ERA: Entity–Relationship Aware Video Summarization with Wasserstein GAN

Guande Wu
guandewu@nyu.edu
Jianzhe Lin
jianzhelin@nyu.edu
Claudio T. Silva
csilva@nyu.edu

New York University
New York, USA

Abstract

Video summarization aims to simplify large-scale video browsing by generating concise, short summaries that diver from but well represent the original video. Due to the scarcity of video annotations, recent progress for video summarization concentrates on unsupervised methods, among which the GAN-based methods are most prevalent. This type of methods includes a summarizer and a discriminator. The summarized video from the summarizer will be assumed as the final output, only if the video reconstructed from this summary cannot be discriminated from the original one by the discriminator. The primary problems of this GAN-based methods are two-folds. First, the summarized video in this way is a subset of original video with low redundancy and contains high priority events/entities. This summarization criterion is not enough. Second, the training of the GAN framework is not stable. This paper proposes a novel Entity–relationship Aware video summarization method (ERA) to address the above problems. To be more specific, we introduce an Adversarial Spatio-Temporal network to construct the relationship among entities, which we think should also be given high priority in the summarization. The GAN training problem is solved by introducing the Wasserstein GAN and two newly proposed video-patch/score-sum losses. In addition, the score-sum loss can also relieve the model sensitivity to the varying video lengths, which is an inherent problem for most current video analysis tasks. Our method substantially lifts the performance on the target benchmark datasets and exceeds the current state-of-the-art. We hope our straightforward yet effective approach will shed some light on the future research of unsupervised video summarization. The code is available online[1].

1 Introduction

As a primary source of recording information, video data on social networks are becoming the dominating form of information exchange. However, with the explosion of video data on different platforms (Youtube, Instagram, etc.), processing (cataloging, captioning, searching, etc.) these large number of videos manually according to their categories and subject matter would be frustrating and unintelligent. Therefore, the storage and compression had attracted

[1] https://github.com/jnzs1836/ERA-VSum

researchers' attention. How to efficiently keep and browse these videos needed a rethinking. One plausible solution was summarizing the long video into a concise synopsis with the most salient and representative content. Such a synopsis could be hyper-lapse[16], montage[20, 21], storyboards[10, 19] and skims[11, 12, 31]. In this study, we focused on the video skims.

The existing method for obtaining a video storyboard was through video summarization. Because most of the accessible videos online were with no annotations, and it would be time-consuming to obtain these annotations through human labeling, the unsupervised video summarization (UVS) model was more practical. A most famous UVS model was realized by adversarial networks [24]. The notion of the framework was that the summary generator was trained to fool a discriminator, which tried its best to distinguish the features reconstructed from the summary. Many follow-up works were proposed in recent years, but two significant problems still existed and were ignored by researchers, from the perspectives of both the loss criterion and the model training.

The criterion of defining model loss in existing models to generate the summarized video is to find frames which: a). contains the entities and events with high priority from the video. b). are with low repetition and redundancy. However, such a criterion might not be feasible to summarize practical scenarios when a trivial accident happens. For example, two kids collide with each other cannot be assumed as a key event but should be an essential accident. In this paper, we assume all these accidents to be associated with entity-relationship. Only when entities interact with each other will the accident happens. Therefore, we propose an entity-relationship aware video summarization method. The relationship of different entities is modeled by a novel Spatio-Temporal network, and changes of relationship will be easily captured and extracted in this way. In addition, different from existing methods, we introduce a novel feature extraction module to extract the scene context to help with the construction of entity-relationship. A more detailed comparison of the proposed criterion and the traditional ones can be found in Fig. 1.

The model training of the adversarial network based method is another problem. The discriminator in this type of model is not stable, while the existing methods rarely consider this. To deal with this problem, We discern two significant issues of the discriminator. Firstly, the BCE loss used in the discriminator can evoke extra training difficulty as it suffers from the vanishing gradient when there is little overlap between the generated and original samples. To solve this BCE loss problem, we instead use the earth moving distance in Wasserstein GAN [4] to formulate the loss. Another issue is the varying video length. The sparsity of feedback from the discriminator varies, which will mislead the generator. To deal with this problem, we introduce a novel patch mechanism to monitor this sparsity.

The rest of this paper is organized as follows. The related work is presented in section II, which is followed by our proposed ERA model as in section III. We present our experimental results in section IV and give a conclusion in section V.

2 Related Works

In this section, we will first generally review the existing unsupervised video summarization (UVS) models. Then we will introduce the Spatio-Temporal Graph network which is our baseline model.

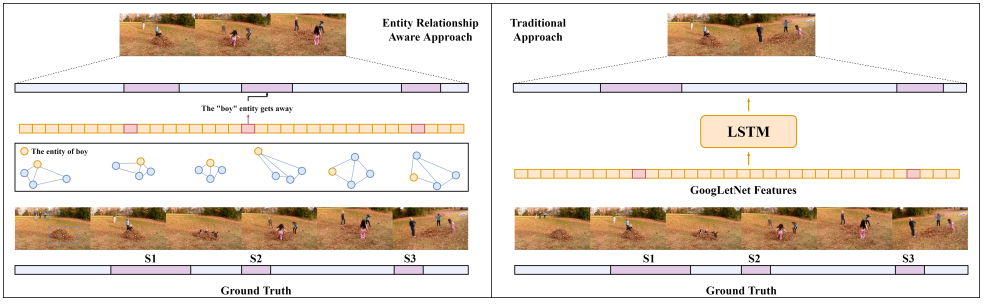


Figure 1: The clip of first 20 seconds of Kid's playing video in SumMe. The ground-truth summary has three shots in the period and we denote them as $S1, S2, S3$. When the traditional method can capture $S1$ and $S3$, it fall shorts in capturing $S2$. $S2$ describes that the body runs away from the leave stack because he got "attacked" by other kids. The features extracted from GoogLeNet may fail to capture the boy's movement. By comparison, the Spatio-Temporal Graph captures the change of the boy's relative position.

2.1 Unsupervised Video Summarization

Unsupervised video summarization methods learn a video summary with the absence of the ground-truth labels. Earlier works explored various heuristics representing the frame importance and guiding the summarization. Ngo et al. proposed to summarize the video based on video structures and video highlights[26]. However, this method relied heavily on prior knowledge, which was not realistic in real-life scenarios. Gygli [11] et al. employed a segment-level visual interesteness score and selected the optimal subset of the segments based on the scores. This work was further extended to multi-objective based optimal subset selection as in [12]. However, the feature extraction and representation parts of these works were still weak. Recent work introduced the deep learning (DL) module for the representation of input videos [32, 33, 38, 45]. A most representative branch of DL-based UVS methods was based on a generative adversarial manner, which replaced the human-defined heuristics with a learned discriminator [1, 17, 18, 24, 42]. This type of method included a feature generator (summarizer) and a discriminator. The summarized video from the generator would be assumed as the final output, only if the video reconstructed from this summary could not be discriminated from the original one by the discriminator. The earliest attempt of this type of method was made by Mahasseni et al., who introduced the LSTM based GAN for UVS for the first time [24]. Following it, CSNet added the local chunks and global stride (view) to the input features. These features enhanced the generator. Another similar work can be found in [46] and [15], in which multiple features/attentional features were introduced to improve the performance of the feature generator. Unlike the existing feature generator, we merged the object-level and scene-level features on the generator side in our work. The idea of merging different sources of features had been explored by Kanafani et al. [18] However, their approach only considered the visual features extracted from two vision models pre-trained on ImageNet. By comparison, we constructed a spatiotemporal graph of the detected objects and extracted the object-level features. Park et al. [28] also exploit graph-based approach for video summarization. However, their work focuses on the relationships between the frames without the object-level relationships. Also, we introduce a novel

discriminator, which was a rarely touched area in former works. Our work differed from the previous studies by replacing the discriminator with a critic used in Wassertein GAN and introducing a video-patch mechanism.

2.2 Spatio-Temporal Graph for the Video

A key characteristic of video data is the associated spatial and temporal semantics [22, 30]. Spatiotemporal graph, which models the characteristics of objects and their relationships by a graph structure, can be to learn this spatio-temporal correlation [5, 8, 35]. The sptiotemporal graph at the very beginning was used to learn human activity[5] and detect events [8] in the videos. An early attempt was also made on video summarization by Zhang et al. while their work relies on the hand-crafted features and does not utilize the deep learning approach[44]. Recently, spatiotemporal graph models had been extended to more general video processing applications with deep learning. For examples, Wang et al. introduced Graph Convolution Network to process the spatiotemporal graph and perform action recognition[37]; Yan et al. modeled human body joints as a spatiotemporal graph and performed pose estimation based on a spatiotemporal Graph Convolution Network[41]. Other applications included action/object localization[9, 25, 36], video captioning[27], human re-identification [23], and gaze prediction[7], etc. In our work, we introduce the deep spatiotemporal graph to UVS task for the first time, and it is used to correlate the object-level features. Our work mainly follows Wang's work [37]. However, the object-level features can be noisy and sometimes unavailable in video summarization, making the spatiotemporal graph far from enough to capture all the cues in a particular video. Thus, besides the object-level features, we also complement the spatiotemporal graph with scene features, and make the prediction based on these two features.

3 Method

In this section, we will describe our proposed ERA framework. We base our methods upon the adversarial unsupervised framework proposed by Mahasseni et al.[24] It consists a generator of VAE for reconstructing the summary-based visual features and a discriminator of LSTM. The VAE further comprises of three modules i.e. Summarizer, Encoder LSTM, Decoder LSTM. We first propose a Spatio-Temporal Network-based summarizer with a score-sum loss to explicitly capture the entity relationships and realize ERA concept. Our Encoder LSTM and Decoder LSTM are identical to the original version. Then to deal with the training difficulty, we replace the discriminator with the critic proposed in Wassertein GAN[4] and introduce a video-patch mechanism.

3.1 Spatio-Temporal Graph Convolution Network (STGCN)

To model the entities' relationships and capture their changes in the video, we incorporate a Spatio-Temporal Graph where each vertex represents an entity, and each edge models the entities' relationship. We construct the graph by 1. extracting entities from each frame via Fast R-CNN; 2. inferring the entities' relationships by a set of heuristics. Then the graph is fed into the Graph Convolution Network (GCN) to learn a graph representation. Finally, we can obtain entity-relationship aware features by performing temporal pooling[27] on the graph representation. The entity-relationship aware features can be noisy in the particular video

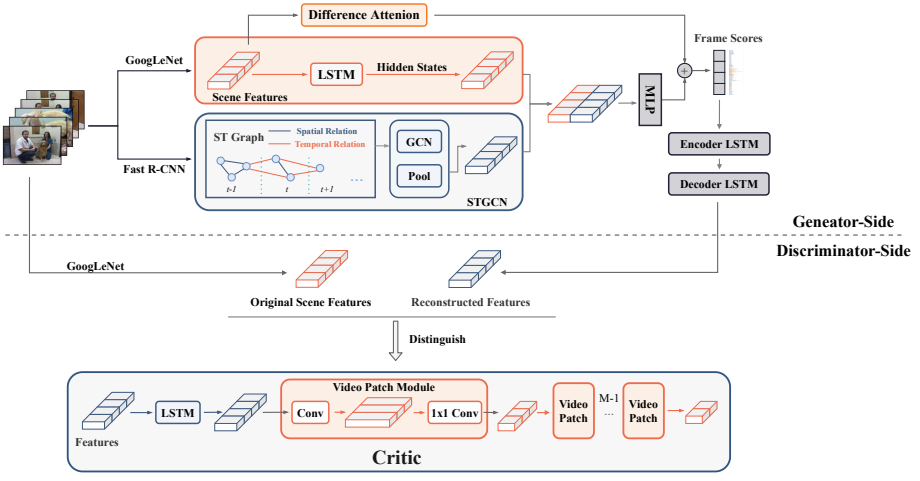


Figure 2: Our proposed methods can be categorized into generator-side and discriminator-side. The generator-side model combines the STGCN, LSTM and Difference Attention to predict the frame scores before the frame scores are exploited to reconstruct the video features by an encoder-decoder structure. The discriminator exploits the critic model proposed by W-GAN[2] and introduces a video patch module to distinguish the original and reconstructed features at a patch level.

due to the limited object detection accuracy and the sparsity of entities. To address it, we complement the features with other sources of features in the score prediction. Specifically, we combine the visual features extracted by GoogLeNet similar to [1, 24] and the difference attention proposed by [17].

3.1.1 Spatio-Temporal Graph

Given a video of T frames, we first run Fast R-CNN on each frame to extract the entities and their features. We represent the entity feature set by $\Omega = \{\mathbf{o_1^1}, \mathbf{o_2^1}, \mathbf{o_{N_1}^1}, ..., \mathbf{o_i^t}, ..., \mathbf{o_{N_t}^t}, ..., \mathbf{o_{N_T}^T}\}$ where o_i^t represents the feature vector of i-th entity in t frame. Besides, N_t denotes the number of entities in frame t. Based on the extracted entities, we define a graph as

$$G = (\Omega, E) \tag{1}$$

where $E = \{w_{i,j}\}$ is a set of edges between the different entities. We can also represent E as an adjacency matrix of the entities. We specify the edge weights by the following spatial and temporal graphs.

Spatial Graph The different entities can be related in the spatial domain. To model the intra-frame entities' relationships, we propose a spatial graph where the entities within the same frame are connected. Since the spatial relationships heavily depend on the spatial proximity, we weight the relationships by the value of Intersection Over Unions(IOU) similar to the previous works [27].

We denote the IOU between the entities o_i^t and o_j^t of the frame t as σ_{ij}^t. Then we assign

the edge weight by the normalized IOU as follows:

$$G_{i,j}^{S^t} = \frac{exp(\sigma_{ij}^t)}{\Sigma_{j=1}^{N_t} exp(\sigma_{ij}^t)} \tag{2}$$

where $G_{i,j}^{S^t}$ is the edge weight between the entities o_i^t and o_j^t in the frame t.

Temporal Graph The same entities can appear in different frames with changing positions, shapes and poses. To capture the inter-frame correlation, we construct a temporal graph where the entities in two adjacent frames are linked according to their feature similarity. Following the previous works [27, 37], we derive the edge weight by the cosine similarity of the entity features as follows:

$$G_{i,j}^{T^t} = \frac{exp(cos(\mathbf{o_i^t}, \mathbf{o_j^{t+1}}))}{\Sigma_{j=1}^{N_{t+1}} exp(cos(\mathbf{o_i^t}, \mathbf{o_j^{t+1}}))} \tag{3}$$

where $G_{i,j}^{T^t}$ is the edge weight between the entities o_i^t and o_j^{t+1}.

Spatio-Temporal Graph After obtaining the intra-frame spatial graph and inter-frame temporal graph, we combine them together to form a the adjacency matrix E.

$$E = \begin{bmatrix} G_1^s & G_{12}^t & 0 & \dots & 0 \\ 0 & G_2^s & G_{23}^t & \dots & 0 \\ 0 & 0 & G_3^s & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \dots & G_T^s \end{bmatrix} \tag{4}$$

3.1.2 Graph Convolution and Score Prediction

Following [37], we apply a graph convolution network on the Spatio-Temporal graph to learn a graph representation. Then we perform a temporal pooling[27] on the obtained representation and extract the frame features as $F_o = \{\mathbf{f_o^1}, \mathbf{f_o^2}, ...\mathbf{f_o^t}, ..., \mathbf{f_o^T}\}$.

Since the entity-relationship aware feature are noisy and unstable, we fusion it with another two sources of features. The first source of features is the scene features extracted from the pool5 layer of GoogLeNet. We denote it as $F_s = \{\mathbf{f_s^0}, \mathbf{f_s^1}, ..., \mathbf{f_s^t}..., \mathbf{f_s^T}\}$. Then, we concatenate F_s and F_o into a merged feature set F^*, which will be fed into a MLP to predict frame scores $\mathbf{s}^* = [s_1^*, s_2^*, ..., s_T^*]^T$. Another source of features is the difference attention proposed by [17]. We derive the final frame scores by take average of the frame scores $\mathbf{s^d} = [s_1^d, s_2^d, ..., s_t^d, ..., s_T^d]^T$ calculated by the difference attention and the score $\mathbf{s}^*$.

3.2 Score-Sum Loss

An training issue is that the summarizer tends to assign high scores to all the frames. The sparsity loss term partially address the issue in [24]. However, the loss term is calculated against a fixed summary rate σ (in this case, $\sigma = 15\%$), which is not always the case. Thus, we propose a score-sum loss to penalize the summarizer for assigning high scores as follows:

$$L_{sum} = \frac{\Sigma s_t}{\sqrt{T}} \tag{5}$$

where s_t refers to the score of frame t. The loss is only used in training the summarizer.

3.3 Wasserstein GAN with Video Patch Mechanism

3.3.1 Wasserstein GAN

Though used widely, GANs are often criticized as difficult to train. One of the reasons stems from the vanishing gradient issue caused by the Jensen-Shannon (JS) divergence. To address the issue, Wasserstein GAN replaces the discriminator with a critic regularized by a gradient penalty[4]. Following it, we also employ a critic to minimize the loss function below:

$$L(c) = ||E(c(x)) - E(c(x'))||_2 + \lambda(||\nabla c||_2 - 1)^2 \tag{6}$$

where x is the original features and x' is the reconstructed features. c represents the critic function and ∇c represents c's gradients. λ is a hyper-parameter for the gradient penalty.

3.3.2 Video Patch Mechanism

Another training issue comes from the varying video lengths. The sparsity of feedback from the discriminator varies according to the different video lengths, which will mislead the generator. We address the issue by introducing a video-patch mechanism following the notion of PatchGAN[12]. Given a sequence of video features $H = \{h_1, h_2, ..., h_t, ..., h_T\}$ with a feature size of K(either reconstructed or original), we employ a 1-D convolution network to reduce the sequence length and patch the frames. The convolution network consists of M building blocks, each of which can reduce the sequence length to a fifth. The building block consists of two 1-D convolution layers. The first layer reduces the sequence length with a stride of five and double the feature dimension. Then the second convolution layer performs 1×1 convolution on the sequence to reduce the feature dimension to K. Thus, the building block can output a shorter sequence with same feature size as $H^{m+1} = \{h_1^m, h_2^m, ..., h_{[\frac{T}{5}]}^m\}$ where m refers the m-th building block. After M building blocks, we can obtain an aggregated sequence of hidden features, $H^M = \{h_1^M, h_2^M, ..., h_{[\frac{T}{5^M}]}^M\}$. Each element in the sequence can have a receptive field of 5^M and thus attend to a patch of 5^M frames in the video.

4 Experiments

4.1 Experiment Settings

Implementations Following the previous works[1, 24], we downsample the videos to 2 fps. We exploit Fast R-CNN provided by Detectron2 [39] to extract the entity-level features. We employ a three-layer Graph Convolution Network with the shortcut connections between the layers [13] to process the spatiotemporal graph. We train our model with Adam optimizers with a learning rate of 1e-4 and 0.1 times after ten epochs.

Datasets and Evaluation Metric We evaluate our approach on two widely used benchmark datasets i.e. SumMe[11] and TVSum[34]. We use the standard 5-fold cross-validation for both datasets. For a fair comparison, we first employ the randomly generated data splits available from [1], which are also used by [2, 3, 18, 29] However, [18] reports that a non-trivial number of the videos are not part of any test set of the five data splits. Thus, we also generate non-overlapping splits where all the videos occur in the test splits precisely once. We assess the result by the harmonic F-measure used in [1, 2, 18, 43]. It compares the machine-generated summary with the multiple user-annotated summaries to compute a

Dataset	Method	F1		F1'		F1*	
		Avg	Max	Avg	Max	Avg	Max
SumMe	SUM – Ind_{LU} [40]	-	51.9	22.1	46.0	19.1	42.5
	CSNet [17]	-	51.3	22.7	48.1	19.0	43.2
	SUM-GAN-AAE [3]	-	48.9	22.8	47.1	19.2	41.5
	MCSF [18]	-	46.0	21.0	46.0	17.4	41.7
	DSR-RL-GRU [29]	-	50.3	22.6	50.3	18.2	40.3
	AC-SUM-GAN [2]	-	50.8	22.9	**50.8**	19.0	43.9
	ERA (Ours)	-	-	**23.2**	48.8	**19.3**	**46.3**
TVSum	SUM – Ind_{LU} [40]	61.5	-	58.7	80.7	56.6	79.3
	CSNet [17]	58.8	-	56.4	77.7	54.4	77.4
	SUM-GAN-AAE[3]	58.3	-	57.7	**81.6**	55.2	77.9
	MCSF [18]	59.1	-	59.1	81.2	58.3	78.1
	DSR-RL-GRU [29]	60.2	-	60.2	81.3	58.3	79.3
	AC-SUM-GAN [2]	60.6	-	**60.6**	81.2	57.8	80.8
	ERA (Ours)	-	-	58.0	81.5	**58.9**	**81.4**

Table 1: Comparison with the state-of-the-art unsupervised approaches: The experiments are conducted with the splits of [1](F_1') and non-overlapping splits (F_1^*). F_1 refers the reported results in the corresponding papers.

set of F-measures. By taking the average and maximum value of the F-measure set, we can derive two different F-measures (Avg and Max in Table-1) to evaluate the machine summary.

4.2 Quantitative Analysis

We compare our method with six state-of-the-art unsupervised methods i.e. SUM – Ind_{LU}[40], CSNet[17], SUM-GAN-AAE[3] and MCSF[18], DSR-RL-GRU[29] and AC-SUM-GAN[2]. The official implementations for MCSF[2] , SUM-GAN-AAE[3], DSR-RL-GRU[4] and AC-SUM-GAN[5] are available online. We exploit the unofficial implementations of SUM – Ind_{LU} provided by [18] and verify its identity to the original paper. Then, we reimplement the CSNet since the authors did not provide their source code. We compare the different approaches based on two versions of the data splits described above. For the overlapping splits, we quote the experiment results of MCSF, SUM-GAN-AAE, and SUM – Ind_{LU} from [18]. [2, 29] also employ the same splits but they only present the average F-measure for TVSum and maximum F-measure for SumMe. Thus, we cite these available F-measures directly and rerun their official implementations to get the rest F-measures. It can be observed that our approach outperforms all the competitors on the non-overlapping splits of the two datasets. Furthermore, we observe that the improvement on TVSum (0.6%) is not as significant as it on SumMe(3.1%). An explanation can be that TVSum videos contain more discontinuous scenes, hindering the temporal relationships between the objects in different scenes.

[2]https://gitlab.uni-hannover.de/hussainkanafani/unsupervised-video-summarization
[3]https://github.com/e-apostolidis/SUM-GAN-AAE
[4]https://github.com/phaphuang/DSR-RL
[5]https://github.com/e-apostolidis/AC-SUM-GAN

Exp.	STGCN	Diff	SSum	F1
0				39.36
1	✓			38.97
2		✓		36.94
3			✓	39.35
4	✓	✓		39.56
5	✓		✓	43.17
6		✓	✓	39.65
7	✓	✓	✓	**46.25**

Table 2: Ablation Study for Generator-Side Approaches

Model	SUM-GAN	STGCN
GAN	39.22	41.23
WGAN	**40.55**	42.66
WGAN + Patch Loss	39.74	**46.25**

Table 3: Ablation study Discriminator-Side Approaches

4.3 Ablation Study

Our proposed approaches can be generally divided into two categories, i.e. generator-side and discriminator-side. To analyze the effects of them, we conduct two individual ablation studies for the two sides correspondingly. We also adopt SumMe as our ablation study dataset similar to [17].

4.3.1 Ablation Study for the Generator-Side Approaches

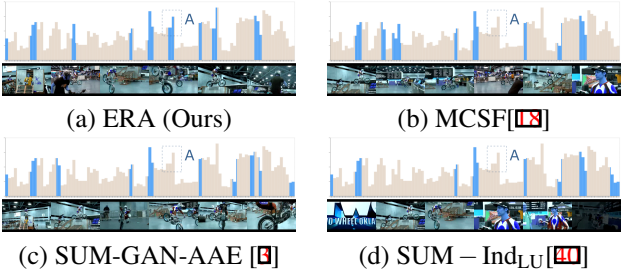
Our generator-side approaches include the Spatio-Temporal Network and the score-sum loss. However, since our model also incorporates the difference attention proposed by [17], it is necessary to analyze the usability and necessarity of the mechanism. Thus, we conduct the ablation study by adding the three approaches step by step and studying their effects. Results are provided in Table 2, from which we can obtain the following key findings.

Solely using an approach can be ineffective. Exp. 1, 2, 3 show that the models with the sole approach can not even outperform the baselines. An explanation stems from the noisy entity-aware features. Thus, it is not reliable to depend solely on entity-aware features.

Difference attention is effective but not necessary. Exp. 5 and 7, show that difference attention boosts the performance. However, the comparison between Exp. 5 and other experiments, also prove that the model can achieve promising performance without the module.

4.3.2 Ablation Study for the Discriminator-Side Approaches

Our discriminator-side improvement involves the W-GAN framework and video-patch mechanism. To verify the effectiveness of them, we train our model and the baseline $SUM-GAN$ by incorporating different discriminators i.e. vanilla discriminator, critic of W-GAN and critic with patch mechanism. The experiment result is delivered in Table 3. From the experiments on the baseline, we only observe minor improvement made by our proposed W-GAN and video-patch mechanism. However, we observe a steadier and quicker training process



(a) ERA (Ours) (b) MCSF[13]

(c) SUM-GAN-AAE [3] (d) SUM − Ind_{LU}[10]

Figure 3: Qualitative Analysis on TVSum Video-45

when using them. Based on the experiments on ERA, we find using both W-GAN and patch mechanism can help our model achieve much better results. We think the difference can be caused by the noisy entity-aware features. Since the used features are noise, ERA is more likely to generate the low-quality summary in the early training stage. The reconstructed features based on the low-quality summary can have minor support with the original features. Thus, the vanishing gradiant problem of JS Divergence haunts ERA's training progress. By contrast, W-GAN addresses the issue and can promote the performance of ERA.

4.4 Qualitative Analysis

To illustrate the selection patterns of our summarization model, we visualize the selected frames and the ground-truth frame scores of the Video-45 in TVSum, shown in Figure 3. Our method covers the peaks of the video, confirming it can capture the key-shots of the videos. For example, our method is the only one to capture the peak *A* in the video.

5 Conclusion

In this work, we study unsupervised video summarization (UVS) with adversarial learning. A novel Entity–Relationship Aware (ERA) video summarization is proposed in this paper. The method is made up of two parts, the generator and the discriminator. For the generator, we propose a novel Spatio-Temporal Graph Convolutional Network to model the entity-level features. For the discriminator, we employ Wasserstein GAN and propose a patch mechanism to deal with the varying video length. The effectiveness of the proposed ERA is verified on the TVSum and SumMe datasets.

6 Acknowledgement

Guande Wu is supported by an NYU School of Engineering Fellowship. This research is partially funded by C2SMART, a Tier 1 University Center awarded by U.S. Department of Transportation under contract 69A3351747124, and NSF awards CNS-1229185, CCF-1533564, CNS-1544753, CNS-1730396, and CNS-1828576. Silva is partially funded by DARPA. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NSF and DARPA.

References

[1] Evlampios Apostolidis, Alexandros I Metsai, Eleni Adamantidou, Vasileios Mezaris, and 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, pages 17–25, 2019.

[2] Evlampios Apostolidis, Eleni Adamantidou, Alexandros I Metsai, Vasileios Mezaris, and Ioannis Patras. Ac-sum-gan: Connecting actor-critic and generative adversarial networks for unsupervised video summarization. IEEE Transactions on Circuits and Systems for Video Technology, 2020.

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

[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In International Conference on Machine Learning (ICML), pages 214–223. PMLR, 2017.

[5] William Brendel and Sinisa Todorovic. Learning spatiotemporal graphs of human activities. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 778–785. IEEE, 2011.

[6] Chao-Yeh Chen and Kristen Grauman. Efficient activity detection with max-subgraph search. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1274–1281. IEEE, 2012.

[7] Lifeng Fan, Wenguan Wang, Siyuan Huang, Xinyu Tang, and Song-Chun Zhu. Understanding human gaze communication by spatio-temporal graph reasoning. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 5724–5733, 2019.

[8] Xu Geng, Yaguang Li, Leye Wang, Lingyu Zhang, Qiang Yang, Jieping Ye, and Yan Liu. Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pages 3656–3663, 2019.

[9] Pallabi Ghosh, Yi Yao, Larry Davis, and Ajay Divakaran. Stacked spatio-temporal graph convolutional networks for action segmentation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pages 576–585, 2020.

[10] Dan B Goldman, Brian Curless, David Salesin, and Steven M Seitz. Schematic storyboarding for video visualization and editing. Acm Transactions on Graphics (ToG), 25 (3):862–871, 2006.

[11] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos. In Proceedings of the European Conference on Computer Vision (ECCV), pages 505–520. Springer, 2014.

[12] Michael Gygli, Helmut Grabner, and Luc Van Gool. Video summarization by learning submodular mixtures of objectives. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3090–3098, 2015.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770–778, 2016.

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1125–1134, 2017.

[15] Zhong Ji, Kailin Xiong, Yanwei Pang, and Xuelong Li. Video summarization with attention-based encoder–decoder networks. IEEE Transactions on Circuits and Systems for Video Technology, 30(6):1709–1717, 2019.

[16] Neel Joshi, Wolf Kienzle, Mike Toelle, Matt Uyttendaele, and Michael F Cohen. Real-time hyperlapse creation via optimal frame selection. ACM Transactions on Graphics (TOG), 34(4):1–9, 2015.

[17] Yunjae Jung, Donghyeon Cho, Dahun Kim, Sanghyun Woo, and In So Kweon. Discriminative feature learning for unsupervised video summarization. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pages 8537–8544, 2019.

[18] Hussain Kanafani, Junaid Ahmed Ghauri, Sherzod Hakimov, and Ralph Ewerth. Unsupervised video summarization via multi-source features. In Proceedings of the 2021 International Conference on Multimedia Retrieval, page 466–470, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450384636. doi: 10.1145/3460426.3463597.

[19] Atsushi Kanehira, Luc Van Gool, Yoshitaka Ushiku, and Tatsuya Harada. Aware video summarization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 7435–7444, 2018.

[20] Hong-Wen Kang, Yasuyuki Matsushita, Xiaoou Tang, and Xue-Quan Chen. Space-time video montage. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), volume 2, pages 1331–1338. IEEE, 2006.

[21] Aditya Khosla, Raffay Hamid, Chih-Jen Lin, and Neel Sundaresan. Large-scale video summarization using web-image priors. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2698–2705, 2013.

[22] Haojie Liu, Han Shen, Lichao Huang, Ming Lu, Tong Chen, and Zhan Ma. Learned video compression via joint spatial-temporal correlation exploration. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, pages 11580–11587, 2020.

[23] Jiawei Liu, Zheng-Jun Zha, Wei Wu, Kecheng Zheng, and Qibin Sun. Spatial-temporal correlation and topology learning for person re-identification in videos. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 4370–4379, 2021.

[24] Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. Unsupervised video summarization with adversarial lstm networks. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 202–211, 2017.

[25] Effrosyni Mavroudi, Benjamín Béjar Haro, and René Vidal. Representation learning on visual-symbolic graphs for video understanding. In Proceedings of the European Conference on Computer Vision (ECCV), pages 71–90. Springer, 2020.

[26] Chong-Wah Ngo, Yu-Fei Ma, and Hong-Jiang Zhang. Automatic video summarization by graph modeling. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 104–109. IEEE, 2003.

[27] Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. Spatio-temporal graph for video captioning with knowledge distillation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 10870–10879, 2020.

[28] Jungin Park, Jiyoung Lee, Ig-Jae Kim, and Kwanghoon Sohn. Sumgraph: Video summarization via recursive graph modeling. In Proceedings of the European Conference on Computer Vision (ECCV), pages 647–663. Springer, 2020.

[29] Aniwat Phaphuangwittayakul, Yi Guo, Fangli Ying, Wentian Xu, and Zheng Zheng. Self-attention recurrent summarization network with reinforcement learning for video summarization task. In 2021 IEEE International Conference on Multimedia and Expo (ICME), pages 1–6. IEEE, 2021.

[30] N. Pissinou, I. Radev, K. Makki, and W.J. Campbell. Spatio-temporal composition of video objects: representation and querying in video database systems. IEEE Transactions on Knowledge and Data Engineering, 13(6):1033–1040, 2001. doi: 10.1109/69.971195.

[31] Danila Potapov, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. Category-specific video summarization. In Proceedings of the European Conference on Computer Vision (ECCV), pages 540–555. Springer, 2014.

[32] Mrigank Rochan and Yang Wang. Video summarization by learning from unpaired data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 7902–7911, 2019.

[33] Mrigank Rochan, Linwei Ye, and Yang Wang. Video summarization using fully convolutional sequence networks. In Proceedings of the European Conference on Computer Vision (ECCV), pages 347–363, 2018.

[34] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 5179–5187, 2015.

[35] Yicong Tian, Rahul Sukthankar, and Mubarak Shah. Spatiotemporal deformable part models for action detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2642–2649, 2013.

[36] Mingui Wang, Di Cui, Lifang Wu, Meng Jian, Yukun Chen, Dong Wang, and Xu Liu. Weakly-supervised video object localization with attentive spatio-temporal correlation. Pattern Recognition Letters, 145:232–239, 2021.

[37] Xiaolong Wang and Abhinav Gupta. Videos as Space-Time Region Graphs. In Proceedings of the European Conference on Computer Vision (ECCV), volume 11209, pages 413–431. Springer, 2018. doi: 10.1007/978-3-030-01228-1_25.

[38] Huawei Wei, Bingbing Ni, Yichao Yan, Huanyu Yu, Xiaokang Yang, and Chen Yao. Video summarization via semantic attended networks. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 32, 2018.

[39] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

[40] Gökhan Yalınız. Unsupervised video summarization with independently recurrent neural networks and multiple rewards. 2019.

[41] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, 2018.

[42] Li Yuan, Francis EH Tay, Ping Li, Li Zhou, and Jiashi Feng. Cycle-sum: cycle-consistent adversarial lstm networks for unsupervised video summarization. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pages 9143–9150, 2019.

[43] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In Proceedings of the European Conference on Computer Vision (ECCV), pages 766–782. Springer, 2016.

[44] Shu Zhang, Yingying Zhu, and Amit K Roy-Chowdhury. Context-aware surveillance video summarization. IEEE Transactions on Image Processing, 25(11):5469–5478, 2016.

[45] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. Hsa-rnn: Hierarchical structure-adaptive rnn for video summarization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 7405–7414, 2018.

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