Greedy Salient Dictionary Learning for Activity Video Summarization*

Ioannis Mademlis, Anastasios Tefas, and Ioannis Pitas

Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece

Abstract. Automated video summarization is well-suited to the task of analysing human activity videos (e.g., from surveillance feeds), mainly as a pre-processing step, due to the large volume of such data and the small percentage of actually important video frames. Although key-frame extraction remains the most popular way to summarize such footage, its successful application for activity videos is obstructed by the lack of editing cuts and the heavy inter-frame visual redundancy. Salient dictionary learning, recently proposed for activity video key-frame extraction, models the problem as the identification of a small number of video frames that, simultaneously, can best reconstruct the entire video stream and are salient compared to the rest. In previous work, the reconstruction term was modelled as a Column Subset Selection Problem (CSSP) and a numerical, SVD-based algorithm was adapted for solving it, while video frame saliency, in the fastest algorithm proposed up to now, was also estimated using SVD. In this paper, the numerical CSSP method is replaced by a greedy, iterative one, properly adapted for salient dictionary learning, while the SVD-based saliency term is retained. As proven by the extensive empirical evaluation, the resulting approach significantly outperforms all competing key-frame extraction methods with regard to speed, without sacrificing summarization accuracy. Additionally, computational complexity analysis of all salient dictionary learning and related methods is presented.

Keywords: Key-frame Extraction · Dictionary Learning · Column Subset Selection Problem · Video Summarization.

1 Introduction

Videos depicting human activities may come from different sources, such as surveillance feeds or movie/TV shooting sessions. They typically extend to many hours of footage which must be manually browsed in order to retain the most interesting parts. Video summarization algorithms may help in automating a large part of this tedious and labour-intensive process, by producing a short summary of the video input. However, activity videos, which can be considered as temporal concatenations of consecutive activity segments, share certain properties which make automated summarization difficult compared to other video types (such as movies [17]): lack of clear editing cuts,

* The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement numbers 287674 (3DTVS) and 316564 (IMPART).

static camera and static background resulting in heavy visual redundancy between video frames, as well as increased subjectivity in identifying important video frames (there is no clear way to proclaim a specific part of a human action as more representative than another one). Potential sources of such videos are surveillance cameras, capture sessions in TV/movie production, etc.

Video summarization algorithms are expected to achieve a balance between different needs, such as sufficient summary compactness (lack of redundancy), conciseness, outlier inclusion, semantic representativeness and content coverage. Despite the fact that many different ways to summarize a video exist (e.g., skimming [17], shot selection [16], synopsis [27], or temporal video segmentation [26, 28]), *key-frame extraction*, i.e., producing a temporally ordered subset of the original video frame set that in some sense contains the most important and representative visual content, remains the most widely applicable video summarization method. In fact, it is unavoidable if the selected subset of original video frames must be retained in unprocessed form, since in such a case video synopsis (which results in synthetic video frames, each one aggregating content from multiple original video frames) cannot be applied. Moreover, simple temporal video segmentation does not result in a summary per se, but is only a substitute of shot cut/boundary detection [4], while skimming requires key-frame extraction as an initial step. Thus, in the context of this paper, the terms "video summarization" and "key-frame extraction", as well as the terms "video summary" and "key-frame set", are hereafter used as synonymous (although this is not true in general). In actual method deployment, the extracted key-frames could be temporally extended to key-segments and then concatenated, so as to form a video skim.

Although supervised key-frame extraction methods, attempting to implicitly learn how to produce static summaries from human-created manual video summaries, have recently appeared due to the success of deep learning [32], they suffer from the subjectiveness inherent in the problem (different persons may produce widely differing summaries for the same video source) and the lack of manual activity video summaries readily available for training in most use-cases. Indeed, no specific video frame of an activity video segment can be reliably considered as more important than another one from the same segment. A more natural summarization goal would be for the algorithm to select one key-frame per actual activity segment, with the video frames belonging to the same segment considered as fully interchangeable. Therefore, this paper focuses on unsupervised key-frame extraction for activity video summarization and employs an objective evaluation metric that takes the above into account.

Two main algorithm families have emerged for unsupervised key-frame extraction over the years. The first one consists in distance-based data partitioning via video frame clustering, under the assumption that video shooting focuses more on important video frames [33]. The number of clusters is either pre-defined by the user or may depend proportionally on the video length [9]. The cluster medoids are selected as key-frames, in a manner dependent on the underlying clustering algorithm. The second algorithm family consists in dictionary-of-representatives approaches, where the original video frames are assumed to be approximately composed of linear combinations of a representative subset of them. These "dictionary frames" are detected and employed as key-frames [10].

In all cases, video frames are either represented by a raw, vectorized form of their unaltered pixel values (e.g., in [8]), or they are initially described by low-level/mid-level global or local image descriptors [13, 14], with sparse local descriptors typically aggregated under a representation scheme such as Bag-of-Features (BoF) [7]. High-level semantic video frame representations, learnt via deep neural networks, have also been tested [23].

Video frame clustering implicitly models summarization as a frame sampling problem, where criteria such as compactness, outlier inclusion and video content coverage should be met. Scene semantics are not considered and semantics extraction is entirely offloaded to the underlying, employed video frame description/representation scheme. Although clustering is a baseline key-frame extraction method, it still dominates the relevant literature due to its simplicity, straightforward problem modelling and relatively good accuracy.

In contrast, dictionary-of-representatives methods inherently consider scene semantics in an unsupervised manner, since they decompose the video into isolated visual building blocks. In [6, 22] the video summarization problem is formulated as sparse dictionary learning, with extracted key-frames ideally enabling optimal linear reconstruction of the original video from the selected dictionary. In both cases, the outliers are entirely disregarded.

In [10] a similar approach is followed, via sparse modeling representative selection. In [8] RPCA-KFE is presented, a key-frame extraction algorithm that takes into account both the contribution to video reconstruction and the distinctness of each video frame. The idea is to select as a summary the subset of video frames that simultaneously minimizes the aggregate reconstruction error and maximizes the total distinctness. However, the distinctness term is defined very inflexibly and is bound to the reconstruction term in a complementary manner.

Very recently, salient dictionary learning was proposed as a way to generalize dictionary-of-representatives approaches for activity key-frame extraction [19]. The key-frame set is extracted by simultaneously optimizing the desired summary for maximum reconstructive ability and maximum saliency. Activity videos are especially suited to such an approach, since human activities can be easily decomposed into approximately linear combinations of elementary actions [1], but on the other hand they contain a significant number of uninteresting/non-salient video frames that, nonetheless, convey large reconstructive advantage (e.g., video frames solely depicting the static background, or containing mostly human body poses common to multiple activity segments).

Following preliminary work in [18], where no saliency term was considered, the Column Subset Selection Problem (CSSP) was selected to model the reconstruction term. This was a novel application of the CSSP, mainly employed for feature selection tasks up to that point. In [19] a fast, randomized, SVD-based two-stage algorithm for solving the CSSP was adopted from [3] and adapted to salient dictionary learning, while video frame saliency was computed using a dense inter-frame distance matrix. In [20] that saliency term was replaced with a much faster to compute Regularized SVD-based Low-Rank Approximation approach, resulting in state-of-the-art summarization accuracy at near-real-time speeds. However, with regard to the reconstruction term, a black

box of non-negligible computational cost remained in the form of the deterministic second stage from [3].

This work further explores the possibilities opened up by CSSP-based modelling and adopts from [11] a different, non-randomized CSSP solution for the reconstruction term. It is a greedy, iterative algorithm, adapted here to salient dictionary learning and coupled with the fast, SVD-based saliency term from [20]. Computational complexity analysis of all salient dictionary learning and related methods is presented for the first time. Extensive empirical evaluation of the proposed method is performed under the setup described in [20]. The results indicate high speed gains while retaining state-of-the-art summarization accuracy, making the proposed algorithm especially suitable for big data pre-processing.

2 Method Preliminaries

Below, an input video composed of N frames is represented as a matrix $\mathbf{D} \in \mathbb{R}^{V \times N}$. Each column vector $\mathbf{d}_j, 0 \leq i < N$, describes a video frame. Moreover, we assume that the desired summary is a matrix $\mathbf{S} \in \mathbb{R}^{V \times C}, C << N$ containing an ordered set of video key-frames. Its columns are indicated by a binary-valued frame selection vector $\mathbf{s} \in \mathbb{N}^N$.

2.1 The Column Subset Selection Problem

In the methods this paper improves upon (the no-saliency, dictionary-of-representatives algorithm [18] and the salient dictionary learning algorithms [19] [20]), the Column Subset Selection Problem (CSSP) [3] was selected for algebraically modelling the reconstruction term.

Given $\mathbf{D}$ and a parameter $C << N$, the CSSP consists in selecting a subset of exactly C columns of $\mathbf{D}$, which will form a new $V \times C$ matrix $\mathbf{S}$ that captures as much of the information contained in the original matrix as possible. The goal is to construct a matrix $\mathbf{S} \in \mathbb{R}^{V \times C}$ such that the quantity:

$$\|\mathbf{D} - (\mathbf{S}\mathbf{S}^{+})\mathbf{D}\|_F \tag{1}$$

is minimized. $\| \cdot \|_F$ is the Frobenius matrix norm and $\mathbf{S}^{+}$ is the pseudoinverse of $\mathbf{S}$. Obviously, $\mathbf{S}$ is entirely defined by $\mathbf{D}$ and the frame selection vector $\mathbf{s}$.

The CSSP was deemed to be especially suitable for modelling key-frame extraction, since it results in a small number C of unaltered columns of the original matrix that are significant in a dictionary-of-representatives sense, with C being strictly user-defined. However, the problem is considered to be NP-hard [2]. A fast, numerical, randomized CSSP method operating in two stages [3] was employed as a main building block in [19] and [20]. The method relied on the SVD decomposition of $\mathbf{D}$.

2.2 Salient Dictionary Learning for Activity Summarization

Salient dictionary learning entails joint optimization of a "reconstruction term" and a "saliency term", so as to avoid a summary that contains many uninteresting video

frames (e.g., depicting the static background) and does not include any outliers. The related objective is defined in [19] using the CSSP for reconstruction and a vector $\mathbf{p}$ for saliency:

$$\min_{\mathbf{s}} : \|\mathbf{D} - \mathbf{S}\mathbf{S}^{+}\mathbf{D}\|_F - \alpha c \mathbf{s}^T \mathbf{p}, \tag{2}$$

where $\alpha \in [0, 1]$ is a user-provided parameter regulating the contribution of the saliency component and c is a scaling factor to bring per-video frame saliency value down to the scale of the dictionary component. $\mathbf{p} \in \mathbb{R}^N$ is a precomputed per-frame saliency vector, assigning a scalar saliency value to each video frame.

In [19], the approximate CSSP algorithm from [3] was coupled with a simple saliency term. Initially, the saliency term produced a precomputed, per-frame saliency vector $\mathbf{p}$. Subsequently, video data matrix $\mathbf{D}$ was suitably transformed in a manner that took into account per-frame saliency, before applying the numerical algorithm from [3] to the modified matrix. The above method implicitly solved the objective from Eq. (2).

In [20] the saliency term was replaced with a Regularized SVD-based Low Rank Approximation method that significantly reduced the computational overhead. This was due to the fact that no inter-frame distance matrix needed to be constructed, while the SVD decomposition of the video data matrix was also required by the employed CSSP algorithm and, therefore, readily available.

3 Greedy Salient Dictionary Learning

Although the method in [20] is faster and, in general, equally or more accurate than competing methods, it is still burdened by the second, deterministic stage of the numerical CSSP algorithm from [3] (Rank-Revealing QR decomposition [5] was employed in both [19] and [20]). Given that the required runtime is a quadratic function of N (as shown below), minimizing the per-frame computational cost of salient dictionary learning is essential for successful deployment in big activity video data analysis.

Towards this end, the possibilities opened up for activity video key-frame extraction by CSSP modelling were explored. In this paper, the numerical, two-stage CSSP solution for the reconstruction term is entirely replaced by an efficient, iterative, deterministic, greedy method [11], adopted from recent CSSP literature and described below.

At each iteration of the algorithm a single video frame is added to the summary, so as to greedily minimize the reconstruction error, until the key-frame set contains exactly C key-frames. The following quantities are defined for the t-th iteration:

1. $\mathbf{s}^{t-1}$: the currently extracted key-frame set/summary binary selection vector, prescribing the current summary $\mathbf{S}^{t-1}$. It holds that $\|\mathbf{s}^{t-1}\|_0 = t - 1$.
2. $\overline{\mathcal{R}}^{t-1}$: the set of the temporal indices of all video frames not contained in $\mathbf{S}^{t-1}$. It contains $N - (t - 1)$ elements, all in the interval $[0, N - 1]$.
3. l^t: the temporal index of the video frame $\mathbf{d}_{:l_t}$ that is actually selected for inclusion in $\mathbf{S}^t$ during iteration t. Obviously, $l^t \in \overline{\mathcal{R}}^{t-1}$, but $l^t \notin \overline{\mathcal{R}}^{t}$.

The method recursively updates two vectors, $\mathbf{f}, \mathbf{g} \in \mathbb{R}^N$. Each one keeps track of a scalar score for each video frame $\mathbf{d}_{:i}, 0 \leq i < N$. At the start of the t-th iteration, the most suitable l^t is selected for addition to the extracted key-frame set/summary in the following manner:

$$l^t = \arg\max_i \frac{f_i^{t-1}}{g_i^{t-1}}, \quad i \in \overline{\mathcal{R}}^{t-1}, \tag{3}$$

where f_i^{t-1}, g_i^{t-1} is the i-the entry of current vector $\mathbf{f}, \mathbf{g}$, respectively. Subsequently, $\mathbf{f}^t$ and $\mathbf{g}^t$ are computed, by updating $\mathbf{f}^{t-1}$ and $\mathbf{g}^{t-1}$ based on the value of l^t. The formulas for initializing and updating $\mathbf{f}$ and $\mathbf{g}$ can be found in [11].

In order to adapt the above method to the proposed framework, $\tilde{\mathbf{p}} \in \mathbb{R}^N$ is initially precomputed once. It is a slightly modified version of $\mathbf{p}$ from [20], with its entries (the per-frame saliency factors) normalized into the interval $[0, 1]$. Subsequently, the greedy CSSP algorithm is iteratively executed as described above, but Equation (3) is modified in the following manner:

$$l^t = \arg\max_i \left((1-\alpha)\frac{f_i^{t-1}}{g_i^{t-1}} + \alpha \tilde{p}_i \frac{f_i^{t-1}}{g_i^{t-1}} \right), \quad i \in \overline{\mathcal{R}}^{t-1}. \tag{4}$$

where $\tilde{p}_i$ is the i-th entry of $\tilde{\mathbf{p}}$. Thus, at each iteration, vectors $\mathbf{f}$ and $\mathbf{g}$ are updated based on the reconstructive advantage currently conveyed by each video frame, but the actual selection of a candidate video frame for inclusion in the summary also depends on its precomputed saliency and the provided saliency contribution parameter α. The algorithm is completed after C iterations.

4 Computational Complexity Analysis

Below, the computational complexity of all CSSP-based activity video key-frame extraction methods is briefly presented for the first time. In [18], the CSSP objective is directly employed as a fitness function under a genetic algorithm, with no saliency term considered. Since the computation of $\mathbf{S}^+$ runs in $\mathcal{O}(min\{VC^2, V^2C\})$, the entire reconstruction term is dominated by the matrix multiplications in Eq. (1). Therefore, assuming population size P and G generations, total method complexity is either $\mathcal{O}(PGV^2N)$, if $V < C$, or $\mathcal{O}(PGVCN)$, if $V > C$.

In [19], the employed CSSP algorithm runs in $\mathcal{O}(min\{VN^2, V^2N\})$ [3], while the time complexity of the proposed inter-frame distance matrix-based saliency term is $\mathcal{O}(VN^2)$. The proposed adaptation of the CSSP method to salient dictionary learning runs in $\mathcal{O}(VN)$. Thus, the overall method complexity is $\mathcal{O}(VN^2)$.

In [20], the CSSP algorithm from [3] is retained, but a different SVD-based saliency term is proposed. Given that the SVD decomposition of $\mathbf{D}$ can be used for both the reconstruction and the saliency term, the complexity of the latter is $\mathcal{O}(VN)$. Finally, the adaptation to salient dictionary learning runs in $\mathcal{O}(VN)$, as in [19]. Thus, the overall method complexity is $\mathcal{O}(min\{VN^2, V^2N\})$.

In this paper, the initialization of $\mathbf{f}$ and $\mathbf{g}$ runs in $\mathcal{O}(VN^2)$, while the main, iterative CSSP algorithm runs in $\mathcal{O}(VNC)$ [11]. The saliency term from [20] is retained and its computation is dominated by the SVD decomposition: $\mathcal{O}(min\{VN^2, V^2N\})$. Since

the per-frame saliency vector only needs to be derived once, replacing Equation (3) with Equation (4) does not alter its time complexity. Thus, given that $C << N$, the overall method complexity is $\mathcal{O}(VN^2)$.

For comparison purposes, the time complexities of [9], [22] and [8] are $\mathcal{O}(VCN)$, $\mathcal{O}(CNV^2)$ and $\mathcal{O}(VCN^2)$, respectively.

5 Evaluation

In order to empirically evaluate the proposed algorithm, extensive comparisons were made against a baseline clustering approach [9], random video frame sampling over a million iterations, as well as competing state-of-the-art methods [8, 18–20, 22], using three human activity video datasets. The empirical evaluation setup is identical to the one found in [20]. All method implementations were in MATLAB, except [18] which was written in C++, using fast linear algebra libraries OpenBLAS [30] and Armadillo [25]. All experiments were performed on a high-end desktop PC.

Although video descriptors that have been learnt via a neural network are becoming the norm in video summarization [21], we employed a combination of traditional, hand-crafted low- and mid-level descriptors for video frame representation. Thus, three different feature descriptors were extracted for each video: LMoD [15], SIFT [13] and Improved Dense Trajectories (IDT) [29], aggregated per video frame under the Improved Fisher Vector (IFV) approach [24]. IFV codebook size was empirically set to 8, 24 and 32 visual words for IDT, SIFT and LMoD, respectively, leading to total dimensionality of video frame representation (after concatenation) $V = 17568$. In the case of [8], vectorized raw image pixel values were employed for video frame representation, due to the nature of the algorithm.

Single-view subsets of three publicly available, annotated, multi-view activity video datasets were employed. The datasets were slightly processed to better suit an activity video summarization task (e.g., several videos, each one depicting a single activity, were temporally concatenated, so as to form a long video composed of multiple consecutive activities). In each case, a specific camera angle was chosen from the original multi-view dataset for all activity sessions. The processed versions are briefly described below:

1. The IMPART video dataset [28], depicting 3 actors in 2 different settings: an outdoor one and a living-room. A total of 116 indoor and 214 outdoor activity sessions with static camera are included, where the actors perform a series of activities one after another, moving along approximately fixed trajectories via predefined waypoints. 4 different activity types were performed, namely “Walk”, “Hand-wave”, “Run” and “Other”. The dataset consists of 6 video files with a resolution of 720 $\times$ 540 pixels and mean duration of about 4542 video frames.
2. The IXMAS dataset [31], depicting 10 actors in an indoor setting. A total of 467 activity sessions with static camera are included, where the actors perform a series of activities one after another, with varying/unconstrained body poses. In total, 11 different activities were performed. The dataset consists of 4 video files with a resolution of 390 $\times$ 290 pixels and mean duration of about 9055 video frames.

This is the most challenging dataset, due to the low video resolution, the relatively high number of video frames and activity segments, as well as the very high visual similarity between video frames belonging to different activity segments.

3. The i3DPOST dataset [12], depicting 8 actors in a blue-screen backdrop. A total of 104 activity sessions with static camera are included, where either the actors perform a series of activities one after another, moving along approximately fixed trajectories, or two actors interact. In total, 12 different activities were performed. The dataset consists of 3 video files with a resolution of 640 $\times$ 480 pixels and mean duration of about 5358 video frames.

The objective Independence Ratio (IR) metric was employed for summarization accuracy evaluation, as in [18–20]. IR scores bypass the subjective, or semi-subjective, nature of traditionally employed video summarization metrics, by treating any two video frames belonging to the same activity segment as interchangeable and, from the aspect of its empirical evaluation, reducing activity video summarization to a variant of temporal video segmentation. Given a summary $\mathbf{s}$ of an input video $\mathbf{D}$, the number I_s of extracted key-frames derived from actually different activity segments (hereafter called *independent key-frames*) is used as an indirect indication of summarization success. Obviously, I_s equals the number of different activity segments represented in the summary $\mathbf{s}$. Thus, the IR score is defined as follows:

$$IR(\mathbf{s}) = \frac{I_s}{C}, \tag{5}$$

where C is the total number of requested key-frames. IR scores indicate the percentage of extracted key-frames derived from actually different activity segments, among the entire extracted key-frame set (computed using the ground truth).

Tables 1 and 2 present the mean IR scores obtained by all competing methods, across all datasets, as well as the mean execution times per video frame. For [8, 19, 20] and the proposed method, only the highest IR results across five tested values of the saliency contribution parameter ($\alpha = 0, 0.25, 0.50, 0.75, 1.00$) are reported per dataset.

Algorithm [8] completely fails to handle activity summarization, simple clustering from [9] performs relativily well, while the proposed method achieves state-of-the-art IR accuracy on two out of three datasets, at the lowest computational penalty. On IMPART, [20] seems to be faster (although with significantly lower IR score), but this stems from the fact that [20] achieved its best IR score (for that particular dataset) with $\alpha = 0$, i.e., without computing the saliency term at all.

Table 1. Mean IR for all competing methods across all datasets (higher is better).

	Random	**Proposed**	[20]	[18]	[19]	[9]	[22]	[8]
IMPART	58.86%	**77.17%**	72.16%	75.85	72.02%	72.94%	68.03%	50.17%
i3DPOST	59.01%	**77.78%**	75.64%	72.56%	74.39%	72.65%	65.81%	44.87%
IXMAS	59.40%	65.72	**66.38%**	62.00%	66.22%	65.29%	66.16%	46.66%

Table 2. Mean runtime per video frame (in milliseconds) for all competing methods across all datasets (lower is better).

	Proposed	[20]	[18]	[19]	[9]	[22]	[8]
IMPART	28.86	**17.90**	552.92	232.21	76.85	4043.82	427.84
i3DPOST	**31.67**	42.05	517.80	262.26	70.01	2544.20	385.35
IXMAS	**49.07**	80.82	734.34	461.15	225.45	8594.31	891.95

Table 3. Mean IR and runtime per video frame for the fastest methods, on the IMPART dataset.

α	**Proposed**-IR	[20]-IR	**Proposed**-Time	[20]-Time
0.00	75.21%	**72.16%**	**1.26**	**17.90**
0.25	75.18%	69.86%	28.77	45.96
0.50	76.00%	70.40%	28.91	45.28
0.75	**77.17%**	68.80%	28.86	44.98
1.00	70.30%	56.09%	28.13	36.14

Therefore, Table 3 details the evaluation results of [20] and the proposed method for all tested values of α, on the IMPART dataset. As it can be seen, the proposed method is significantly faster for any given α, while for $\alpha = 1$ (where the corresponding, adapted CSSP algorithm is executed as a reconstruction term, but only video frame saliency is actually taken into account for key-frame selection) the proposed method achieves almost 14% better IR score than [20]. On IXMAS, [20] performs slightly better than the proposed method, at a significantly higher computational cost.

6 Conclusions

A fast approach to salient dictionary learning for activity video key-frame extraction is proposed. The method retains the SVD-based saliency term from the fastest relevant algorithm available up to now, but replaces the numerical CSSP method employed for reconstruction with a greedy, iterative algorithm, properly adapted to salient dictionary learning. The result is a very rapid approach that, in general, achieves state-of-the-art summarization accuracy, while simultaneously significantly outperforming all competing methods in terms of speed. The proposed algorithm seems especially suitable for pre-processing large video streams, while greater performance gains are expected in the future, by employing neurally derived video descriptors and by integrating constraints in the optimization problem.

References

1. Aggarwal, J.K., Ryoo, M.S.: Human activity analysis: A review. ACM Computing Surveys **43**(3), 16:1–16:43 (2011)
2. Arai, H., Maung, C., Schweitzer, H.: Optimal column subset selection by A-star search. In: AAAI Conference on Artificial Intelligence (2015)

3. Boutsidis, C., Mahoney, M.W., Drineas, P.: An improved approximation algorithm for the Column Subset Selection Problem. In: Symposium on Discrete Algorithms. pp. 968–977 (2009)
4. Cernekova, Z., Pitas, I., Nikou, C.: Information theory-based shot cut/fade detection and video summarization. IEEE Transactions on Circuits and Systems for Video Technology **16**(1), 82–91 (2006)
5. Chan, T.F., Hansen, P.C.: Low-rank revealing QR factorizations. Numerical Linear Algebra with Applications **1**(1), 33–44 (1994)
6. Cong, Y., Yuan, J., Luo, J.: Towards scalable summarization of consumer videos via sparse dictionary selection. IEEE Transactions on Multimedia **14**(1), 66–75 (2012)
7. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: European Conference on Computer Vision (ECCV). pp. 1–2 (2004)
8. Dang, C., Radha, H.: RPCA-KFE: Key frame extraction for video using robust principal component analysis. IEEE Transactions on Image Processing **24**(11), 3742–3753 (2015)
9. De Avilla, S.E.F., Lopes, A.P.B., Luz, A.L.J., Araujo, A.A.: VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method. Pattern Recognition Letters **32**(1), 56–68 (2011)
10. Elhamifar, E., Sapiro, G., Vidal, R.: See all by looking at a few: Sparse modeling for finding representative objects. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2012)
11. Farahat, A.K., Ghodsi, A., Kamel, M.S.: Efficient greedy feature selection for unsupervised learning. Knowledge and Information Systems **35**(2), 285–310 (2013)
12. Gkalelis, N., Kim, H., Hilton, A., Nikolaidis, N., Pitas, I.: The i3DPOST multi-view and 3D human action/interaction database. In: Proceedings of the IEEE Conference for Visual Media Production (CVMP). pp. 159–168 (2009)
13. Lowe, D.G.: Object recognition from local scale-invariant features. In: International Conference on Computer Vision (ICCV). pp. 1150–1157. IEEE (1999)
14. Mademlis, I., Nikolaidis, N., Pitas, I.: Stereoscopic video description for key-frame extraction in movie summarization. In: European Signal Processing Conference (EUSIPCO). pp. 819–823. IEEE (2015)
15. Mademlis, I., Tefas, A., Nikolaidis, N., Pitas, I.: Compact video description and representation for automated summarization of human activities. In: INNS Conference on Big Data. pp. 18–28. Springer (2016)
16. Mademlis, I., Tefas, A., Nikolaidis, N., Pitas, I.: Movie shot selection preserving narrative properties. In: Proceedings of the IEEE International Workshop on Multimedia Signal Processing (MMSP) (2016)
17. Mademlis, I., Tefas, A., Nikolaidis, N., Pitas, I.: Multimodal stereoscopic movie summarization conforming to narrative characteristics. IEEE Transactions on Image Processing **25**(12), 5828–5840 (2016)
18. Mademlis, I., Tefas, A., Nikolaidis, N., Pitas, I.: Summarization of human activity videos via low-rank approximation. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2017)
19. Mademlis, I., Tefas, A., Pitas, I.: Summarization of human activity videos using a salient dictionary. In: Proceedings of the IEEE International Conference on Image Processing (ICIP) (2017)
20. Mademlis, I., Tefas, A., Pitas, I.: Regularized SVD-based video frame saliency for activity summarization. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2018)
21. Mahasseni, B., Lam, M., Todorovic, S.: Unsupervised video summarization with adversarial lstm networks. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

22. Mei, S., Guan, G., Wang, Z., Wan, S., He, M., Feng, D.D.: Video summarization via minimum sparse reconstruction. Pattern Recognition **48**(2), 522–533 (2015)
23. Otani, M., Nakashima, Y., Rahtu, E., Heikkilä, J., Yokoya, N.: Video summarization using deep semantic features. arXiv preprint arXiv:1609.08758 (2016)
24. Perronnin, F., Sánchez, J., Mensink, T.: Improving the Fisher Kernel for large-scale image classification. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 143–156. Springer (2010)
25. Sanderson, C., Curtin, R.: Armadillo: a template-based C++ library for linear algebra. Journal of Open Source Software (2016)
26. Sener, F., Yao, A.: Unsupervised learning and segmentation of complex activities from video. arXiv preprint arXiv:1803.09490 (2018)
27. Song, X., Sun, L., Lei, J., Tao, D., Yuan, G., Song, M.: Event-based large scale surveillance video summarization. Neurocomputing **187**, 66–74 (2016)
28. Theodoridis, T., Tefas, A., Pitas, I.: Multi-view semantic temporal video segmentation. In: Proceedings of the IEEE International Conference on Image Processing (ICIP) (2016)
29. Wang, H., Schmid, C.: Action recognition with Improved Trajectories. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2013)
30. Wang, Q., Zhang, X., Zhang, Y., Yi, Q.: AUGEM: automatically generate high performance dense linear algebra kernels on x86 CPUs. In: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. ACM (2013)
31. Weinland, D., Ronfard, R., Boyer, E.: Free viewpoint action recognition using motion history volumes. Computer Vision and Image Understanding **104**(2), 249–257 (2006)
32. Zhang, K., Chao, W.L., Sha, F., Grauman, K.: Video summarization with Long Short-Term Memory. In: Proceedings of the European Conference on Computer Vision (ECCV). Springer (2016)
33. Zhuang, Y., Rui, Y., Huang, T., Mehrotra, S.: Adaptive key frame extraction using unsupervised clustering. In: International Conference on Image Processing (ICIP). pp. 866–870. IEEE (1998)