Video Summarization in a Multi-View Camera Network

Rameswar Panda*, Abir Das†, Amit K. Roy-Chowdhury*
*Electrical and Computer Engineering Department, University of California, Riverside
†Computer Science Department, University of Massachusetts, Lowell
Email: rpand002@ucr.edu, abir.das@email.ucr.edu, amitrc@ece.ucr.edu

***Abstract*—While most existing video summarization approaches aim to extract an informative summary of a single video, we propose a novel framework for summarizing multi-view videos by exploiting both intra- and inter-view content correlations in a joint embedding space. We learn the embedding by minimizing an objective function that has two terms: one due to intra-view correlations and another due to inter-view correlations across the multiple views. The solution can be obtained directly by solving one Eigen-value problem that is linear in the number of multi-view videos. We then employ a sparse representative selection approach over the learned embedding space to summarize the multi-view videos. Experimental results on several benchmark datasets demonstrate that our proposed approach clearly outperforms the state-of-the-art.**

I. Introduction

Network of surveillance cameras are everywhere nowadays. A major problem is to figure out how to extract useful information from the videos captured by these cameras. Fig. 1 depicts an illustrative example where a network of cameras, with both overlapping and non-overlapping fields of view (fovs) are capturing videos from a region. The basic question that we want to explore in such scenario is: *can we get an idea of the video content without watching all the videos entirely?*

Much progress has been made in developing a variety of ways to summarize a single video, by exploring different design criteria (representativeness [15], [6], [28], [4], interestingness [21], [13]) in an unsupervised manner, or developing supervised algorithms [17], [12], [11], [26]). However, with some exceptions of [8], [20], [24], [16], summarizing multi-view videos still remains as a challenging problem because of large amount of inter-view content correlations along with intra-view correlations present in such videos.

In this paper, we focus on the task of summarizing multi-view videos, and illustrate how a new representation that exploits multi-view correlations can effectively generate a more informative summary while comparing with the prior multi-view works. Our work builds upon the idea of subspace learning, which typically aims to obtain a latent subspace shared by multiple views by assuming that these views are generated from this latent subspace. Specifically, our key idea is the following: by viewing two or more multi-view videos as actually being one large video, making inference about multi-view videos reduces to making inference about a single video in the latent subspace.

Our approach works as follows. First, we embed all the frames in an unified low dimensional latent space such that the locations of the frames preserve both intra- and inter-view correlations (Section II). This is achieved by minimizing an objective function that has two terms: one due to intra-view correlations and another due to inter-view correlations across the multiple views. The solution can be obtained by solving an eigen-value problem that is linear in size of the multi-view videos. Then, we employ a sparse representative selection approach over the embedding to produce multi-view summaries (Section III). Specifically, we formulate the task of finding summaries as a sparse coding problem where the dictionary is constrained to have a fixed basis (dictionary to be the matrix of same data points) and the nonzero rows of sparse coefficient matrix represent the multi-view summaries.

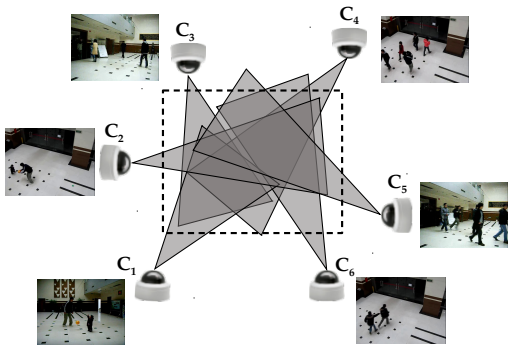

Fig. 1. An illustration of a multi-view camera network where six cameras C_1, C_2, …, C_6 are observing an area (black rectangle) from different viewpoints. Since the views are roughly overlapping, information correlations across multiple views along with correlations in each view should be taken into account for generating a concise multi-view summary.

Contributions. The contributions of our work can be summarized as follows.

(1) We propose a multi-view frame embedding which is able to preserve both intra and inter-view correlations without assuming any prior correspondences/alignment between the multi-view videos.

(2) We propose a sparse representative selection method over the learned embedding to summarize the multi-view videos, which provides *scalability* in generating summaries. In particular, this allows us to generate summaries of different lengths as per the user request *(analyze once, generate many)*.

(3) The proposed method is a *generalized framework* which makes sparse coding feasible in summarizing both single and multi-view videos. We demonstrate the generalizability of our framework with extensive experiments on three publicly available multi-view datasets (11 videos) and two single view datasets (100 videos).

Related Work. Most of the previous summarization techniques are designed for single-view videos. Various strategies have been studied, including clustering [1], [5], [23], [9], attention modeling [21], saliency based regression model [17], super frame segmentation [13], kernel temporal segmentation [26], crowd-sourcing [15], submodular maximization [12], and point process [11]. Interested readers can check [22], [27] for a more comprehensive summary. However, they usually do not perform well for summarizing multi-view videos since they cannot exploit the large inter-view correlations.

To address the challenges encountered in a multi-view camera network, some state-of-the-art approaches use random walk over spatio-temporal shot graphs [8] and rough sets [20] to summarize multi-view videos. A very recent work in [16] uses bipartite matching constrained optimum path forest clustering to solve the problem of summarizing multi-view videos. In [25], stochastic neighbor embedding with sparse coding is employed to summarize multi-view videos. An online method for summarization can also be found in [24]. The work in [18] and [19] also addresses a similar problem of summarization in multi-camera settings with non-overlapping field of views. By contrast, the approach that we describe here seeks to find summary from a multi-view network as shown in Fig. 1. Moreover, our approach does not require a priori knowledge of field of view.

II. Multi-view Frame Embedding

Problem Statement. Consider a set of K different videos captured from different cameras, in a D-dimensional space where $X^{(k)} = \{x_i^{(k)} \in \mathbb{R}^D, i = 1, \cdots, N_k\}, k = 1, \cdots, K$. Each x_i represents the feature descriptor (e.g., color, texture) of a video frame in D-dimensional feature space. As the videos are captured non-synchronously, the number of frames in each video might be different and hence there is no optimal one-to-one correspondence that can be assumed. We use N_k to denote the number of frames in k-th video and N to denote the total number of frames in all videos.

Given the multi-view videos, our goal is to find an embedding for all the frames into a joint latent space while satisfying some constraints. Specifically, we are seeking a set of embedded coordinates $Y^{(k)} = \{y_i^{(k)} \in \mathbb{R}^d, i = 1, \cdots, N_k\}, k = 1, \cdots, K$, where, d ($<< D$) is the dimensionality of the embedding space, with the following two constraints: (1) *Intra-view correlations.* The content correlations between frames of a video should be preserved in the embedding space. (2) *Inter-view correlations.* The frames from different videos with high feature similarity should be close to each other in the resulting embedding space as long as they do not violate the intra-view correlations present in an individual view.

Modeling Multi-view Correlations. To achieve an embedding that preserves the above two constraints, we need to consider feature similarities between two frames in an individual video as well as across two different videos.

Inspired by the recent success of sparse representation coefficient based methods to compute data similarities in subspace clustering [7], we adopt such coefficients in modeling multi-view correlations. Our proposed approach has two nice properties: (1) the similarities computed via sparse coefficients are robust against noise and outliers since the value not only depends on the two frames, but also depends on other frames that belong to the same subspace, and (2) it simultaneously carries out the adjacency construction and similarity calculation within one step unlike kernel based methods that usually handle these tasks independently with optimal choice of several parameters.

Intra-view Similarities. Intra-view similarity should reflect spatial arrangement of feature descriptors in each view. Based on the *self-expressiveness property* [7] of an individual view, each frame can be sparsely represented by a small subset of frames that are highly correlated in the dataset. Mathematically, for k-th view, it can be represented as

$$x_i^{(k)} = X^{(k)} c_i^{(k)}, \ c_{ii}^{(k)} = 0, \tag{1}$$

where $c_i^{(k)} = [c_{i1}^{(k)}, c_{i2}^{(k)}, ..., c_{iN_k}^{(k)}]^T$, and the constraint $c_{ii}^{(k)} = 0$ eliminates the trivial solution of representing a frame with itself. The coefficient vector $c_i^{(k)}$ should have nonzero entries for a few frames that are correlated and zeros for the rest. However, in (1), the representation of x_i in the dictionary X is not unique in general. Since we are interested in efficiently finding a nontrivial sparse representation of x_i, we consider the tightest convex relaxation of the l_0 norm, i.e.,

$$\min \ ||c_i^{(k)}||_1 \ \text{ s.t. } \ x_i^{(k)} = X^{(k)} c_i^{(k)}, \ c_{ii}^{(k)} = 0, \tag{2}$$

It can be rewritten in matrix form for all frames in a view as

$$\min \ ||C^{(k)}||_1 \ \text{ s.t. } X^{(k)} = X^{(k)} C^{(k)}, \ \text{diag}(C^{(k)}) = 0, \tag{3}$$

where $C^{(k)} = [c_1^{(k)}, c_2^{(k)}, ..., c_{N_k}^{(k)}]$ is the sparse coefficient matrix whose i-th column corresponds to the sparse representation of the frame $x_i^{(k)}$. The coefficient matrix obtained from the above l_1 sparse optimization essentially characterizes the frame correlations and thus it is natural to utilize as intra-view similarities. This provides an immediate choice of the intra-view similarity matrix as $C_{intra}^{(k)} = |C^{(k)}|^T$ where i-th row of matrix $C_{intra}^{(k)}$ represents the similarities between the i-th frame to all other frames in the view.

Inter-view Similarities. Since all cameras are focusing on roughly the same fovs from different viewpoints, all views have apparently a single underlying structure. Following this assumption in a multi-view setting, we find the correlated frames across two views on solving a similar l_1 sparse optimization like in intra-view similarities. Specifically, we calculate the pairwise similarity between m-th and n-th view by solving the following optimization problem:

$$\min \ ||C^{(m,n)}||_1 \ \text{ s.t. } \ X^{(m)} = X^{(n)} C^{(m,n)}, \tag{4}$$

where $C^{(m,n)} \in \mathbb{R}^{N_n \times N_m}$ is the sparse coefficient matrix whose i-th column corresponds to the sparse representation of the frame $x_i^{(m)}$ using the dictionary X. Ideally, after solving the proposed optimization problem in (4), we obtain a sparse representation for a frame in m-th view whose nonzero elements correspond to frames from n-th view that belong to the same subspace. Finally, the inter-view similarity matrix between m-th and n-th view can be represented as

$C_{inter}^{(m,n)} = |C^{(m,n)}|^T$ where i-th row of matrix $C_{inter}^{(m,n)}$ represent similarities between i-th frame of m-th view and all other frames in the n-th view.

Objective Function. The aim of embedding is to correctly match the proximity score between two frames x_i and x_j to the score between corresponding embedded points y_i and y_j respectively. Motivated by this observation, we reach the following objective function on the embedded points Y.

$$\begin{aligned}\mathcal{F}(Y^{(1)},...,Y^{(K)}) &= \sum_k \mathcal{F}_{\text{intra}}(Y^{(k)}) + \\ &\sum_{\substack{m,n \\ m\neq n}} \mathcal{F}_{\text{inter}}(Y^{(m)}, Y^{(n)}) \\ &= \sum_k \sum_{i,j} ||y_i^{(k)} - y_j^{(k)}||^2 C_{intra}^{(k)}(i,j) + \\ &\sum_{\substack{m,n \\ m\neq n}} \sum_{i,j} ||y_i^{(m)} - y_j^{(n)}||^2 C_{inter}^{(m,n)}(i,j)\end{aligned} \tag{5}$$

where k, m and $n = 1, \cdots, K$. $\mathcal{F}_{\text{intra}}(Y^{(k)})$ is the cost of preserving local correlations within $X^{(k)}$ and $\mathcal{F}_{\text{inter}}(Y^{(m)}, Y^{(n)})$ is the cost of preserving correlations between $X^{(m)}$ and $X^{(n)}$. The first term says that if two frames $(x_i^{(k)}, x_j^{(k)})$ of a view are similar, which happens when $C_{intra}^{(k)}(i,j)$ is larger, their locations in the embedded space, $y_i^{(k)}$ and $y_j^{(k)}$ should be close to each other. Similarly, the second term tries to preserve the inter-view correlations by bringing embedded points $y_i^{(m)}$ and $y_i^{(n)}$ close to each other if the pairwise proximity score $C_{inter}^{(m,n)}(i,j)$ is high. The above objective function (5) can be rewritten using one similarity matrix defined over the whole set of frames as

$$\mathcal{F}(Y) = \sum_{m,n} \sum_{i,j} ||y_i^{(m)} - y_j^{(m)}||^2 C_{total}^{(m,n)}(i,j) \tag{6}$$

where the total similarity matrix is defined as

$$C_{total}^{(m,n)}(i,j) = \begin{cases} C_{intra}^{(k)}(i,j) & \text{if } m = n = k \\ C_{inter}^{(m,n)}(i,j) & \text{otherwise} \end{cases} \tag{7}$$

This construction defines a $N \times N$ similarity matrix where the diagonal blocks represent the intra-view similarities and off-diagonal blocks represent inter-view similarities. Note that an interesting fact about our total similarity matrix construction in (7) is that since each l_1 optimization is solved individually, a fast parallel computing strategy can be easily adopted for efficiency. However, the matrix in (7) is not symmetric since in l_1 optimization (2,4), a frame x_i can be represented as a linear combination of some frames including x_j, but x_i may not be present in the sparse representation of x_j. But, ideally, a similarity matrix should be symmetric in which frames belonging to the same subspace should be connected to each other. Hence, we reformulate (6) with a symmetric similarity matrix $W = C_{total} + C_{total}^T$ as

$$\mathcal{F}(Y) = \sum_{m,n} \sum_{i,j} ||y_i^{(m)} - y_j^{(m)}||^2 W^{(m,n)}(i,j) \tag{8}$$

With the above formulation, we make sure that two frames x_i and x_j get connected to each other either x_i and x_j is in the sparse representation of the other. Furthermore, we normalize W as $w_i \leftarrow w_i / ||w_i||_\infty$ to make sure the weights in the similarity matrix are of same scale.

Given this construction, the objective function (8) reduces to the problem of Laplacian embedding [2] of frames defined by the similarity matrix W. So, the optimization problem can be written as

$$Y^* = \underset{Y}{\text{argmin}}\ tr\big(Y^T L Y\big) \quad s.t. \quad Y^T D Y = I, \tag{9}$$

where L is the laplacian matrix of W, I is the identity matrix. The first constraint eliminates the arbitrary scaling and avoids degenerate solutions. Minimizing this objective function is a generalized eigenvector problem: $Ly = \lambda Dy$ and the optimal solution can be obtained by the bottom d nonzero eigenvectors. The required embedding of the frames are given by the row vectors of Y.

III. Sparse Representative Selection

Once the frame embedding is obtained, our next goal is to find an optimal subset of all the embedded frames, such that each frame can be described as weighted linear combination of a few of the frames from the subset. The subset is then referred as the informative summary of the multi-view videos. Therefore, our natural goal is to establish a frame level sparsity which can be induced by performing l_1 regularization on rows of the sparse coefficient matrix [4], [6]. By introducing the row sparsity regularizer, the summarization problem can now be succinctly formulated as

$$\min_Z \quad ||Z||_{2,1} \quad \text{s.t.} \quad Y = YZ, \ \text{diag}(Z) = 0 \tag{10}$$

where $Z \in \mathbb{R}^{N \times N}$ is the sparse coefficient matrix and $\|Z\|_{2,1} \triangleq \sum_{i=1}^{N} \|Z^i\|_2$ is the row sparsity regularizer i.e., sum of l_2 norms of the rows of Z. The first constraint i.e., self-expressiveness property in summarization is logical as the representatives for summary should come from the original frame set whereas the second constraint is introduced to avoid the numerically trivial solution ($Z = I$) in practice by forcing the diagonal elements to be zeros. Minimization of (10) leads to a sparse solution for Z in terms of rows, *i.e.*, the sparse coefficient matrix Z contains few nonzero rows which constitute the video summary. Notice that both of the sparse optimization in (3) and (10) look similar; however, the nature of sparse regularizer in both formulations are completely different. In (3), the objective of l_1 regularizer is to induce element wise sparsity in a column whereas in (10), the objective of $l_{2,1}$ regularizer is to induce row level sparsity in a matrix.

The objective functions in (5) and (10) are quite general. One can easily notice that our framework can be extended to summarize single view videos by removing the inter-view similarities in (5). Hence, our proposed embedding with sparse representative selection can summarize both single as well as multi-view videos whereas the prior sparse coding based methods [4], [6] can summarize only single-view videos. Moreover, our approach is computationally efficient as the sparse coding is done in lower-dimensional space and at the same time, it preserves the locality and correlations among the original frames which has a great impact on the summarization output.

IV. Solving the Sparse Optimization Problems

We solve all the sparse optimization problems using an Alternating Direction Method of Multipliers (ADMM) framework [3]. Due to space limitation, we only present the optimization procedure to solve (10). However, the same procedure can be easily extended to solve other sparse optimizations (3, 4). Using Lagrange multipliers, the optimization problem (10) can be written as

$$\min_{Z} \|Z\|_{2,1} + \frac{\lambda}{2}\|Y - YZ\|_F^2 \quad s.t. \;\; diag(Z) = 0 \tag{11}$$

where λ is the regularization parameter that balances the weight of the two terms. To facilitate the optimization, we consider an equivalent form of (11) by introducing an auxiliary variable A:

$$\min_{Z,A} \|Z\|_{2,1} + \frac{\lambda}{2}\|Y - YA\|_F^2 \quad s.t. \;\; A = Z, \;\; diag(Z) = 0 \tag{12}$$

ADMM tries to solve (12) by iteratively updating A and Z shown in Algo. 1, where the shrinkage-thresholding operator $S_\mu(z)$ acting on each row of the given matrix is defined as

$$S_\mu(z) = \max\left\{||z||_2 - \mu, 0\right\}\frac{z}{\|z\|_2}. \tag{13}$$

Algorithm 1 An ADMM solver for (11)

Input: Embedded feature matrix Y
Initialization: Initialize A, Z, B to zero and $\lambda, \rho > 0$
while *not converged* **do**
  $A \leftarrow (\lambda Y^T Y + \rho I)^{-1}(\lambda Y^T Y + \rho Z - B)$;
  $A \leftarrow A - \text{diag}(\text{diag}(A))$;
  $Z \leftarrow S_{\frac{1}{\rho}}(A + B/\rho)$ (row-wise);
  $Z \leftarrow Z - \text{diag}(\text{diag}(Z))$;
  $B \leftarrow B + \rho(A - Z)$;
end while
Output: Sparse coefficient matrix Z.

Since the problem (11) is convex, Algo. 1 is guaranteed to converge by the existing ADMM theory [10].

Remark 1. We do not require to compute $(\lambda Y^T Y + \rho I)^{-1}$ in each iteration for updating A. More specifically, it is unchanged during iterations of Algo. 1. Thus, one can precompute the required Cholesky factorizations to avoid redundant computations for efficiently solving those linear systems.

V. Experiments

Datasets. We conduct rigorous experiments using five publicly available datasets: (i) Office dataset captured with 4 stably-held web cameras in an indoor environment [8], (ii) Campus dataset taken with 4 handheld ordinary video cameras in an outdoor scene [8], (iii) Lobby dataset captured with 3 cameras in a large lobby area [8], (iv) Open Video Project (OV) dataset of 50 videos [1] and (v) YouTube dataset of 50 videos [1]. These datasets are extremely diverse: while the first three datasets consists of multi-view videos with overall 360 degree coverage of the scene, the last two datasets contains single view videos of several genres.

Features. We utilize Pycaffe with the "BVLC CaffeNet" pretrained model [14] to extract a 4096-dim CNN feature vector (i.e. the top layer hidden unit activations of the network) for each video frame. We use deep features, as they are the state-of-the-art visual features and have shown best performance on various recognition tasks.

Performance Measures. We compare all the approaches using three quantitative measures, including Precision, Recall and F-measure ($\frac{2\times Precision\times Recall}{Precision+Recall}$) [8]. For all these metrics, the higher value indicates better summarization quality.

Other Details. The regularization parameters λ is taken as λ_0/γ where $\gamma > 1$ and λ_0 is analytically computed from the data [6]. In Algo. 1, the stop criteria is defined as following:

$$||A^{(t)} - Z^{(t)}||_\infty \leq \epsilon \; or \; t \geq 2000 \tag{14}$$

where t is the iteration number and ϵ is set to 10^{-7} throughout the experiments.

Multi-view Video Summarization: This experiment aims at evaluating our proposed framework in summarizing multi-view videos compared to the state-of-the-art including both methods for single view and multi-view video summarization.

Compared Methods. We compare our approach with total of seven existing approaches including four baseline methods (ConcateAttention [21], ConcateSparse [6], AttentionConcate [21], SparseConcate [6]) that use single-view summarization approach over multi-view datasets to generate summary and three methods (RandomWalk [8], RoughSets [20], Bipartite-OPF [16]) which are specifically designed for multi-view video summarization. Note that the first two baselines (ConcateAttention, ConcateSparse) concatenate all the views into a single video and then applie a summarization approach, whereas in the other two baselines (AttentionConcate, SparseConcate), an approach is first applied to each view and then the resulting summaries are combined along the time line to form a single multi-view summary. [4] also uses the same objective function as in [6] for summarizing consumer videos. The only difference lies in the algorithm used to solve the objective function (proximal vs ADMM). Hence, we compared only with [6]. The purpose of comparing single view methods is to show that techniques that attempt to find informative summary from single-view videos usually do not produce an optimal set of representatives while summarizing multi-view videos. We employ the ground truth of important events reported in [8] for a fair comparison. In our approach, an event is taken to be correctly detected if we get a representative frame from the set of ground truth frames between the start and end of the event.

Results. Table. I show the summarization results on all three multi-view datasets. The analysis of the results for both Office and Lobby dataset are quite interesting in two aspects. First, our approach produces summaries with same precision as RandowWalk for both of the datasets. However, the improvement in recall value indicates the ability of our method in keeping more important information in the summary compared to RandomWalk. One such illustrative example for the Office dataset is presented in Fig. 3. Second, our performance is similar to the recently published baseline BipartiteOPF for

TABLE I

PERFORMANCE COMPARISON WITH SEVERAL BASELINES INCLUDING BOTH SINGLE AND MULTI-VIEW METHODS APPLIED ON THE THREE MULTI-VIEW DATASETS. **P**: PRECISION IN PERCENTAGE, **R**: RECALL IN PERCENTAGE AND **F**: F-MEASURE. OURS PERFORM THE BEST.

Methods	Office			Campus			Lobby		
	P	R	F	P	R	F	P	R	F
ConcateAttention [21]	100	38	55.07	56	48	51.86	31	95	81.98
ConcateSparse [6]	100	54	66.99	59	45	50.93	93	65	76.69
AttentionConcate [21]	100	46	63.01	40	28	32.66	100	70	82.21
SparseConcate [6]	94	58	71.34	58	52	54.49	88	70	77.87
RandomWalk [8]	100	61	76.19	70	55	61.56	100	77	86.81
RoughSets [20]	100	61	76.19	69	57	62.14	97	74	84.17
BipartiteOPF [16]	100	69	81.79	75	69	71.82	100	79	88.26
Ours	**100**	**73**	**84.48**	**84**	**69**	**75.42**	**100**	**79**	**88.26**

Fig. 2. Some summarized events for the *Lobby* dataset. Top row: summary produced by SparseConcate [6], Middle row: summary produced by ConcateSparse [6], and Bottom row: summary produced by our approach. It is clearly evident from both top and middle rows that both of the single-view baselines produce a lot of redundant events in summarizing multi-view videos, however, our approach (bottom row) produces meaningful representatives by exploiting the content correlations via an embedding. Redundant events are marked with same color borders. Note that, although the frames with same color border look somewhat visually distinct, they essentially represent same events as per the ground truth in [8].

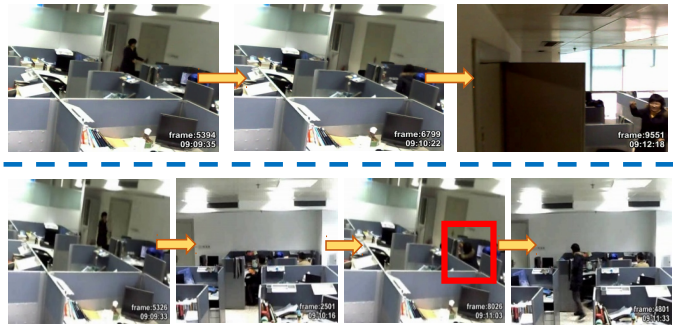

Fig. 3. Sequence of events detected related to activities of a member (A_0) inside the Office dataset. Top row: Summary produced by method [8], and Bottom row: Summary produced by our approach. The event of looking for a thick book to read (as per the ground truth in [8]) is missing in the summary produced by method [8] where as it is correctly detected by our approach (3rd frame: bottom row). This indicates our method captures video semantics in more informative way compared to [8].

Lobby dataset but we improved around 5% in terms recall and 3% in terms of F-measure for the Office dataset.

Notice that for all methods, including ours, performance on Campus dataset is not that good as compared to other two datasets. This is obvious since the Campus dataset contains many trivial events as it was captured in an outdoor environment, thus making the summarization more difficult. Nevertheless, for this challenging dataset, F-measure of our approach is about 15% better than that of RandomWalk and 5% better than that of BipartiteOPF. Overall, on all datasets, our approach outperforms all the baselines in terms of F-measure. This corroborates the fact that sparse representative selection coupled with multi-view frame embedding produces better summaries in contrast to the state-of-the-art methods.

Furthermore, while comparing with several mono-view summarization approaches (ConcateAttention, ConcateSparse, AttentionConcate, SparseConcate), Table. I reveals that summaries produced using these methods contain a lot of redundancies (simultaneous presence of most of the events) since they fail to exploit the complicated inter-view frame correlations present in multi-view videos. However, our proposed framework significantly outperforms these methods in terms of precision, recall and F-measure due to its ability to model multi-view correlations. Limited to the space, we only present a part of the summarized events for the Lobby dataset as illustrated in Fig. 2.

Generalization to Single-view Video Summarization: The objective of this experiment is to validate the generalizabilty of our framework in summarizing single view videos along with multi-view videos. In particular, the basic question that we want to explore in this experiment is: *does the learned embedding also help in summarizing single-view videos?*

Compared Methods. We contrast our approach with several baselines covering a wide variety of single-view methods as follows: (1) DT [23] that model a video using a delaunay triangulation to extract the key frames, (2) STIMO [9]: uses a fast clustering algorithm with advanced user customization, (3) VSUMM [5]: uses an improved k-means algorithm clustering and then the centroids are deemed as key frames, (4) VISON [1]: this method extract key frames based on local maximum in the frame similarity curve, (5) a sparse coding approach that does not consider frame correlations (Sparse) [4], [6]. We follow the standard procedure in [5] to

TABLE II
PERFORMANCE OF VARIOUS SINGLE VIEW VIDEO SUMMARIZATION METHODS ON BOTH OV AND YOUTUBE DATASETS.

Methods	OV			YouTube		
	P	R	F	P	R	F
DT [23]	67.7	53.2	57.6	40.7	42.8	42.3
STIMO [9]	60.3	72.2	63.4	46.2	43.1	45.6
VSUMM [5]	70.6	75.8	70.3	58.3	57.6	56.8
VISON [1]	70.1	82.0	75.5	50.1	51.5	49.2
Sparse [6]	79.5	83.4	78.2	65.7	63.8	61.2
Ours	**81.6**	**84.8**	**80.4**	**67.0**	**66.5**	**64.6**

obtain the mean performance measures on comparing with all user-created summaries.

Results. Table. II shows results on both OV and YouTube datasets. Without surprise, Sparse [6] performs better as compared to other clustering based methods [23], [9], [5], [1] since it selects key frames based on how representative a particular frame is in the reconstruction of the original video. However, our method performs even better compared to Sparse. We believe the improvement can be attributed to frame embedding that exploits frame correlations in sparse representative selection.

Scalability in Generating Summaries: The aim of this experiment is to demonstrate the scalability of our approach in generating summaries of different length based on the user constraints without any further analysis of the input videos. Such property makes sense in surveillance systems as one user may want to see only 5 most important events of the day whereas at the same time, another user may want to see only 2 most important events that occurred in the whole day.

Results. Apart from indicating the representatives for the summary, the non-zero rows of Z also provides information about the relative importance of the representatives for describing the whole videos. A higher ranking representative frame takes part in the reconstruction of many frames in the multi-view videos as compared to a lower ranked frame. This provides scalability to our approach as the ranked list can be used as a scalable representation to provide summaries of different lengths (*analyze once, generate many*). Fig. 4 shows the generated summaries of length 3, 4 and 7 most important events (as determined by the method described above) for the Office dataset.

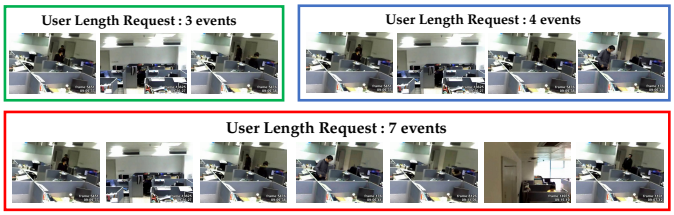

Fig. 4. The figure shows an illustrative example of scalability in generating summaries of different length based on the user constraints for the Office dataset. Each event is represented by a key frame and are arranged according to the l_2 norms of corresponding non-zero rows of Z.

VI. CONCLUSIONS

In this paper, we present a novel framework for summarizing multi-view videos in a camera network by exploiting the content correlations via an joint embedding. The embedding formulation introduced encodes both intra and inter-view correlations in a unified latent subspace. We then employ a sparse coding method over the embedding that provides scalability in generating the summaries. We show the effectiveness of our framework through rigorous experimentation on five datasets.

Acknowledgments: This work was partially supported by NSF grant IIS-1316934 and 1330110.

REFERENCES

[1] J. Almeida, N. J. Leite, and R. da S. Torres. VISON: VIdeo Summarization for ONline applications. *PRL*, 2012. 2, 4, 5, 6
[2] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, 2001. 3
[3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 2011. 4
[4] Y. Cong, J. Yuan, and J. Luo. Towards Scalable Summarization of Consumer Videos Via Sparse Dictionary Selection. *TMM*, 2012. 1, 3, 4, 5
[5] S. E. F. de Avila, A. P. B. Lopes, A. da Luz Jr., and A. de A. Arajo. VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method. *PRL*, 2011. 2, 5, 6
[6] E. Elhamifar, G. Sapiro, and R. Vidal. See all by looking at a few: Sparse modeling for finding representative objects. In *CVPR*, 2012. 1, 3, 4, 5, 6
[7] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *TPAMI*, 2013. 2
[8] Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z.-H. Zhou. Multi View Video Summmarization. *TMM*, 12(7):717–729, 2004. 1, 2, 4, 5
[9] M. Furini, F. Geraci, M. Montangero, and M. Pellegrini. Stimo: Still and moving video storyboard for the web scenario. *Multimed Tools and Appl*, 2010. 2, 5, 6
[10] R. Glowinski and P. Le Tallec. *Augmented Lagrangian and operator-splitting methods in nonlinear mechanics*. SIAM, 1989. 4
[11] B. Gong, W. Chao, K. Grauman, and F. Sha. Diverse sequential subset selection for supervised video summarization. In *NIPS*, 2014. 1, 2
[12] M. Gygli, L. V. Gool, and H. Grabner. Video summarization by learning submodular mixtures of objectives. In *CVPR*, 2015. 1, 2
[13] M. Gygli, H. Grabner, H. Riemenschneider, and L. V. Gool. Creating summaries from user videos. In *ECCV*, 2014. 1, 2
[14] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint*, 2014. 4
[15] A. Khosla, R. Hamid, C. J. Lin, and N. Sundaresan. Large-scale video summarization using web-image priors. In *CVPR*, 2013. 1, 2
[16] S. Kuanar, K. Ranga, and A. Chowdhury. Multi-view video summarization using bipartite matching constrained optimum-path forest clustering. *TMM*, 2015. 1, 2, 4, 5
[17] Y. Lee, J. Ghosh, and K. Grauman. Discovering important people and objects for egocentric video summarization. In *CVPR*, 2012. 1, 2
[18] C. D. Leo and B. S. Manjunath. Multicamera video summarization from optimal reconstruction. In *ACCV Workshop*, 2011. 2
[19] C. D. Leo and B. S. Manjunath. Multicamera Video Summarization and Anomaly Detection from Activity Motifs. *TOSN*, 2014. 2
[20] P. Li, Y. Guo, and H. Sun. Multi key-frame abstraction from videos. In *ICIP*, 2011. 1, 2, 4, 5
[21] Y. F. Ma, X. S. Hua, and H. J. Zhang. A Generic Framework of User Attention Model and Its Application in Video Summarization. *TMM*, 2005. 1, 2, 4, 5
[22] A. G. Money and H. Agius. Video summarisation: A conceptual framework and survey of the state of the art. *JVCIR*, 2008. 2
[23] P. Mundur, Y. Rao, and Y. Yesha. Keyframe-based video summarization using delaunay clustering. *IJDL*, 2006. 2, 5, 6
[24] S.-H. Ou, C.-H. Lee, V. Somayazulu, Y.-K. Chen, and S.-Y. Chien. On-Line Multi-View Video Summarization for Wireless Video Sensor Network. *JSTSP*, 2015. 1, 2
[25] R. Panda, A. Das, and A. K. Roy-Chowdhury. Embedded sparse coding for summarizing multi-view videos. In *ICIP*, 2016. 2
[26] D. Potapov, M. Douze, Z. Harchaoui, and C. Schmid. Category-specific video summarization. In *ECCV*, 2014. 1, 2
[27] B. Truong and S. Venkatesh. Video abstraction: A systematic review and classification. *TOMCCAP*, 2007. 2
[28] B. Zhao and E. P. Xing. Quasi real-time summarization for consumer videos. In *CVPR*, 2014. 1