Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Full length article

Unsupervised video summarization with adversarial graph-based attention network[☆]

Jeshmitha Gunuganti [a,*], Zhi-Ting Yeh [a], Jenq-Haur Wang [b], Mehdi Norouzi [a]

[a] *Department of Electrical Engineering and Computer Science, University of Cincinnati, Cincinnati, OH, 45221, United States*

[b] *Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei, 106344, Taiwan*

ARTICLE INFO

Keywords:
Video summarization
Video structure analysis
Video keyframe extraction
Video highlight detection
Generative adversarial network
Graph neural network

ABSTRACT

Video summarization aims to select a subset of video segments that best capture the video storyline. Our study seeks to train an encoder to transform the raw frame features extracted from pre-trained CNN models into representations that embody importance and guide the selection of the video segment. Our main idea is to use graph modeling and attention mechanisms to train the encoder adversarially. The graph representation enables the model to learn the relationship among frames, revealing the intrinsic structure of a video. The attention mechanism allows the model to capture the magnitude of these relationships. In the proposed model, an attention-based encoder is trained using a graph-based generator that reconstructs videos using the encoded features and a discriminator that guides the generator, distinguishing the original and reconstructed video. Thus, by leveraging graph attention and refining mechanisms, the proposed model offers distinct advantages over existing methods, including enhanced summarization accuracy, improved preservation of temporal coherence, and the ability to capture complex semantic linkages within video content. These advancements are substantiated through a comprehensive ablation study, which demonstrates the efficacy of our model using various evaluation metrics — Kendall and Spearman coefficients. The proposed model is evaluated on TVSum and SumMe datasets and achieves results on par with supervised models that used similar encoders and achieved state-of-the-art results compared to other unsupervised models.

1. Introduction

Given the rapid growth in video data created, shared, and viewed every minute (Users stream 694K hours of video on YouTube every minute [1]), the need for video processing tools to help users process/retrieve information becomes essential. Video summaries representing informative video frames/segments can help users browse videos quickly and efficiently.

Researchers have tried to automate video summarization using supervised, unsupervised, and semi-supervised approaches. Supervised models [2–11] rely on human annotations in the form of key frames or key segments and are trained by minimizing the loss defined as the difference between predicted frame importance scores and the ground truth. However, considering the time and effort needed, these human annotations are expensive to collect. Moreover, summarizing a video is highly subjective since annotators have different perspectives and motives for retrieving information. Hence, the ground truth scores or summaries given by different annotators can vary significantly, and training models using them can be misleading. Whereas using an unsupervised approach as proposed in [12–20] does not require human intervention, labeled ground truths, and its result is capable of generalizing in a broader range of domains.

This paper introduces an unsupervised model to summarize videos following previous studies by Mahasseni et al. [17] and Park et al. [9]. Mahasseni et al. [17] employed Generative Adversarial Networks to summarize videos for the first time. Park et al. [9] achieved state-of-the-art results approaching video summarization as a graph modeling problem, validating the importance of understanding the intrinsic structure of the video for encoding semantic relationships among key frames. Fig. 1 depicts our approach to predicting frame importance scores by modeling a video as a graph by minimizing feature differences. The encoded features are the bases for key frame extraction and estimation of frame importance score. Through evaluation, we show that the proposed model achieves competitive results in both summary F-score and

[☆] This paper has been recommended for acceptance by Zicheng Liu.

[*] Corresponding author.
E-mail address: gunugaja@mail.uc.edu (J. Gunuganti).

https://doi.org/10.1016/j.jvcir.2024.104200
Received 5 February 2023; Received in revised form 28 May 2024; Accepted 4 June 2024
Available online 14 June 2024

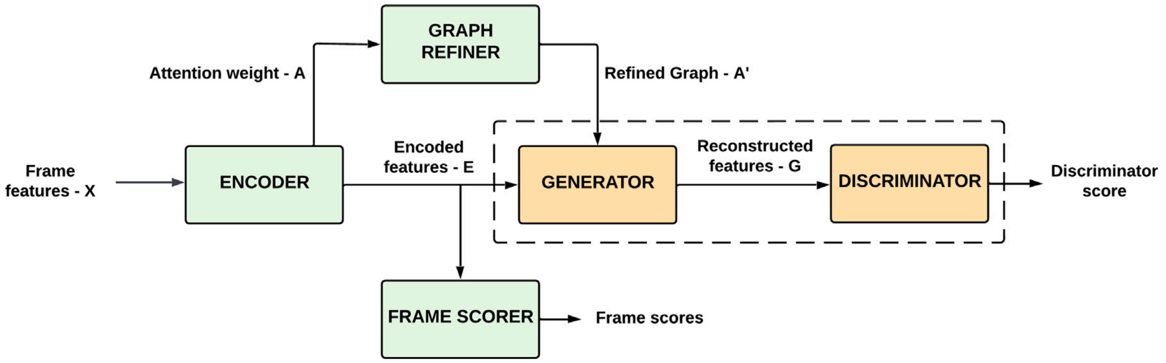

Fig. 1. Our approach: To predict the frame importance scores from the encoded features formed by minimizing the difference between the original video feature and reconstructed video feature given by discriminator.

frame-level importance score correlation metrics suggested by Otani et al. [21]. Our contributions are summarized below:

- To the best of our knowledge, our work is the first to integrate graph and attention mechanisms to perform unsupervised video summarization. Moreover, rather than a fully connected graph, we introduce a graph-refining process to reduce the complexity of the model while capturing the relations in the video.
- The proposed model outperforms existing methods on the SumMe dataset when trained on other datasets — TVSum, OVP, and YouTube proving its transfer capability in learning the structure of a video and generating generic video summaries
- The experimental results show that our model achieves the state-of-the-art result on TVSum performing consistently over all three settings — canonical, augmented, and transfer, unlike other existing models whose performance decreases significantly on transfer setting verifying the impact of using attention-based graph modeling to interpret a video.

The paper is organized as follows: The related work is presented in Section 2, followed by our proposed model in Section 3. We discuss our experiments, evaluation protocol, and results in Section 4. Moreover, conclude in Section 5.

2. Related work

Given that a video is a sequence, supervised approaches use Recurrent Neural Networks (RNN) like Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) to model the frame dependencies. Zhang et al. [2] proposed a bidirectional LSTM-based model that captures temporal dependencies between the past and the future frames. Then, frame importance scores are predicted using a Multi-Layer Perceptron (MLP) that takes in the LSTM hidden states and visual features. Kulesza et al. [3] enhanced the results of [2] using the Determinantal Point Process (DPP), which is a probabilistic model for diverse subset selection. Lebron Casas et al. [4] built upon vsLSTM and dppLSTM models proposed by Zhang et al. [2], incorporating the attention network to capture the user's emphasis along the video. The attention mechanism is added to the relevance and similarity branches of vsLSTM and dppLSTM, respectively. Considering the limitations of RNNS in modeling long-term dependencies, Wang et al. [8] proposed a stacked memory network consisting of multiple LSTM layers, each augmented with a hierarchical memory layer. Ji et al. [5] proposed an attentive encoder–decoder network in which the encoder encodes the frame dependencies using a BiLSTM, and the decoder is a combination of two attention-based LSTM networks, A-AVS and M-AVS, applying additive and multiplicative objective functions, respectively. The sequential nature of the RNNs, precluding parallelization and making them computationally complex [22], led to the introduction of the Transformer by Vaswani et al. [22]. Employing a Transformer-encoder network, Fajtl et al. [7] proposed a model, VASNet, which uses a single global, soft self-attention layer that models the frame dependencies and a dual-layer fully connected network that takes the weighted features as input and predicts the frame importance scores. Liu et al. [6] combined a shot-level reconstruction model and multi-head attention network to extract video key frames for the first time.

Weakly supervised approaches break through the need for extensive labeled data utilizing various methods, such as integrating video metadata in the training procedure or using weakly annotated labels [23]. Panda et al. [24] proposed a 3D CNN model to categorize videos and then train a model that ranks segments based on the relevance of the video category to the segments.

In contrast to the supervised approaches, traditional unsupervised techniques try to summarize videos using clustering algorithms [12, 13], where frames or shots with similar features are grouped in the same cluster. Kuanar et al. [25] proposed a key frame extraction method using dynamic Delaunay graph clustering with an iterative edge pruning mechanism. Parihar et al. [26] presented a multiview summarization method, which first utilizes Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) on initial frames to eliminate redundancy. Second, shot boundaries are identified by measuring frame similarity using the Jaccard index and the Dice score. Finally, multi-level K-Means were used to extract the key frames. Uchihashi et al. [12] hierarchically clustered the shots using the Euclidean distance among the high-variance Discrete Cosine Transformation (DCT) coefficients. Cluster weight was assigned based on the fraction of the video occupied, and the shot importance score was estimated using shot length and the cluster weight to which it belongs. Jadon et al. [13] clustered the frames using K-means and Gaussian mixture model based on various features: histograms, scale-invariant feature transformations (SIFT), and deep features extracted from pre-trained VGG16 [27] and RestNet16 [28] models. The frames closest to the center of each cluster were chosen as key frames. The video's sequential order may not be retained while using the K-means for clustering. Thus, Lai et al. [29] proposed a time-constrained clustering algorithm that retains the sequential frame order in the video. A key frame was selected from each cluster based on the color, motion, and texture features. Naveed Ejaz et al. [30] proposed an adaptive aggregation formula that extracts key frames by combining metrics from color histograms, RGB channels, and moments of inertia. Hannane et al. [31] performed key frame extraction using global orientation features and an adaptive mean shift algorithm. They also proposed a linear algorithm to eliminate redundant frames based on visual and temporal information.

More recent unsupervised approaches are built on the principle that video summaries should contain the information required to reconstruct the original video [12–19,32]. Therefore, generative models are trained to generate video summaries while trying to reconstruct the original video with minimum information loss. VAE (Variational Auto-Encoder) and GAN (Generative Adversarial Network) are two

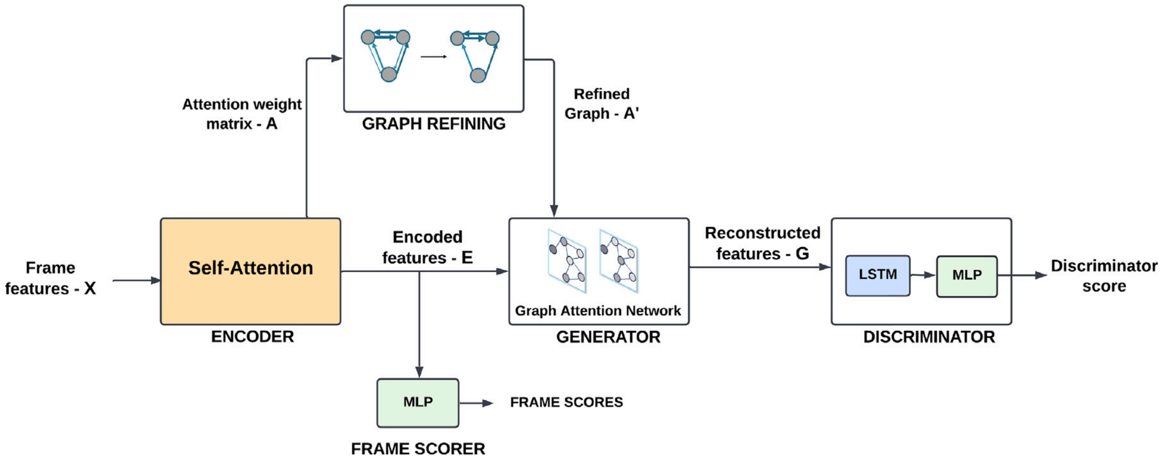

Fig. 2. Model architecture: Transforms the frame features using the attention-based encoder when decoded (by the generator), leads to a reconstructed video representation that is difficult to distinguish from the original video (by the discriminator).

generative models that are highly effective in capturing complex data distributions, as well as latent representations [33,34]. VAE consists of an encoder and decoder and aims to model the posterior distribution close to the prior distribution. In contrast, a GAN comprises of a generator and a discriminator, trained competitively [35]. VAEs are typically easier to train and tend to produce expected but blurry/low-quality samples in image-generation tasks. The GANs, on the other hand, can generate higher-quality images but experience training instabilities and require oversight to achieve meaningful results. Larsen et al. [36] proposed VAE-GAN to maintain the benefits of both models and outperform conventional VAEs. Mahasseni et al. [17] employed VAE-GAN architecture for video summarization in which the summarizer and discriminator, built using LSTMs, were trained adversarially, minimizing the distance between the original video and the summary. Apostolidis et al. [14] built upon [17] by suggesting a stepwise, label-based approach for training and later introduced an attention layer between the encoder and decoder in their subsequent work [15]. Apostolidis et al. [19] used a combination of self-attention and BiLSTM to capture frame dependencies. A conditional feature selector is used to guide the GAN model in focusing on essential regions of the video. Similar to Mahasseni et al. [17], Li Yuan et al. [37] proposed a model involving cyclic-GAN that is trained by maximizing the mutual information between the original video and the summary.

2.1. Differences from existing algorithms

The proposed model differentiates itself from existing models primarily by integrating graph representation with the self-attention mechanism to reconstruct the video, combined with a graph refinement process introduced. Unlike the traditional models, which use frame-level features, the proposed model uses graph-based video representations that can capture the complex relationships between the frames. Also the attention mechanism used to encode the features focuses on capturing the strength of relationships between the frames. Both combined enhance the summarization process. Furthermore, after the encoding, the graph refining process reduces the model complexity by preserving only the essential relations, leading to higher accuracy and temporal coherence. Moreover, the unsupervised approach eliminates the need for human annotations, which makes the model capable of generalizing and performing consistently over different domains when trained with a wide variety of data.

3. Method

The previous implementation of adversarial learning in video summarization [15,17] integrated the video summary in the learning pipeline as an input to the generator, which can reconstruct the video such that the discriminator cannot discriminate the reconstructed video from the original video. We separated the video summarizer into a feature encoder and a frame scorer in the proposed network, considering that we can assign the frame importance scores based on the encoded features outside the adversarial learning pipeline. Our goal is to transform the frame features via an encoder, which, when given as input to the generator, can reconstruct the video so that the discriminator cannot discriminate it from the original video. Fig. 2 illustrates the architecture of the proposed model. We employed both encoded frame features and the structure of the video as a graph to decode/reconstruct the original video representations.

To implement the proposed model, we utilized a generative model that combines VAE and GAN [36]. The VAE captures the probability distributions of latent attributes. Hence, the encoder acts as a variational inference network that encodes input features X to latent representations E by approximating a posterior distribution, as shown in Eq. (1). While the decoder/generator reconstructs input features as G by decoding the latent variables E back to the data distribution, as shown in Eq. (2). The discriminator takes X, and G as inputs and tries to distinguish them by assigning labels of 1 and 0, respectively.

$$Encoder(X) = E \sim q_{\theta}(E|X) \tag{1}$$

$$Generator(E) = G \sim p_{\phi}(X|E) \tag{2}$$

The objective of the GANs is to encourage the discriminator to best distinguish between the original and generated features while improving the generator to fit the original data distribution.

3.1. Encoder

Given the simplicity and success of the attention-based encoder in the supervised video summarization [7], we chose a multi-head self-attention network for the encoder, which transforms the frame features X extracted from a pre-trained GoogLeNet [38]. The output of the encoder is the attended frame features E and an attention matrix interpreted as a directed graph A, which represents the strength of the relationship among video frames. The matrix A represents a fully connected graph where the element at ith row and jth column denotes the relevance of frame j to frame i in the video.

3.2. Graph refining

Reconstructing frame features based on the relation among all frames represented by a fully connected graph can be expensive and redundant. So, the connections between frames that are less critical (unrelated) are eliminated. This process, called graph refining, converts a fully connected weighted directed graph A into a less dense unweighted directed graph A' by eliminating β number of connections to each frame. β is calculated using Eq. (3)

$$\beta = (1 - \alpha) * N \tag{3}$$

where, N is the total number of frames in a video, α is the retaining factor; Here, $\alpha = 0.15$

Thus, each frame in the refined graph A' is connected to only the top 15% of the related frames.

3.3. Generator

In the proposed model, the generator reconstructs frame features using a graph, in which nodes are the attended frame features and edges represent the relationship among frames within the video. Thus, we designed the generator using a 2-layered graph attention network [39], which uses encoded features E and refined graph connections A' to reconstruct the frame features G. Every node feature E goes through a learnable linear transformation W, and the graph attention coefficient is computed using Eq. (4)

$$e_{ij} = LeakyReLU\left(a^T[WE_i \parallel WE_j]\right) \tag{4}$$

Here, the attention mechanism a, which is a single-layer feed-forward network with Leaky ReLU activation applied on the concatenated embeddings of neighboring nodes i and j. The graph attention coefficient e_{ij} represents the edge weight of the link between node i and node j. The reconstructed node feature, G_i, is calculated, as shown in Eq. (6) using the neighboring node features, E_j and the computed edge weights, e_{ij} going through softmax (in Eq. (5))

$$\alpha_{ij} = softmax(e_{ij}) = \frac{e^{e_{ij}}}{\Sigma_j e^{e_{ij}}} \tag{5}$$

$$G_i = \Sigma_j \alpha_{ij} W E_j \tag{6}$$

where $j \in$ {first-order neighbors of node i}.

3.4. Discriminator

The discriminator aims to distinguish between the original and reconstructed frame features. We employed LSTM to encode the frame features into a single feature vector representing a video following a previous study by Mahasseni et al. [17]. A 2-layer linear network then uses the video feature vector to score the realness of the generated/original video.

3.5. Frame scorer

The encoder transforms the raw visual features into encoded features, which incorporate importance and capture the video storyline so that the generator can use them to reconstruct the original features. Using a linear layer, the frame scorer assigns importance scores S to the encoded features, E, based on their similarity, minimizing the sparsity loss.

3.6. Training

The training process consists of multiple steps as described in algorithm 1.

Algorithm 1 Pseudo code for training each epoch of the model

```
Retrieve the input image features
Perform training for Encoder:
    Encode input features using Encoder
    Generate features using Generator
    Compute reconstruction loss ($L_{\text{recon}}$)
    Update Encoder parameters using reconstruction loss gradient
Perform training for Frame Scorer:
    Encode input features using Encoder
    Compute frame scores using Frame Scorer
    Compute sparsity loss ($L_{\text{sparsity}}$)
    Update Frame Scorer parameters using sparsity loss gradient
Perform training for Discriminator:
    Compute discriminator loss for real features:
        Encode input features using Encoder
        Obtain discriminator output for real features
        Compute discriminator loss ($L_{\text{original}}$) between real features and original labels
        Backpropagate gradients
    Compute critic loss for generated features:
        Encode input features using Encoder
        Generate features using Generator
        Obtain discriminator output for generated features
        Compute discriminator loss ($L_{\text{summary}}$) between generated features and summary labels
        Backpropagate gradients
    Update discriminator parameters
    Clip discriminator parameters if necessary to enforce Lipschitz constraint
Perform training for Generator:
    Generate features using Generator
    Compute generator loss ($L_{\text{gen}}$) for generated features
    Update Generator parameters using generator loss gradient
```

The parameters of the Encoder, Generator, Discriminator, and Frame scorer, $\{\theta_e, \theta_g, \theta_d, \theta_s\}$, are optimized through an iterative training procedure by minimizing multiple loss functions similar to [17]:

1. Update θ_e by minimizing the L_{recon}.
2. Update θ_s by minimizing the $L_{sparsity}$.
3. Update θ_d after accumulating the gradients from both $L_{original}$ and $L_{summary}$.
4. Update θ_g by minimizing the L_{gen}.

The steps above are repeated in a sequence in every epoch. Loss functions are detailed below.

Reconstruction loss, L_{recon}: The reconstruction loss is the euclidean distance between the reconstructed video feature vector h_{recon} and the original video feature vector h_{origin}, as shown in Eq. (7)

$$L_{recon} = \|h_{recon} - h_{origin}\|_2 \tag{7}$$

The video feature vector is the final hidden state of the LSTM used in the discriminator. Reconstruction loss is used to train the encoder.

Sparsity loss, $L_{sparsity}$: The Sparsity loss is used to train the frame scorer by balancing the mean of the assigned frame importance scores [17], as shown in Eq. (8)

$$L_{sparsity} = \| \frac{1}{N} \Sigma_{i=1}^{N} s_i - \sigma \| \tag{8}$$

where, s_i denotes the predicted importance score of frame i, $\sigma = 0.15$, representing the video summary length, 15%.

Generator loss, L_{gen}: The Generator aims at reconstructing the encoded frame features, E as close as possible to the original frame features, confusing the discriminator. For training the Generator, we adopted a label-based approach suggested by [15]:

$$L_{gen} = (L_o - D(G(E)))^2 \tag{9}$$

The output of the Generator, $G(E)$, passes through the Discriminator to get the probability assigned to the reconstructed video vector, $D(G(E))$ and then compared to L_o, which is one, the probability assigned to the original video vector, as shown in Eq. (9)

Table 1
Performance comparison of variants of the proposed model using Kendall's coefficient and Spearman's coefficient.

Model	Kendall's τ	Spearman's ρ
Single-head self-attention + Graph refining	0.083	0.11
Multi-head self-attention + Graph refining	**0.094**	**0.123**
Multi-head self-attention + Positional encoding + Graph refining	0.045	0.059
Multi-head self-attention without graph refining	0.06	0.078

Table 2
Rank correlation coefficients: Kendall's τ and Spearman's ρ on TVSum dataset compared to existing methods.

Model	Kendall's τ	Spearman's ρ
Random [21]	0.00	0.00
Human [21]	0.177	0.204
dppLSTM [2]	0.042	0.055
DR-DSN [43]	0.020	0.026
RSGN [20]	0.048	0.052
SumGraph [9]	0.094	0.138
DSAVS [44]	0.080	0.087
Multi-head encoder + Graph refining	**0.094**[a]**/0.107**	**0.123**[a]**/0.138**

[a] Five full-fold results.

Discriminator loss: The discriminator aims to distinguish the original video and the reconstructed video accurately. Thus, it is trained using both the original frame features and the encoded frame features:

$$L_{summary} = (L_g - D(G(E)))^2 \tag{10}$$

$$L_{original} = (L_o - D(X))^2 \tag{11}$$

$L_{summary}$ is the distance between the probability assigned to the reconstructed video by the discriminator, $D(G(E))$, and the desired label, L_g, which here is 0, as shown in Eq. (10). $L_{original}$ is the distance between the probability assigned to the original video frame features, X, by the discriminator and the desired label, L_o, which here is 1, as shown in Eq. (11). The discriminator is trained by accumulating the gradients from both $L_{original}$ and $L_{summary}$.

4. Experiments

4.1. Datasets

The performance of the proposed model is evaluated on two existing benchmark datasets, TVSum [40], and SumMe [41]. The SumMe dataset consists of 25 videos of various events such as cooking, sports, etc. Each video is 1 to 6 min long and annotated by 15 to 18 people. Annotators created a snippet of the original video (Video summary) such that its length is between 5% to 15% of the original video length. The TVSum dataset contains 50 videos belonging to 10 categories. Each video is 1 to 5 min long and annotated by 20 people. Annotators assigned importance scores ranging from 1 to 5 every two seconds to the video frames. The OVP [42] and the YouTube [42] datasets are also used to augment the training data following previous studies [2]. The OVP dataset contains 50 videos, 1 to 4 min long. The YouTube dataset contains 39 videos, 1–10 min long. Five annotators selected the key frames in these two datasets.

4.2. Evaluation metrics

Video is split into shots based on visual similarity among consecutive frames [51,52]. A video summary – video skim – is formed by selecting a subset of video shots that are important [32]. Shot importance is calculated by averaging the predicted frame importance scores within the video shot [40,41]. Maximizing the video summary total score while keeping the video summary length fixed, the shot selection procedure can be formed as a 0/1 Knapsack problem [2,7,9,17,19]. The generated video summary is evaluated using the F-score based on Precision and Recall, as shown in Eqs. (12)–(14)

$$Precision = \frac{Duration\ of\ A \cap B}{Duration\ of\ A} \tag{12}$$

$$Recall = \frac{Duration\ of\ A \cap B}{Duration\ of\ B} \tag{13}$$

$$F - score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{14}$$

A is the summary generated based on the predicted frame scores, and B is the user summary generated based on the ground truth scores or ground truth summaries. The overlap between A and B is measured by comparing frames selected in both summaries. The F-score of each video is reported as the average F-score of the predicted summary and all user summaries for the TVSum dataset. For the SumMe dataset, the maximum of F-scores between the predicted summary and all user summaries is reported to be consistent with previous studies [2,9,14–16,19,43,45–48].

Otani et al. [21] demonstrated that the randomly generated frame importance scores could lead to video summaries comparable to the state-of-the-art summaries in terms of the summary F-score, thereby raising doubts about the shot selection procedure (Knapsack), emphasizing the significance of shot boundary detection, and suggesting to use rank correlation coefficients evaluating frame scores. The issues mentioned were corroborated by us in a different study [53].

Therefore, in this study, in addition to providing the summary F-scores, we compare the predicted frame importance scores to the human annotations using two rank correlation coefficients, Kendall's τ [54] using Eq. (15) and Spearman's ρ [55] using Eq. (16)

$$\tau = \frac{n_c - n_d}{n(n-1)/2} \tag{15}$$

where, n_c, n_d denoted as number of concordant pairs and discordant pairs respectively, n denotes total number of frames.

$$\rho = 1 - \frac{6\Sigma_{i=0}^{n} d_i^2}{n(n^2-1)} \tag{16}$$

where d_i is the distance between two ranks, n denotes total number of frames.

4.3. Implementation details

We used the same frame features extracted by Zhang et al. [2] to represent video frames, ensuring fairness while comparing the proposed model to the previous studies; videos were down-sampled to 2 fps, and frame features (dimension of 1024) were extracted from the pool5 layer of the GoogLeNet [38] model pre-trained on the ImageNet dataset [56].

The encoder consists of four heads and four parallel self-attention networks. The generator comprises of two stacked layers of graph attention networks, the discriminator consists of an LSTM and two linear layers, and the frame scorer is a linear layer.

Adam optimizer is used to update the model parameters. The learning rate for the generator and discriminator is set to 10^{-7}. The learning rate for the encoder and frame scorer is set to a higher value of 10^{-5}.

Video summaries are generated using video shot boundaries identified by Kernel Temporal Segmentation (KTS) [57]. Video summary length is set to 15% of the total video length following previous studies [2,9,15,17,19].

Table 3
F-scores of the generated summaries for TVSum and SumMe in C — Canonical, A — Augmented and T — transfer settings.

Paper	Method	TVSum			SumMe		
		C	A	T	C	A	T
SUM-GAN [17]	LSTM	51.7	59.5		39.1	43.4	
DR-DSN [43]	Reinforcement learning + Diversity based reward	57.6[a]	58.4[a]	57.8[a]	41.4[a]	42.8[a]	42.4[a]
SUM-FCN [45]	FCSN + LSTM	52.7			41.5		39.5
Unpaired [46]	Unpaired data + FCSN	55.6		55.7	47.5		41.6
CSNet [47]	Chunk-stride network + difference attention	58.8	59.0	59.2	51.3	52.1	45.1
AC-GAN [16]	Conditional GAN + self-attention	58.5[a]	58.9[a]	57.8[a]	46[a]	47.0[a]	44.5[a]
SUM-GAN-sl [14]	Stepwise label training on SUM-GAN [17]	58			47.3		
SUM-GAN-AAE [15]	Attention mechanism + SUM-GAN-sl [14]	58.3/54.7[a]			48.9[a]		
AC-SUM-GAN [19]	Reinforcement learning + Actor–Critic	60.6			50.8		
SumGraph [9]	Graph convolutional network	59.3	61.2	57.6	49.8	52.1	47.0
Interp-SUM [48]	Transformer + CNN + Piecewise linear interpolation	59.1			47.7		
RSGN [20]	BiLSTM + GCN + Reinforcement	58.0	59.1	59.7	42.3	43.6	41.2
DSAVS [44]	CNN + Self-attention + 2 layer LSTM	59.1			46.6		
VOGNet [49]	Variational Graph Autoencoder (VGAE) + Graph attention network	60.8			49.8		
UVS-DRLI [50]	Transformer + Pointwise convolution 1D network + Piecewise linear interpolation	59.86			51.66		
Proposed model	**Multi-head encoder + Graph refining**	**59**[a]/**61.4**	58.8[a]/60.6	58.8[a]/**60.5**	47.6[a]/**51.8**	**47**[a]/**52.6**	**47**[a]/**51.4**

[a] Five full-fold results.

4.4. Ablation study

Table 1 presents the results obtained on comparing the performance of different variants of our proposed model using correlation coefficients.

To examine the impact of each component in the proposed model, we run a variety of experiments. These studies show that the model with multi-head self-attention and graph refining performs the best out of all the versions, indicating the significance of both graph modeling and multi-headed attention mechanisms in predicting frame-level importance scores.

Impact of single vs. multi-head self-attention: Firstly, we use a single-head self attention as the encoder. In the direction of improving the encoder, we experiment with multi-head self attention as the encoder. The results show that the correlation coefficients improved significantly on using multi-head self attention.

Impact of positional embedding: To preserve the sequential nature of the frames in the video, we add positional encoding vectors generated using sine and cosine functions at different frequencies, as shown in Eqs. (17) and (18)

$$PE_{(pos,i)} = sin\left(\frac{pos}{10000^{\frac{i}{D}}}\right) \ if\ i\ is\ even. \tag{17}$$

$$PE_{(pos,i)} = cos\left(\frac{pos}{10000^{\frac{i}{D}}}\right) \ if\ i\ is\ odd. \tag{18}$$

where, $PE_{(pos,i)}$ — represents the position encoding at the frame position 'pos' in the sequence, and hidden dimensionality 'i' within the feature vector. D — represents the dimension of the features.

Thus, we studied the influence of the positional encoding in self attention on the performance of the model. On adding positional encoding to the multi-head self attention encoder, the performance of the model reduced by almost 50%.

Impact of graph refining: To evaluate the effectiveness of the graph refining, we evaluated the model's performance without graph refining. The results show that the performance is significantly lower when compared to the models using graph refining with single and multi-head self-attention. Thus, in spite of reducing the complexity of the model, performing graph refining improved the performance of the model as well.

4.5. Results

We evaluated the proposed model in five full folds ensuring that all videos are selected in the test split once as suggested by [16]. Evaluation splits are chosen so that each split's test data includes videos from all the video categories. Additionally, evaluation results on random splits are provided to have a fair comparison to the previous studies, which did not utilize full folds.

Table 2 compares the performance of the proposed model with the existing methods in terms of rank correlation coefficients, which measure the accuracy of the estimated frame importance scores and are not affected by the performance of the summary generation procedure. The usage of graph refining leads to higher correlation coefficients affirming that the combination of the attention weights and the refined graph enhances the encoder's performance and the state-of-art frame importance scores.

Table 3 provides a comprehensive analysis of how our proposed model's performance on TVSum and SumMe compares with that of existing state-of-the-art unsupervised models using F-score in three distinct settings — Canonical, Augment, and Transfer. The OVP and YouTube datasets were used in both Augment and Transfer settings for the training subsets while testing subsets were kept unchanged in all folds.

Even though the F-score is lower in the full five-fold training, it is more reliable, given that the models are evaluated on all videos without duplication or exceptions. However, we have provided the F-score for both random splits and the full folds since some previous studies did not use full-fold training.

The results in Table 3 demonstrate that the proposed model performs consistently across various settings, especially when the result from Canonical settings are compared with the results of the Transfer settings. This consistency shows that the proposed model captures a video's structure from different categories through effective encoding, generalizes well, and can be applied to unseen videos.

Fig. 3 demonstrates how the predicted frame importance scores track the ground truth scores closely for video 20 of the TVSum dataset. In this video, the narrator describes the special chicken sandwiches made at their restaurant and gives a glimpse into the process of making them. Figs. 4 and 5 show the video summary generated using the ground truth scores and the video summary using the predicted frame importance scores, respectively. The images represent the content of video shots; the ground truth summary includes the restaurant's name

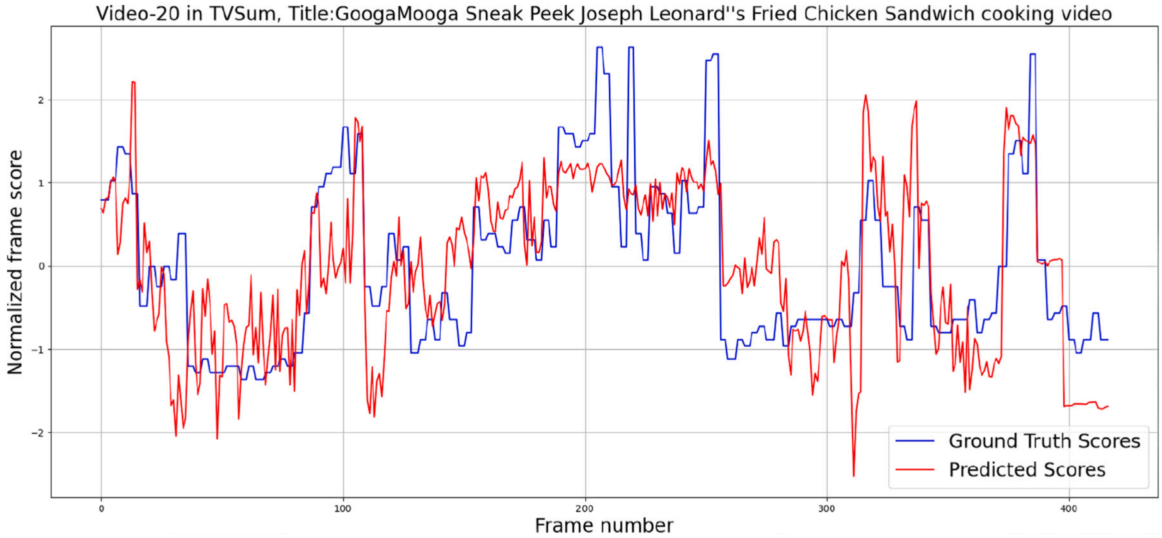

Fig. 3. TVSum:Video-20-Distribution of ground truth scores vs. predicted frame importance scores.

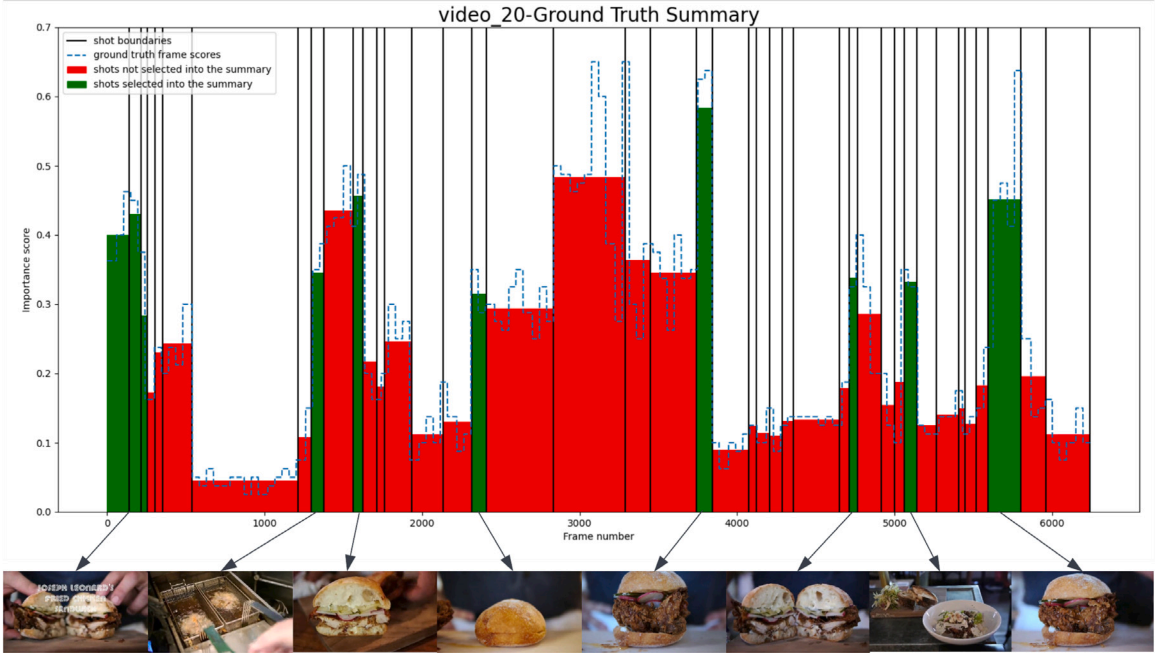

Fig. 4. Video-20: Ground truth summary.

and the staged process of making the sandwich. The summary generated using the predicted frame importance scores closely tracks the ground truth, includes matching video shots.

Furthermore, Table 4 compares the proposed model with the state-of-the art VASNet model [7] which utilized the same self-attention as the encoder but trained using supervised learning. The comparison shows that our proposed model outperforms VASNet on TVSum and SumMe datasets. Achieving results comparable with supervised models that utilized similar encoders once again verifies the proposed methodology's effectiveness.

5. Discussion and conclusion

This paper proposes a model for unsupervised video summarization that relies solely on self-attention for frame encoding. The objective is to train an encoder that can encode visual frame features for estimating frame importance scores. The encoder is trained adversarially and incrementally using a graph-based generator and a discriminator.

Table 4
Comparison of proposed model with VASNet.

Method	TVSum		SumMe	
	C	A	C	A
VASNet [7] (supervised)	61.4	62.3	49.7	51
Proposed model	**61.4**	60.6	**51.8**	**52.6**

Our result demonstrated that the unsupervised training of the attention-based encoder could generate results on par with the supervised training of a similar encoder. Furthermore, the multi-head self-attention encoder trained using the proposed graph refining method outperforms the state-of-the-art unsupervised models when evaluated using F-score, Kendall, and Spearman correlation coefficients.

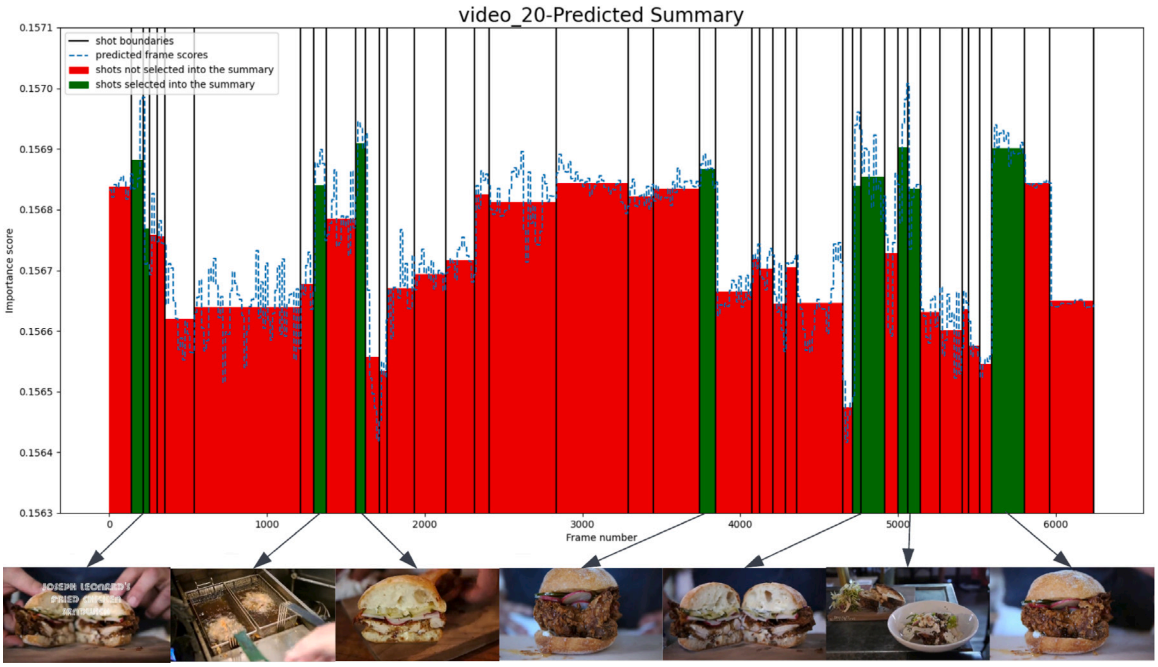

Fig. 5. Video-20: Predicted summary.

In conclusion, though our proposed model exhibits promising results, it is important to acknowledge its limitations. Our model has been trained on a relatively smaller dataset, whereas unsupervised models showcase strong performance when trained on large datasets. Hence, moving forward, we aim to fine-tune the trained model on a variety of datasets in the future and optimize the model to summarize videos on different video subjects. In our approach, we are using only visual features for video summarization. However, contextual cues, including audio and text, influence the importance of individual frames. Incorporating these additional modalities can significantly increase the quality of summarization.

Moreover, the complexity of our model results in longer training times. However, after the training, the prediction is efficient, requiring only the encoder and frame scorer. Furthermore, as explained in [21], existing summarization frameworks have drawbacks in converting scores into summaries. Although this aspect is not directly related to our research focus, it emphasizes the need for continued refinement in summarization evaluation methodologies. Additionally, while the model can summarize videos of any length, it can cause increased computing complexity. Longer video sequences also make it harder to preserve temporal coherence and capture complex semantic linkages which might lead to reduced accuracy in generated summaries. Moreover, evaluating summaries from long videos becomes more subjective and context-dependent.

CRediT authorship contribution statement

Jeshmitha Gunuganti: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Zhi-Ting Yeh:** Validation, Formal analysis, Data curation. **Jenq-Haur Wang:** Validation, Supervision, Project administration. **Mehdi Norouzi:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

We would like to thank Dr. Badri N Vellambi and Sina Eghbal for reviewing the paper and providing valuable suggestions, which allowed us to improve the quality of the paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.jvcir.2024.104200.

References

[1] DOMO, Data never sleeps 9.0, 2020, https://www.domo.com/learn/infographic/%20data-never-sleeps-9.

[2] Ke Zhang, Wei-Lun Chao, Fei Sha, Kristen Grauman, Video summarization with long short-term memory, in: European Conference on Computer Vision, Springer, 2016, pp. 766–782.

[3] Alex Kulesza, Ben Taskar, Determinantal point processes for machine learning, 2012, arXiv preprint arXiv:1207.6083.

[4] Luis Lebron Casas, Eugenia Koblents, Video summarization with LSTM and deep attention models, in: International Conference on MultiMedia Modeling, Springer, 2019, pp. 67–79.

[5] Zhong Ji, Kailin Xiong, Yanwei Pang, Xuelong Li, Video summarization with attention-based encoder–decoder networks, IEEE Trans. Circuits Syst. Video Technol. 30 (6) (2019) 1709–1717.

[6] Yen-Ting Liu, Yu-Jhe Li, Fu-En Yang, Shang-Fu Chen, Yu-Chiang Frank Wang, Learning hierarchical self-attention for video summarization, in: 2019 IEEE International Conference on Image Processing, ICIP, IEEE, 2019, pp. 3377–3381.

[7] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, Paolo Remagnino, Summarizing videos with attention, in: Asian Conference on Computer Vision, Springer, 2018, pp. 39–54.

[8] Junbo Wang, Wei Wang, Zhiyong Wang, Liang Wang, Dagan Feng, Tieniu Tan, Stacked memory network for video summarization, in: Proceedings of the 27th ACM International Conference on Multimedia, 2019, pp. 836–844.

[9] Jungin Park, Jiyoung Lee, Ig-Jae Kim, Kwanghoon Sohn, Sumgraph: Video summarization via recursive graph modeling, in: European Conference on Computer Vision, Springer, 2020, pp. 647–663.

[10] Feng Mao, Xiang Wu, Hui Xue, Rong Zhang, Hierarchical video frame sequence representation with deep convolutional graph network, in: Proceedings of the European Conference on Computer Vision (ECCV) Workshops, 2018.

[11] Yassir Saquil, Da Chen, Yuan He, Chuan Li, Yong-Liang Yang, Multiple pairwise ranking networks for personalized video summarization, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 1718–1727.

[12] Shingo Uchihashi, Jonathan Foote, Summarizing video using a shot importance measure and a frame-packing algorithm, in: 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258), Vol. 6, IEEE, 1999, pp. 3041–3044.

[13] Shruti Jadon, Mahmood Jasim, Unsupervised video summarization framework using keyframe extraction and video skimming, in: 2020 IEEE 5th International Conference on Computing Communication and Automation, ICCCA, IEEE, 2020, pp. 140–145.

[14] Evlampios Apostolidis, Alexandros I Metsai, Eleni Adamantidou, Vasileios Mezaris, Ioannis Patras, A stepwise, label-based approach for improving the adversarial training in unsupervised video summarization, in: Proceedings of the 1st International Workshop on AI for Smart TV Content Production, Access and Delivery, 2019, pp. 17–25.

[15] Evlampios Apostolidis, Eleni Adamantidou, Alexandros I Metsai, Vasileios Mezaris, Ioannis Patras, Unsupervised video summarization via attention-driven adversarial learning, in: International Conference on Multimedia Modeling, Springer, 2020, pp. 492–504.

[16] Xufeng He, Yang Hua, Tao Song, Zongpu Zhang, Zhengui Xue, Ruhui Ma, Neil Robertson, Haibing Guan, Unsupervised video summarization with attentive conditional generative adversarial networks, in: Proceedings of the 27th ACM International Conference on Multimedia, 2019, pp. 2296–2304.

[17] Behrooz Mahasseni, Michael Lam, Sinisa Todorovic, Unsupervised video summarization with adversarial lstm networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 202–211.

[18] Yujia Zhang, Xiaodan Liang, Dingwen Zhang, Min Tan, Eric P Xing, Unsupervised object-level video summarization with online motion auto-encoder, Pattern Recognit. Lett. 130 (2020) 376–385.

[19] Evlampios Apostolidis, Eleni Adamantidou, Alexandros I Metsai, Vasileios Mezaris, Ioannis Patras, Ac-sum-gan: Connecting actor-critic and generative adversarial networks for unsupervised video summarization, IEEE Trans. Circuits Syst. Video Technol. 31 (8) (2020) 3278–3292.

[20] Bin Zhao, Haopeng Li, Xiaoqiang Lu, Xuelong Li, Reconstructive sequence-graph network for video summarization, IEEE Trans. Pattern Anal. Mach. Intell. 44 (5) (2021) 2793–2801.

[21] Mayu Otani, Yuta Nakashima, Esa Rahtu, Janne Heikkila, Rethinking the evaluation of video summaries, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 7596–7604.

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, Illia Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).

[23] Yiyan Chen, Li Tao, Xueting Wang, Toshihiko Yamasaki, Weakly supervised video summarization by hierarchical reinforcement learning, in: Proceedings of the ACM Multimedia Asia, 2019, pp. 1–6.

[24] Rameswar Panda, Abir Das, Ziyan Wu, Jan Ernst, Amit K Roy-Chowdhury, Weakly supervised summarization of web videos, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3657–3666.

[25] Sanjay K. Kuanar, Rameswar Panda, Ananda S. Chowdhury, Video key frame extraction through dynamic Delaunay clustering with a structural constraint, J. Vis. Commun. Image Represent. 24 (7) (2013) 1212–1227.

[26] Anil Singh Parihar, Joyeeta Pal, Ishita Sharma, Multiview video summarization using video partitioning and clustering, J. Vis. Commun. Image Represent. 74 (2021) 102991.

[27] Karen Simonyan, Andrew Zisserman, Very deep convolutional networks for large-scale image recognition, 2014, arXiv preprint arXiv:1409.1556.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[29] Jie-Ling Lai, Yang Yi, Key frame extraction based on visual attention model, J. Vis. Commun. Image Represent. 23 (1) (2012) 114–125.

[30] Naveed Ejaz, Tayyab Bin Tariq, Sung Wook Baik, Adaptive key frame extraction for video summarization using an aggregation mechanism, J. Vis. Commun. Image Represent. 23 (7) (2012) 1031–1040.

[31] Rachida Hannane, Abdessamad Elboushaki, Karim Afdel, MSKVS: Adaptive mean shift-based keyframe extraction for video summarization and a new objective verification approach, J. Vis. Commun. Image Represent. 55 (2018) 179–200.

[32] Evlampios Apostolidis, Eleni Adamantidou, Alexandros I Metsai, Vasileios Mezaris, Ioannis Patras, Video summarization using deep neural networks: A survey, Proc. IEEE 109 (11) (2021) 1838–1863.

[33] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, Generative adversarial nets, Adv. Neural Inf. Process. Syst. 27 (2014).

[34] Diederik P. Kingma, Max Welling, Auto-encoding variational bayes, 2013, arXiv preprint arXiv:1312.6114.

[35] Yuexin Cao, Zhengchen Liu, Minchuan Chen, Jun Ma, Shaojun Wang, Jing Xiao, Nonparallel emotional speech conversion using VAE-GAN, in: INTERSPEECH, 2020, pp. 3406–3410.

[36] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, Ole Winther, Autoencoding beyond pixels using a learned similarity metric, in: International Conference on Machine Learning, PMLR, 2016, pp. 1558–1566.

[37] Li Yuan, Francis EH Tay, Ping Li, Li Zhou, Jiashi Feng, Cycle-SUM: Cycle-consistent adversarial LSTM networks for unsupervised video summarization, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 9143–9150.

[38] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.

[39] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, Graph attention networks, 2017, arXiv preprint arXiv:1710.10903.

[40] Yale Song, Jordi Vallmitjana, Amanda Stent, Alejandro Jaimes, Tvsum: Summarizing web videos using titles, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5179–5187.

[41] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, Luc Van Gool, Creating summaries from user videos, in: European Conference on Computer Vision, Springer, 2014, pp. 505–520.

[42] Sandra Eliza Fontes De Avila, Ana Paula Brandao Lopes, Antonio da Luz Jr., Arnaldo de Albuquerque Araújo, VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method, Pattern Recognit. Lett. 32 (1) (2011) 56–68.

[43] Kaiyang Zhou, Yu Qiao, Tao Xiang, Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, 2018.

[44] Sheng-Hua Zhong, Jingxu Lin, Jianglin Lu, Ahmed Fares, Tongwei Ren, Deep semantic and attentive network for unsupervised video summarization, ACM Trans. Multimed. Comput Commun. Appl. (TOMM) 18 (2) (2022) 1–21.

[45] Mrigank Rochan, Linwei Ye, Yang Wang, Video summarization using fully convolutional sequence networks, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 347–363.

[46] Mrigank Rochan, Yang Wang, Video summarization by learning from unpaired data, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 7902–7911.

[47] Yunjae Jung, Donghyeon Cho, Dahun Kim, Sanghyun Woo, In So Kweon, Discriminative feature learning for unsupervised video summarization, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 8537–8544.

[48] Ui-Nyoung Yoon, Myung-Duk Hong, Geun-Sik Jo, Interp-SUM: Unsupervised video summarization with piecewise linear interpolation, Sensors 21 (13) (2021) 4562.

[49] Jing Zhang, Guangli Wu, Shanshan Song, Video summarization generation based on graph structure reconstruction, Electronics 12 (23) (2023) 4757.

[50] Ui Nyoung Yoon, Myung Duk Hong, Geun-Sik Jo, Unsupervised video summarization based on deep reinforcement learning with interpolation, Sensors 23 (7) (2023) 3384.

[51] Costas Cotsaces, Nikos Nikolaidis, Ioannis Pitas, Video shot detection and condensed representation. a review, IEEE Signal Process. Mag. 23 (2) (2006) 28–37.

[52] Gautam Pal, Dwijen Rudrapaul, Suvojit Acharjee, Ruben Ray, Sayan Chakraborty, Nilanjan Dey, Video shot boundary detection: a review, in: Emerging ICT for Bridging the Future-Proceedings of the 49th Annual Convention of the Computer Society of India CSI Volume 2, Springer, 2015, pp. 119–127.

[53] Changwei Li, -Based Video Summarization Using Attention Networks (Ph.D. thesis), University of Cincinnati, 2022.

[54] Maurice G. Kendall, The treatment of ties in ranking problems, Biometrika 33 (3) (1945) 239–251.

[55] Daniel Zwillinger, Stephen Kokoska, CRC Standard Probability and Statistics Tables and Formulae, Crc Press, 1999.

[56] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, Li Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, Ieee, 2009, pp. 248–255.

[57] Danila Potapov, Matthijs Douze, Zaid Harchaoui, Cordelia Schmid, Category-specific video summarization, in: European Conference on Computer Vision, Springer, 2014, pp. 540–555.