computers

MDPI

Article

Video Summarization Based on Feature Fusion and Data Augmentation †

Theodoros Psallidas [1,2] and Evaggelos Spyrou [1,*]

[1] Department of Informatics & Telecommunications, University of Thessaly, 35100 Lamia, Greece; tpsallidas@uth.gr
[2] Institute of Informatics & Telecommunications, National Center for Scientific Research–"Demokritos", 15310 Athens, Greece
* Correspondence: espyrou@uth.gr
† This paper is an extended version of our paper published in 2022 17th International Workshop on Semantic and Social Media Adaptation & Personalization (SMAP), Online Event, 3–4 November 2022.

Abstract: During the last few years, several technological advances have led to an increase in the creation and consumption of audiovisual multimedia content. Users are overexposed to videos via several social media or video sharing websites and mobile phone applications. For efficient browsing, searching, and navigation across several multimedia collections and repositories, e.g., for finding videos that are relevant to a particular topic or interest, this ever-increasing content should be efficiently described by informative yet concise content representations. A common solution to this problem is the construction of a brief summary of a video, which could be presented to the user, instead of the full video, so that she/he could then decide whether to watch or ignore the whole video. Such summaries are ideally more expressive than other alternatives, such as brief textual descriptions or keywords. In this work, the video summarization problem is approached as a supervised classification task, which relies on feature fusion of audio and visual data. Specifically, the goal of this work is to generate dynamic video summaries, i.e., compositions of parts of the original video, which include its most essential video segments, while preserving the original temporal sequence. This work relies on annotated datasets on a per-frame basis, wherein parts of videos are annotated as being "informative" or "noninformative", with the latter being excluded from the produced summary. The novelties of the proposed approach are, (a) prior to classification, a transfer learning strategy to use deep features from pretrained models is employed. These models have been used as input to the classifiers, making them more intuitive and robust to objectiveness, and (b) the training dataset was augmented by using other publicly available datasets. The proposed approach is evaluated using three datasets of user-generated videos, and it is demonstrated that deep features and data augmentation are able to improve the accuracy of video summaries based on human annotations. Moreover, it is domain independent, could be used on any video, and could be extended to rely on richer feature representations or include other data modalities.

Keywords: data augmentation; deep visual features; video skimming; video summarization

Citation: Psallidas, T.; Spyrou, E. Video Summarization Based on Feature Fusion and Data Augmentation. *Computers* **2023**, *12*, 186. https://doi.org/10.3390/computers12090186

Academic Editor: Leandros Maglaras

Received: 8 August 2023
Revised: 2 September 2023
Accepted: 8 September 2023
Published: 15 September 2023

1. Introduction

During the last few years and mainly due to the rise of social media and video sharing websites, there has been an exponential increase in the amount of user-generated audio visual content. The average user captures and shares several aspects of her/his daily life moments, such as (a) personal videos, e.g., time spent with friends and/or family and hobbies; (b) activity videos, e.g., sports and other similar activities; (c) reviews, i.e., sharing opinions regarding products, services, movies etc.; and (d) how-to videos, i.e., videos created by users in order to teach other users to fulfill a task. Of course, apart from those that are creating content, a plethora of users daily consume massive amounts of such content. Notably, YouTube users daily watch more than 1 billion hours of visual content,

while creating and uploading more than 500 h [1]. Of course, these numbers are expected to further increase within the next few years, resulting in overexposure to massive amounts of data, which in turn may prohibit users from capturing relevant information. The latter becomes more difficult in case of lengthy content, thus necessitating aid to this task [1].

Video summarization is a promising solution to the aforementioned problem, aiming to extract the most relevant segments from a given video, in order to create a shorter, more informative version of the original video, which is engaging, while preserving its main content and context [2]. Specifically, a generated summary typically consists of a set of representative frames (i.e., the "keyframes") or a set of video fragments. These parts should be kept in their original order, while the summary should be of a much shorter duration than the original video, while including its most relevant elements. Video summarizing applications include efficient browsing and retrieval of visual art movies (such as films and documentaries), TV shows, medical videos, surveillance videos, and so forth.

Video summarization techniques may be categorized into four main categories [1], which differ based on their output, i.e., the actual summary that is delivered to its end user [2]. Specifically, these categories are [3–6] (a) a collection of video frames (keyframes), (b) a collection of video segments, (c) graphical cues, and (d) textual annotations. Summaries belonging to (a) and (b) are frequently referred to as "static" and "dynamic", respectively. Note that a dynamic summary preserves the audio and the motion of videos, whereas a static summary consists of a collection of still video frames. In addition, graphical cues are rarely employed in conjunction with other methodologies. As expected, users tend to prefer dynamic summaries over static ones [7]. Video summarization techniques may be also categorized as (a) unimodal approaches, i.e., those that are based only on the visual content of the video, and (b) multimodal approaches, i.e., those that are using more than one of the available modalities, such as audio, textual, semantic (i.e., depicted objects, scenes, people, etc.) video content [8]. Depending on the training approach that is used, summarization techniques may be categorized as (a) supervised, i.e., those that are based on datasets that have been annotated by human annotators, in either a per-frame or a per-fragment basis; (b) unsupervised, i.e., those that do not rely on some kind of ground truth data, but instead use a large corpus of available data so as to "learn" the important parts; and (c) weakly supervised, i.e., those that do not need exact, full annotations, but instead are based on weak labels, which are imperfect yet able to create powerful predictive models.

In this work, a supervised methodology for the creation of brief dynamic summaries of user-generated audiovisual content is proposed, which falls into a subcategory known as "video skimming" [9,10]; i.e., the goal is to produce an output video consisting of bits of the input video, structured so that the original temporal order is preserved. This sort of summary is critical since it allows for a better comprehension [9] of the original video by its users. The herein presented video summarization approach faces the problem as a classification task. Specifically, the herein presented work relies on annotated ground truth on a per-segment basis. For this, a custom dataset that has been collected and annotated in the context of prior work [1] is mainly used. A set of user-generated activity videos from YouTube was manually collected and was annotated by a collaborative annotation process, involving several users, which were asked to provide binary annotations on a per-fragment basis. Specifically, video segments of 1 s duration were annotated as being "informative" or "noninformative"; i.e., according the the opinion of the human annotators, the former should be part of the summary, while the latter should be omitted. Each video was annotated by more than three users, while the "informativeness" of a given video was decided upon a voting procedure.

The approach for selecting the most informative parts of videos is as follows: It begins with the extraction of (a) handcrafted audio and visual features and (b) deep visual features. The former are extracted using well-known and widely used features, which have been proven reliable and effective in several classification tasks. For the extraction of the latter, the well-defined pretrained model VGG19 [11] is used, which also has been successfully applied into a plethora of computer vision applications. Note that the proposed approach

is multimodal, since the handcrafted features, apart from the visual properties of videos, also capture the aural properties, as well as semantic features (e.g., faces and objects). To produce the visual summaries, both handcrafted and deep feature representations are then fused, and experiments with a set of well-known supervised classification approaches are conducted. To verify the proposed methodology and to assess whether the deep features are able to provide a boost of performance over the use of solely handcrafted features, experiments on three datasets are presented. Specifically, apart from the aforementioned custom dataset that has been created, experiments using two well-known, publicly available datasets, namely, TVSum [12] and SumMe [13], are also presented, while the proposed approach is also compared with a plethora of state-of-the-art techniques. To further improve the outcome of the proposed approach, experiments with augmentation of the training dataset are conducted. Specifically, when experimenting with a given dataset, the remaining two are used as training data. The experimental results indicate that (a) deep features, when fused with handcrafted ones, are able to provide an increase of performance, and (b) augmentation of the training dataset in some cases also provides a significant performance boost.

The remainder of this work is organized as follows: Section 2 includes recent relevant research works from the broader field of video summarization, focusing on those that use deep features. Then, in Section 3, detailed descriptions of the video summarization datasets used in this study are presented. In Section 4, the proposed classification methodology is presented, which comprises handcrafted audio and video extraction, deep feature extraction, and augmentation of a training dataset. Model training, experimental results, and comparisons with the state of the art are presented and discussed in Section 5. Finally, conclusions and future work plans are presented in Section 6.

2. Related Work

In our previous work [1], a user-generated video summary method that was based on the fusion of audio and visual modalities was proposed. Specifically, the video summarization task was addressed as a binary, supervised classification problem, relying on audio and visual features. The proposed model was trained to recognize the "important" parts of audiovisual content. A key component of this approach was its dataset, which consists of user-generated, single camera videos and a set of extracted attributes. Each video included a per-second annotation indicating its "importance".

The fundamental purpose of a video summarization approach, according to [14], is to create a more compact version of the original video, without sacrificing much semantic information, while making it relatively complete for the viewer. In this work, the authors introduced SASUM, a unique approach that, in contrast with previous algorithms that focused just on the variety of the summary, extracted the most descriptive elements of the video while summarizing it. SASUM, in particular, comprised a frame selector and video descriptors to assemble the final video so that the difference between the produced description and the human-created description was minimized. In [15], a user-attention-model-based strategy for keyframe extraction and video skimming was developed. Audio, visual, and linguistic features are extracted, and an attention model is created based on the motion vector field, resulting in the creation of a motion model. Three types of maps based on intensity, spatial coherence, and temporal coherence are created and are then combined to create a saliency map. A static model was also used to pick important backdrop regions and extract faces and camera attention elements. Finally, audio, voice, and music models were developed. To construct an "attention" curve, the aforementioned attention components were linearly fused. Keyframes were extracted from local maxima of this curve within pictures, whereas skim segments were chosen based on a variety of factors.

Based on deep features extracted by a convolutional neural network (CNN), the authors of [16] trained a deep adversarial long short-term memory (LSTM) network consisting of a "summarizer" and a "discriminator" to reduce the distance between ground truth movies and their summaries. The former, in particular, was made up of a selector and an

encoder that picked out relevant frames from the input video and converted them to a deep feature vector. The latter was a decoder that distinguished between "original" and "summary" frames. The proposed deep neural network aimed to deceive the discriminator by presenting the video summary as the original input video, thinking that both representations are identical. Otani et al. [17] proposed a deep video feature extraction approach with the goal of locating the most interesting areas of the movie that are necessary for video content analysis. In [18], the authors focused primarily on building a computational model based on visual attention for summarizing videos from television archives. In order to create a static video summary, their computer model employed a number of approaches, including face identification, motion estimation, and saliency map calculation. The above computational model's final video summary was a collection of important frames or saliency pictures taken from the initial video.

The methodology proposed by [19] used as input sequences original video frames and produced their projected significance scores. Specifically, they adopted a framework for sequence-to-sequence learning so as to formulate the task of summarization, addressing the problems of short-term attention deficiency and distribution inconsistency. Extensive tests on benchmark datasets indicated that the suggested ADSum technique is superior to other existing approaches. A supervised methodology for the automatic selection of keyframes of important subshots of videos is proposed in [20]. These keyframes serve as a summary, while the core concept of this approach is the description of the variable-range temporal dependence between video frames using long short-term memory networks (LSTM), taking into consideration the sequential structure that is essential for producing insightful video summaries. In the work of [21], the specific goal of video summarization was to make it easier for users to acquire movies by creating brief and informative summaries that are diverse and authentic to the original videos. To summarize movies, they used a deep summarization network (DSN), which selected the video frames to be included in summaries, based on probability distributions. Specifically, it forecast a probability per video frame, indicating how likely it is to be selected. Note that, within this process, labels were not necessary; thus, the DNS approach may operate completely unsupervised. In [22], a unique video-summarizing technique called VISCOM was introduced, which is based on the color occurrence matrices from the video, which were then utilized to characterize each video frame. Then, from the most informative frames of the original video, a summary was created. In order to make the aforementioned video-summarizing model robust, VISCOM is tested on a large number of videos from a range of genres.

The authors of [23] present a new approach to supervised video summarization using keyshots. They introduce a soft, self-attention mechanism that is both conceptually simple and computationally efficient. Existing methods in the field utilize complex bidirectional recurrent networks such as BiLSTM combined with attention, which are difficult to implement and computationally intensive compared with fully connected networks. In contrast, the proposed method employs a self-attention-based network that performs the entire sequence-to-sequence transformation in a single forward pass and backward pass during training. The results demonstrate that this approach achieves state-of-the-art performance on two widely used benchmarks in the field, namely, TVSum and SumMe.

Finally, the authors in [24] introduce a novel method for supervised video summarization that addresses the limitations of existing recurrent neural network (RNN)–based approaches, particularly in terms of modeling long-range dependencies between frames and parallelizing the training process. The proposed model employs self-attention mechanisms to determine the significance of video frames. Unlike previous attention-based summarization techniques that model frame dependencies by examining the entire frame sequence, this method integrates global and local multihead attention mechanisms to capture different levels of granularity in frame dependencies. The attention mechanisms also incorporate a component that encodes the temporal position of video frames, which is crucial for video summarization. The results show that the model outperforms exist-

ing attention-based methods and is competitive with other top-performing supervised summarization approaches.

3. Datasets

In this section, the datasets that have been used for the experimental evaluation of this work are presented, specifically, (a) a custom dataset comprising user-generated videos obtained from YouTube [1]; the SumMe dataset, which comprises videos of various sources, such as movies, documentaries, and sports [13]; and (c) the TVSum dataset, which comprises videos of various genres, such as documentaries, vlogs, and egocentric videos [12].

3.1. Custom Dataset

The first dataset that was utilized in this work was initially proposed in our previous work [1] and consists of 409 user-generated videos that were collected from YouTube. In order to collect these videos, the following criteria were adopted: (a) absence of video edits and music scores over the original audio source of the videos and (b) single-camera configuration, e.g., an action camera or a smartphone's camera. Specifically, it comprises videos from the following 14 video categories: car review, motorcycle racing, kayaking, climbing, fishing, spearfishing, snack review, skydiving, roller coasters, theme park review, downhill, mountain bike, survival in the wild, and primitive building.

This dataset was constructed motivated by the intention to propose a summarization approach that could be effectively applied to unedited, "raw" videos. Therefore, the majority of videos in this dataset are based on outdoor activities, such as action and extreme sports. It consists of 409 videos of a total duration of 56.3 h. Their duration ranges between 15 s and ~15 min, while their average duration is ~8.25 min.

The videos were annotated by 22 human annotators in order to construct the ground truth video summaries. The annotation was binary; users were asked to note the time intervals that they found informative/interesting. Note that within each video, a maximum number of informative intervals was not imposed, while annotators were able to freely label the timestamps of each informative time interval, making the whole process completely subjective. Each video was annotated by more than 1 annotator (typically by 3–4 annotators), while their opinions were aggregated upon using a simple majority-based aggregation rule. In particular, each segment of 1 s was considered to be included in the summary (i.e., characterized as "informative") if at least 60% of the annotators agreed on that decision. Note that videos that either were annotated by less than 3 annotators or did not contain any informative segments were discarded. Upon this process, the dataset ended up comprising 336 videos, while it was split into two sets, i.e., a set of 268 training and a set of 68 test videos, comprising 127,972 and 31,113 samples, respectively.

3.2. Public Datasets

The SumMe dataset is a video summarization dataset introduced by Gygli et al. [13]. Specifically, it comprises 25 videos from various sources, such as movies, documentaries, sports events, and online videos. Each video is encoded in MP4 format and has a varying duration. The dataset includes 390 human summaries, while it has been annotated by at least 15 users. Note that annotation is not binary, i.e., ground truth values per frame within the range 0–1. However, since the proposed methodology requires a binary annotation for each second of the video, the ground truth values required a conversion by applying a value of 0.6 as the agreed-upon threshold for aggregation; i.e., any values greater than 0.6 were transformed to 1, indicating informative parts, while values below the threshold were transformed to 0, indicating noninformative parts. Upon this process, SumMe comprises 2687 and 793 training and test samples, respectively.

The TVSum dataset was proposed by Sum et al. [12] and comprises 50 videos of various genres, including news, "how-to", documentaries, vlogs, and egocentric videos. Each video in the dataset is accompanied by 1000 annotations of shot-level importance

scores obtained using a crowdsourcing approach, ending up with 20 annotators per video. Each video is divided into a set of 2 s shots, and scores for each shot are provided for each annotator. Specifically, the annotation is not binary, while users rate the informativeness of each shot compared with other shots from the same video, by providing a score ranging from 1 (low) to 5 (high). In this work, both 1 s halves of each 2 s segment were considered being of equal informativeness; i.e., the same score for both was used, and upon imposing a threshold of 3 (i.e., the median score value) and binarizing, each 1 s segment was ultimately marked as either "informative" or "noninformative". Upon this process, TVSum comprises 10,454 and 2093 training and test samples, respectively.

4. Methodology

The goal of this work is to provide a solution to the problem of video summarization by constructing dynamic visual summaries from unedited, raw videos and extend the methodology that has been presented and evaluated in the context of our previous work [1], wherein video summarization was first tackled as a supervised classification task. The proposed approach falls under the broad area of "video skimming", aiming to select "interesting" fragments of the original video and merge them in order to create a temporally shortened version of it. Specifically, a given video stream is analyzed at a segment level; from each segment of 1 s duration, audio and visual features are extracted. These segments are then classified by supervised classification algorithms as being either "informative" (i.e., adequately interesting so as to be included in the produced video summary) or "noninformative" (i.e., not containing any important information, and thus should not be part of the produced video summary) [1].

For this, supervised binary classifiers that are trained on feature representations of audio, visual, or combined modalities, which include handcrafted features fused with deep visual features, are used. Specifically, deep convolutional neural networks [25] that have been trained on massive datasets and are currently considered as the most advanced technique for the problem of deep visual feature extraction have been applied to the problem at hand. Thus, the final feature vector is formed upon concatenation of both handcrafted audio and visual feature vectors and also the deep visual feature vector. Moreover, in order to train models so as to be more robust, a data augmentation approach, by incorporating features extracted from other datasets within the training process, is also proposed. In Figure 1, a visual overview of the feature extraction and classification pipeline is presented.

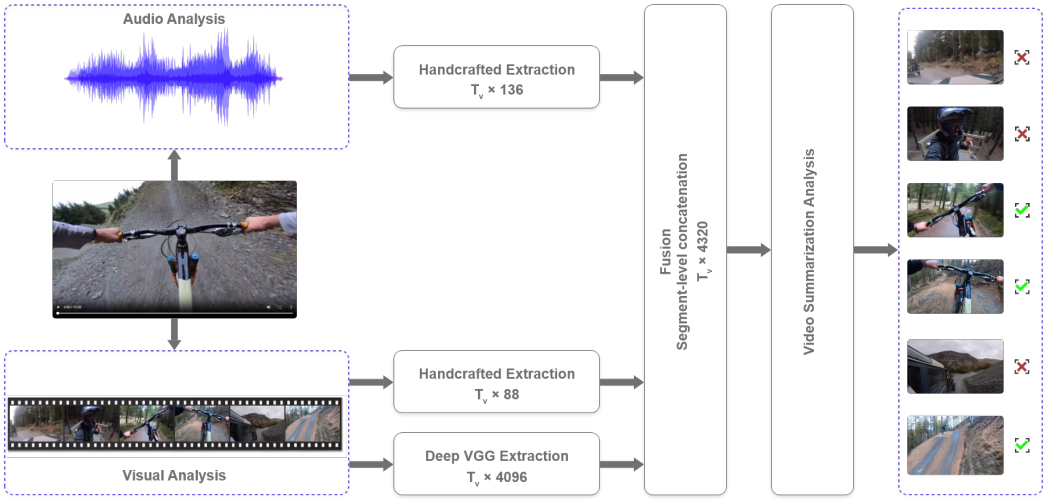

Figure 1. Feature extraction method flow diagram for both audio and visual modalities.

4.1. Feature Extraction

As it has already been mentioned, in this work, two modalities of information extracted from audiovisual content, i.e., the audio and the visual modality, are used. Specifically, handcrafted features are extracted. These have been frequently used in audio and/or visual classification and clustering tasks, such as music information retrieval, auditory scene analysis, video classification, and picture retrieval, to accomplish a rich feature representation in both modalities, accompanied with deep visual features with the use of deep, pretrained models. Our intention was to add as many aspects of audio and visual information as possible. In the following, the proposed feature extraction methodology is presented in depth.

4.2. Audiovisual Handcrafted Feature Extraction

As it has already been mentioned, the goal of this work is to extract both aural and visual features and use them for summarization. Regarding the first modality, the goal is to extract both perceptual and harmonic information of audio signals using a set of handcrafted audio characteristics. Specifically, statistics such as mean and standard deviation of several features are used. These have been proved to be helpful in event recognition tasks, so as to generate feature vectors that capture a low-level representation of an entire audio segment. For this, segment-level audio features per audio clip are extracted using the `pyAudioAnalysis` Python library [26] and `ffmpeg` (note: https://ffmpeg.org, accessed on 15 July 2023).

Initially, short-term audio feature extraction is performed, and then segment-level feature statistics are calculated, so as to create the audio segment representation. Short-term processing is carried out for each segment of the audio signal, with each short-term window having 68 short-term features. The audio signal is divided into segment-level windows (either overlapping or nonoverlapping) (i.e., 34 features and 34 deltas). Segment windows may range from 0.5 s to several seconds, depending on what defines a homogeneous segment in each application area, whereas short-term windows commonly range from 10 to 200 ms. Time domain, frequency domain, and cepstral domain are the three categories of short-term features that are extracted; they are presented in Table 1. Note that each short-term feature sequence's mean and standard deviation comprise 136 features in total, which form the audio feature vector.

Table 1. Extracted short-term audio features.

Index	Name	Description
1	Zero Crossing Rate	Rate of sign changes of the frame
2	Energy	Sum of squares of the signal values, normalized by frame length
3	Entropy of Energy	Entropy of subframes' normalized energies. A measure of abrupt changes
4	Spectral Centroid	Spectrum's center of gravity
5	Spectral Spread	Spectrum's second central moment of the spectrum
6	Spectral Entropy	Entropy of the normalized spectral energies for a set of subframes
7	Spectral Flux	Squared difference between the normalized magnitudes of the spectra of the two successive frames
8	Spectral Rolloff	The frequency below in which 90% of the magnitude distribution of the spectrum is concentrated.
9–21	MFCCs	Mel frequency cepstral coefficients: a cepstral representation with mel-scaled frequency bands
22–33	Chroma Vector	A 12-element representation of the spectral energy in 12 equal-tempered pitch classes of Western-type music
34	Chroma Deviation	Standard deviation of the 12 chroma coefficients

Lastly, for the visual modality, a wide range of visual features are extracted in order to capture the visual properties of video sequences. Since this modality is considered to be more important as it has been experimentally demonstrated in previous works [1,27], the goal here was to extract a richer representation compared with the audio modality. Specifically, every 0.2 s, 88 visual features from the corresponding frame are extracted; these are presented in Table 2 and may be grouped depending on their feature level representation, such as, low (i.e., simple color aggregates), mid (i.e., optical flow features), and high (i.e., objects and faces).

Table 2. Extracted visual features.

Index	Type	Description
1–45	Color	8-bin histograms of red/green/blue/grayscale/saturation values, 5-bin histogram of the max-by-mean ratio for each RGB triplet
46	Frame Differences	Average absolute difference between two successive frames in gray scale
47–48	Facial	Detection/number of faces, average ratio of the faces' bounding box areas divided by frame size, using the Viola–Jones approach [28]
49–51	Optical Flow	Optical flow estimation using Lucas–Kanade [29]; extraction of (a) average magnitude, (b) standard deviation of the angles of the flow vectors, and (c) the ratio of the magnitude of the flow vectors by the deviation of the angles of the flow vectors (the latter measures the possibility of a camera tilt movement)
52	Shot Duration	Using a shot detection method, extraction of the length of the shot (in s) that a given frame belongs to
53–88	Object Related	Utilizing the single-shot multibox detector [30] technique, 12 objects are recognized and then 3 statistics per frame are extracted: (a) number of items detected, (b) average detection confidence, and (c) average ratio of the objects' area to the frame's area. Thus, $3 \times 12 = 36$ object-related features are extracted. Note that objects belonging to the following categories are recognized: person, car, animal, accessory, sport, kitchen, food, furniture, electronic device, interior device

4.3. Deep Visual Feature Extraction

In order to enhance the visual feature representation, deep visual features, by using already-trained deep neural networks, are extracted. Specifically, knowledge from transfer learning [31] is a technique for utilizing previously trained models for some task into a new task. Tasks such as classification, regression, and clustering may benefit from transfer learning. Thus, since the proposed approach tackles video summarization as a classification task, a visual domain adaptation approach [32] is adopted. The deep visual features are extracted upon analyzing the first frame of each 1 s segment of the video, depending on the frames per second (fps) retrieved from each video's metadata.

The frame vector of each second of the video is given as input in the VGG19 pretrained model, which is a VGG model [11] version that consists of 19 layers in total (16 convolution layers, 3 fully connected layers, 5 max pooling layers, and 1 SoftMax layer). In this work, the last 4 layers from VGG19 were removed in order to reduce the domain specification of the model [33] and using the corresponding intermediate output of the network as the feature representation for the visual modality, resulting in a total of 4096 features.

4.4. Dataset Augmentation

The final step, in order to enhance the predictive power and the robustness of the proposed methodology, is the augmentation of the training dataset. This was achieved by incorporating additional data from the SumMe and the TVSum datasets. This aug-

mentation step resulted in broadening the diversity of training data, contributing to the overall generalization ability of the model. The training datasets used in case of SumMe and TVSum summarization tasks were also augmented. Specifically, both datasets were partitioned into distinct training and testing subsets by allocating 80% and 20% of the available videos, respectively. When testing with a given dataset, its training data were augmented with the remaining two datasets. Note that prior to the augmentation step, the transformation approaches that have been previously described in Section 3.2 in case of SumMe and TVSum, were followed so as to obtain binary annotations for each 1 s video segment.

5. Results and Experiments

This section presents experimental results of the proposed video summarization approach using the datasets that have been previously presented in Section 3. As in our prior work [1], the following classifiers for the classification of the fused features resulting upon the combination of handcrafted and deep visual features are presented: (a) naive Bayes (NB), (b) k-nearest neighbors (kNN), (c) logistic regression (LR), (d) decision trees (DT), (e) random forests (RF), and (f) XGBoost (XGB). The classifiers of cases of (a)–(d) have been implemented using the scikit-learn library [34], with the appropriate parameter adjustments. Specifically, in the case of kNN adjusted, the k parameter, which stands for the number of nearest neighbors, was adjusted. Moreover, in the case of LR experiments involved the C inverse regularization parameter. In the case of DT, the split quality measure criterion for DT (i.e., Gini impurity, entropy, etc.) was varied. It should be noted that RF is a collection of DT models that may be seen of as an improvement of bagging. For their construction, the balanced classifier from [35] was utilized and experiments involved the maximum tree depth. Finally, for the XGB, the classifier from [36] was used. In terms of the split quality measure criterion and the quantity of tree estimators, both tree classifiers have been optimized. In the case of kNN, k was set equal to 5, while in the case of LR, the value of C was set equal to 10. The DT classifier adopts the "entropy" criterion for its information gain strategy and also limits the maximum tree depth to 6 in order to prevent overfitting. Accordingly, the RF classifier adopts the Gini impuriry criterion, while the maximum number of trees is 400. Similarly, also in the case of the XGBoost classifier, the maximum number of trees is set to 400.

To validate the effectiveness of the proposed approach, the macro averaged F1-score, i.e., the macro average of the individual class-specific F1-scores, is computed. Note that F1-score is the harmonic mean of recall and precision, per class; therefore, the F1 macro average provides an overall normalized metric for the general classification performance. Additionally, note that the F1 macro average provides a general evaluation metric of video summarization, formulated as a classification task, taking into account the class imbalance of the task and aggregating precision and recall.

In Table 3, the F1-scores for the six different classification methods applied to the herein used custom dataset are depicted, and for the cases of (a) using only handcrafted features, (b) fusing handcrafted and deep features, (c) fusing handcrafted and deep features and also augmenting the training dataset with SumMe and TVSum. Accordingly, in Tables 4 and 5, results for SumMe and TVSum are presented. Moreover, in Table 6, comparisons of the proposed approach with several approaches that are based on supervised learning techniques are presented.

In the case of the custom dataset, it may be observed that deep features provided a boost of performance in most cases, while the overall best F1-score in that case was increased, compared with the use of handcrafted features only. Specifically, 5 out of 6 classifiers demonstrated a higher F1-score, while kNN showed equivalent performance. The best performance using handcrafted features was observed in the case of using XGB and was 62.3, while in the case of handcrafted features and using the same classifier, it was 59.7. Additionally, in the case of the fusion of handcrafted and deep features, XGB demonstrated the best overall performance, which in that case was 63.7. However, by augmenting the

training dataset with all available samples of SumMe and TVSum, performance was further improved, reaching a highest F1-score equal to 65.8, when using the RF classifier. Note that in this case, 5 out 6 classifiers exhibited improved performance, with the only exception being the DT, which showed almost equivalent performance.

Table 3. Video summarization performance at the custom dataset. Numbers indicate the macro-averaged F1-score. Bold indicates the best overall result. HF, DF, and DA denote the use of handcrafted features, deep features, and data augmentation, respectively.

	HF	**DF**	**HF + DF**	**HF + DF + DA**
Naive Bayes	51.6	52.9	53.9	60.6
KNN	57.7	54.8	57.2	58.4
Logistic Regression	49.4	54.4	57.3	62.9
Decision Tree	45.6	41.3	62.1	61.5
Random Forest	60.6	58.2	61.5	**65.8**
XGBoost	62.3	59.7	63.7	64.8

Table 4. Video summarization performance using the SumMe dataset. Numbers indicate the macro-averaged F1-score. Bold indicates the best overall result. HF, DF, and DA denote the use of handcrafted features, deep features, and data augmentation, respectively. In this case, DA considers the custom dataset and TVSum.

	HF	**DF**	**HF + DF**	**HF + DF + DA**
Naive Bayes	50.6	46.7	49.2	35.2
KNN	46.1	38.9	48.2	**58.4**
Logistic Regression	40.7	43.9	50.6	51.5
Decision Tree	46.8	51.7	51.2	46.2
Random Forest	44.3	44.5	43.2	55.9
XGBoost	46.9	50.4	48.7	57.3

Table 5. Video summarization performance using the TVSum dataset. Numbers indicate the macro-averaged F1-score. Bold indicates the best overall result. HF, DF, and DA denote the use of handcrafted features, deep features, and data augmentation, respectively. In this case, DA considers the custom dataset and SumMe.

	HF	**DF**	**HF + DF**	**HF + DF + DA**
Naive Bayes	41.8	55.4	46.6	52.8
KNN	52.2	46.9	54.3	52.8
Logistic Regression	47.7	57.3	57.0	50.2
Decision Tree	45.2	47.1	47.9	47.0
Random Forest	56.4	57.6	**60.3**	56.9
XGBoost	47.5	50.4	55.4	55.1

Continuing with the SumMe dataset, it may be observed that the fusion of handcrafted with deep features provided a boost of performance in 4 out of 6 cases, while the overall best score was also increased in this dataset. A slight decrease in performance was observed in the cases of NB and RF. Specifically, the overall best F1-score was 50.6 in the case of using only handcrafted features and the NB classifier and was increased to 51.2 when adding the deep features and the DT. Notably, by using only deep features, slightly increased performance was achieved, i.e., 51.7 by also using the DT. By augmenting the dataset with the custom one and TVSum, a further performance boost was observed. The F1-score was increased in 4 out of 6 classifiers, while best F1-score was equal to 58.4. Notably, the NB exhibited a high drop of performance. A smaller drop of performance was also observed in the case of DT.

Table 6. Comparisons of the proposed approach with state-of-the-art, supervised methods using SumMe and TVSum datasets. Numbers denote F1-score. The best result per dataset is indicated in bold.

Research Work	Dataset	
	SumMe	**TVSum**
Zhang et al. [20]	38.6	54.7
Elfeki and Borji [37]	40.1	56.3
Lebron Casas and Koblents [38]	43.8	-
Zhao et al. [39]	42.1	57.9
Ji et al. [40]	44.4	61.0
Huang and Wang [41]	46.0	58.0
Rochan et al. [42]	48.8	58.4
Zhao et al. [43]	44.1	59.8
Yuan et al. [44]	47.3	58.0
Feng et al. [45]	40.3	**66.8**
Zhao et al. [46]	44.3	60.2
Ji et al. [47]	45.5	63.6
Li et al. [48]	52.8	58.9
Chu et al. [49]	47.6	61.0
Liu et al. [50]	51.8	60.4
Wang et al. [51]	58.3	64.5
Apostolidis et al. [24]	55.6	64.5
Proposed approach	**58.4**	60.3

Finally, in the case of TVSum, the addition of deep features led to an increase in F1-score. The best overall F1-score in both cases was observed when using the RF classifier. In the case of using only handcrafted features, the best F1-score was 56.4, which was increased to 57.6 using only deep features and then to 60.3 by fusing both types of features. All 6 classifiers exhibited an increase in performance due to the addition of the deep features. Notably, in the case of TVSum, the augmentation of the training dataset was unable to provide a further increase in the best overall performance. Specifically, 5 out of 6 classifiers showed a drop of performance, while the best overall F1-score decreased to 56.9, again when using the RF.

To compare the proposed approach with state-of-the-art research works that are based on supervised learning, TVSum and SumMe were used, which, as already mentioned, are popular video summarization datasets. These comparisons are depicted in Table 6. As it could be observed, the proposed approach achieved the best performance in SumMe; it ranked eighth overall. These results clearly indicate its potential. Note that almost all approaches presented in Table 6 are based on deep architectures, while the herein proposed one is based on traditional machine learning algorithms, while it uses deep architectures only for feature extraction and not for classification.

6. Conclusions and Future Work

In this paper, previous research work on video summarization has been extended by (a) introducing pretrained models as transfer learning feature extractors in order to represent the audiovisual content of videos in a more intuitive form, disengaged from the exclusive use of handcrafted features, and (b) augmenting the training dataset by using other datasets in the training procedure. The performance was evaluated using a custom dataset comprising YouTube user-generated video clips that had been collected and annotated on a 1 s basis as "informative" and "noninformative". This process was performed by human annotators, who annotated as "informative" video segments which to their opinion were suitable to be part the final video summary. Two publicly available datasets were also used, namely, TVSum and SumMe. Video summarization was formulated as a binary clas-

sification task, and feature vector representations were extracted from 1 s video segments of the audiovisual streams of videos. Both handcrafted and deep features were fused into a single vector, and experiments with a set of six well-known classifiers were performed, with and without augmentation of training dataset. It has been experimentally shown that the herein proposed approach achieved better performance than most contemporary research works, and it has been verified that, in most cases, deep features, when fused with the handcrafted ones, are able to provide a boost of performance. The latter may be further improved by augmenting the training dataset with examples from other datasets.

Plans for future work include the enrichment of the custom dataset with several other datasets comprising more videos from various and heterogeneous domains. We believe that such an augmentation may lead to a further increase in performance as it will allow training of a more robust video summarization model. Moreover the proposed approach could be extended from a fully to a weakly supervised methodology. Additionally, as several discrepancies among annotators were observed, it would be interesting to further experiment with the voting scheme and the aggregation approach, and also try to detect and exclude "mischievous" annotators, in an overall effort to enhance the objectivity of our models. It will be also interesting to exploit speech and textual modalities that may be present within the aural and the visual part of the video sequences. In addition to the deep feature extraction, deep sequence classification approaches with attention mechanisms have been extremely promising, leading to better results; therefore, it is within our immediate plans to investigate such methodologies.

Author Contributions: Conceptualization, T.P. and E.S.; methodology, T.P. and E.S.; software, T.P.; validation, E.S.; data curation, T.P.; writing—original draft preparation, T.P. and E.S.; writing—review and editing, E.S.; visualization, T.P.; supervision, E.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The SumMe dataset is available at https://gyglim.github.io/me/vsum/index.html (accessed on 31 July 2023). The TVSum dataset is available at https://github.com/yalesong/tvsum (accessed on 31 July 2023). Our dataset is available at https://github.com/theopsall (accessed on 31 July 2023).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	convolutional neural network
DA	data augmentation
DSN	deep summarization network
DF	deep features
DT	decision tree
HF	handcrafted features
kNN	k-nearest neighbor
LR	logistic regression
LSTM	long short-term memory
NB	naive Bayes
RNN	recurrent neural networks
RF	random forest
XGB	XGBoost

References

1. Psallidas, T.; Koromilas, P.; Giannakopoulos, T.; Spyrou, E. Multimodal summarization of user-generated videos. *Appl. Sci.* **2021**, *11*, 5260. [CrossRef]
2. Money, A.G.; Agius, H. Video summarisation: A conceptual framework and survey of the state of the art. *J. Vis. Commun. Image Represent.* **2008**, *19*, 121–143. [CrossRef]

3. Chen, B.C.; Chen, Y.Y.; Chen, F. Video to Text Summary: Joint Video Summarization and Captioning with Recurrent Neural Networks. In Proceedings of the BMVC, London, UK, 4–7 September 2017.
4. Li, Y.; Merialdo, B.; Rouvier, M.; Linares, G. Static and dynamic video summaries. In Proceedings of the 19th ACM International Conference on Multimedia, Scottsdale, AZ, USA, 28 November–1 December 2011; pp. 1573–1576.
5. Lienhart, R.; Pfeiffer, S.; Effelsberg, W. The MoCA workbench: Support for creativity in movie content analysis. In Proceedings of the Third IEEE International Conference on Multimedia Computing and Systems, Hiroshima, Japan, 17–23 June 1996; pp. 314–321.
6. Spyrou, E.; Tolias, G.; Mylonas, P.; Avrithis, Y. Concept detection and keyframe extraction using a visual thesaurus. *Multimed. Tools Appl.* **2009**, *41*, 337–373. [CrossRef]
7. Li, Y.; Zhang, T.; Tretter, D. *An Overview of Video Abstraction Techniques*; Technical Report HP-2001-191; Hewlett-Packard Company: Palo Alto, CA, USA, 2001
8. Apostolidis, E.; Adamantidou, E.; Metsai, A.I.; Mezaris, V.; Patras, I. Video summarization using deep neural networks: A survey. *Proc. IEEE* **2021**, *109*, 1838–1863. [CrossRef]
9. Sen, D.; Raman, B. Video skimming: Taxonomy and comprehensive survey. *arXiv* **2019**, arXiv:1909.12948.
10. Smith, M.A.; Kanade, T. *Video Skimming for Quick Browsing Based on Audio and Image Characterization*; School of Computer Science, Carnegie Mellon University: Pittsburgh, PA, USA, 1995
11. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
12. Song, Y.; Vallmitjana, J.; Stent, A.; Jaimes, A. Tvsum: Summarizing web videos using titles. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5179–5187.
13. Gygli, M.; Grabner, H.; Riemenschneider, H.; Van Gool, L. Creating summaries from user videos. In Proceedings of the Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; pp. 505–520.
14. Wei, H.; Ni, B.; Yan, Y.; Yu, H.; Yang, X.; Yao, C. Video summarization via semantic attended networks. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–3 February 2018; Volume 32.
15. Ma, Y.F.; Lu, L.; Zhang, H.J.; Li, M. A user attention model for video summarization. In Proceedings of the Tenth ACM International Conference on Multimedia, Juan-les-Pins, France, 1–6 December 2002; pp. 533–542.
16. Mahasseni, B.; Lam, M.; Todorovic, S. Unsupervised video summarization with adversarial lstm networks. In Proceedings of the IEEE con ference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 202–211.
17. Otani, M.; Nakashima, Y.; Rahtu, E.; Heikkilä, J.; Yokoya, N. Video summarization using deep semantic features. In Proceedings of the Asian Conference on Computer Vision, Taipei, Taiwan, 20–24 November 2016; Springer: Cham, Switzerland, 2016; pp. 361–377.
18. Jacob, H.; Pádua, F.L.; Lacerda, A.; Pereira, A.C. A video summarization approach based on the emulation of bottom-up mechanisms of visual attention. *J. Intell. Inf. Syst.* **2017**, *49*, 193–211. [CrossRef]
19. Ji, Z.; Jiao, F.; Pang, Y.; Shao, L. Deep attentive and semantic preserving video summarization. *Neurocomputing* **2020**, *405*, 200–207. [CrossRef]
20. 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, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 766–782.
21. Zhou, K.; Qiao, Y.; Xiang, T. Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–3 February 2018; Volume 32.
22. Mussel Cirne, M.V.; Pedrini, H. VISCOM: A robust video summarization approach using color co-occurrence matrices. *Multimed. Tools Appl.* **2018**, *77*, 857–875. [CrossRef]
23. Fajtl, J.; Sokeh, H.S.; Argyriou, V.; Monekosso, D.; Remagnino, P. Summarizing videos with attention. In Proceedings of the Computer Vision—ACCV 2018 Workshops: 14th Asian Conference on Computer Vision, Perth, Australia, 2–6 December 2018; Revised Selected Papers 14; Springer International Publishing: Berlin/Heidelberg, Germany, 2019; pp. 39–54.
24. Apostolidis, E.; Balaouras, G.; Mezaris, V.; Patras, I. Combining global and local attention with positional encoding for video summarization. In Proceedings of the 2021 IEEE international symposium on multimedia (ISM), Naple, Italy, 29 November–1 December 2021; pp. 226–234.
25. Hertel, L.; Barth, E.; Käster, T.; Martinetz, T. Deep convolutional neural networks as generic feature extractors. In Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN), Killarney, Ireland, 12–17 July 2015; pp. 1–4.
26. Giannakopoulos, T. pyaudioanalysis: An open-source python library for audio signal analysis. *PLoS ONE* **2015**, *10*, e0144610. [CrossRef]
27. Psallidas, T.; Vasilakakis, M.D.; Spyrou, E.; Iakovidis, D.K. Multimodal video summarization based on fuzzy similarity features. In Proceedings of the 2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), Nafplio, Greece, 26–29 June 2022; pp. 1–5.
28. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, Kauai, HI, USA, 8–14 December 2001; Volume 1.
29. Lucas, B.D.; Kanade, T. An iterative image registration technique with an application to stereo vision. In Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81), Vancouver, BC, Canada, 24–28 August 1981
30. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 21–37.

31. Tammina, S. Transfer learning using vgg-16 with deep convolutional neural network for classifying images. *Int. J. Sci. Res. Publ. (IJSRP)* **2019**, *9*, 143–150. [CrossRef]
32. Wang, M.; Deng, W. Deep visual domain adaptation: A survey. *Neurocomputing* **2018**, *312*, 135–153. [CrossRef]
33. Yu, W.; Yang, K.; Bai, Y.; Xiao, T.; Yao, H.; Rui, Y. Visualizing and comparing AlexNet and VGG using deconvolutional layers. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016.
34. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
35. Lemaître, G.; Nogueira, F.; Aridas, C.K. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *J. Mach. Learn. Res.* **2017**, *18*, 559–563.
36. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM sigkdd International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
37. Elfeki, M.; Borji, A. Video summarization via actionness ranking. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 754–763.
38. Lebron Casas, L.; Koblents, E. Video summarization with LSTM and deep attention models. In Proceedings of the International Conference on Multimedia Modeling, Bangkok, Thailand, 5–7 February 2018; Springer International Publishing: Cham, Switzerland, 2018; pp. 67–79.
39. Zhao, B.; Li, X.; Lu, X. Hierarchical recurrent neural network for video summarization. In Proceedings of the 25th ACM International Conference on Multimedia, Mountain View, CA, USA, 23–27 October 2017; pp. 863–871.
40. Ji, Z.; Xiong, K.; Pang, Y.; Li, X. Video summarization with attention-based encoder–decoder networks. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 1709–1717. [CrossRef]
41. Huang, C.; Wang, H. A novel key-frames selection framework for comprehensive video summarization. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 577–589. [CrossRef]
42. Rochan, M.; Ye, L.; Wang, Y. Video summarization using fully convolutional sequence networks. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 347–363.
43. Zhao, B.; Li, X.; Lu, X. Hsa-rnn: Hierarchical structure-adaptive rnn for video summarization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7405–7414.
44. Yuan, Y.; Li, H.; Wang, Q. Spatiotemporal modeling for video summarization using convolutional recurrent neural network. *IEEE Access* **2019**, *7*, 64676–64685. [CrossRef]
45. Feng, L.; Li, Z.; Kuang, Z.; Zhang, W. Extractive video summarizer with memory augmented neural networks. In Proceedings of the 26th ACM International Conference on Multimedia, Seoul, Republic of Korea, 22–26 October 2018; pp. 976–983.
46. Zhao, B.; Li, X.; Lu, X. TTH-RNN: Tensor-train hierarchical recurrent neural network for video summarization. *IEEE Trans. Ind. Electron.* **2020**, *68*, 3629–3637. [CrossRef]
47. Ji, Z.; Zhao, Y.; Pang, Y.; Li, X.; Han, J. Deep attentive video summarization with distribution consistency learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 1765–1775. [CrossRef]
48. Li, P.; Ye, Q.; Zhang, L.; Yuan, L.; Xu, X.; Shao, L. Exploring global diverse attention via pairwise temporal relation for video summarization. *Pattern Recognit.* **2021**, *111*, 107677. [CrossRef]
49. Chu, W.T.; Liu, Y.H. Spatiotemporal modeling and label distribution learning for video summarization. In Proceedings of the 2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP), Kuala Lumpur, Malaysia, 27–29 September 2019; pp. 1–6.
50. Liu, Y.T.; Li, Y.J.; Yang, F.E.; Chen, S.F.; Wang, Y.C.F. Learning hierarchical self-attention for video summarization. In Proceedings of the 2019 IEEE int ernational conf erence on im age proc essing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 3377–3381.
51. Wang, J.; Wang, W.; Wang, Z.; Wang, L.; Feng, D.; Tan, T. Stacked memory network for video summarization. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 836–844.