Summary Transfer: Exemplar-based Subset Selection for Video Summarization

Ke Zhang*, Wei-Lun Chao*
University of Southern California
Los Angeles, CA 90089
{zhang.ke,weilunc}@usc.edu

Fei Sha
University of California
Los Angeles, CA 90095
feisha@cs.ucla.edu

Kristen Grauman
University of Texas at Austin
Austin, TX 78701
grauman@cs.utexas.edu

Abstract

Video summarization has unprecedented importance to help us digest, browse, and search today's ever-growing video collections. We propose a novel subset selection technique that leverages supervision in the form of human-created summaries to perform automatic keyframe-based video summarization. The main idea is to nonparametrically transfer summary structures from annotated videos to unseen test videos. We show how to extend our method to exploit semantic side information about the video's category/genre to guide the transfer process by those training videos semantically consistent with the test input. We also show how to generalize our method to subshot-based summarization, which not only reduces computational costs but also provides more flexible ways of defining visual similarity across subshots spanning several frames. We conduct extensive evaluation on several benchmarks and demonstrate promising results, outperforming existing methods in several settings.

1. Introduction

The amount of video data has been explosively increasing due to the proliferation of video recording devices such as mobile phones, wearable and ego-centric cameras, surveillance equipment, and others. According to YouTube statistics, about 300 hours of video are uploaded every minute [2]. To cope with this video data deluge, *automatic video summarization* has emerged as a promising tool to assist in curating video contents for fast browsing, retrieval, and understanding [14, 29, 45, 47], without losing important information.

Video can be summarized at several levels of abstraction: keyframes [10, 24, 27, 34, 48], segment or shot-based skims [23, 30, 35, 36], story-boards [6, 9], montages [16, 41] or video synopses [39]. In this paper, we focus on developing learning algorithms for selecting keyframes or subshots from a video sequence. Namely, the input is a video and its subshots and the output is an (ordered) subset of the frames or subshots in the video.

Inherently, summarization is a structured prediction problem where the decisions on whether to include or exclude certain frames or subshots into the subset are interdependent. This is in sharp contrast to typical classification and recognition tasks where the output is a single label.

The structured nature of subset selection presents a major challenge. Current approaches rely heavily on several heuristics to decide the desirability of each frame: representativeness [15, 17, 36], diversity or uniformity [28, 48], interestingness and relevance [16, 24, 30, 32, 36]. However, combining those frame-based properties to output an optimal subset remains an open and understudied problem. In particular, researchers are hampered by the lack of knowledge on the "global" criteria human annotators presumably optimize when manually creating a summary.

Recently, initial steps investigating *supervised* learning for video summarization have been made. They demonstrate promising results [10, 12], often exceeding the conventional unsupervised clustering of frames. The main idea is to use a training set of videos and human-created summaries as targets to adapt the parameters of a subset selection model to optimize the quality of the summarization. If successful, a strong form of supervised learning would extract high-level semantics and cues from human-created summaries to guide summarization.

Supervised learning for structured prediction is a challenging problem in itself. Existing parametric techniques typically require a complex model with sufficient annotated data to represent highly complicated decision regions in a combinatorially large output space. In this paper, we explore a *nonparametric* supervised learning approach to summarization. Our method is motivated by the observation that *similar videos share similar summary structures*. For instance, suppose we have a collection of videos of wedding ceremonies inside churches. It is quite likely good summaries for those videos would all contain frames portraying brides proceeding to the altar, standing of the grooms

* Equal contributions

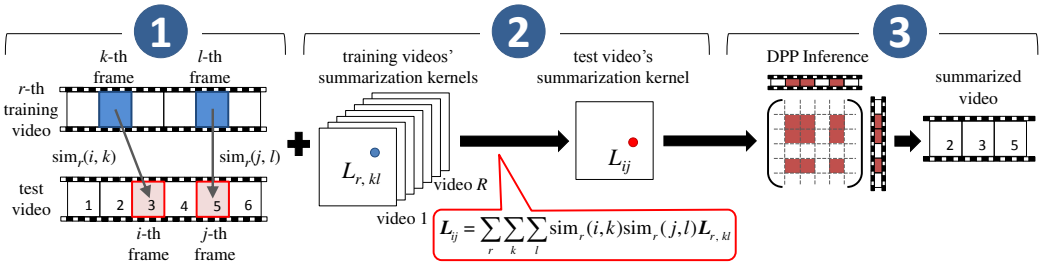

Figure 1. The conceptual diagram of our approach, which leverages the intuition that *similar videos share similar summary structures*. The main idea is nonparametric structure transfer, ie, transferring the subset structures in the human-created summaries (blue frames) of the training videos to a new video. Concretely, for each new video, we first compute frame-level similarity between training and test videos (i.e., $\mathsf{sim}(\cdot,\cdot)$, cf. eq. (4)). Then, we encode the summary structures in the training videos with kernel matrices made of binarized pairwise similarity among their frames. We combine those structures, factoring the pairwise similarity between the training and the test videos, into a kernel matrix that encodes the summary structure for the test video, cf. eq. (7). Finally, the summary is decoded by inputting the kernel matrix to a probabilistic model called the determinantal point process (DPP) to extract a globally optimal subset of frames.

and their best men, the priests' preaching, the exchange of rings, etc. Thus, if one such video is annotated with human-created summaries, a clever algorithm could essentially "copy and paste" the relative positions of the extracted frames in the annotated video sequence and apply them to an unannotated one to extract relevant frames. Note that this type of transferring summary structures across videos need *not* assume a precise matching of visual appearance in corresponding frames — there is no need to have the same priest as long as the frames of the priests in each video are sufficiently different from other frames to be "singled out" as possible candidates.

The main idea of our approach centers around this intuition, that is, *non-parametric learning from exemplar videos* to transfer summary structures to novel input videos. In recent years, non-parametric methods in the vision literature have shown great promise in letting the data "speak for itself", though thus far primarily for traditional categorization or regression tasks (e.g., label transfer for image recognition [26, 42] or scene completion [13]).

How can summarization be treated non-parametrically? A naive application of non-parametric learning to video summarization would treat keyframe selection as a binary classification problem—matching each frame in the unannotated test video to the nearest human-selected keyframes in some training video, and deciding independently per frame whether it should be included in or excluded from the summary. Such an approach, however, conceptually fails on two fronts. First, it fails to account for the relatedness *between* a summary's keyframes. Second, it limits the system to inputs having very similar frame-level matches in the annotated database, creating a data efficiency problem.

Therefore, rather than transfer simple relevance labels, our key insight is to transfer the *structures* implied by subset selection. We show how kernel-based representations of a video's frames (subshots) can be used to detect and align the meta-cues present in selected subsets. In this way, we compose novel summaries by borrowing recurring structures in exemplars for which we have seen both the source video and its human-created summary. A conceptual diagram of our approach is shown in Fig. 1.

In short, our main contributions are an original modeling idea that leverages non-parametric learning for structured objects (namely, selecting subsets from video sequences), a summarization method that advances the frontier of supervised learning for video summarization, and an extensive empirical study validating the proposed method and attaining far better summarization results than competing methods on several benchmark datasets.

The rest of the paper is organized as follows. In section 3, we describe our approach of nonparametric structure transfer. We report and analyze experimental results in section 4 and conclude in section 5.

2. Related Work

A variety of video summarization techniques have been developed in the literature [33, 43]. Broadly speaking, most methods first compute visual features at the frame level, then apply some selection criteria to prioritize frames for inclusion in the output summary.

Keyframe-based methods select a subset of frames to form a summary, and typically use low-level features like optical flow [44] or image differences [48]. Recent work also injects high-level information such as object tracks [27] or "important" objects [24], or takes user input to generate a storyboard [9]. In contrast, video skimming techniques first segment the input into subshots using shot boundary detection. The summary then consists of a selected set of representative subshots [23, 35, 36].

Selection criteria for summaries often aim to retain diverse and representative frames [15, 17, 28, 36, 48]. Another strategy is to predict object and event saliency [16, 24, 32, 36], or to pose summarization as an anomaly detection problem [19, 49]. When the camera is known to be

stationary, background subtraction and object tracking offer valuable cues about the salient entities in the video [5, 39].

Whatever the above choices, existing methods are almost entirely *unsupervised*. For example, they employ clustering to identify groups of related frames, and/or manually define selection criteria based on intuition for the problem. Some prior work includes supervised learning components (e.g., to generate regions with learned saliency metrics [24], train classifiers for canonical viewpoints [17], or recognize fragments of a particular event category [38]), but they do not learn the subset selection procedure itself.

Departing from unsupervised methods, limited recent work formulates video summarization as a subset selection problem [10, 12, 18, 46]. This enables supervised learning, exploiting knowledge encoded in human-created summaries. In [12], a submodular function optimizes a global objective function of the desirability of selected frames, while [10] uses a probabilistic model that maximizes the probability of the ground-truth subsets.

The novelty of our approach is to learn non-parametrically from exemplar training videos to transfer summary structures to test videos. In contrast to previous parametric models [10, 12], non-parametric learning generalizes to new videos by directly exploiting patterns in the training data. This has the advantage of generalizing locally within highly nonsmooth regions: as long as a test video's "neighborhood" contains an annotated training video, the summary structure of that training video will be transferred. In contrast, parametric techniques typically require a complex model with sufficient annotated data to parametrically represent those regions. Our non-parametric approach also puts design power into flexible kernel functions, as opposed to relying strictly on combinations of hand-crafted criteria (e.g., frame interestingness, diversity, etc.).

3. Approach

We cast the process of extracting a summary from a video as selecting a subset of items (i.e., video frames) from a ground set (i.e., the whole video). Given a corpus of videos and their human-created summaries, our learning algorithm learns the optimal criteria for subset selection and applies them to unseen videos to extract summaries.

The first step is to decide on a subset selection model that can output a structure (i.e., an ordered subset). For such structured prediction problems, we focus on the *determinantal point process* (DPP) [22] which has the benefits of being more computationally tractable than many probabilistic graphical models [20]. Empirically, DPP has been successfully applied to documentation summarization [21], image retrieval [7] and more recently, to video summarization [3, 10].

We will describe first DPP and how it can be used for video summarization. We then describe our main approach in detail, as well as its several extensions.

3.1. Background

Let $\mathcal{Y} = \{1, 2, \cdots, \mathsf{N}\}$ denote a (ground) set of N items, such as video frames. The ground set has 2^{N} subsets. The DPP over the N items assigns a probability to each of those subsets. Let $\boldsymbol{y} \subseteq \mathcal{Y}$ denote a subset and the probability of selecting it is given by

$$P(\boldsymbol{y}; \boldsymbol{L}) = \frac{\det(\boldsymbol{L}_{\boldsymbol{y}})}{\det(\boldsymbol{L} + \boldsymbol{I})}, \tag{1}$$

where $\boldsymbol{L}$ is a symmetric, positive semidefinite matrix and $\boldsymbol{I}$ is an identity matrix of the same size of $\boldsymbol{L}$. $\boldsymbol{L}_{\boldsymbol{y}}$ is the principal minor (sub-matrix) with rows and columns from $\boldsymbol{L}$ indexed by the integers in $\boldsymbol{y}$.

DPP can be seen conceptually as a fully connected N-node Markov network where the nodes correspond to the items. This network's node-potentials are given by the diagonal elements of $\boldsymbol{L}$ and the edge potentials are given by the off-diagonal elements in $\boldsymbol{L}$. Note that those "potentials" cannot be arbitrarily assigned — to ensure they form a valid probabilistic model, the matrix $\boldsymbol{L}$ needs to be positive semidefinite. Due to this constraint, $\boldsymbol{L}$ is often referred to as a kernel matrix whose elements can be interpreted as measuring the pairwise compatibility.

Besides computational tractability which facilitates parameter estimation, DPP has an important modeling advantage over standard Markov networks. Due to the celebrated Hammersley-Clifford Theorem, Markov networks cannot model distributions where there are zero-probability events. On the other hand, DPP is capable of assigning zero probability to absolutely impossible (or inadmissible) instantiations of random variables.

To see its use for video summarization, suppose there are two frames that are identical. For keyframe-based summarization, any subset containing such identical frames should be ruled out by being assigned zero probability. This is impossible in Markov networks — no matter how small, Markov networks will assign strictly positive probabilities to an exponentially large number of subsets containing identical frames. For DPP, since the two items are identical, they lead to two identical columns/rows in the matrix $\boldsymbol{L}$, resulting a determinant zero (thus zero probability) for those subsets. Thus, DPP naturally encourages selected items in the subset to be diverse, an important objective for summarization and information retrieval [22].

The mode of the distribution is the most probable subset

$$\boldsymbol{y}^* = \arg\max_{\boldsymbol{y}} P(\boldsymbol{y}; \boldsymbol{L}) = \arg\max_{\boldsymbol{y}} \det(\boldsymbol{L}_{\boldsymbol{y}}). \tag{2}$$

This is an NP-hard combinatorial optimization problem, and there are several approaches to obtaining approximate solutions [8, 22].

The most crucial component in a DPP is its kernel matrix $\boldsymbol{L}$. To apply DPP to video summarization, we define the ground set as the frames in a video and identify the most desired summarization as the MAP inference result of eq. (2). We compute $\boldsymbol{L}$ with a bivariate function over two frames — we dub it the *summarization kernel*:

$$L_{ij} = \boldsymbol{\phi}(\boldsymbol{v}_i)^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{v}_j) \tag{3}$$

where $\boldsymbol{\phi}(\cdot)$ is a function of the features $\boldsymbol{v}_i$ (or $\boldsymbol{v}_j$) computed on the i-th (or the j-th) frames. There are several choices. For instance, $\boldsymbol{\phi}(\cdot)$ could be an identity function, a nonlinear mapping implied by a Gaussian RBF kernel, or the output of a neural network [10].

As each different video needs to have a different kernel, $\boldsymbol{\phi}(\cdot)$ needs to be identified from a sufficiently rich function space so it generalizes from modeling the training videos to new ones. If the videos are substantially different, this generalization can be challenging, especially when there are not enough annotated videos with human-created summaries. Our approach overcomes this challenge by directly using the training videos' kernel matrices, as described below.

3.2. Non-parametric video summarization

Our approach differs significantly from existing summary methods, including those based on DPPs. Rather than learn a single function $\boldsymbol{\phi}(\cdot)$ and discard the training dataset, we construct $\boldsymbol{L}$ for every unannotated video by comparing it to the annotated training videos. This construction exploits two sources of information: 1) how similar the new video is to annotated ones, and 2) how the training videos are summarized. The former can be inferred directly by comparing visual features at each frame, while the latter can be "read off" from the human-created training summaries.

We motivate our approach with an idealized example that provides useful insight. Let us assume we are given a training set of videos and their summaries $\mathcal{D} = \{(\mathcal{Y}_r, \boldsymbol{y}_r)\}_{r=1}^{\mathsf{R}}$ and a new video $\mathcal{Y}$ to be summarized. Suppose this new video is very similar — we define similarity more precisely later — to a particular $\mathcal{Y}_r$ in $\mathcal{D}$. Then we can reasonably assume that the summary $\boldsymbol{y}_r$ might work well for the new video. As a concrete example, consider the case where both $\mathcal{Y}$ and $\mathcal{Y}_r$ are videos for wedding ceremonies inside churches. We anticipate seeing similar events across both videos: brides proceeding to the altars, priests delivering speeches, exchanging rings etc. Moreover, similarity in their summaries exists on the higher-order *structural* level: the relative positions of the summary frames of $\boldsymbol{y}_r$ in the sequence $\mathcal{Y}$ are an informative prior on where the frames of the summary $\boldsymbol{y}$ should be in the new video $\mathcal{Y}$. Specifically, as long as we can link the test video to the training video by identifying similar frames,[1] we can "copy down"—transfer—the positions of $\boldsymbol{y}_r$ and lift the corresponding frames in $\mathcal{Y}$ to generate its summary $\boldsymbol{y}$.

While this intuition is conceptually analogous to the familiar paradigm of nearest-neighbor classification, our approach is significantly different. The foremost is that, as discussed in section 1, we cannot select frames independently (by nonparametrically learning its similarity to those in the training videos). We need to transfer summary structures which encode interdependencies of selecting frames. Therefore, a naive solution of representing videos with fix-length descriptors in Euclidean space and literally pretending their summaries are "labels" that can be transferred to new data is flawed.

The main steps of our approach are outlined in Fig. 1. We describe them in detail in the following.

Step 1: Frame-based visual similarity To infer similarity across videos, we experiment with common ones in the computer vision literature for calculating frame-based similarity from visual features $\boldsymbol{v}_i$ and $\boldsymbol{v}_k$ extracted from the corresponding frames:

$$\begin{aligned}
\mathsf{sim}^1(i,k) &= \boldsymbol{v}_i^{\mathrm{T}}\boldsymbol{v}_k \\
\mathsf{sim}^2(i,k) &= \exp\{-\|\boldsymbol{v}_i - \boldsymbol{v}_k\|_2^2/\sigma\} \\
\mathsf{sim}^3(i,k) &= \exp\{-(\boldsymbol{v}_i - \boldsymbol{v}_k)^{\mathrm{T}}\boldsymbol{\Omega}(\boldsymbol{v}_i - \boldsymbol{v}_k)\},
\end{aligned} \tag{4}$$

where σ and $\boldsymbol{\Omega}$ are adjustable parameters (constrained to be positive or positive definite, respectively). These forms of similarity measures are often used in vision tasks and are quite flexible, e.g., one can learn the kernel parameters for sim^3. However, they are not the focus of our approach — we expect more sophisticated ones will only benefit our learning algorithm. We also expect high-level features (such as interestingness, objectness, etc.) could also be beneficial. In section 3.4 we discuss a generalization to replace frame-level similarity with subshot-level similarity.

Step 2: Summarization kernels for training videos The summarization kernels $\{\boldsymbol{L}_r\}_{r=1}^{\mathsf{R}}$ are not given to us in the training data. However, note that the crucial function of those kernels is to ensure that when used to perform the MAP inference in eq. (2) to identify the summary on the training video $\mathcal{Y}_r$, it will lead to the correct summarization $\boldsymbol{y}_r$ (which is in the training set). This prompts us to define the following *idealized summarization kernels*

$$\boldsymbol{L}_r = \alpha_r \begin{bmatrix} \delta(1 \in \boldsymbol{y}_r) & 0 & \cdots & 0 \\ 0 & \delta(2 \in \boldsymbol{y}_r) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \delta(\mathsf{N}_r \in \boldsymbol{y}_r) \end{bmatrix} \tag{5}$$

[1] This task is itself not trivial, of course, but it does have the benefit of a rich literature on image matching and recognition work, including efficient search strategies.

or more compactly,

$$\boldsymbol{L}_r = \alpha_r \mathsf{diag}(\{\delta(n \in \boldsymbol{y}_r)\}_{n=1}^{\mathsf{N}_r}), \tag{6}$$

where diag turns a vector into a diagonal matrix, N_r is the number of frames in $\mathcal{Y}_r$ and $\alpha_r > 1$ is an adjustable parameter. The structure of $\boldsymbol{L}_r$ is intuitive: if a frame is in the summary $\boldsymbol{y}_r$, then its corresponding diagonal element is α_r, otherwise 0. It is easy to verify that $\boldsymbol{L}_r$ indeed gives rise to the correct summarization. Note that $\alpha_r > 1$ is required. If $\alpha_r = 1$, any subset of $\boldsymbol{y}_r$ is a solution to the MAP inference problem (and we will be getting a shorter summarization). If $\alpha_r < 1$, the empty set would be the summary (as the determinant of an empty matrix is 1, by convention).

Step 3: Transfer summary structure Our aim is now to transfer the structures encoded by the idealized summarization kernels from the training videos to a new (test) video $\mathcal{Y}$. To this end, we synthesize $\boldsymbol{L}$ for new video $\mathcal{Y}$ out of $\{\boldsymbol{L}_r\}$.

Let i and j index the video frames in $\mathcal{Y}$, with k and l for a specific training video $\mathcal{Y}_r$. Specifically, the "contribution" from $\mathcal{Y}_r$ to $\boldsymbol{L}$ is given by

$$r_{ij} = \sum_k \sum_l \mathsf{sim}_r(i,k)\mathsf{sim}_r(j,l)L_{r,kl} \tag{7}$$

where $L_{r,kl}$ is the element in $\boldsymbol{L}_r$, and $\mathsf{sim}_r(\cdot,\cdot)$ measures frame-based (visual) similarity between frames of $\mathcal{Y}$ and $\mathcal{Y}_r$.

Fig. 1 illustrates graphically how frame-based similarity enables transfer of structures in training summaries. We gain further insights by examining the case when the frame-based similarity $\mathsf{sim}_r(\cdot,\cdot)$ is sharply peaked — namely, there are very good matches between specific pairs of frames (an assumption likely satisfied in the running example of summarizing wedding ceremony videos)

$$\begin{aligned} \mathsf{sim}_r(i,m) &\gg \mathsf{sim}_r(i,k), \quad \forall\, k \neq m \\ \mathsf{sim}_r(j,n) &\gg \mathsf{sim}_r(j,l), \quad \forall\, l \neq n. \end{aligned} \tag{8}$$

Under these conditions,

$$r_{ij} \approx \mathsf{sim}_r(i,m)\mathsf{sim}_r(j,n)L_{r,mn}. \tag{9}$$

Intuitively, if $\mathcal{Y}$ and $\mathcal{Y}_r$ are precisely the same video (and the video frames in $\mathcal{Y}_r$ are sufficiently visually different), then the matrix $\boldsymbol{L}$ would be very similar to $\boldsymbol{L}_r$. Consequently, the summarization $\boldsymbol{y}_r$, computed from $\boldsymbol{L}_r$, would be a good summary for $\mathcal{Y}$.

To include all the information in the training data, we sum up the contributions from all $\mathcal{Y}_r$ and arrive at

$$L_{ij} = \sum_r r_{ij}. \tag{10}$$

We introduce a few shorthand notions. Let $\boldsymbol{S}_r$ be a $\mathsf{N} \times \mathsf{N}_r$ matrix whose elements are $\mathsf{sim}_r(i,k)$, the frame-based similarity between N frames in $\mathcal{Y}$ and N_r frames in $\mathcal{Y}_r$. The kernel matrix $\boldsymbol{L}$ is thus

$$\boldsymbol{L} = \sum_r \boldsymbol{S}_r \boldsymbol{L}_r \boldsymbol{S}_r^{\mathrm{T}}. \tag{11}$$

Note that, $\boldsymbol{L}$ is for the test video with N frames — there is no need for all the videos have the same length.

Step 4: Extracting summary Once we have computed $\boldsymbol{L}$ for the new video, we use the MAP inference eq. (2) to extract the summary as the most probable subset of frames.

3.3. Learning

Our approach requires adjusting parameters such as $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \cdots, \alpha_{\mathsf{R}}\}$ for the ideal summarization kernels and/or $\boldsymbol{\Omega}$ for computing frame-based visual similarity eq. (4). We use maximum likelihood estimation to estimate those parameters. Specifically, for each video in the training dataset, we pretend it is a new video and formulate a kernel matrix

$$\hat{\boldsymbol{L}}_q = \sum_r \boldsymbol{S}_r^q \boldsymbol{L}_r \boldsymbol{S}_r^{q\mathrm{T}}, \forall, q = 1, 2, \cdots, \mathsf{R}. \tag{12}$$

We optimize the parameters such that the ground-truth summarization $\boldsymbol{y}_q$ attains the highest probability under $\hat{\boldsymbol{L}}_q$,

$$\boldsymbol{\alpha}^* = \arg\max_{\boldsymbol{\alpha}} \sum_{q=1}^{\mathsf{R}} \log P(\boldsymbol{y}_q; \hat{\boldsymbol{L}}_q). \tag{13}$$

We can formulate a similar criterion to learn the $\boldsymbol{\Omega}$ parameter for $\mathsf{sim}^3(\cdot,\cdot)$. We carry out the optimization by gradient descent. In our experiments, we set σ for sim^2 to be 1, with features normalized to have unit norm. Additional details are in the Suppl. and omitted here for brevity.

3.4. Extensions

Category-specific summary transfer Video datasets labeled with semantically consistent categories have been emerging [38, 40]. We view categories as valuable prior information that can be exploited by our nonparametric learning algorithm. Intuitively, videos from the same category (activity type, genre, etc.) are likely to be similar in part, not only in visual appearance but also in high-level semantic cues (such as how key events are temporally organized), resulting in a similar summary structures. We extend our method to take advantage such optional side information in two ways:

- hard transfer. We compare the new video from a category c only to the training videos from the same

category c. Mathematically, for each video category, we learn a category-specific set of $\boldsymbol{\alpha}^{(c)} = \{\alpha_1^{(c)}, \alpha_2^{(c)}, \cdots, \alpha_{\mathsf{R}}^{(c)}\}$ such that $\alpha_r^{(c)} > 0$ only when the training video r belongs to category c.

- soft transfer. We relax the requirement in hard transfer such that $\alpha_r^{(c)} > 0$ even if the rth training video is not from the category c. Note that while we utilize structural information from all training videos, the way we use them still depends on the test video's category.

Subshot-based summary transfer Videos can also be summarized at the level of subshots. As opposed to selecting keyframes, subshots contain short but contiguous frames, giving a glimpse of a key event. We next extend our subset selection algorithm to select a subset of subshots.

To this end, in our conceptual diagram as in Fig. 1, we replace computing frame-level similarity with subshot-level similarity, where we compare subshots between the training videos and the new video. We explore two possible ways to compute the frame-set to frame-set similarity:

- Similarity between averaged features. We represent the subshots using the averaged frame-level feature vectors within each subshot. We then compute the similarity using the previously defined $\mathsf{sim}(\cdot,\cdot)$.
- Maximum similarity. We compute pairwise similarity between frames within the subshots and select the maximum value as the similarity between the subshots.

Both of these two approaches reduce the reliance on frame-based similarity defined in the global frame-based descriptors of visual appearance, loosening the required visual alignment for discovering a good match—especially with the latter max operator, in principle, since it can ignore many unmatchable frames in favor of a single strong link within the subshots. Moreover, the first approach can significantly reduce the computational cost of nonparametric learning as the amount of pairwise-similarity computation now depends on the number of subshots, which is substantially smaller than the number of frames.

3.5. Implementation and computation cost

Computing $\boldsymbol{S}_r$ in eq. (11) is an $O(\mathsf{N} \times \sum_r \mathsf{N}_r)$ operation. For long videos, several approaches will reduce the cost significantly. First, it is a standard procedure to downsample the video (by a factor of 5-30) to reduce the number of frames for keyframe-based summarization. Our subshot-based summarization can also reduce the computation cost, cf. section 3.4. Generic techniques should also help — $\mathsf{sim}(\cdot,\cdot)$ computes various forms of distances among visual feature vectors. Thus, many fast search techniques apply, such as locality sensitive hashing or tree structures for nearest neighbor searches.

4. Experiment

We validate our approach on five benchmark datasets. It outperforms competing methods in many settings. We also analyze its strengths and weaknesses.

4.1. Setup

Data For *keyframe-based* summarization, we experiment on three video datasets: the **Open Video Project (OVP)** [1, 4], the **YouTube** dataset [4], and the **Kodak** consumer video dataset [31]. All the 3 datasets were used in [10] and we follow the procedure described there to preprocess the data, and to generate training ground-truths from multiple human-created summaries. For the YouTube dataset, in the following, we report results on 31 videos after discarding 8 videos that are neither "Sports" nor "News" such that we can investigate category-specific video summarization (cf. sec. 3.4). In Suppl., we report results on the original dataset.

For *subshot-based* summarization (cf. sec. 3.4), we experiment on three video datasets: the portion of **MED** with 160 annotated summaries [38], **SumMe** [11] and **YouTube**. Videos in MED are pre-segmented into subshots with the Kernel Temporal Segmentation (KTS) method [38] and we observe those subshots. For SumMe and YouTube, we apply KTS to generate our own sets of subshots. MED has 10 well-defined video categories allowing us to experiment with category-specific video summarization on it too. SumMe does not have semantic category meta-data. Instead, its video contents have a large variation in visual appearance and can be classified according to shooting style: still camera, egocentric, or moving. Table 1 summarizes key characteristics of those datasets with details in the Suppl.

Features For OVP/YouTube/Kodak/SumMe, we encode each frame with an ℓ^2-normalized 8192-dimensional Fisher vector [37], computed from SIFT features. For OVP/YouTube/Kodak, we also use color histograms. We also experimented with features from a pre-trained convolutional nets (CNN), details in the Suppl. For MED, we use the provided 16512-dimensional Fisher vectors.

Evaluation metrics As in [10, 11, 12] and other prior work, we evaluate automatic summarization results (A) by comparing them to the human-created summaries (B) and reporting the standard metrics of F-score (F), precision (P), and recall (R) — their definitions are given in the Suppl.

For OVP/YouTube/Kodak, we follow [10] and utilize the VSUMM package [4] for finding matching pairs of frames. For SumMe, we follow the procedure in [11, 12]. More details are in the Suppl.

Implementation details For each dataset, we randomly choose 80% of the videos for training and use the remain-

Table 1. Key characteristics of datasets used in our empirical studies. Most videos in these datasets have a duration from 1 to 5 minutes.

Dataset	# of video	# of category	# of Training videos	# of Test video	Type of summarization	Evaluation metrics F-score in matching
Kodak	18	-	14	4	keyframe	selected frames
OVP	50	-	40	10	keyframe	
Youtube	31	2	31	8	keyframe; subshot	selected frames; frames in selected subshots
SumMe	25	-	20	5	subshot	frames in selected subshots
MED	160	10	128	32	subshot	matching selected subshots

ing 20% for testing, repeating for 5 or 100 rounds so that we can calculate averaged performance and standard errors. To report existing methods' results, we use prior published numbers when possible. We also implement the VSUMM approach [4] and obtained code from the authors for seqDPP [10] in order to apply them to several datasets. We follow the practices in [10] so that we can summarize videos sequentially. For MED, we implement KVS [38] ourselves.

4.2. Main Results

We summarize our key findings in this section. For more details, please refer to the Suppl.

In Table 2, we compare our approach to both supervised and unsupervised methods for video summarization. We report published results in the table as well as results from our own implementation of some methods. Only the best variants of all methods are quoted and presented; others are deferred to the Suppl.

On all but one of the five datasets (OVP), our nonparametric learning method achieves the best results. In general, the supervised methods achieve better results than the unsupervised ones. Note that even for datasets with a variety of videos that are not closely visually similar (such as SumMe), our approach attains the best result—it indicates our method of transferring summary structures is effective, able to build on top of even crude frame similarities.

4.3. Detailed analysis

Advantage of nonparametric learning Nonparametric learning enjoys the property of generalizing locally. That is, as long as a test video has enough correctly annotated training videos in its neighborhood, the summary structures of those videos will transfer. A parametric learning method, on the other end, needs to learn both the locations of those local regions and how to generalize within local regions. If there are not enough annotated training videos covering the whole range of data space, it could be difficult for a parametric learning method to learn well.

We design a simple example to illustrate this point. As it is difficult to assess "similarity" to derive nearest neighbors for video, we use a video's category to delineate those "semantically near". Specifically, we split YouTube's 31 videos into two piles, according to their categories "Sports" or "News". We then construct seqDPP, a parametric learning model [10], using all the 31 videos, as well as "Sports" or "News" videos only to summarize test videos from either category. We then contrast to our method in the same setting. Table 3 displays the results.

Table 3. Advantage of nonparametric summary transfer

Type of test video	seqDPP		Ours	
	all	same as test	all	same as test
Sports	52.8	54.5	53.5	54.4
News	67.9	67.7	66.9	69.1

Table 4. Video category information helps summarization

Setting	YouTube	MED	SumMe
w/o category	60.0	28.9	39.2
category-specific hard	61.5	30.4	40.9
category-specific soft	60.6	30.7	40.2

The results on the "News" category convincingly suggest that the nonparametric approach like ours can leverage the semantically-close videos to outperform the parametric approach with the same amount of annotated data—or even more data. (Note that, the difference on the "Sports" category is nearly identical.)

Advantage of exploiting category prior Table 3 already alludes to the fact that exploiting category side-information can improve summarization (cf. contrasting the column of "same as test" to "all"). Now we investigate this advantage in more detail. Table 4 shows how our nonparametric summary transfer can exploit video category information, using the method explained in section 3.4.

Particularly interesting are our results on the SumMe dataset, which itself does not provide semantically meaningful categories. Instead, we generate two "fake" categories for it. We first collapse the 10 video categories in the dataset TVSum[2] [40] into two super-categories (details in Suppl.) — these two super-categories are semantically similar within each other, though they do not have obvious visual similarity to videos in the SumMe.

We then build a binary classifier trained on TVSum videos but classify the videos in SumMe as "super-category I" and "super-category II" and then proceed as if they are ground-truth categories, as in MED and YouTube. Despite this independently developed dichotomy, the results on SumMe improve over using all video data together.

[2]We choose this one as it has raw videos for us to extract features and have a larger number of labeled videos for us to build a category classifier

Table 2. Performance (F-score) of various video summarization methods. Numbers followed by citations are from published results. Others are from our own implementation. Dashes denote unavailable/inapplicable dataset-method combinations.

	Unsupervised						Supervised			
	VSUMM$_1$ [4]	VSUMM$_2$ [4]	DT [34]	STIMO [6]	KVS [38]	Video MMR [25]	SumMe [11]	Submodular [12]	seqDPP [10]	**Ours**
Kodak	69.5	67.6	-	-	-	-	-	-	78.9	**82.3**
OVP	70.3	69.5	57.6	63.4	-	-	-	-	**77.7**	76.5
YouTube	58.7	59.9	-	-	-	-	-	-	60.8	**61.8**
MED	28.9	28.8	-	-	20.6	-	-	-	-	**30.7**
SumMe	32.8	33.7	-	-	-	26.6	39.3	39.7	-	**40.9**

Table 5. Subshot-based summarization on YouTube

Category -specific	Frame-based	Subshot-based	
		Mean-Feature	Max-similarity
No	60.0	60.7	60.9
Yes	61.5	61.6	61.8

Subshot-based summarization In section 3.4, we discuss an extension to summarize video at the level of selecting subshots. This extension not only reduces computational cost (as the number of subshots is significantly smaller than that of frames), but also provides additional means of measuring similarity across videos beyond frame-level visual similarity inferred from global frame-based descriptors. Next we examine how such flexibility can ultimately improve keyframe-based summarization. Concretely, we first perform subshot summarization, then pick the middle frame in each selected subshot as the output keyframes. This allows us to directly compare to keyframe-based summarization using the same F-score metric.

Table 5 shows the results. Subshot-based summarization clearly improves frame-based — this is very likely due to the more robust similarity measures now computed at the subshot-level. The improvement is more pronounced when a category prior is not used. One possible explanation is that measuring similarity on videos from the same categories is easier and more robust, whereas across categories it is noisier. Thus, when a category prior is not present, the subshot-based similarity measure benefits summarization more.

Other detailed analysis in Suppl. We summarize other analysis results as follows. We show how to improve frame-level similarity by learning better metrics. We also show deep features, powerful for visual category recognition, is not particularly advantageous comparing to shallow features. We also show how category prior can still be exploited even we do not know the true category of test videos.

4.4. Qualitative analysis

Fig. 2 shows a failure case by our method. Here the test video depicts a natural scene, while its closest training video depicts beach activities. There is a visual similarity (e.g., in the swath of sky). However, semantically, these two videos do not seem to be relevant and it is likely difficult for the transfer to occur. In particular, our results miss the last two frames where there are a lot of grass. This suggests one weakness in our approach: the formulation of our summarization kernel for the test video does not directly consider the relationship between its own frames — instead, they interact through training videos. Thus, one possible direction to avoid unreliable neighbors in the training videos is to rely on the visual property of the test video itself. This suggests future work on a hybrid approach with parametric and non-parametric aspects that complement each other.

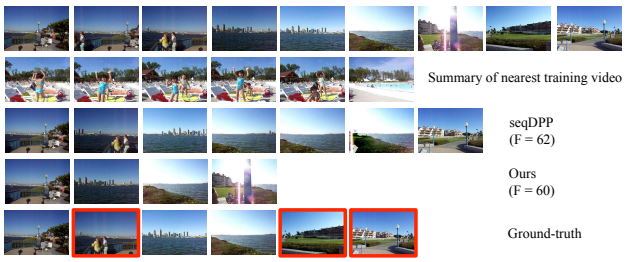

Figure 2. A failure case by our approach. Our summary misses the last two frames from the ground-truth (red-boxed) as the test video (nature scene) transfers from the nearest video with a semantically different category (beach activity). See text.

Please see the Suppl. for more qualitative analysis and example output summaries.

5. Conclusion

We propose a novel supervised learning technique for video summarization. The main idea is to learn non-parametrically to transfer summary structures from training videos to test ones. We also show how to exploit side (semantic) information such as video categories and propose an extension for subshot-based summarization. Our method achieves promising results on several benchmark datasets, compared to an array of nine existing techniques.

Acknowledgements

KG is partially supported by NSF IIS - 1514118. Others are partially supported by USC Annenberg and Viterbi Graduate Fellowships, NSF IIS - 1451412, 1513966, and CCF-1139148.

References

[1] Open video project. http://www.open-video.org/. 6

[2] Youtube statistics: https://www.youtube.com/yt/press/statistics.html. 1

[3] W.-L. Chao, B. Gong, K. Grauman, and F. Sha. Large-margin determinantal point processes. In *UAI*, 2015. 3

[4] S. E. F. de Avila, A. P. B. Lopes, A. da Luz, and A. de Albuquerque Araújo. Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern Recognition Letters*, 32(1):56–68, 2011. 6, 7, 8

[5] S. Feng, Z. Lei, D. Yi, and S. Z. Li. Online content-aware video condensation. In *CVPR*, 2012. 3

[6] M. Furini, F. Geraci, M. Montangero, and M. Pellegrini. Stimo: Still and moving video storyboard for the web scenario. *Multimedia Tools and Applications*, 46(1):47–69, 2010. 1, 8

[7] J. Gillenwater, A. Kulesza, and B. Taskar. Discovering diverse and salient threads in document collections. In *EMNLP*, 2012. 3

[8] J. Gillenwater, A. Kulesza, and B. Taskar. Near-optimal MAP inference for determinantal point processes. In *NIPS*, 2012. 3

[9] D. B. Goldman, B. Curless, D. Salesin, and S. M. Seitz. Schematic storyboarding for video visualization and editing. *ACM Transactions on Graphics*, 25(3):862–871, 2006. 1, 2

[10] B. Gong, W.-L. Chao, K. Grauman, and F. Sha. Diverse sequential subset selection for supervised video summarization. In *NIPS*, 2014. 1, 3, 4, 6, 7, 8

[11] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool. Creating summaries from user videos. In *ECCV*, 2014. 6, 8

[12] M. Gygli, H. Grabner, and L. Van Gool. Video summarization by learning submodular mixtures of objectives. In *CVPR*, 2015. 1, 3, 6, 8

[13] J. Hays and A. A. Efros. Scene completion using millions of photographs. *ACM Transactions on Graphics*, 26(3):4, 2007. 2

[14] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 1

[15] R. Hong, J. Tang, H.-K. Tan, S. Yan, C. Ngo, and T.-S. Chua. Event driven summarization for web videos. In *SIGMM Workshop*, 2009. 1, 2

[16] H.-W. Kang, Y. Matsushita, X. Tang, and X.-Q. Chen. Space-time video montage. In *CVPR*, 2006. 1, 2

[17] A. Khosla, R. Hamid, C.-J. Lin, and N. Sundaresan. Large-scale video summarization using web-image priors. In *CVPR*, 2013. 1, 2, 3

[18] G. Kim, L. Sigal, and E. P. Xing. Joint summarization of large-scale collections of web images and videos for storyline reconstruction. In *CVPR*, 2014. 3

[19] J. Kim and K. Grauman. Observe locally, infer globally: a space-time mrf for detecting abnormal activities with incremental updates. In *CVPR*, 2009. 2

[20] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009. 3

[21] A. Kulesza and B. Taskar. Learning determinantal point processes. In *UAI*, 2011. 3

[22] A. Kulesza and B. Taskar. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2–3), 2012. 3

[23] R. Laganière, R. Bacco, A. Hocevar, P. Lambert, G. Païs, and B. E. Ionescu. Video summarization from spatio-temporal features. In *ACM TRECVid Video Summarization Workshop*, 2008. 1, 2

[24] Y. J. Lee, J. Ghosh, and K. Grauman. Discovering important people and objects for egocentric video summarization. In *CVPR*, 2012. 1, 2, 3

[25] Y. Li and B. Merialdo. Multi-video summarization based on videommr. In *WIAMIS Workshop*, 2010. 8

[26] C. Liu, J. Yuen, and A. Torralba. Nonparametric scene parsing via label transfer. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(12):2368–2382, 2011. 2

[27] D. Liu, G. Hua, and T. Chen. A hierarchical visual model for video object summarization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(12):2178–2190, 2010. 1, 2

[28] T. Liu and J. R. Kender. Optimization algorithms for the selection of key frame sequences of variable length. In *ECCV*, 2002. 1, 2

[29] Y. Lu, X. Bai, L. Shapiro, and J. Wang. Coherent parametric contours for interactive video object segmentation. In *CVPR*, 2016. 1

[30] Z. Lu and K. Grauman. Story-driven summarization for egocentric video. In *CVPR*, 2013. 1

[31] J. Luo, C. Papin, and K. Costello. Towards extracting semantically meaningful key frames from personal video clips: from humans to computers. *Circuits and Systems for Video Technology, IEEE Transactions on*, 19(2):289–301, 2009. 6

[32] Y.-F. Ma, L. Lu, H.-J. Zhang, and M. Li. A user attention model for video summarization. In *ACM Multimedia*, 2002. 1, 2

[33] A. G. Money and H. Agius. Video summarisation: A conceptual framework and survey of the state of the art. *Journal of Visual Communication and Image Representation*, 19(2):121–143, 2008. 2

[34] P. Mundur, Y. Rao, and Y. Yesha. Keyframe-based video summarization using delaunay clustering. *International Journal on Digital Libraries*, 6(2):219–232, 2006. 1, 8

[35] J. Nam and A. H. Tewfik. Event-driven video abstraction and visualization. *Multimedia Tools and Applications*, 16(1-2):55–77, 2002. 1, 2

[36] C.-W. Ngo, Y.-F. Ma, and H. Zhang. Automatic video summarization by graph modeling. In *ICCV*, 2003. 1, 2

[37] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007. 6

[38] D. Potapov, M. Douze, Z. Harchaoui, and C. Schmid. Category-specific video summarization. In *ECCV*, 2014. 3, 5, 6, 7, 8

[39] Y. Pritch, A. Rav-Acha, A. Gutman, and S. Peleg. Webcam synopsis: Peeking around the world. In *ICCV*, 2007. 1, 3

[40] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes. Tvsum: Summarizing web videos using titles. In *CVPR*, 2015. 5, 7

[41] M. Sun, A. Farhadi, B. Taskar, and S. Seitz. Salient montages from unconstrained videos. In *ECCV*, 2014. 1

[42] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(11):1958–1970, 2008. 2

[43] B. T. Truong and S. Venkatesh. Video abstraction: A systematic review and classification. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 3(1):3, 2007. 2

[44] W. Wolf. Key frame selection by motion analysis. In *ICASSP*, 1996. 2

[45] Z. Wu, Y. Fu, Y.-g. Jiang, and L. Sigal. Harnessing object and scene semantics for large-scale video understanding. In *CVPR*, 2016. 1

[46] J. Xu, L. Mukherjee, Y. Li, J. Warner, J. M. Rehg, and V. Singh. Gaze-enabled egocentric video summarization via constrained submodular maximization. In *CVPR*, 2015. 3

[47] Y. Yang, G. Sundaramoorthi, and S. Soatto. Self-occlusions and disocclusions in causal video object segmentation. In *ICCV*, 2015. 1

[48] H. J. Zhang, J. Wu, D. Zhong, and S. W. Smoliar. An integrated system for content-based video retrieval and browsing. *Pattern recognition*, 30(4):643–658, 1997. 1, 2

[49] B. Zhao and E. P. Xing. Quasi real-time summarization for consumer videos. In *CVPR*, 2014. 2