Proc. ACM MM 2020. Author's accepted version. The final publication is at https://doi.org/10.1145/3394171.3413632

Performance over Random: A Robust Evaluation Protocol for Video Summarization Methods

Evlampios Apostolidis
CERTH-ITI
Thermi, Greece, 57001
Queen Mary University of London
apostolid@iti.gr

Eleni Adamantidou
CERTH-ITI
Thermi, Greece, 57001
adamelen@iti.gr

Alexandros I. Metsai
CERTH-ITI
Thermi, Greece, 57001
alexmetsai@iti.gr

Vasileios Mezaris
CERTH-ITI
Thermi, Greece, 57001
bmezaris@iti.gr

Ioannis Patras
Queen Mary University of London
London, UK, E14NS
i.patras@qmul.ac.uk

ABSTRACT

This paper proposes a new evaluation approach for video summarization algorithms. We start by studying the currently established evaluation protocol; this protocol, defined over the ground-truth annotations of the SumMe and TVSum datasets, quantifies the agreement between the user-defined and the automatically-created summaries with F-Score, and reports the average performance on a few different training/testing splits of the used dataset. We evaluate five publicly-available summarization algorithms under a large-scale experimental setting with 50 randomly-created data splits. We show that the results reported in the papers are not always congruent with their performance on the large-scale experiment, and that the F-Score cannot be used for comparing algorithms evaluated on different splits. We also show that the above shortcomings of the established evaluation protocol are due to the significantly varying levels of difficulty among the utilized splits, that affect the outcomes of the evaluations. Further analysis of these findings indicates a noticeable performance correlation among all algorithms and a random summarizer. To mitigate these shortcomings we propose an evaluation protocol that makes estimates about the difficulty of each used data split and utilizes this information during the evaluation process. Experiments involving different evaluation settings demonstrate the increased representativeness of performance results when using the proposed evaluation approach, and the increased reliability of comparisons when the examined methods have been evaluated on different data splits.

CCS CONCEPTS

• **Computing methodologies** → **Video summarization**; • **General and reference** → **Evaluation**; **Metrics**; • **Applied computing** → *Mathematics and statistics.*

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413632

KEYWORDS

Video summarization, Performance over Random, Evaluation protocol, Random performance, Human performance, Covariance, Pearson correlation coefficient

ACM Reference Format:
Evlampios Apostolidis, Eleni Adamantidou, Alexandros I. Metsai, Vasileios Mezaris, and Ioannis Patras. 2020. Performance over Random: A Robust Evaluation Protocol for Video Summarization Methods. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413632

1 INTRODUCTION

Nowadays, we are experiencing a constantly growing engagement of users with devices (e.g. smart-phones, wearables etc.) that carry powerful video recording sensors and allow instant upload of the captured video on the Web. Huge amounts of video content are uploaded on video sharing platforms (e.g. YouTube, DailyMotion, Vimeo), social networks (e.g. Facebook, Twitter, Instagram) and online repositories of media and news organizations every single hour. This tremendous growth of the available video material has rapidly increased the needs for technologies that allow users to navigate within endless collections of videos in a time-efficient manner, and find the piece of video content that they are looking for. Part of the response to this demand was the development of techniques for automatic video summarization. These methods generate a concise synopsis that conveys the important parts of the full-length video; based on this, viewers can have a quick overview of the whole story without having to watch the entire content.

Several approaches for automatic video summarization have been proposed over the last couple of decades. Early methods targeted the extraction of a set of representative keyframes, forming a static summary of the video content (a.k.a. video storyboard). While keyframes enable quick visual inspection of the entire video content, they are limited in that all motion information is lost. To address this restriction, most recent video summarization algorithms aim to select the key parts/fragments of the video, and create a dynamic summary of the video content (a.k.a. video skim). These summaries improve the viewing experience as they offer a more advanced and narrative way for presenting the story in the video.

Proc. ACM MM 2020. Author's accepted version. The final publication is at https://doi.org/10.1145/3394171.3413632

With regards to the assessment of the generated video summaries, early works involved visual inspection of the produced summaries and the collection and analysis of human responses about their quality. These laborious procedures have been replaced by the creation of ground-truth data and the definition of more subjective evaluation protocols. The most commonly used datasets are SumMe [17] and TVSum [39]. These datasets provide a set of videos along with multiple human annotations for each video. In SumMe the annotations indicate the selected video fragments that form the video summary, while in TVSum they correspond to values signifying the importance of each frame of the video. The evaluation relies on quantifying the alignment between the user-defined and the automatically-generated summaries with F-Score, and most commonly, it involves the use of a small set of randomly-created training/testing splits of the utilized dataset. Since the introduction of SumMe and TVSum, the above coarsely-described evaluation protocol has been widely adopted in the relevant literature [2, 3, 7, 13, 14, 19, 22, 23, 26, 27, 29, 30, 34, 36–38, 40, 42–47].

In this work we examine the established evaluation approach from a perspective that is aligned with our view regarding the characteristics of an optimal evaluation protocol for video summarization. More specifically, such a protocol should be applicable to a small set of data splits and provide results that are highly representative of the algorithm's performance. In this way, the evaluation outcomes on a few data splits (e.g. on a set of 5 splits, as is commonly the case in the literature) would be generalizable to any large set of data splits, that typically enables more safe conclusions about a method's performance. This would allow reliable comparisons among algorithms that have not been assessed on the exact same set of data splits. To our knowledge, whether the established evaluation approach has these properties has not been investigated thus far. Nevertheless, such a study is particularly important for assessing the reliability of the reported performances and comparisons in summarization papers.

In the following we review the relevant literature, focusing on the utilized evaluation approaches (Section 2). Then, we perform a large-scale experiment using the SumMe and TVSum datasets and the publicly-available software implementations of five summarization methods ([2, 3, 14, 41, 46]), to assess the established evaluation protocol according to the aspects discussed above. This study indicates limitations of this protocol w.r.t. the representativeness of evaluation outcomes and the reliability of performance comparisons. Both of these shortcomings are directly associated to the observed varying level of difficulty among the utilized data splits in the evaluation (Section 3). Further experimentation combined with a deeper analysis of the aforementioned findings leads to a new evaluation protocol. This protocol takes under consideration an estimate of the difficulty of each of the utilized data splits and uses this knowledge when assessing the performance of an algorithm (Section 4). Extensive experiments using both datasets and considering different evaluation settings demonstrate the merits of the proposed evaluation protocol (Section 5).

The main contributions of this work are:

- An experiment using five summarization methods and a large set of data splits, that identifies shortcomings of the established evaluation protocol w.r.t. the representativeness of the evaluation outcomes and the reliability of performance comparisons, that relate to the large differences in the difficulty of the data splits used by different methods.
- A new evaluation approach, called "Performance over Random" (PoR), that takes under consideration estimates of how challenging each used data split is, and thus, mitigates the observed weaknesses of the established evaluation protocol.
- Experiments using the SumMe and TVSum datasets and considering four different evaluation settings, that document the enhanced representativeness and reliability of the proposed evaluation approach.

2 RELEVANT LITERATURE

2.1 Evaluating video storyboards

Most of the early video summarization techniques created a static summary of the video content with the help of representative keyframes. The latter (known as video storyboard) gives a quick overview of the story and facilitates content indexing, retrieval and navigation tasks. The utilized evaluation protocols in these methods targeted the assessment of the created keyframe-based summaries. A typical approach, applied in [8], involved independent users that assess both the relevance of each individual keyframe (using a [1-5] scale) and the quality of the entire summary w.r.t. redundant or missing information. In the same direction, [11] evaluated video summaries through a user study that aimed to assess the quality of the summary based on criteria that relate to the informativeness, enjoyability and rank of the summary. Differently to the above, [5] estimated the efficiency of the produced summary using the Fidelity measure [20] and the Shot Reconstruction Degree criterion [28]. Fidelity represents the minimum distance of a video frame from the set of selected keyframes and it is computed for all frames. Then, the average value of these computed scores is subtracted from a constant representing the largest possible value that the frame difference measure can assume, to form the average fidelity score for the summary. Shot Reconstruction Degree estimates how efficiently the entire frame sequence can be reconstructed from the keyframe set based on an interpolation algorithm. The work of [9] was the first to introduce an evaluation approach (called "Comparison of User Summaries (CUS)") that involved the manual generation of user summaries through a keyframe selection process. These summaries are compared with the results of a video summarization approach, to give an estimate about the quality of the automatically-created summaries. Comparison is performed at the keyframe-basis using color histograms, the Manhattan distance and an experimentally defined similarity threshold, while the final outcome relies on the number of matched and mis-matched keyframes. This approach was used in [1, 4, 12, 21]. A similar methodology was adopted in [31]. Each keyframe of the created summary is compared with frames in the ground-truth summary using color features, the Bhattacharya distance and a predefined threshold. Evaluation relies on Precision (P), Recall (R) and F-Score (F). Precision is computed as the ratio of the number of similar (matched) frames to the total number of frames in the automatically-created summary (A), Recall is defined as the ratio of the number of similar (matched) frames to the total number of frames in the user summary (U), and F-Score is

the harmonic mean of Precision and Recall:

$$F = 2\,\frac{P\,R}{P+R},\ \text{with}\ P = \frac{\#matches}{\#A\ frames}\ and\ R = \frac{\#matches}{\#U\ frames} \quad (1)$$

The above protocol was utilized in the supervised video summarization approach of [15]. For each utilized dataset, 80% of the videos were used for training and the remaining 20% for testing. The created summary for a test video was compared against the available user summaries for this video and evaluated using the aforementioned metrics. The mean value of the computed scores represented the algorithm's performance on the test video, and the average of these scores for all test videos the overall method's performance. Finally, a variation of this protocol was used in [10, 16, 33]; in addition to their visual similarity, two frames are a match only if they are no more than Δ frames apart.

2.2 Evaluating video skims

Most recent algorithms tackle video summarization by creating a dynamic video summary (a.k.a. video skim). For this, they select the most representative video fragments and join them in a sequence to form a shorter video. Skims offer a more natural story narration and, compared to storyboards, significantly enhance the expressiveness and informativeness of the summary. The evaluation methodologies of these works assess the quality of video skims (key-fragment-based summaries) according to their alignment with human preferences. A first attempt was made in [17], where an evaluation approach along with a new benchmark dataset for video summarization was introduced. The SumMe dataset contains 25 videos of $1-6$ mins length, covering holidays, events and sports. Multiple $(15-18)$ user-generated summaries exist for each video, with length between 5% and 15% of the original video length. To enable matching between key-fragment-based summaries (i.e. to compare the user-generated with the automatically-defined summary), videos are first segmented into consecutive and non-overlapping fragments. Then, based on the determined scores for the fragments of a given video (through the analysis), an optimal subset of them (key-fragments) is selected and forms the summary. The alignment of this summary with the user summaries for this video is evaluated by computing F-Score in a pairwise manner. In particular, the F-Score for the summary of the i^{th} video is computed as follows:

$$F_i = \frac{1}{W_i}\sum_{j=1}^{W_i} 2\,\frac{P_{i,j}\,R_{i,j}}{P_{i,j}+R_{i,j}} \quad (2)$$

where W_i is the number of available user-generated summaries for the i^{th} test video, $P_{i,j}$ and $R_{i,j}$ are the Precision and Recall against the j^{th} user summary, and they are both computed on a per-frame basis. This methodology was adopted also in [18, 35] and [39]. The latter work [39] introduces another benchmarking dataset, called TVSum. This dataset contains 50 videos of various genres, including news, how-to's, documentaries, and user-generated content. As in [17], the shots of the videos were defined through automatic video segmentation. Then, shot- and frame-level importance scores were obtained via crowd-sourcing. Based on the results of a video summarization algorithm, the computed (frame- or fragment-level) scores are used to define the sequence of selected video fragments and produce the summary. Similar to [17], the length of the summary equals to 15% of the original video duration and the agreement of the summaries is quantified by F-Score.

The evaluation approach and benchmarking datasets of [17] and [39] were jointly used to evaluate the performance of the algorithm in [41]. After defining a new segmentation for the videos of both datasets, Zhang et al. evaluated the efficiency of their method on both datasets based on the multiple user-annotated summaries for each video. Moreover, they documented the needed conversions from frame-level importance scores to key-shot-based summaries in the Supplementary Material of [41]. The typical settings about the data split into training and testing (80% for training and 20% for testing) and the summary length ($\leq$ 15% of video duration) were used, and the evaluation was based on F-Score. Experiments were conducted 5 times and the authors report the average performance and the standard deviation (STD). The above described evaluation protocol - with slight variations that relate to the number of experiments using different randomly created splits of the data (5-splits; 10-splits; "few"-splits; 5-fold cross validation), the way that the computed F-Scores from the pairwise comparisons with the user summaries are taken under consideration (maximum value is kept for SumMe according to [18]; average value is kept for TVSum) to form the F-Score for a given test video, and the way the average performance of these multiple runs is indicated (mean of highest performance for each run; best mean performance at the same training epoch for all runs) - has been adopted by the vast majority of the state-of-the-art works on video summarization (see [2, 3, 7, 13, 14, 19, 22, 23, 26, 27, 29, 30, 34, 36–38, 40, 42–47]). Hence, it can be seen as the currently established benchmarking approach for assessing the performance of video summarization algorithms. Last but not least, a different evaluation approach was proposed in [32]. This method is independent of any predefined fragmentation of the video. The user-generated frame-level importance scores for the TVSum videos are considered as rankings, and two rank correlation coefficients, namely Kendall τ [24] and Spearman ρ [25] coefficients, are used to evaluate the summary. However, these metrics can be used only on datasets that follow the TVSum annotations and methods that produce the same type of results (i.e. frame-level importance scores). This methodology was used (in addition to the established evaluation protocol) to assess the efficiency of the algorithm in [6].

3 A STUDY ON THE ESTABLISHED EVALUATION PROTOCOL

Our study focuses on the key-fragment-based evaluation protocol of [41] that was discussed in Section 2.2 and is used by the majority of SoA video summarization approaches. Our aim is to assess the representativeness of results when the evaluation is based on a small set of randomly-created splits and the reliability of performance comparisons that use different data splits for each algorithm. In this context we evaluate five publicly-available video summarization algorithms (two supervised: dppLSTM [41], VASNet [14]; and three unsupervised: DR-DSN [46], SUM-GAN-sl [3], SUM-GAN-AAE [2]) using the established protocol and a fixed set of 5 randomly-generated data splits of the SumMe and TVSum datasets (that simulates the evaluation conditions of most SoA works). These methods are, to our knowledge, the only ones

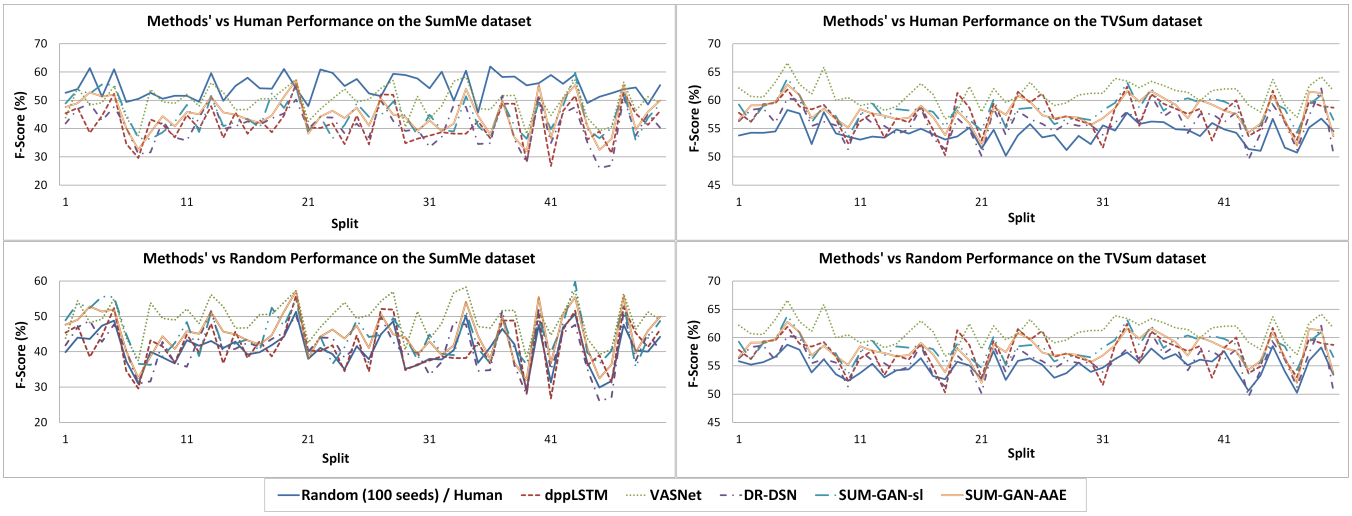

Figure 1: Visualized performance for the tested summarization methods, the random and the human summarizer in the SumMe (left side) and TVSum (right side) datasets. Many similarities can be observed between the performance curves.

	SumMe			TVSum		
Splits	5	50	Rep.	5	50	Rep.
dppLSTM [41]	40.8	41.7	38.6	59.6	57.4	54.7
VASNet [14]	**44.2**	**43.1**	**49.7**	**63.1**	**59.6**	**61.4**
DR-DSN [46]	38.7	41.1	41.4	57.6	56.0	57.6
SUM-GAN-sl [3]	43.9	40.9	47.3	59.2	57.2	58.0
SUM-GAN-AAE [2]	41.0	41.4	48.9	58.7	56.2	58.3

Table 1: Comparison (F-Score (%)) of five publicly-available video summarization approaches in SumMe and TVSum datasets, using 5 and 50 randomly-generated splits. Column "Rep." reports the score from the relevant paper. Best score shown in bold, second-best is underlined.

for which implementations are publicly available, and thus allow us to run our experiments. Then, we examine the extent to which the evaluation outcomes are generalizable on a significantly larger set of 50 splits by randomly creating 45 additional splits. Finally, we compare our findings with the performances reported in the corresponding papers and assess the reliability of comparisons that do not consider common evaluation conditions (i.e. the exact same data splits) for all methods but simply rely on the reported results. In both cases, splitting into training and testing data was based on the typical approach in most SoA works; i.e. 80% of data used for training and the remaining 20% used for testing.

The results of these evaluations, along with the reported performances in the relevant papers (see column "Rep.") are presented in Table 1. In most cases there is a noticeable difference between the results obtained using the small and the large set of splits. These differences do not necessarily indicate performance reduction in the large set, and are often larger than differences between the methods. Furthermore, the methods' rankings are quite different on the small and large set of splits, and do not match the ranking based on the reported results. The above remarks point out a serious lack of reliability of comparisons that do not use the exact same set of data splits. To identify the reasons for the varying performance of all tested algorithms in the different evaluation settings, we grouped the recorded values on a per-split basis. The result is depicted in Fig. 1 and makes it obvious that there is a noticeable variability in the performance of the examined algorithms over the set of splits. Moreover, this variability follows a quite similar pattern for all methods, i.e. the performance curve of a summarization method is similar to the curves of the other algorithms. The above observations point to different levels of difficulty for the used splits, a fact that clearly affects the outcomes of the performance evaluation.

To summarize, the established evaluation protocol has some serious shortcomings. The randomly-created splits of data for evaluating the performance of a video summarization algorithm exhibit dissimilar difficulty which significantly affects the evaluation outcomes. As a consequence, the obtained results are not representative of the algorithm's performance, and the comparisons that rely on these results are of limited reliability.

4 PERFORMANCE OVER RANDOM: AN APPROACH TO MITIGATE THE OBSERVED DEFICIENCIES

Aiming to reduce the impact of the utilized data splits, we investigate the existence of a potential association between the methods' performance and a measure of how challenging each split is. For this, we considered the performance of: i) a random summarizer and ii) an average human summarizer. To estimate the performance of a random summarizer for a given video of a test split, frame-level importance scores (ranging in $[0, 1]$) were randomly assigned based on a uniform distribution of probabilities. The corresponding fragment-level scores were computed by averaging the scores of the frames within each video fragment and were then used to select the key video fragments and form the summary of the random summarizer using the Knapsack algorithm and a predefined summary length budget ($\leq$ 15% of video duration). The generated summary

Algorithm 1 Evaluating random summarizer on a set of test videos

Notation: N is the number of test videos; for the j^{th} test video (with $j \in [1, N]$): M_j is the number of frames, C_j is the number of video segments, $\{\mathbf{S_{j,k}}\}_{k=1}^{M_j}$ are the frame-level importance scores, $\{\mathbf{T_{j,u}}\}_{u=1}^{C_j}$ are the fragment-level importance scores, u_f and u_l are the indices of the first and last frame of the u^{th} fragment, LB_j is the length budget for the video summary, $\mathbf{VS_j}$ is the generated summary, L_j is the number of ground-truth user summaries, $\mathbf{US_{j,q}}$ is the q^{th} user summary, and $P_{j,q}$, $R_{j,q}$, $F_{j,q}$ are Precision, Recall and F-Score values after comparing $\mathbf{VS_j}$ with $\mathbf{US_{j,q}}$.

Input: A set of test videos.

Output: $\mathcal{F}$, the average F-Score (%) representing the performance of the random summarizer on the set of test videos.

for $i = 1 \rightarrow 100$ **do**
  for $j = 1 \rightarrow N$ **do**
    # generate frame-level importance scores based on the Random Uniform Distribution (RUD)
    $\{\mathbf{S_{i,j,k}}\}_{k=1}^{M_j} = \text{RUD}(M_j)$
    for $u = 1 \rightarrow C_j$ **do**
      # compute fragment-level importance scores
      $T_{i,j,u} = \text{Avg}(\{\mathbf{S_{i,j,k}}\}_{k=u_f}^{u_l})$
    # create the video summary using the Knapsack algorithm
    $\mathbf{VS_{i,j}} = \text{Knapsack}(\{\mathbf{T_{i,j,u}}\}_{u=1}^{C_j}, LB_j)$
    for $q = 1 \rightarrow L_j$ **do**
      # compute Precision, Recall and F-Score through pair-wise comparisons with the user summaries
      $P_{i,j,q} = \frac{\mathbf{VS_{i,j}} \cap \mathbf{US_{j,q}}}{||VS_{i,j}||}$, $R_{i,j,q} = \frac{\mathbf{VS_{i,j}} \cap \mathbf{US_{j,q}}}{||US_{j,q}||}$,
      $F_{i,j,q} = 2\ \frac{P_{i,j,q}\ R_{i,j,q}}{P_{i,j,q}+R_{i,j,q}}$
    # compute F-Score for the j^{th} video
    $F_{i,j} = \text{Avg}(\{\mathbf{F_{i,j,q}}\}_{q=1}^{L_j})$ in TVSum
    $F_{i,j} = \text{Max}(\{\mathbf{F_{i,j,q}}\}_{q=1}^{L_j})$ in SumMe
  # compute average F-Score for the set of test videos
  $F_i = \text{Avg}(\{\mathbf{F_{i,j}}\}_{j=1}^{N})$
compute average F-Score for all iterations
$\mathcal{F} = \text{Avg}(\{\mathbf{F_i}\}_{i=1}^{100}) \cdot 100$

through the above process was compared with the available user summaries in a pair-wise manner and its alignment with each different user summary was measured by computing Precision, Recall and F-Score values. After the end of these comparisons the final F-Score for the given video was formed by averaging the computed F-Scores in the case of TVSum and keeping the maximum F-Score in the case of SumMe. The F-Score for the entire test split was calculated by averaging the computed F-Scores for each individual video of the split. This procedure was performed 100 times for each test split and the overall average score was kept as a measure of the performance of the random summarizer. This process is presented in Alg. 1 [1].

[1]Python implementation publicly available at: https://github.com/e-apostolidis/PoR-Summarization-Measure

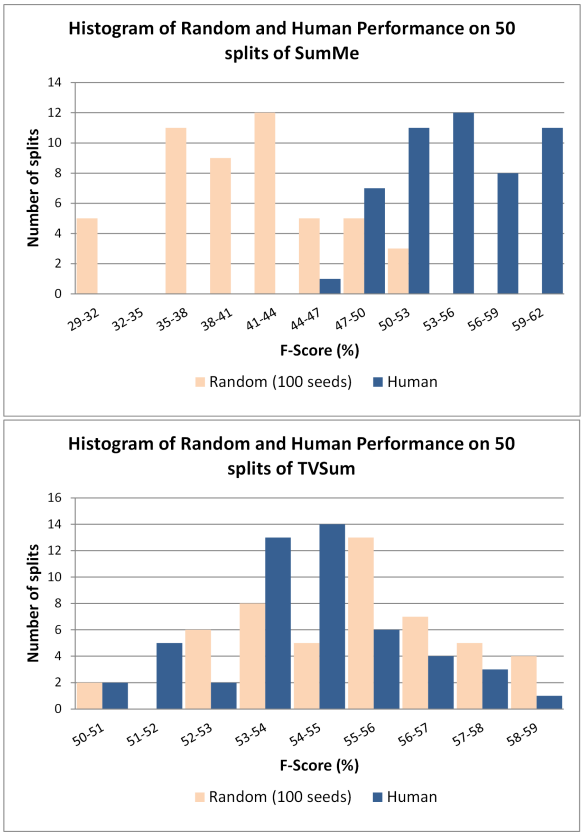

Figure 2: Histograms of random and human performance on 50 randomly-generated splits of SumMe and TVSum.

In the second case (i.e. for assessing human performance) we used the existing human-generated summaries for each video; each human annotator's performance on the videos of a test split was measured through a "leave-one-out" approach where the summaries of the examined annotator were evaluated against the summaries of all the remaining ones. The performance of the random and human summarizer on the set of 50 splits is illustrated in Fig. 1. Once again, there is a noticeable variance in the performance of both summarizers over the used splits, which strengthens our claim regarding different levels of difficulty for these splits. In Fig. 2 we plot the histogram of F-Scores attained on the various data splits, for the random and the human summarizers. This figure shows that there is indeed a noticeable difference with regards to how challenging a data split is.

To quantify the degree of correlation between each method's ([2, 3, 14, 41, 46]) performance and the performance of the random and the average human summarizer, we report the "Covariance" and the "Pearson correlation coefficient". The results presented in Table 2 show that, according to both measures and in both datasets, the performance correlation among the tested methods and the random summarizer is stronger than the correlation with the average human summarizer.

Driven by the above observations we design a new evaluation approach that estimates the level of difficulty of each data split, and

Metric	Covariance				Pearson correlation coefficient			
Dataset	SumMe		TVSum		SumMe		TVSum	
Reference	Rand.	Hum.	Rand.	Hum.	Rand.	Hum.	Rand.	Hum.
dppLSTM [41]	**29.30**	4.28	**3.99**	2.72	**0.83**	0.15	**0.72**	0.51
VASNet [14]	**21.58**	5.43	3.10	**3.13**	**0.78**	0.25	0.76	**0.80**
DR-DSN [46]	**27.40**	7.38	**4.39**	3.02	**0.79**	0.27	**0.84**	0.60
SUM-GAN-sl [3]	**25.21**	7.33	**3.74**	3.13	**0.76**	0.28	**0.87**	0.76
SUM-GAN-AAE [2]	**28.76**	6.55	**3.91**	3.15	**0.86**	0.25	**0.84**	0.71

Table 2: Estimates on the correlation of the methods' performance with the efficiency of a random and a human summarizer in the SumMe and TVSum datasets, according to Covariance and Pearson correlation coefficient. Higher values are better. Values in bold indicate which of the two summarizers (random, human) correlates better with the results of a given method for the same dataset and correlation measure.

exploits this information during the assessment of a video summarization method. This approach aims to: a) reduce the impact of the utilized data splits in the performance evaluation, b) increase the representativeness of evaluation outcomes about the performance of a summarization algorithm, and c) enhance the reliability of comparisons that rely on individual performance evaluations using different data splits.

For a given summarization method and a data split, the proposed evaluation approach contains the following steps: 1) Implement the process in Alg. 1 and compute $\mathcal{F}$, the performance of a random summarizer for the used data split. 2) Measure the performance S of the summarization method on the data split based on the evaluation protocol of [41] that estimates performance using F-Score (%). 3) For each testing epoch compute "Performance over Random" (PoR) as $PoR = \frac{S}{\mathcal{F}} \cdot 100$

Given the above, a PoR score below 100 indicates performance worse than the baseline (random) and a score above 100 indicates performance higher than the baseline (random).

In the next section we present the results of the conducted experiments on the different evaluation approaches, w.r.t. the representativeness of the evaluation outcomes and the reliability of performance comparisons that do not rely on constant evaluation conditions for all considered methods. In addition to the PoR approach discussed above, we examined also the "Performance over Human" (PoH) evaluation methodology that is defined as $PoH = \frac{S}{\mathcal{H}} \cdot 100$, where $\mathcal{H}$ is the estimated average human performance on the used data split.

5 EXPERIMENTS

5.1 Representativeness of performance evaluation

First we studied the representativeness of the outcomes of each different evaluation approach after testing the considered summarization algorithms on: i) 50 fixed splits and ii) 20 fixed split-sets (of 5 data splits each) of the utilized datasets, thus repeating, in the latter case, 20 times the evaluation process of most SoA works. For both evaluation settings, we initially examined the extent to which the performance of the random and human summarizers varies, by computing the mean and standard deviation. The results in Table 3 confirm our previous remarks about the varying difficulty of the different data splits and show that a difference in difficulty is observed also in the case where a few data splits are used together (i.e. as a split-set) for evaluation. Hence, performance evaluations made based on the use of a few data splits do not allow for safe comparisons. To be representative, the results of a given evaluation approach need to vary as little as possible across different data splits/split-sets, and for measuring this we compute the Relative Standard Deviation (RSD) of the scores of each summarization approach. RSD is defined as the ratio of the standard deviation σ to the mean value μ, $c_v = \sigma \div \mu$ and it was preferred against the typical Mean and Standard Deviation measures, as it is independent of the unit in which the measurement has been taken. The results in Table 4 show similar RSD values for F-Score and PoH in most cases, and remarkably smaller RSD values for PoR. This indicates that PoR is more representative of an algorithm's performance, compared to the estimates made using the F-Score and PoH evaluation approaches; i.e. the PoR score is less affected by which data split/split-set is used.

5.2 Reliability of performance comparisons

As discussed in Section 3, comparing summarization methods that have not been evaluated on the same data splits is not reliable. However, this comparison methodology is adopted by the majority of the SoA video summarization works; i.e. the comparisons rely on the reported values in the corresponding papers and the data splits used in each paper are completely unknown. In order to assess how robust each evaluation approach is to such comparisons, we manually created 20 mixed groups of split-sets following the procedure depicted in Fig. 3. We then used the results on these mixed split-sets to rank the five compared summarization methods from best to worst. Hence, our estimates on the robustness of each evaluation approach and, consequently, the reliability of performance comparisons, are based on 20 different comparisons that simulate the established comparison methodology in the bibliography.

Specifically, we studied the overall ranking and the variation of each summarization method's ranking for the different evaluation approaches when the assessment is based on the 20 fixed split-sets of each dataset, and we examined how these values are affected when using the 20 mixed split-sets. The variation was quantified by computing the standard deviation of a method's ranking (i.e. from 1, for the best among the 5 methods in a given experiment, to 5 for the worst) over the group of split-sets. The results reported in Table 5, show that the three evaluation approaches exhibit the

Eval. Setting	Using 50 fixed splits		Using 20 fixed split-sets	
Dataset	SumMe	TVSum	SumMe	TVSum
Metric	Mean ± STD	Mean ± STD	Mean ± STD	Mean ± STD
Random	40.85 ± 5.38	55.07 ± 1.96	40.81 ± 2.08	55.12 ± 0.95
Human	54.84 ± 4.21	54.27 ± 1.88	54.86 ± 1.92	54.31 ± 1.00

Table 3: Mean and standard deviation (STD) of the Random and Human performance on the utilized data splits/split-sets of the SumMe and TVSum datasets. Values denote F-Score.

Eval. setting	Using 50 fixed splits						Using 20 fixed split-sets					
Dataset	SumMe			TVSum			SumMe			TVSum		
Metric	F1	PoR	PoH	F1	PoR	PoH	F1	PoR	PoH	F1	PoR	PoH
dppLSTM [41]	16.13	**9.16**	16.51	5.02	**3.50**	4.40	5.52	**3.17**	5.33	2.11	**1.41**	2.14
VASNet [14]	10.69	**8.58**	11.58	3.46	2.43	**2.20**	4.31	**3.72**	5.11	2.26	**1.00**	1.04
DR-DSN [46]	15.99	**9.83**	15.93	4.88	**2.75**	3.96	6.26	**3.76**	5.31	1.91	**0.85**	1.33
SUM-GAN-sl [3]	14.08	**9.78**	13.73	3.84	**1.87**	2.55	6.30	**4.51**	6.25	1.99	**0.78**	0.94
SUM-GAN-AAE [2]	14.10	**7.14**	14.18	4.18	**2.28**	3.04	5.81	**2.70**	5.41	2.01	**0.75**	1.27

Table 4: Relative Standard Deviation of the examined metrics on both datasets and for the considered evaluation settings. F1 denotes F-Score. Lower values are better.

Dataset	SumMe						TVSum			
STD of ranking on	Fixed 20 split-sets			Mixed 20 split-sets			Fixed 20 split-sets	Mixed 20 split-sets		
the used split-sets	F1	PoR	PoH	F1	PoR	PoH	F1 / PoR / PoH	F1	PoR	PoH
dppLSTM [41]	1.18	**1.12**	1.17	1.23	**1.15**	1.35	0.91	1.25	**0.88**	1.21
VASNet [14]	0.73	0.73	0.73	0.81	**0.52**	0.83	0.00	0.52	0.22	**0.00**
DR-DSN [46]	0.88	**0.70**	0.85	1.33	**1.05**	1.20	0.00	1.15	**0.22**	0.69
SUM-GAN-sl [3]	**1.24**	1.26	1.25	1.36	**1.18**	1.40	0.67	0.94	0.75	**0.66**
SUM-GAN-AAE [2]	0.89	**0.68**	0.89	1.23	**0.88**	1.14	0.50	1.04	**0.76**	1.00
Average	0.98	**0.90**	0.98	1.19	**0.96**	1.18	0.42	0.98	**0.57**	0.71

Table 5: Standard deviation of each method's ranking on the groups of fixed and manually mixed split-sets of the SumMe (left side) and TVSum (right side) datasets. F1 denotes F-Score. Lower values are better. Small differences are due to the PoR and PoH calculation involving estimates about the difficulty of each used data split, and averaging the performance over the entire split-set may lead to a slightly different ranking of the compared methods.

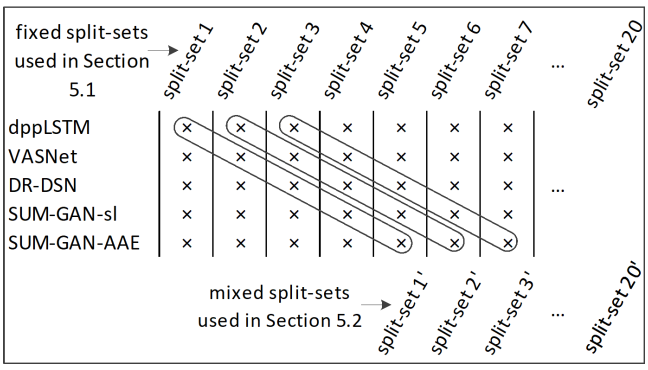

Figure 3: The applied mechanism for creating mixed groups of split-sets of the data.

same (in TVSum) or similar (in SumMe) behavior in the case of fixed split-sets. This means that when PoR and PoH are used to compare methods under a common evaluation setting, they usually lead to the same findings with F-Score. Small differences in the ranking are observed in a few cases when the compared algorithms exhibit highly-similar performance; this is due to the PoR and PoH calculation involving estimates about the difficulty of each used data split (i.e. a small difference in a difficult split can become more pronounced than a larger difference in a less-demanding split), and averaging the performance over the entire split-set may lead to a slightly different ranking of the compared methods. When moving from fixed to mixed split-sets there is a considerable increase in the average ranking deviation of the F-Score (by ≈ 21% in SumMe and ≈ 133% in TVSum) and PoH (by ≈ 21% in SumMe and ≈ 69% in TVSum) approaches. On the contrary, the PoR evaluation protocol exhibits a more robust performance, leading to much smaller increase of the average ranking deviation (by ≈ 7% in SumMe and ≈ 36% in TVSum). The above findings clearly show that the introduced PoR evaluation approach is much more robust compared to the F-Score alone, and therefore more appropriate to compare future works on video summarization.

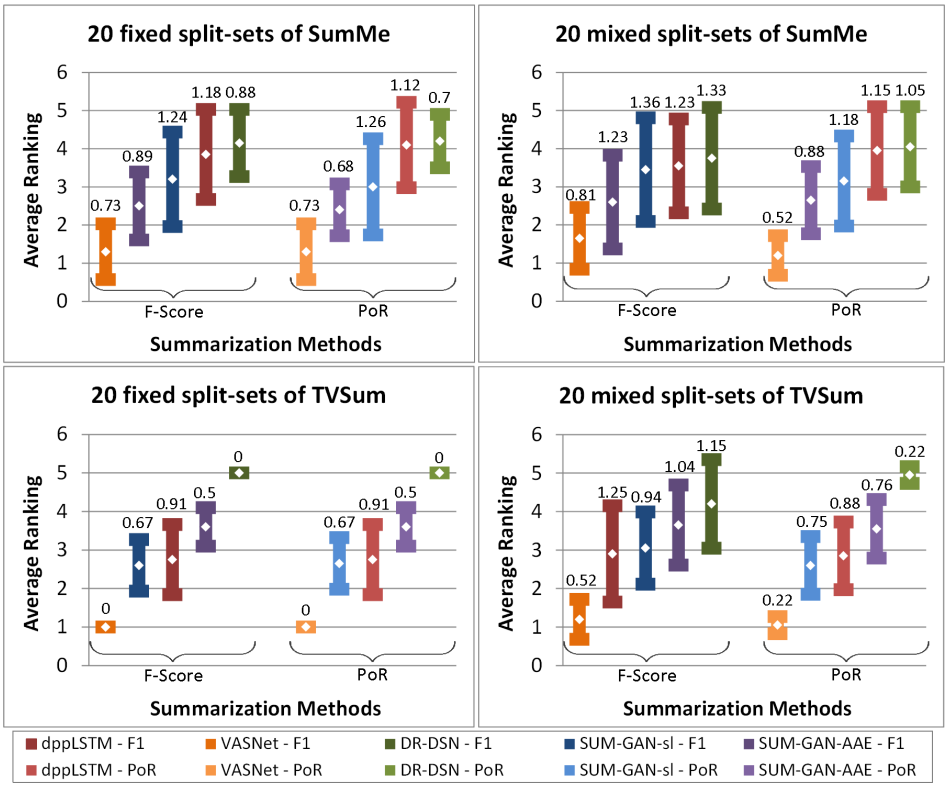

Figure 4: Average ranking and STD of each summarization approach for the fixed and mixed split-sets of the SumMe (top row) and TVSum (bottom row) datasets, according to the F-Score and PoR metrics.

In Fig. 4, the standard deviation values reported in Table 5 are plotted to visualize the improved reliability of the PoR-based ordering compared to the one based on F-Score. As seen on the left side of Fig. 4, when the assessment is based on the exact same group of split-sets, the average ranking of summarization methods (over the 20 fixed split-sets) is the same for both evaluation protocols. However, when the evaluation relies on different groups of split-sets for each algorithm, the average ranking may differ (as in TVSum). This difference is justified by the different level of difficulty of each split-set that is taken under consideration by the PoR approach, thus affecting the importance of per-split-set differences among the compared summarization methods, and resulting to different rankings. Even more importantly, in the mixed split-sets setting the standard deviation of the average ranking values differs significantly. From the right side of Fig. 4 we observe that the PoR evaluation protocol leads to lower standard deviation values than the F-Score, indicating the ability of the introduced PoR evaluation approach to provide more reliable results about the relative performance (ordering) of the compared summarization techniques.

6 CONCLUSIONS

In this work we examined how video summarization methods are evaluated. The main conclusion of our early experiments was that not all random splits of training/testing data within a standard dataset are equal, in terms of their difficulty; and, as just a handful of (different) random splits are used for evaluation in each paper, this significantly affects the reliability of performance evaluations and comparisons. To mitigate this weakness we introduced a new evaluation protocol, PoR, which takes under consideration estimates about the level of difficulty of each used split, and we experimentally documented its merit. The code for applying this protocol was made publicly-available (see footnote 1), to assist researchers when evaluating their video summarization techniques and to allow for more fair comparisons between video summarization works in the future.

ACKNOWLEDGMENTS

This work was supported by the EU Horizon 2020 research and innovation programme under grant agreement H2020-780656 ReTV. The work of Ioannis Patras was supported by EPSRC under grant No. EP/R026424/1.

REFERENCES

[1] Jurandy Almeida, Neucimar J. Leite, and Ricardo da S. Torres. 2012. VISON: VIdeo Summarization for ONline Applications. *Pattern Recogn. Lett.* 33, 4 (March 2012), 397–409.

[2] Evlampios Apostolidis, Eleni Adamantidou, Alexandros I. Metsai, Vasileios Mezaris, and Ioannis Patras. 2020. Unsupervised Video Summarization via Attention-Driven Adversarial Learning. In *Proc. of the MultiMedia Modeling 2020*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 492–504.

[3] Evlampios Apostolidis, Alexandros I. Metsai, Eleni Adamantidou, Vasileios Mezaris, and Ioannis Patras. 2019. A Stepwise, Label-based Approach for Improving the Adversarial Training in Unsupervised Video Summarization. In *Proc. of the 1st Int. Workshop on AI for Smart TV Content Production, Access and Delivery* (Nice, France) *(AI4TV '19)*. Association for Computing Machinery, New York, NY, USA, 17–25.

[4] Edward J. Y. C. Cahuina and Guillermo C. Chavez. 2013. A New Method for Static Video Summarization Using Local Descriptors and Video Temporal Segmentation. In *Proc. of the 2013 XXVI Conf. on Graphics, Patterns and Images*. 226–233.

[5] Vasileios Chasanis, Aristidis Likas, and Nikolaos Galatsanos. 2008. Efficient Video Shot Summarization Using an Enhanced Spectral Clustering Approach. In *Proc. of the Artificial Neural Networks - ICANN 2008*, Véra Kůrková, Roman Neruda, and Jan Koutník (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 847–856.

[6] Yiyan Chen, Li Tao, Xueting Wang, and Toshihiko Yamasaki. 2019. Weakly Supervised Video Summarization by Hierarchical Reinforcement Learning. In *Proc. of the ACM Multimedia Asia* (Beijing, China) *(MMAsia '19)*. Association for Computing Machinery, New York, NY, USA.

[7] Wei-Ta Chu and Yu-Hsin Liu. 2019. Spatiotemporal Modeling and Label Distribution Learning for Video Summarization. In *Proc. of the 2019 IEEE 21st Int. Workshop on Multimedia Signal Processing (MMSP)*. 1–6.

[8] Sandra E. F. de Avila, Antonio da Luz Jr., Arnaldo de A. Araújo, and Matthieu Cord. 2008. VSUMM: An Approach for Automatic Video Summarization and Quantitative Evaluation. In *Proc. of the 2008 XXI Brazilian Symposium on Computer Graphics and Image Processing*. 103–110.

[9] Sandra E. F. de Avila, Ana P. B. Lopes, Antonio da Luz Jr., and Arnaldo de A. Araújo. 2011. VSUMM: A Mechanism Designed to Produce Static Video Summaries and a Novel Evaluation Method. *Pattern Recognition Letters* 32, 1 (Jan. 2011), 56–68.

[10] Mahmut Demir and H. Isil Bozma. 2015. Video Summarization via Segments Summary Graphs. In *Proc. of the 2015 IEEE Int. Conf. on Computer Vision Workshop (ICCVW)*. 1071–1077.

[11] Naveed Ejaz, Irfan Mehmood, and Sung Wook Baik. 2014. Feature Aggregation Based Visual Attention model for Video Summarization. *Computers and Electrical Engineering* 40, 3 (2014), 993 – 1005. Special Issue on Image and Video Processing.

[12] Naveed Ejaz, Tayyab Bin Tariq, and Sung Wook Baik. 2012. Adaptive Key Frame Extraction for Video Summarization Using an Aggregation Mechanism. *Journal of Visual Communication and Image Representation* 23, 7 (Oct. 2012), 1031–1040.

[13] Mohamed Elfeki and Ali Borji. 2019. Video Summarization Via Actionness Ranking. In *Proc. of the 2019 IEEE Winter Conf. on Applications of Computer Vision (WACV)*. 754–763.

[14] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. 2019. Summarizing Videos with Attention. In *Proc. of the 2018 Asian Conf. on Computer Vision (ACCV) Workshops*, Gustavo Carneiro and Shaodi You (Eds.). Springer International Publishing, Cham, 39–54.

[15] Boqing Gong, Wei-Lun Chao, Kristen Grauman, and Fei Sha. 2014. Diverse Sequential Subset Selection for Supervised Video Summarization. In *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger (Eds.). Curran Associates, Inc., 2069–2077.

[16] Genliang Guan, Zhiyong Wang, Shaohui Mei, Max Ott, Mingyi He, and David Dagan Feng. 2014. A Top-Down Approach for Video Summarization. *ACM Transactions on Multimedia Computing, Communications, and Applications* 11, 1 (Sept. 2014), 21.

[17] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. 2014. Creating Summaries from User Videos. In *Proc. of the 2014 European Conf. on Computer Vision (ECCV)*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (Eds.). Springer International Publishing, Cham, 505–520.

[18] Michael Gygli, Helmut Grabner, and Luc Van Gool. 2015. Video Summarization by Learning Submodular Mixtures of Objectives. In *Proc. of the 2015 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 3090–3098.

[19] Xufeng He, Yang Hua, Tao Song, Zongpu Zhang, Zhengui Xue, Ruhui Ma, Neil Robertson, and Haibing Guan. 2019. Unsupervised Video Summarization with Attentive Conditional Generative Adversarial Networks. In *Proc. of the 27th ACM Int. Conf. on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 2296–2304.

[20] Hyun Sung Chang, Sanghoon Sull, and Sang Uk Lee. 1999. Efficient Video Indexing Scheme for Content-Based Retrieval. *IEEE Transactions on Circuits and Systems for Video Technology* 9, 8 (1999), 1269–1279.

[21] Hugo Jacob, Flávio L. Pádua, Anisio Lacerda, and Adriano C. Pereira. 2017. A Video Summarization Approach Based on the Emulation of Bottom-up Mechanisms of Visual Attention. *Journal of Intelligent Information Systems* 49, 2 (Oct. 2017), 193–211.

[22] Zhong Ji, Kailin Xiong, Yanwei Pang, and Xuelong Li. 2019. Video Summarization with Attention-Based Encoder-Decoder Networks. *IEEE Transactions on Circuits and Systems for Video Technology* (2019), 1–1.

[23] Yunjae Jung, Donghyeon Cho, Dahun Kim, Sanghyun Woo, and In-So Kweon. 2019. Discriminative Feature Learning for Unsupervised Video Summarization. In *Proc. of the 2019 AAAI Conf. on Artificial Intelligence*.

[24] Maurice G. Kendall. 1945. The Treatment of Ties in Ranking Problems. *Biometrika* 33, 3 (1945), 239–251.

[25] Stephen Kokoska and Daniel Zwillinger. 2000. *CRC Standard Probability and Statistics Tables and Formulae*. Crc Press.

[26] Shamit Lal, Shivam Duggal, and Indu Sreedevi. 2019. Online Video Summarization: Predicting Future to Better Summarize Present. In *Proc. of the 2019 IEEE Winter Conf. on Applications of Computer Vision (WACV)*. 471–480.

[27] Xuelong Li, Bin Zhao, and Xiaoqiang Lu. 2017. A General Framework for Edited Video and Raw Video Summarization. *IEEE Transactions on Image Processing* 26, 8 (2017), 3652–3664.

[28] Tie-Yan Liu, Xu-Dong Zhang, Jian Feng, and Kwok-Tung Lo. 2004. Shot Reconstruction Degree: A Novel Criterion for Key Frame Selection. *Pattern Recognition Letters* 25 (2004), 1451–1457.

[29] Yen-Ting Liu, Yu-Jhe Li, Fu-En Yang, Shang-Fu Chen, and Yu-Chiang F. Wang. 2019. Learning Hierarchical Self-Attention for Video Summarization. In *Proc. of the 2019 IEEE Int. Conf. on Image Processing (ICIP)*. 3377–3381.

[30] Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. 2017. Unsupervised Video Summarization with Adversarial LSTM Networks. In *Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2982–2991.

[31] Karim M. Mahmoud, Nagia M. Ghanem, and Mohamed A. Ismail. 2013. Unsupervised Video Summarization via Dynamic Modeling-Based Hierarchical Clustering. In *Proc. of the 12th Int. Conf. on Machine Learning and Applications*, Vol. 2. 303–308.

[32] Esa Rahtu Mayu Otani, Yuta Nakahima and Janne Heikkilä. 2019. Rethinking the Evaluation of Video Summaries. In *Proc. of the 2019 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

[33] Shaohui Mei, Genliang Guan, Zhiyong Wang, Shuai Wan, Mingyi He, and David [Dagan Feng]. 2015. Video Summarization via Minimum Sparse Reconstruction. *Pattern Recognition* 48, 2 (2015), 522 – 533.

[34] Jingjing Meng, Suchen Wang, Hongxing Wang, Junsong Yuan, and Yap-Peng Tan. 2018. Video Summarization Via Multiview Representative Selection. *IEEE Transactions on Image Processing* 27, 5 (2018), 2134–2145.

[35] Mayu Otani, Yuta Nakashima, Esa Rahtu, Janne Heikkilä, and Naokazu Yokoya. 2017. Video Summarization Using Deep Semantic Features. In *Proc. of the 2017 Asian Conf. on Computer Vision (ACCV)*, Shang-Hong Lai, Vincent Lepetit, Ko Nishino, and Yoichi Sato (Eds.). Springer International Publishing, Cham, 361–377.

[36] Mrigank Rochan and Yang Wang. 2019. Video Summarization by Learning From Unpaired Data. In *Proc. of the 2019 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 7894–7903.

[37] Junbo Wang, Wei Wang, Zhiyong Wang, Liang Wang, Dagan Feng, and Tieniu Tan. 2019. Stacked Memory Network for Video Summarization. In *Proc. of the 27th ACM Int. Conf. on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 836–844.

[38] Huawei Wei, Bingbing Ni, Yichao Yan, Huanyu Yu, Xiaokang Yang, and Chen Yao. 2018. Video Summarization via Semantic Attended Networks. In *Proc. of the 2018 AAAI Conf. on Artificial Intelligence*. 216–223.

[39] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. 2015. TVSum: Summarizing Web Videos Using Titles. In *Proc. of the 2015 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 5179–5187.

[40] Li Yuan, Francis Eng Hock Tay, Ping Li, Li Zhou, and Jiashi Feng. 2019. Cycle-SUM: Cycle-Consistent Adversarial LSTM Networks for Unsupervised Video Summarization. In *Proc. of the 2019 AAAI Conf. on Artificial Intelligence*.

[41] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. 2016. Video Summarization with Long Short-Term Memory. In *Proc. of the 2016 European Conf. on Computer Vision (ECCV)*, Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (Eds.). Springer International Publishing, Cham, 766–782.

[42] Yujia Zhang, Michael Kampffmeyer, Xiaoguang Zhao, and Min Tan. 2019. DTR-GAN: Dilated Temporal Relational Adversarial Network for Video Summarization. In *Proc. of the ACM Turing Celebration Conf. - China* (Chengdu, China) *(ACM TURC '19)*. Association for Computing Machinery, New York, NY, USA.

[43] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. 2017. Hierarchical Recurrent Neural Network for Video Summarization. In *Proc. of the 25th ACM Int. Conf. on Multimedia* (Mountain View, California, USA) *(MM '17)*. Association for Computing Machinery, New York, NY, USA, 863–871.

[44] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. 2018. HSA-RNN: Hierarchical Structure-Adaptive RNN for Video Summarization. In *2018 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 7405–7414.

[45] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. 2019. Property-Constrained Dual Learning for Video Summarization. *IEEE Transactions on Neural Networks and Learning Systems* (2019), 1–12.

[46] Kaiyang Zhou, Yu Qiao, and Tao Xiang. 2018. Deep Reinforcement Learning for Unsupervised Video Summarization with Diversity-Representativeness Reward. In *Proc. of the 2018 AAAI Conf. on Artificial Intelligence*.

[47] Kaiyang Zhou, Tao Xiang, and Andrea Cavallaro. 2018. Video Summarisation by Classification with Deep Reinforcement Learning. In *Proc. of the 2018 British Machine Vision Conferenve (BMVC)*.