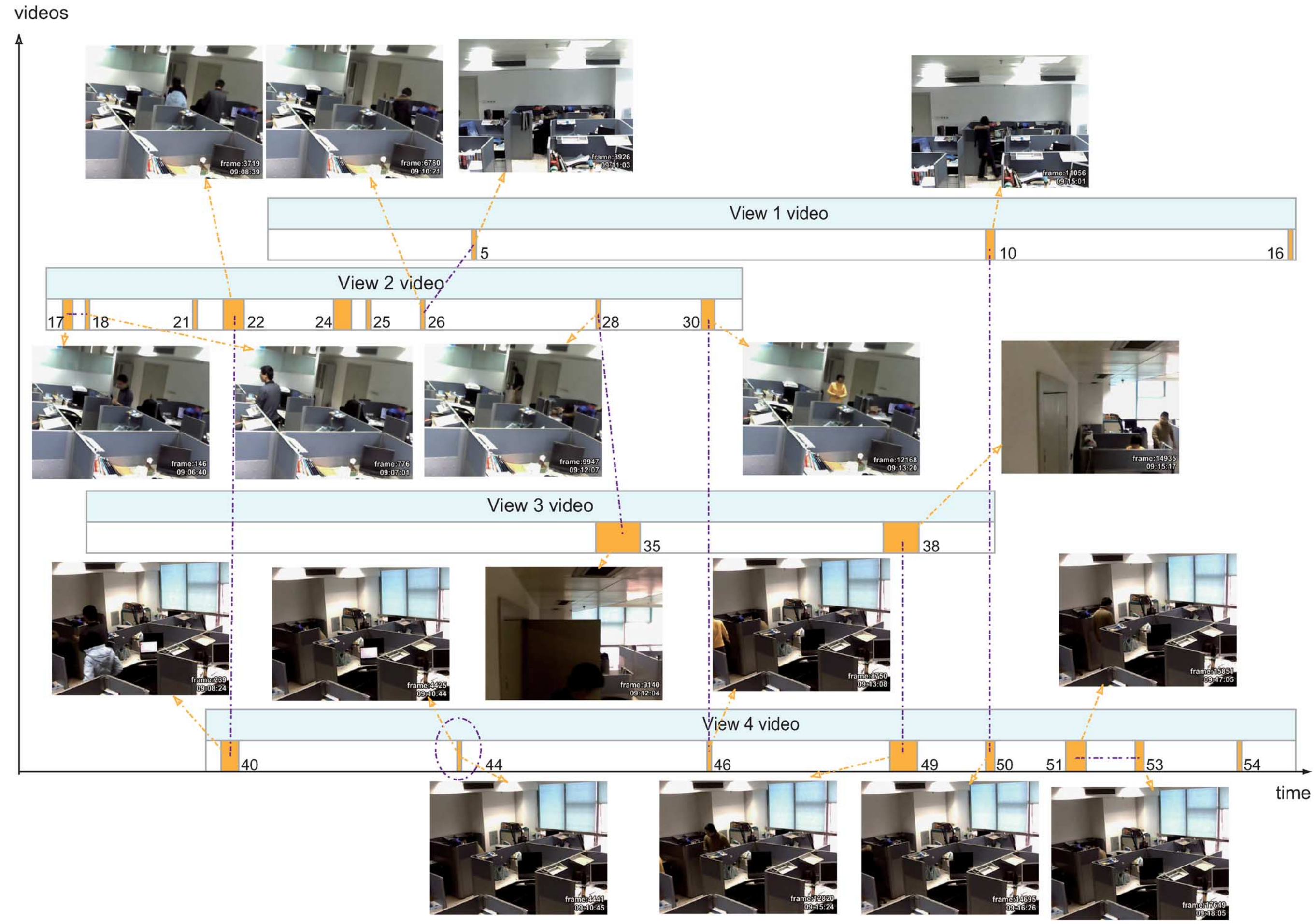


Fig. 5. Multi-view video storyboard. Without losing generality, the multi-view office1 videos with four views are given for illustration. The blue rectangles denote original multi-view videos. Each shot in summary is represented with a yellow box, by clicking on which the corresponding shot can be displayed. Each shot in summary is assigned a number indicating its order in those shots resulting from the video parsing process. Here, we give the numbers for the convenience of further discussion. Dashed lines connect those shots with strong correlations. The middle frames of a few resulting shots, which allow the quick browse of the summary, are demonstrated here.

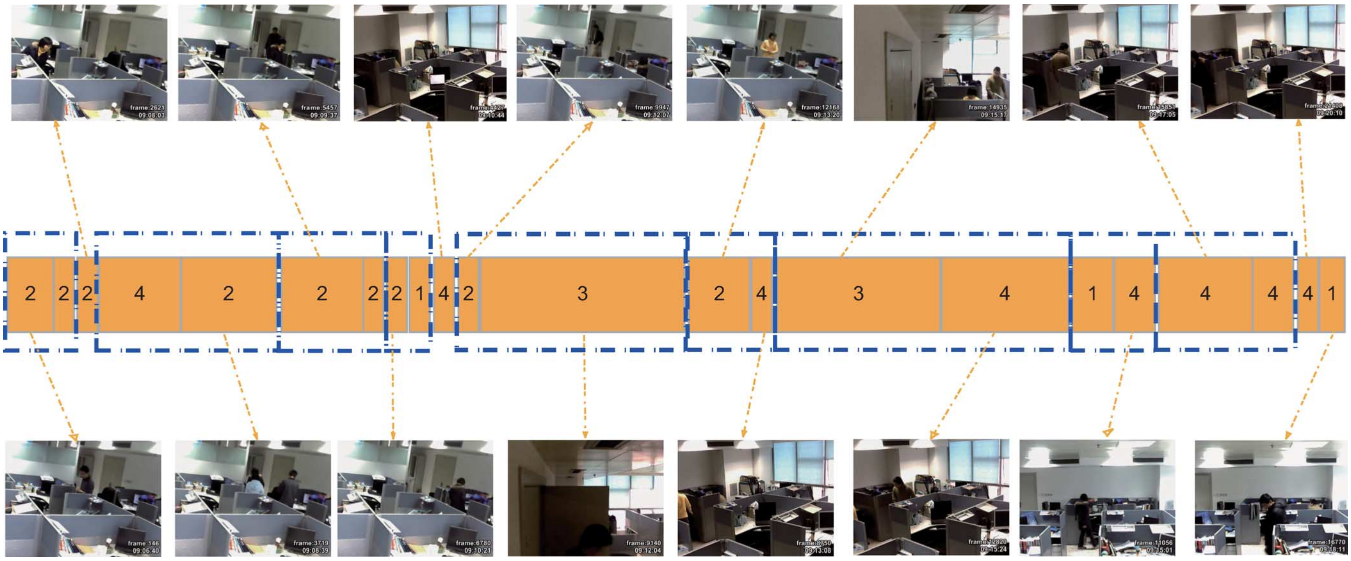


Fig. 6. Event-board assembles the event-centered summarized shots in temporal order. Each shot is represented with a box and the number in the box illustrates the view to which the shot belongs. Dashed blue boxes represent those events that are recorded by more than one shot or different views. By clicking on the boxes, the summarized shots can be displayed. Some representative frames, usually the middle frames of the shots, are showed for quick preview of the summary.

is integrated into the constraint that controls total length of summary. λ_2 is used to adjust information coverage. Increasing λ_1 and λ_2 simultaneously will generate a long and meanwhile informative summary.

The multi-view badminton videos are summarized into three levels, according to the length and information entropy set for the summary. The parameter λ_1 is set to 0.035, 0.075, and 0.15, respectively, on the 1st, 2nd, and 3rd level. λ_2 is set to 0.6, 0.65, and 0.7 accordingly. Obviously, the high-level summary covers most part of low-level summary, while reasonable disparity is due to the different optimization procedures involved. The low-level summary comprises the most highly repeated actions, such as serve, smash, and dead bird. Such statistics can be used for badminton training. The high level summary in contrast appends more amazing rally, e.g., the shots 67, 79, 124, 135, and 154 on level 3.

Other examples of multi-level summarization include the office lobby and road videos. We summarize both of them into two levels by setting λ_2 to 0.6 and 0.7, respectively. In general, the videos containing many events with different shot importance values are more suitable for multi-level summarization. For such videos, the low-level summary contains the shots which are enough to describe most of the original video events. The high-level compact summary, by contrast, comprises the events which are more active or salient.

There are some discussions about the choice of λ_1 and λ_2. Intuitively, λ_2 is used to control importance value of the summary. In our method, shot importance is evaluated by the entropy defined in terms of low-level features and updated by high-level semantics. The total entropy of those shots that are discarded for their lower activities is too low to be taken into account. Therefore, we can relatively safely assume that all reserved shots contain most information of multi-view videos. λ_2 thus can be regarded as the minimum percent information to be preserved in summary. In implementation, λ_2 is given by user. For λ_1, we try it from 0.05 to 1 with an increment of 0.05, and select the one ensuring a solution for (15) as λ_1.

Computational complexity of our method mainly depends on the lengths, resolutions, and activities of the multi-view videos. The major cost is spent on video parsing and graph construction, which take about 15 min for the office1 example. In contrast, summarization with random walks-based clustering and multi-objective optimization is fast. This step spends less than 1 min, since the graph constructed only has nearly 60 nodes. Video summarization is often used as a post-processing tool. Our method can be accelerated by high-performance computing system.

A. *Comparison With Mono-View Summarization*

We compare our method with previous mono-view video summarization methods. The summaries produced by our method and previous ones are shown in the demo webpage.

We implement the video summarization method presented in [11] and apply it to each view of the multi-view office1, campus, and office lobby videos. For each multi-view video, we combine the resulting shots along the timeline to form a single video summary. For a fair comparison, we also use the above method to summarize the single video formed by combining the multi-view videos along the timeline, and generate a dynamic single video summary. As the summary is extracted around crests of attention curve, the method does not provide a mechanism to remove content redundancy among multi-views. It is obvious that the summaries produced by the method contain much redundant information. There exist significant temporal overlaps among summarized multi-views shots. Most events are simultaneously recorded in the summaries.

By using our multi-view summarization method, such redundancy is largely reduced in contrast. Some events are recorded by the most informative summarized shots, while the most important events are reserved in multi-view summaries. Some events that are ignored by previous method—for instance the events recorded from 1st to 5th second, 14th to 18th second, and 39th to 41st second in our office1 single video summary—are reserved by our method in contrast. This is determined by our shot clustering algorithm and multi-objective optimization operated on the spatio-temporal shot graph. Such property of our method facilitates generating a short-length, yet highly informative summary.

We also compare our algorithm against a graph-based summarization method. A single video is first formed by combining the multi-view videos along the timeline. We then construct the graph according to the method given in [10]. Final summary is produced by using normalized cut-based event clustering and highlight detection [14]. Normalized cut widely employed by previous methods often suffers from the "small cut" problem. This can be problematic when the method uses heuristic criterion to select highlight from event clusters as summary. That is, some important events with short durations are missed. Our method, however, can meet different summarization objectives by using the multi-objective optimization. Important events with much higher importance are reserved in multi-views, while some important events with shot durations are preserved as well.

To quantitatively compare our method with previous ones, we use precision and recall to measure the performance. We invited five graduate students who remained unknown about our research to define the ground-truth video summaries. Each shot is labeled as a ground-truth shot only if the five guys agree with each other. For the office1 multi-view videos, totally 26 shots are labeled as ground-truth shots. The ground-truth summary of campus videos includes 29 shots. Precision and recall scores of the methods are shown in Table II. Accurately controlling the summary lengths is difficult. The summaries of different methods are all around 50 s, except the campus summary obtained by the graph method [10], [14] is 109 s. The second/sixth row is the data computed by applying the method [11] to each view video separately. The third/seventh row is generated by applying it to the single video formed by first combining each view. Generally, for the office1 multi-view videos, from the precision scores, summaries obtained by each method belong to the ground-truth. In contrast, precisions of the four methods computed on the campus videos are all around 50%. The campus videos contain many trivial events. It is challenging to generate an unbiased summary using the methods. The last column of the table indicates that our method is superior to others in terms of recall. This suggests that our method is more effective in removing content redundancy.

B. *User Study*

To further evaluate the effectiveness of our method, we have carried out a user study. The aim is to assess the enjoyability,

TABLE II
PERFORMANCE COMPARISON WITH PREVIOUS METHODS

Data	Method	Length of Summary (s)	Number of Events in Summary	Precision (%)	Recall (%)
office1	User attention method1 [11]	40	10	100	38
	User attention method2 [11]	55	12	100	46
	Graph method [10], [14]	64	7	100	26
	Multi-view method	55	16	100	61
campus	User attention method1 [11]	76	14	56	48.3
	User attention method2 [11]	35	8	40	27.6
	Graph method [10], [14]	109	14	50	48.3
	Multi-view method	42	16	69.6	55.2
office lobby	User attention method1 [11]	184	31	95	72.1
	User attention method2 [11]	179	30	100	69.8
	Graph method [10], [14]	201	25	100	58
	Multi-view method	176	33	100	76.7

informativeness, and usefulness of our multi-view video summary.

The study was conducted offline and online simultaneously. For the offline study, we invited 12 participants to take part in the study in our meeting room. All the participants are undergraduate students ranging in age from 16 to 22. To our knowledge, they remained unknown about our project. Each participant was shown the office1, badminton, campus, and road multi-view videos, together with their summaries. Summaries of badminton at three levels are all given. They were only asked to respond to the questions we raised. The online study was conducted similarly. Participants voluntarily responded to advertisements posted to mailing lists and were not compensated for their time. The link of the project webpage was opened to them. We obtained 23 responses to the office1 and badminton videos, and 27 responses to the campus and road videos from the graduate students ranging in age from 21 to 29.

The questions for evaluating our method are: Q1: How about the enjoyability of the video summary? Q2: Do you think the information encoded in the summary is reliable compared to the original multi-view videos. Q3: Will you prefer the summary to original multi-view videos if stored in your computer?

For Q1 and Q2, the participant was requested to assign two scores ranging from 0 to 100, whereas he/she only needed to respond to Q3 with "yes" or "no". Each participant was required to choose at least one from the office1 and badminton testing examples, and wrote the answers on the answer sheet.

We combine the offline and online answers together. For the usefulness term, we compute the percentage of number of "yes" to all responses in each test. The statistical data of user study are shown in Table III. The results are encouraging. With the increase of information reserved in summary, the users are more satisfied with the summary in terms of informativeness and usefulness. As for enjoyability, users' scores on badminton videos are higher than the score of office videos, even for the same level of information entropy reserved. This is partly attributed to the interestingness of the badminton videos.

C. Limitations

In the current implementation, we use a forward-backward relevance algorithm to parse videos into shots. The algorithm is more suitable for the partition of surveillance videos, for instance, the office videos used in our experiments. We also test some other types of videos captured by nearly stable cameras, and the algorithm works through careful parameter tuning. For summarizing generic multi-view videos, domain-specific techniques are good alternatives to the forward-backward relevance algorithm used for video parsing.

TABLE III
STATISTICAL DATA OF USER STUDY

Multi-view videos	Level	Enjoyability (%)	Info. (%)	Usefulness (%)
office1	Level 1	64	85	85
campus	Level 1	62	82	82
road	Level 1	67	54	64
	Level 2	69	59	67
badminton	Level 1	75	85	71
	Level 2	81	86	80
	Level 3	86	88	89

Video saliency is necessary for summarization to produce compact, yet informative summary. We compute it and evaluate the importance of multi-view shots using a Gaussian entropy fusion scheme. Since multi-view videos, especially surveillance videos, generally contain the single video modality, the fusion scheme only considers visual features. Features in other modalities, for example, audio frequency, texts, and camera motions, which are important cues for the occurrences of salient events, are however ignored. This is a limitation of our current implementation. Such features may play a crucial role in summarization, especially for sports and entertainment videos. We intend to integrate features of multiple modalities, and make a new implementation applicable to generic multi-view video genres.

Our method involves the setting of several parameters, whose values are empirically set through experiments now. Although the summarization results are not very sensitive to the setting of some parameters, it will be even more better if the parameters can be set automatically and adaptively according to video types, activities, lengths, as well as resolutions.

VII. CONCLUSIONS AND FUTURE WORK

In this paper, we propose, to the best of our knowledge, the first attempt at multi-view video summarization. We propose to use the spatio-temporal shot graph, which is based on a hypergraph, to embed the multi-view video structure, cluster the event-centered video shots using random walks, and generate the final summary by multi-objective optimization. The optimization procedure can balance various user requirements. Meanwhile, multi-level summarization can be conveniently achieved. Experiments show that the proposed summarization method is robust to brightness difference among multiple views and conspicuous noises frequently encountered in multi-view videos.

In our current version, the video saliency and shot importance are computed using only the visual features. One future work is to take into account multi-modality features. It is also possible to couple other effective attention detection methods [11], [22] together, and develop multi-view summarization method for specific video genres. Furthermore, the multi-view summary is now represented as a multi-view storyboard or a single video summary. It may be useful to generalize the video collage [53], [54] to the representation of multi-view video summary. This is another future work.

Acknowledgment

The authors would like to thank the reviewers for their helpful comments and suggestions.

References

[1] D. DeMenthon, V. Kobla, and D. Doermann, “Video summarization by curve simplification,” in *Proc. ACM Multimedia*, 1998, pp. 211–218.

[2] A. Hanjalic, R. Lagendijk, and J. Biemond, “A new method for key frame based video content representation,” in *Image Databases and Multi-Media Search*. Singapore: World Scientific, 1998.

[3] A. Hanjalic and H.-J. Zhang, “An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 8, pp. 1280–1289, Dec. 1999.

[4] Z. Li, G. M. Schuster, and A. K. Katsaggelos, “MINMAX optimal video summarization,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 10, pp. 1245–1256, Oct. 2005.

[5] X. Orriols and X. Binefa, “An EM algorithm for video summarization, generative model approach,” in *Proc. IEEE ICCV*, 2001, pp. 335–342.

[6] I. Yahiaoui, B. Merialdo, and B. Huet, “Automatic video summarization,” in *Proc. CBMIR Conf.*, 2001.

[7] N. Babaguchi, “Towards abstracting sports video by highlights,” in *Proc. ICME*, 2000, pp. 1519–1522.

[8] Y. Gong and X. Liu, “Summarizing video by minimizing visual content redundancies,” in *Proc. ICME*, 2001, pp. 607–610.

[9] Y. Gong and X. Liu, “Video summarization and retrieval using singular value decomposition,” *Multimedia Syst.*, vol. 9, pp. 157–168, Aug. 2003.

[10] S. Lu, I. King, and M. R. Lyu, “Video summarization by video structure analysis and graph optimization,” in *Proc. ICME*, 2004, pp. 1959–1962.

[11] Y.-F. Ma, X.-S. Hua, L. Lu, and H.-J. Zhang, “A generic framework of user attention model and its application in video summarization,” *IEEE Trans. Multimedia*, vol. 7, no. 5, pp. 907–919, Oct. 2005.

[12] J. Nam and A. H. Tewfik, “Dynamic video summarization and visualization,” in *Proc. ACM Multimedia*, 1999, pp. 53–56.

[13] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang, “Video summarization and scene detection by graph modeling,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 2, pp. 296–305, Feb. 2005.

[14] Y. Peng and C.-W. Ngo, “Clip-based similarity measure for query-dependent clip retrieval and video summarization,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 5, pp. 612–627, May 2006.

[15] X. Shao, C. Xu, N. C. Maddage, Q. Tian, M. S. Kankanhalli, and J. S. Jin, “Automatic summarization of music videos,” *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 2, no. 2, pp. 127–148, 2006.

[16] M. A. Smith and T. Kanade, “Video skimming and characterization through the combination of image and language understanding techniques,” in *Proc. IEEE CVPR*, 1997, pp. 775–781.

[17] H. Sundaram, L. Xie, and S.-F. Chang, “A utility framework for the automatic generation of audio-visual skims,” in *Proc. ACM Multimedia*, 2002, pp. 189–198.

[18] C. Xu, X. Shao, N. C. Maddage, and M. S. Kankanhalli, “Automatic music video summarization based on audio-visual-text analysis and alignment,” in *Proc. ACM SIGIR*, 2005, pp. 361–368.

[19] J. You, G. Liu, L. Sun, and H. Li, “A multiple visual models based perceptive analysis framework for multilevel video summarization,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 3, pp. 273–285, Mar. 2007.

[20] B. Yu, W.-Y. Ma, K. Nahrstedt, and H.-J. Zhang, “Video summarization based on user log enhanced link analysis,” in *Proc. ACM Multimedia*, 2003, pp. 382–391.

[21] B. T. Truong and S. Venkatesh, “Video abstraction: A systematic review and classification,” *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 3, no. 1, pp. 1–37, Feb. 2007.

[22] Y.-F. Ma and H.-J. Zhang, “Contrast-based image attention analysis by using fuzzy growing,” in *Proc. ACM Multimedia*, 2003, pp. 374–381.

[23] J. Assfalg, M. Bertini, C. Colombo, A. D. Bimbo, and W. Nunziati, “Semantic annotation of soccer videos: Automatic highlights identification,” *Comput. Vis. Image Understand.*, vol. 92, no. 2–3, pp. 285–305, 2003.

[24] Z. Zhao, S. Jiang, and Q. Huang, “Highlight summarization in soccer video based on goalmouth detection,” in *Proc. Asia-Pacific Workshop Visual Information Processing*, 2006.

[25] W. Bailer, E. Dumont, S. Essid, and B. Merialdo, “A collaborative approach to automatic rushes video summarization,” in *Proc. ICIP*, 2008, pp. 29–32.

[26] M. G. Christel, A. G. Hauptmann, W.-H. Lin, M.-Y. Chen, J. Yang, B. Maher, and R. V. Baron, “Exploring the utility of fast-forward surrogates for bbc rushes,” in *ACM Multimedia Workshop TRECVID Video Summarization*, 2008, pp. 35–39.

[27] C.-M. Pan, Y.-Y. Chuang, and W. Hsu, “Ntu trecvid-2007 fast rushes summarization system,” in *Proc. ACM Multimedia Workshop TRECVID Video Summarization*, 2007, pp. 23–29.

[28] C. Xu, N. C. Maddage, and X. Shao, “Automatic music classification and summarization,” *IEEE Trans. Speech Audio Process.*, vol. 13, no. 3, pp. 441–450, May 2005.

[29] J. Lee, J. Oh, and S. Hwang, “Scenario based dynamic video abstractions using graph matching,” in *Proc. ACM Multimedia*, 2005, pp. 810–819.

[30] J.-G. Lou, H. Cai, and J. Li, “A real-time interactive multi-view video system,” in *Proc. ACM Multimedia*, 2005, pp. 161–170.

[31] P. Merkle, A. Smolić, K. Müller, and T. Wiegand, “Efficient prediction structures for multiview video coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1461–1473, Nov. 2007.

[32] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, “High-quality video view interpolation using a layered representation,” in *Proc. ACM SIGGRAPH*, 2004, pp. 600–608.

[33] A. Hanjalic, R. Lagendijk, and J. Biemond, “Automated high-level movie segmentation for advanced video retrieval systems,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 4, pp. 580–588, Jun. 1999.

[34] C.-W. Ngo, T.-C. Pong, and R. T. Chin, “Video partitioning by temporal slice coherency,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 8, pp. 941–953, Aug. 2001.

[35] R. Radhakrishnan, A. Divakaran, and Z. Xiong, “A time series clustering based framework for multimedia mining and summarization using audio features,” in *Proc. 6th ACM SIGMM Workshop Multimedia Information Retrieval*, 2004, pp. 157–164.

[36] T. Xiang and S. Gong, “Activity based video content trajectory representation and segmentation,” in *Proc. British Machine Vision Conf.*, 2004, pp. 177–186.

[37] H.-J. Zhang, A. Kankanhalli, and S. W. Smoliar, “Automatic partitioning of full-motion video,” *Multimedia Syst.*, vol. 1, no. 1, pp. 10–28, Jan. 1993.

[38] U. Sakarya and Z. Telatar, “Graph-based multilevel temporal video segmentation,” *Multimedia Syst.*, vol. 14, no. 5, pp. 277–290, Nov. 2008.

[39] R. Benmokhtar, B. Huet, S.-A. Berrani, and P. Lechat, “Video shots key-frames indexing and retrieval through pattern analysis and fusion techniques,” in *Proc. 10th Int. Conf. Information Fusion*, 2007, pp. 1–6.

[40] G. Ciocca and R. Schettini, “An innovative algorithm for key frame extraction in video summarization,” *J. Real-Time Image Process.*, vol. 1, pp. 69–88, Oct. 2006.

[41] R. H. Shumway and D. S. Stoffer, *Time Series Analysis and Its Applications*. New York: Springer, 2000.

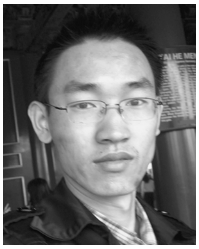
[42] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proc. IEEE CVPR*, 2001, pp. 511–518.

[43] T. Xiang and S. Gong, “Video behaviour profiling and abnormality detection without manual labelling,” in *Proc. IEEE ICCV*, 2005, pp. 1238–1245.

[44] C. Berge, *Hypergraphs*. Amsterdam, The Netherlands: North-Holland, 1989.

[45] D. Zhou, J. Huang, and B. Schölkopf, “Learning with hypergraphs: Clustering, classification, and embedding,” in *Proc. Advances in Neural Information Processing Systems*, 2007.

[46] L. Sun, S. Ji, and J. Ye, “Hypergraph spectral learning for multi-label classification,” in *Proc. 14th ACM SIGKDD Conf.*, 2008, pp. 668–676.
[47] L. Grady, “Random walks for image segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.
[48] F. Wang, Y.-G. Jiang, and C.-W. Ngo, “Video event detection using motion relativity and visual relatedness,” in *Proc. ACM Multimedia*, New York, 2008, pp. 239–248.
[49] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery*, Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge, U.K.: Cambridge Univ. Press, 2007.
[50] L. A. Zadeh, “Fuzzy sets,” *Inf. Control*, vol. 8, no. 3, pp. 338–353, Jun. 1965.
[51] D. Li and S. Chen, “A fuzzy programming approach to fuzzy linear fractional programming with fuzzy coefficients,” *J. Fuzzy Math.*, vol. 4, no. 4, pp. 829–833, 1996.
[52] [Online]. Available: http://en.wikipedia.org/wiki/Knapsack_problem.
[53] X. Liu, T. Mei, X.-S. Hua, B. Yang, and H.-Q. Zhou, “Video collage,” in *Proc. ACM Multimedia*, 2007, pp. 461–462.
[54] T. Mei, B. Yang, S.-Q. Yang, and X.-S. Hua, “Video collage: Presenting a video sequence using a single image,” *Visual Comput.*, vol. 25, no. 1, pp. 39–51, 2009.

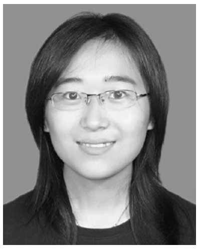


Yanwei Fu received the B.Sc. degree in information and computing sciences from Nanjing University of Technology, Nanjing, China, in 2008. He is now pursuing the M.Sc. degree in the Department of Computer Science and Technology at Nanjing University.

His research interest is in video summarization and machine learning.

Yanwen Guo received the Ph.D. degree in applied mathematics from State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China, in 2006.

He is currently an Associate Professor at the National Key Laboratory for Novel Software Technology, Department of Computer Science and Technology, Nanjing University, Nanjing, China. His main research interests include image and video processing, geometry processing, and face-related applications. He worked as a visiting researcher in the Department of Computer Science and Engineering, the Chinese University of Hong Kong, in 2006 and 2009 and a visiting researcher in the Department of Computer Science, the University of Hong Kong, in 2008.

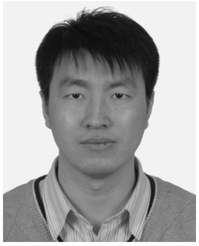


Yanshu Zhu received the B.Sc. degree in computer science from Nanjing University, Nanjing, China, in 2009. She is now pursuing the Ph.D. degree in the Department of Computer Science, The University of Hong Kong.

Her research interest is in image and video analysis.

Feng Liu received the B.S. and M.S. degrees from Zhejiang University, Hangzhou, China, in 2001 and 2004, respectively, both in computer science. He is currently pursuing the Ph.D. degree in the Department of Computer Sciences at the University of Wisconsin-Madison.

His research interests are in the areas of multimedia, computer vision, and graphics.



Chuanming Song received the M.E. and B.E. degrees, both in computer applied technology, from the Liaoning Normal University, Dalian, China, in 2003 and 2006, respectively. He is currently pursuing the Ph.D. degree at Nanjing University, Nanjing, China.

His research interests include scalable image and video coding, and digital watermarking of multimedia.

Zhi-Hua Zhou (S’00–M’01–SM’06) received the B.Sc., M.Sc., and Ph.D. degrees in computer science from Nanjing University, Nanjing, China, in 1996, 1998, and 2000, respectively, all with the highest honors.

He joined the Department of Computer Science and Technology at Nanjing University as an Assistant Professor in 2001, and is currently Cheung Kong Professor and Director of the LAMDA group. His research interests are in artificial intelligence, machine learning, data mining, pattern recognition, information retrieval, evolutionary computation, and neural computation. In these areas, he has published over 70 papers in leading international journals or conference proceedings.

Dr. Zhou has won various awards/honors including the National Science & Technology Award for Young Scholars of China (2006), the Award of National Science Fund for Distinguished Young Scholars of China (2003), the National Excellent Doctoral Dissertation Award of China (2003), and the Microsoft Young Professorship Award (2006). He is an Associate Editor of the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING and *ACM Transactions on Intelligent Systems and Technology*, Associate Editor-in-Chief of *Chinese Science Bulletin*, and on the editorial boards of over ten other journals. He is the Founding Steering Committee Co-Chair of ACML; Steering Committee member of PAKDD and PRICAI; Program Committee Chair/Co-Chair of PAKDD’07, PRICAI’08, and ACML’09; Vice Chair or Area Chair or Senior PC of conferences including IEEE ICDM’06, IEEE ICDM’08, SIAM DM’09, ACM CIKM’09, ACM SIGKDD’10, ECML PKDD’10, and ICPR‘10; and General Chair/Co-Chair or Program Committee Chair/Co-Chair of a dozen of native conferences. He is the Chair of the Machine Learning Society of Chinese Association of Artificial Intelligence, Vice Chair of the Artificial Intelligence & Pattern Recognition Society of China Computer Federation, and the Chair of the IEEE Computer Society Nanjing Chapter.