ADVERSARIAL UNSUPERVISED VIDEO SUMMARIZATION AUGMENTED WITH DICTIONARY LOSS

Michail Kaseris *Ioannis Mademlis* *Ioannis Pitas*

Department of Informatics, Aristotle University of Thessaloniki
{kaseris, imademlis, pitas}@csd.auth.gr

ABSTRACT

Automated unsupervised video summarization by key-frame extraction consists in identifying representative video frames, best abridging a complete input sequence, and temporally ordering them to form a video summary, without relying on manually constructed ground-truth key-frame sets. State-of-the-art unsupervised deep neural approaches consider the desired summary to be a subset of the original sequence, composed of video frames that are sufficient to visually reconstruct the entire input. They typically employ a pre-trained CNN for extracting a vector representation per RGB video frame and a baseline LSTM adversarial learning framework for identifying key-frames. In this paper, to better guide the network towards properly selecting video frames that can faithfully reconstruct the original video, we augment the baseline framework with an additional LSTM autoencoder, which learns in parallel a fixed-length representation of the entire original input sequence. This is exploited during training, where a novel loss term inspired by dictionary learning is added to the network optimization objectives, further biasing key-frame selection towards video frames which are collectively able to recreate the original video. Empirical evaluation on two common public relevant datasets indicates highly favourable results.

Index Terms— Video summarization, Generative Adversarial Networks, Long Short-Term Memory, Dictionary learning, Unsupervised key-frame extraction

1. INTRODUCTION

Video has undoubtedly become the most widespread and preferred source of visual information. However, this vast availability has made it difficult for users to find content that matches their taste, while media production companies need tools that facilitate searching through typically enormous video databases. Hence, overall, obtaining a succinct *summary* of a video is crucial for handily conveying its essence.

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951911 (AI4Media).

This demand is met by automated *video summarization* algorithms, which generate significantly shorter versions of original input videos, called *summaries*. A summary only retains the most important segments of the full sequence, while removing the redundant content. Different forms of summaries exist, but the most popular one is the *key-frame set*. The latter is a subset of temporally ordered "representative" video frames, that have been selected among the complete set of original video frames as the ones that "best" summarize the complete input in a succinct manner. As expected, different definitions of representativeness and optimal summarization can and have been employed over time, leading to diverging methodological approaches.

Early methods were typically unsupervised learning algorithms relying on clustering [1] or on dictionary learning [2]. Typical key-frame selection criteria were *summary diversity* and *reconstructive ability*, the latter being a way of formalizing representativeness as the degree to which the key-frames are jointly able to visually reconstruct all original video frames. Additionally, in various unsupervised approaches, key-frame difference from its temporal neighbours or similar *saliency* criteria were also employed [3, 4, 5, 6]. In certain cases these algorithms processed raw video frames, but typically they were fed manually crafted image/video features [7, 8].

The advent of Deep Neural Networks (DNNs) in recent years led to the popularization of supervised learning approaches to video summarization [9, 10]. However, the high cost of annotating the training data, the increased risk of overfitting to the specific training videos, as well as the subjective nature of the task, which renders it difficult to obtain a satisfying "ground-truth" summary, have led recently to unsupervised deep neural approaches. Typically, these combine several LSTMs [11], including autoencoder and discriminator modules, in the context of a unified architecture [12]. The main idea is that an LSTM *selector network* maps each input video frame representation (typically obtained by a pretrained CNN) onto a normalized scalar importance score, which determines whether it is selected as a key-frame or not. During training, a successively placed autoencoder attempts to reconstruct the representations of all input video frames from the key-frames and a discriminator judges how

convincing this reconstruction looks. Thus, an error signal is passed to the selector, allowing its adversarial training via back-propagation and a variant of gradient descent. Several reworks of this main idea have been presented during the last few years [13, 14, 15, 16], giving rise to very promising empirical results.

This paper addresses the task of unsupervised key-frame extraction by extending a common, state-of-the-art adversarially trained, LSTM-based framework. Assuming that a good summary must be able to reconstruct the original full/complete input sequence, we add an additional novel *dictionary loss* term during training, which directly penalizes the difference of the fixed-length *summary representation* (final hidden state of the LSTM autoencoder's encoding component) from a similar fixed-length *original sequence representation*. The latter is obtained as the final hidden state of a newly introduced, parallel LSTM encoder that independently processes the full/complete video. Before computing this difference within the loss term, the summary representation is first projected into the space defined by a common set of basis vectors that are being simultaneously learnt from the entire training dataset, thus serving as a global visual dictionary [3]. Thus, each summary representation is exhorted towards being a set of linear reconstruction coefficients that are jointly able to reproduce the corresponding original sequence representation. Such a linear reconstruction requirement is added to the corresponding non-linear one enforced by the LSTM autoencoder's traditional reconstruction loss, further pushing the selector towards picking representative key-frames.

This novel, augmented DNN architecture has been evaluated on two commonly used, public datasets and shown to surpass existing unsupervised state-of-the-art methods in typical video summarization metrics.

2. RELATED WORK

A first attempt to tackle unsupervised neural video key-frame extraction is [12], taking the form of a Generative Adversarial Network (GAN) framework. A bi-directional LSTM summarizer is employed to compute the importance of each video frame in the original/full sequence and a LSTM-based autoencoder processes its output, i.e., the summary, to construct "fake" samples for the adversarially trained discriminator. In [14] a subset of key-frames is obtained by maximizing the mutual information metric between the summary and the original video, under the principle of cycle consistency. This method trains two discriminators, instead of one. An extension relying on conditional GANs can be found in [15], where the generator picks conditional features to further guide it towards selecting more important video frames. Additionally, the problem of modeling inter-frame dependencies across long video sequences is handled by utilizing attention mechanisms [10]. Similarly, in [13], a variant of [12] is discussed where summarizer performance is increased by integrating attention within the LSTM autoencoder, along with additional minor improvements.

All unsupervised DNN-based methods contain several neural components that are being jointly trained by a set of component-specific loss functions. Typically, with the exception of [15], the most important loss is the reconstruction term, which forces the summarizer to select key-frames that are able to jointly reconstruct the original full video in a convincing manner, based on a popular definition of the "representativeness" summarization criterion. However, no effort has been expended on improving the reconstructive ability of the summary by imposing additional constraints on key-frame extraction during training, since most existing relevant research concerns the employed neural mechanisms or the adversarial training process.

3. ADVERSARIAL FRAMEWORK FOR VIDEO SUMMARIZATION

The method proposed in this paper contributes a novel loss term to the training of a common, state-of-the-art, adversarial DNN framework for unsupervised video summarization via key-frame extraction. Thus, this Section first discusses the baseline architecture within which the proposed method operates and, subsequently, presents the novel loss term.

3.1. Baseline architecture

In principle, a Generative Adversarial Network (GAN) involves a minimax game between two players, the *generator* and the *discriminator*, where the former is trained to maximally confuse the latter and the discriminator is trained to discern whether a training sample comes from the real data distribution or it is a "fake" sample. In an adversarial framework for unsupervised video summarization by key-frame extraction [12] the generator is typically replaced by a *summarizer* which is fed CNN-derived video frame representations. Both the summarizer and the discriminator are LSTM networks. The baseline framework is briefly detailed below.

We assume that the full/original/complete input video sequence is represented by a matrix $\mathbf{X} \in \mathbb{R}^{M \times T}$, where T is the total number of video frames and M the dimensionality of each video frame. Each column $\mathbf{x}_t \in \mathbb{R}^M, , t = 1, ..., T$ of $\mathbf{X}$ is a video frame representation, extracted using a pre-trained CNN. The columns of $\mathbf{X}$ are successively fed to the summarizer, which is composed of three successive LSTM subnetworks, each one unfolding across T time instances: a *selector*, an *encoder* and a *decoder*. The selector output is a real vector $\mathbf{s} \in [0, 1]^T$, with each entry of $\mathbf{s}$ reflecting the suitability of the corresponding input video frame as a key-frame. Subsequently, each scalar product $s_t \mathbf{x}_t$ is fed to the encoder, which gradually generates a fixed-length representation of the summary $\mathbf{e} \in \mathbb{R}^H$, where H is the LSTM hidden state dimensionality. After $\mathbf{e}$ has been finalized, it is fed to the decoder which also unfolds across T time instances. Thus, overall, the decoder outputs a reconstructed video sequence $\hat{\mathbf{X}} \in \mathbb{R}^{M \times T}$. The columns of $\hat{\mathbf{X}}$ are subsequently fed into the discrimina-

tor, which is a binary LSTM classifier being optimized to distinguish between original videos ("positive examples") and their summary-based reconstructions ("negative examples"). Importantly, at approximately half of all training iterations, the discriminator is fed an original input video as a positive example. Thus, its final hidden state in such an iteration is a fixed-length representation $\phi(\mathbf{X})$ of the corresponding full video $\mathbf{X}$.

During training of the overall architecture, several loss functions are concurrently minimized by different neural components, typically using error back-propagation and a variant of gradient descent. Since there are small variations in the losses used for different implementations of the basic framework, we cite below the loss functions specifically employed in [13], as a representative example.

Reconstruction loss: $\mathcal{L}_{recon} = \|\phi(\mathbf{X}) - \phi(\hat{\mathbf{X}})\|_2^2$. $\mathcal{L}_{recon}$ is used to update θ_s, θ_e and θ_d.

Original video loss: $\mathcal{L}_{orig} = \left(1 - C(\mathbf{X})\right)^2$, which is the MSE between the original video label (i.e., 1) and the discriminator output for original video input. $\mathcal{L}_{orig}$ is used to update θ_c.

Summary loss: $\mathcal{L}_{sum} = \left(C(\hat{\mathbf{X}})\right)^2$, which is the MSE between the summary label (i.e., 0) and the discriminator output for summary-based reconstructed video input. $\mathcal{L}_{sum}$ is used to update θ_c.

Generator loss: $\mathcal{L}_{gen} = \left(1 - C(\hat{\mathbf{X}})\right)^2$, which is the MSE between the original video label (i.e., 1) and the discriminator output for summary-based reconstructed video input. $\mathcal{L}_{gen}$ is used to update θ_d.

Sparsity loss: $\mathcal{L}_{sparsity} = \|\frac{1}{T}\sum_{t=1}^{T} s_t - \sigma\|_2$ is a diversity-inducing regularizer used to update θ_s. Hyperparameter σ represents the desired percentage of original video frames to be retained in the summary.

In this formulation, θ_s, θ_e, θ_d and θ_c refers to the parameter vector of the selector, the encoder, the decoder and the discriminator, respectively. Moreover, $\phi(\mathbf{X})$/$\phi(\hat{\mathbf{X}})$ is the discriminator's final hidden state when it has been fed $\mathbf{X}$/$\hat{\mathbf{X}}$ as input, respectively. In a similar manner, $C(\mathbf{X})$/$C(\hat{\mathbf{X}})$ is the final output of the discriminator when it has been fed $\mathbf{X}$/$\hat{\mathbf{X}}$ as input, respectively.

3.2. Proposed Dictionary Loss

Building upon this basic framework, the proposed method adds a complementary neural component which is only employed during training, i.e., an LSTM autoencoder that also unfolds across T time instances and consists in a LSTM encoder-decoder architecture. This *parallel encoder* runs in parallel to the main network. It successively receives all original video frame representations $\mathbf{x}_t$ as input, encodes the entire original sequence into a final hidden state $\mathbf{h} \in \mathbb{R}^N$ and subsequently decodes it to approximately reproduce the full original video. Obviously, N is the dimensionality of the hidden state of the parallel encoder.

The newly introduced, parallel autoencoder is pretrained in a separate preliminary phase, before training the main DNN, using a typical MSE loss. After it has been optimized, its decoder is discarded and only the parallel encoder is retained for the main training stage. Subsequently, during the inference phase of the trained summarizer DNN, only the selector LSTM is required: the encoder, the decoder, the discriminator and the parallel encoder can be discarded.

The rationale behind the addition of the parallel autoencoder into the overall framework for obtaining a fixed-length representation of the original video, was that the existing $\phi(\mathbf{X})$ which is employed for computing the main reconstruction loss is constructed by the discriminator. Thus, it is a representation adapted to discriminating between original videos and their summary-based reconstructions, but not necessarily an optimal compact representation of the original video content itself. In contrast, $\mathbf{h}$ is a exactly such an original sequence representation, obtained at each iteration of the summarizer training process as the final hidden state of the pretrained parallel encoder.

This allows us to introduce the proposed *dictionary loss* $\mathcal{L}_{\text{dict}}$ as an additional training constraint for updating θ_s and θ_e, besides the traditional reconstruction loss. $\mathcal{L}_{\text{dict}}$ exploits $\mathbf{h}$ and a common matrix $\mathbf{A}$. It is inspired by the dictionary-of-representatives formulation of unsupervised video key-frame extraction [3]. In this generic framework, given original input video $\mathbf{X} \in \mathbb{R}^{M \times T}$, the goal is to find an optimal summary matrix $\mathbf{S} \in \mathbb{R}^{M \times C}, C << T$ and a reconstruction coefficient matrix $\mathbf{B} \in \mathbb{R}^{C \times T}$, so that the columns of $\mathbf{S}$ constitute a subset of columns of $\mathbf{X}$ and the following objective is minimized:

$$\min_{\mathbf{S},\mathbf{B}} : \sum(\|\mathbf{X} - \mathbf{SB}\|)_n, \tag{1}$$

where $\|\cdot\|$ is a matrix norm.

Eq. (1) is not a loss function, but a generic and abstract problem formulation which, up to now, had not been considered at all in the context of the adversarial reconstruction video summarization framework. Thus, the proposed method is a novel loss term that concretizes this general objective for deep neural unsupervised key-frame extraction scenarios which rely on adversarial reconstruction:

$$\mathcal{L}_{\text{dict}} = \|\mathbf{h} - \mathbf{Ae}\|_2, \tag{2}$$

where $\mathbf{e}$ is the final hidden state of the main LSTM encoder (summary representation of training example $\mathbf{X}$) and $\mathbf{A} \in \mathbb{R}^{N \times H}$ is learnt during training from the entire dataset, i.e., over time it converges to a single matrix for all training examples.

At each training iteration, $\mathbf{A}$ transforms the current summary representation to a vector space being simultaneously learnt from all the original videos, therefore $\mathbf{A}$ serves as a global visual dictionary. Thus, the gain from integrating $\mathcal{L}_{\text{dict}}$

into the training process is two-fold. First, each summary representation is exhorted towards being a set of linear reconstruction coefficients that are jointly able to reproduce the corresponding original sequence representation. Such a linear reconstruction requirement is added to the corresponding non-linear one enforced by $\mathcal{L}_{recon}$ [17], further pushing the selector towards picking representative key-frames. Second, $\mathcal{L}_{\text{dict}}$ directly updates only θ_s and θ_e, while traditional $\mathcal{L}_{recon}$ influences θ_d as well, potentially leading to an undesired compensation of suboptimal key-frame selection by overfitting the decoder to the training set.

Overall, the proposed method only affects the training stage: the novel loss term $\mathcal{L}_{\text{dict}}$ is appended to the training objectives set of the adversarial reconstruction video summarization framework (*not* replacing anything), while the novel, pretrained parallel autoencoder is simultaneously appended to the neural architecture so that $\mathcal{L}_{\text{dict}}$ can be computed. The parallel autoencoder can be discarded after training has been completed; therefore, the proposed method induces zero inference runtime overhead to the baseline framework.

4. EVALUATION

The baseline framework codebase was borrowed from [13]. According to typical adversarial unsupervised video summarization evaluation protocols, the videos were downsampled to 2 frames per second, the CNN-derived 1024-dimensional video frame representations were extracted from the pool5 layer of a GoogLeNet [18], pretrained on the ImageNet dataset for image classification, while the hidden state of the involved LSTM modules is 500-dimensional. Each component, including the newly introduced parallel autoencoder, consists in a two-layer LSTM. The parallel encoder, produces a fixed-size (500-dimensional) original sequence representation.

Two Adam optimizers with a learning rate of 10^{-4} were employed, for training the summarizer and for learning matrix $\mathbf{A}$ in the dictionary loss, respectively. 80% of the dataset was used for training and the rest was reserved for validation. According to common practice in relevant literature, the dataset was repeatedly partitioned in 5 random splits and the overall reported performance is the average over all runs. Matrix $\mathbf{A}$ was initialized using Glorot uniform initialization [19].

Finally, given that the selector LSTM produces an importance score/percentage per original video frame during inference, a Knapsack algorithm was implemented [20] to temporally segment the video into subshots (i.e., smaller consecutive video clips). The subshots are sorted from high to low based on the average importance of their video frames, before selecting the final key-frame set [13].

Empirical evaluation was conducted using the F-Score metric F and the commonly employed, public datasets SumMe [21] and TVSum [22]. These contain videos of diverse content, ranging from first- and third-person clips of human activities to television shows and documentaries, with a varying duration of 1 up to 6 minutes. Both datasets provide ground-truth summaries annotated by multiple users.

Table 1 depicts quantitative evaluation results for the proposed method, as well as for competing ones, with desired summary length equal to 15% of the duration of the original full/complete input in all cases.

Method	TVSum	SumMe
SUM-GAN-AAE [13]	58.3%	48.9%
vsLSTM [23]	54.2%	37.6%
dppLSTM [23]	54.7%	38.6%
Cycle-SUM [14]	57.6%	41.9%
ACGAN [15]	58.5%	46.0%
Proposed	$\mathbf{59.3}\%$	$\mathbf{51.0}\%$

Table 1. F-Score results of unsupervised video summarization methods in two public datasets.

As it can be seen, the proposed method surpasses all competing methods and gives rise to gains of 0.8% and 2.1% (measured in F-Score), respectively in the two datasets, in comparison to the next best unsupervised approach per dataset. In comparison to the baseline [13] where the proposed method is appended to, the respective F-Score gains are 1.0% and 2.1%.

For reference purposes, we also report here the F-Score performance of two supervised, neural attention-driven methods: VASNet [24] achieves 49.71% and 61.42%, in the SumMe/TVSum dataset, respectively, while M-AVS [10] achieves 44.4% and 61.0%, respectively.

5. CONCLUSIONS

This paper presented a novel, differentiable loss function inspired by dictionary learning, which is added to the training process of a common adversarial neural video summarization framework for unsupervised key-frame extraction. The proposed dictionary loss exploits a newly introduced, parallel LSTM autoencoder and biases key-frame selection towards video frames which are collectively able to recreate the original sequence, by imposing a linear reconstruction objective on top of the non-linear reconstruction enforced by the main LSTM autoencoder. The method surpasses the state-of-the-art when evaluated on two common public relevant datasets, confirming our underlying hypothesis that the reconstructive ability plays a crucial role in key-frame selection.

6. REFERENCES

[1] "VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method," *Pattern Recognition Letters*, vol. 32, no. 1, pp. 56 – 68, 2011.

[2] E. Elhamifar, G. Sapiro, and R. Vidal, "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.

[3] I. Mademlis, A. Tefas, and I. Pitas, "A salient dictionary learning framework for activity video summarization via key-frame extraction," *Information Sciences*, vol. 432, pp. 319–331, 2018.

[4] I. Mademlis, A. Tefas, and I. Pitas, "Greedy salient dictionary learning with optimal point reconstruction for activity video summarization," in *Proceedings of the IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2018.

[5] I. Mademlis, A. Tefas, and I. Pitas, "Regularized SVD-based video frame saliency for unsupervised activity video summarization," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[6] I. Mademlis, A. Tefas, and I. Pitas, "Summarization of human activity videos using a salient dictionary," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017.

[7] I. Mademlis, A. Tefas, N. Nikolaidis, and I. Pitas, "Compact video description and representation for automated summarization of human activities," in *INNS Conference on Big Data*. Springer, 2016.

[8] I. Mademlis, N. Nikolaidis, and I. Pitas, "Stereoscopic video description for key-frame extraction in movie summarization," in *Proceedings of the European Signal Processing Conference (EUSIPCO)*. IEEE, 2015.

[9] B. Zhao, X. Li, and X. Lu, "TTH-RNN: Tensor-train hierarchical recurrent neural network for video summarization," *IEEE Transactions on Industrial Electronics*, 2020.

[10] Z. Ji, K. Xiong, Y. Pang, and X. Li, "Video summarization with attention-based encoder-decoder networks," *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[12] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial lstm networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[13] E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris, and I. Patras, "Unsupervised video summarization via attention-driven adversarial learning," in *Proceedings of the International Conference on Multimedia Modeling (MMM)*. Springer, 2020.

[14] L. Yuan, F. E.H. Tay, P. Li, L. Zhou, and J. Feng, "Cycle-SUM: Cycle-consistent adversarial LSTM networks for unsupervised video summarization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[15] X. He, Y. Hua, T. Song, Z. Zhang, Z. Xue, R. Ma, N. Robertson, and H. Guan, "Unsupervised video summarization with attentive conditional generative adversarial networks," in *Proceedings of the ACM International Conference on Multimedia*, 2019.

[16] E. Apostolidis, A. I. Metsai, E. Adamantidou, V. Mezaris, and I. Patras, "A stepwise, label-based approach for improving the adversarial training in unsupervised video summarization," in *Proceedings of the International Workshop on AI for Smart TV Content Production, Access and Delivery*, 2019.

[17] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using LSTMs," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2015.

[18] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[19] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feed-forward neural networks," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2010.

[20] D. Potapov, M. Douze, Z. Harchaoui, and C. Schmid, "Category-specific video summarization," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2014.

[21] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.

[22] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "TVSum: Summarizing web videos using titles," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[23] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2016.

[24] J. Fajtl, H. S. Sokeh, V. Argyriou, D. Monekosso, and P. Remagnino, "Summarizing videos with attention," in *Proceedings of the Asian Conference on Computer Vision (ACCV)*. Springer, 2018.