ON THE DIFFICULTIES OF VIDEO SUMMARIZATION: STRUCTURE AND SUBJECTIVITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Video summarization, aiming at selecting a representative set of frames from a video in a limited budget, is a challenging problem in computer vision. First, to summarize a video with complex contents, understanding the storytelling structure is essential, but this fundamental step is still largely under-utilized. Also, summarization is in nature subjective, since each annotator may have different views on what the most important part is within a video. To tackle these difficulties, we propose **Hi**erarchical model for video **Sum**marization (HiSum), discovering semantic hierarchy structure of a video by event boundary detection and taking advantage of it for important frame selection. From extensive experiments on two standard benchmarks and three other new datasets specially designed to take part in subjectivity, we demonstrate that our model achieves the state-of-the-art performance.

1 INTRODUCTION

The video summarization aims to select a subset of frames or clips that represent the core contents from a long video. With the explosively growing video contents on online video sharing platforms such as YouTube or TikTok, the demand for video summarization (Elhamifar & Clara De Paolis Kaluza, 2017; Hussain et al., 2021; Panda et al., 2017) has rapidly grown for fast and scalable content discovery. For instance, video summarization is used for automatic thumbnail generation (Yuan et al., 2017), sports highlight generation (Kolekar & Sengupta, 2006; Khan & Shao, 2022; Lee et al., 2020), and video surveillance systems (Panda & Roy-Chowdhury, 2017; Zhang et al., 2016b; Muhammad et al., 2020).

However, video summarization is one of the most challenging tasks in computer vision. First difficulty is coming from the "video" part. Although we use the same term for any sequence of visual frames, videos are extremely diverse in their formation as well as contents. An example from the simplest side is a 16-frame GIF with a single action, while on the other side we have a few hours long movies or documentaries, conveying complex stories with a variety of producing techniques. Filmmakers carefully organize information for clear and effective storytelling, and this choice is closely related to the contents. Thus, to effectively understand the contents of a video, a machine learning model should be able to first capture this *structure*. Video summarization is challenging since this fundamental step, structurally understanding videos, is still largely under-utilized. Another difficulty is coming from the "summarization" part. Summarization is in nature *subjective* unlike other relatively objective tasks like action recognition, since each person may have different views on what the most "important" part is within a video. To the best of our knowledge, no previous work explicitly raises or tackles this issue in spite of its importance.

Although recent advances in deep learning made some progress (*e.g.,* PGL-SUM (Apostolidis et al., 2021b), MAVS (Feng et al., 2018)), most existing models aim at optimizing the standard benchmark metrics, without comprehensively understanding the underlying storytelling structure of videos. Instead of further over-optimizing on a few specific benchmark metrics, it is a moment to ask the fundamental question: what is a good summary of a video?

The answer to this question should be closely tied with the challenges mentioned above. A good video summarizer should be able to capture the video structure and semantics in human's manner, and based on them mimic human's frame selection process which can be done somewhat subjectively. First, for more human-like video summarization, we revisit how the model understands the video, and propose a **Hi**erarchical model for video **Sum**marization called **HiSum**, which explicitly

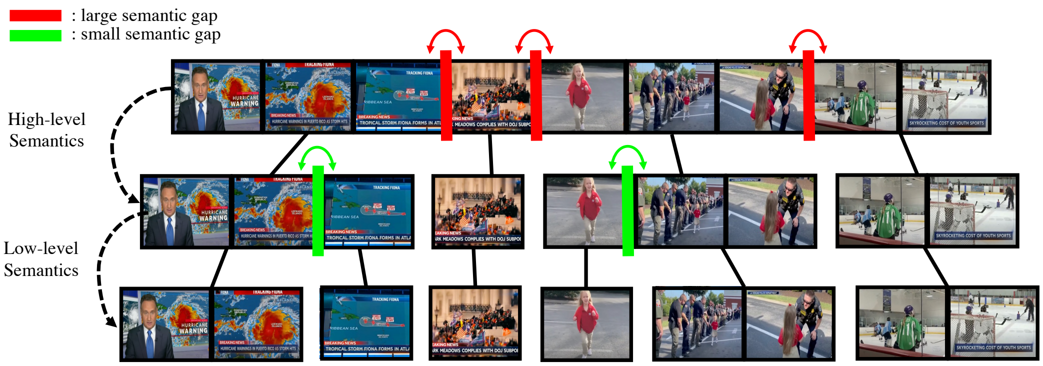

Figure 1: Video hierarchy formation from semantic boundaries. Boundaries with larger semantic gap split the video at a higher level, while smaller gaps split a sub-clip into smaller units.

captures and utilizes underlying hierarchical structure of the video based on multi-granular semantic units. To be specific, semantic structure of a video is captured as a tree, where the biggest topic change is split into multiple nodes (right below the root), and smaller topic change is split at the second level, until the video is split into individual frames, as illustrated in Fig. 1. From this analysis, our model summarizes adaptively to the nature of the target video; for instance, a video containing multiple topics that are relatively equally important is summarized to focus more on coverage, while another video with a more focused topic is summarized to reflect its importance.

Second, the subjectivity is unavoidable, but we suggest two common criteria to judge a summary. First is coverage or *diversity*. It is ideal for a summary to cover various (sub-)topics appearing in the video as much as possible, rather than focusing on a single topic and ignoring all others. For example, a good summary of one hour-long TV news is expected to include most articles uniformly, rather than choosing just one or two of them. Another is reflecting *importance*. Depending on the target video, importance of each sub-topic may not be equal. A good summary is expected to put more emphasis on a topic that people usually agree to be more important. For instance, if we summarize the news on a presidential election day, we may expect the summary allots more budget on the election than others. These two criteria somewhat conflict to each other, and one may be more important than the other depending on the target video. In this paper, we put a foundation stone on this issue by conducting experiments on three new datasets repurposed from existing benchmarks to evaluate if models can capture the labeling tendency towards diversity or importance and summarize according to the tendency.

Last but not least, during extensive experiments, we discover that recent publications have reported the best measured metric on test set, without using a set-aside validation set for model selection. This choice was probably due to the lack of training data, but this way will lead to serious overfitting on the test set and the selected model will not generalize well to unseen videos. Thus, strictly speaking, the recently reported state-of-the-art metrics are not really achieved by anyone. We fix this issue by training, validating, and evaluating most state-of-the-art models including ours on a fair playground free from overfitting. We will release our code upon acceptance.

Our main contributions are summarized as follows:

- We propose a video summarization model (HiSum) which explicitly discovers and utilizes its hierarchical **semantic structure**, mimicking human cognitive process.
- We raise the **subjectivity** issue in video summarization and provide insight how to tackle this by experiments on a novel setting.
- We **reset the standard benchmarks** for video summarization by correcting the prevalent overfitting issue on the test set. From extensive experiments under a fair setting, HiSum achieves the state-of-the-art performance on standard benchmarks.

2 Related Work

Video Summarization. Video summarization task aims to find a subset of frames that semantically represent a long video. Both supervised and unsupervised methods have been widely used to address this problem, training a model to distinguish representative *vs.* redundant frames on human-labeled summary datasets.

Early approaches in supervised learning employed RNN (Zhang et al., 2016a; Zhao et al., 2017; 2020; Zhang et al., 2018b; Yao et al., 2016) to capture temporal dependencies between frames in a long video. Since there was a long-range dependency problem when simply using RNN or LSTM, other studies tried to use additional memory (Feng et al., 2018) which is still one of the state-of-the-art models. Wang et al. (2019) proposed a memory network by stacking multiple LSTM layers and external memory layers. Some approaches additionally exploit spatial information for video summarization. Hussain et al. (2019), for instance, suggested a two-tier framework represented as multiview video summarization, using CNN and Bi-LSTM respectively. Yuan et al. (2019b) utilized a 3D-CNN and feature fusion to model spatio-temporal structure of a video.

Furthermore, attention mechanism has been widely applied recently. Fajtl et al. (2018) employed a self-attention mechanism for the first time, and Ji et al. (2019) proposed an attentive encoder-decoder architecture. To model better long-range temporal dependencies, Apostolidis et al. (2021b) combined global and local multi-head attentions. Ghauri et al. (2021) utilized attention mechanism with spatio-temporal features, extracted from raw frames and their optical flow maps. Jiang & Mu (2022) proposed cross-task learning by employing the moment localization task.

Although there are some previous works utilizing hierarchical models, the main difference is that our proposed model constructs the tree based on semantic changes, explicitly discovered by an event boundary detection (Kang et al., 2022). The majority of the existing hierarchical models have used a fixed length of frames to construct a higher-level unit (Zhang et al., 2018a; 2020), or used semantic units limitedly (*e.g.*, no more than shot-level; Liu et al. (2019); An & Zhao (2022). In particular, SHTVS (An & Zhao, 2022) used shots, which are 5-second-long on average, as the highest-level sub-clips. This design is obviously suboptimal, since no more abstraction is learned above this fine resolution. On the contrary, our model is capable of discovering arbitrary number of semantic levels, making SHTVS as a special case of our model.

Unsupervised approaches in video summarization, on the other hand, mostly utilizes reinforcement learning (RL) (Zhou et al., 2018; Lei et al., 2018) and adversarial training (Fu et al., 2019; Yuan et al., 2019a; Zhang et al., 2019; Apostolidis et al., 2020a;b; Mahasseni et al., 2017; Rochan & Wang, 2019; Yuan et al., 2019a). For instance, Zhou et al. (2018) devised a new reward system, which considers both diversity and representativeness. One example of adversarial training is Zhou et al. (2018), in which they designed a cycle-consistence loss for video summarization to precisely reconstruct the original video from the summary.

Generic Event Boundary Detection. This task aims to find events that humans perceive, such as an action change or environment change, from an untrimmed video. Shou et al. (2021) used a simple linear classifier to predict a boundary among before and after 5 frames from a candidate. Tang et al. (2022) proposed progressive attention to multi-level dense difference maps to learn more complex semantics in data. Rai et al. (2021) exploited the spatio-temporal features using a two-stream inflated 3D convolution architecture. Li et al. (2022) proposed a temporal contrastive model, exploiting temporal dependency between frames and event boundaries in the compressed domain. UBoCo (Kang et al., 2022) devised a recursive temporal self-similarity matrix (TSM) parsing algorithm to detect boundaries and achieved the state-of-the-art in both supervised and unsupervised settings. We adopt UBoCo as our event boundary detection model with slight modification.

Transformers. Transformers (Vaswani et al., 2017) were originally introduced for natural language processing (NLP) tasks. The numerous successes in the NLP domain encourage the computer vision community to adapt Transformers for vision tasks. Following Dosovitskiy et al. (2021); Chen et al. (2020), the Transformer has been used for image and video understanding (Arnab et al., 2021; Bertasius et al., 2021; Liu et al., 2022; Fan et al., 2021; Wu et al., 2021). In this work, we propose a Transformer-based hierarchical structure for video summarization. Deviated from the previous hierarchical Transformers (Zhang et al., 2020; 2018a; An & Zhao, 2022; Zhao et al., 2017), our model generates deeper semantic units via event boundary detection and exploits information from semantic hierarchy via the Transformer.

3 The Proposed Method

In this section, we describe our Hierarchical Video Summarization (HiSum) model, utilizing semantic hierarchy of the video for summarization. We first formally define the task and notations, followed by how semantic hierarchy is established and how the HiSum model use this information.

3.1 Task Definition and Notations

Given a video $\mathbf{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ with N frames, where $\mathbf{x}_t \in \mathbb{R}^{d_1}$ is a feature representing the frame at t, video summarization task aims to find a set of representative $K \ll N$ frames. Equivalently, the task is to generate a binary summary vector $\hat{\mathbf{y}} = \{\hat{y}_1, \cdots, \hat{y}_N\}$, where $\hat{y}_t \in \{0, 1\}$, $\hat{y}_t = 1$ indicates the frame at t is included in the summary, and $\hat{y}_t = 0$ otherwise. To prevent a trivial solution of selecting all frames, this task is accompanied with a condition that $\sum_t \hat{y}_t \leq K$; that is, up to K frames can be chosen. Some datasets alternatively set this condition with the ratio $\rho = K/N$, as each video has different length. Most summarization datasets provide multiple ground truth labels, where each of them was annotated by different raters. At training, most summarization models (including ours) use the average of each rater's binary annotation $\mathbf{y} \in \{0, 1\}^N$ as the ground truth importance score $\mathbf{s} = \{s_1, \cdots, s_N\}$, where $s_t \in [0, 1]$. The model solves a regression problem to predict this frame importance score $\mathbf{s}$.

3.2 Motivation: Semantic Hierarchy of a Video

A video longer than a few minutes is usually composed of multiple smaller semantic units within a structure. For instance, a news article starts with the anchor's introduction, followed by the reporter's comments and supporting recordings. Longer videos tend to have more complicated structures to effectively organize information, potentially with a multi-level hierarchy of deeper than two levels. An example of this is a movie, which usually consists of multiple scenes, and each scene is composed of multiple shots, where each unit is connected to form the overall story organizationally. Obviously, understanding and taking advantage of such a structure lets the model better extract essential information needed for a good summary.

From the perspective of a video as a sequence of frames, we can think of the semantic hierarchy in a top-down manner. When we encounter sufficiently large semantic gap between two adjacent frames, that certain point can be considered as an boundary. However, not all boundaries have same degree of semantic gap. Some boundaries have larger gap while others have milder changes. Considering the various degree of semantic gap between adjacent frames, we can build a hierarchy of the video, where boundaries with greater semantic gaps form a higher level of video hierarchy, while smaller semantic gaps form its lower level. Figure 1 illustrates how semantic boundaries generate a hierarchical structure of a video.

In order to find semantic boundaries, we apply an unsupervised generic event boundary detection model on summarization datasets. Generic event boundaries include change of action, subject, object, environment or combination of them (Shou et al., 2021). UBoCo learns underlying patterns observed near semantic boundaries using Temporal Self-similarity Matrix (TSM), which describes similarity between encoded frame features, without any boundary labels. We slightly adapt the UBoCo to better fit to our problem, detailed in Appendix A.3. We report experimental results based on UBoCo (Kang et al., 2022), but our model is agnostic of this choice and depending on the complexity of the video, other generic boundary detectors (Shou et al., 2021; Rao et al., 2020; Mun et al., 2022) may be used.

HiSum can accommodate with arbitrary number of semantic levels. In this paper, we illustrate with three levels: event-level, shot-level, and frame-level, ordered by highest to lowest. The highest event boundaries are inferred by UBoCo as described above. Then, the second highest level, shot boundaries, are obtained by kernel temporal segmentation (KTS; Potapov et al. (2014)). Shot boundaries divide videos into five-second-long shots on average, which is more fine-grained than generic event boundaries. The lowest level is individual frame level, where every single frame is considered as a semantic unit. We choose this three-level structure since it is fitted to the datasets we use, but the model itself is general enough to extend or shrink to arbitrary number of levels depending on the complexity of the target videos.

3.3 The HiSum Model

Figure 2 shows the overall architecture of our model, HiSum. It takes as input a sequence of visual features, $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$, where $\mathbf{x}_t \in \mathbb{R}^{d_1}$ for each frame at t, extracted from a pre-trained model. Each frame feature first passes through n fully-connected layers, optionally reducing its

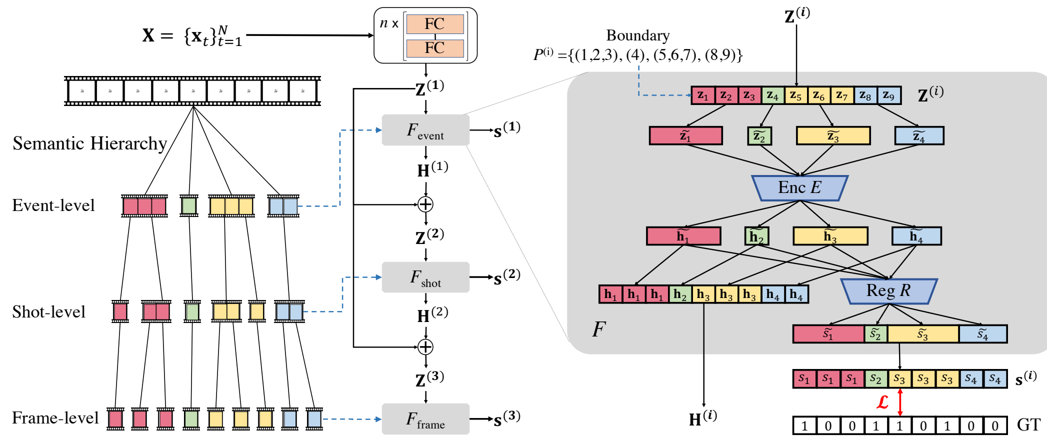

Figure 2: Overview of the HiSum model. Taking video sequence $\boldsymbol{X}$ and semantic hierarchy P as input, it estimates importance score for each semantic unit at each level, *e.g.*, $\boldsymbol{s}^{(0)}, \boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)}$.

dimensionality from $\mathbf{X} \in \mathbb{R}^{d_1}$ to $\mathbf{Z}^{(1)} \in \mathbb{R}^{d_2}$, the input to the highest (event) level module, where $d_2 \leq d_1$. This provides further flexibility to model each frame, optimized for the task.

Starting from the frame features $\mathbf{Z}^{(1)}$, HiSum repeatedly performs learning to represent each semantic unit (*e.g.*, shot, event) and scoring it as a whole if it is to be included in the summary at L different levels. For instance, Fig. 2 illustrates the case where $L = 3$, where we name each semantic level as event, shot, and frame level. At each level, from $\mathbf{Z} \in \mathbb{R}^{N \times d_2}$, a level module F learns to represent semantic units $\mathbf{H} \in \mathbb{R}^{N \times d_2}$, and outputs a score vector $\mathbf{s} \in [0, 1]^N$ indicating the relevance of each semantic unit to be chosen for the summary. For readability, we denote the modules $F^{(i)}$ for $i = 1, 2, 3$ interchangeably as F_{event}, F_{shot}, F_{frame}, respectively.

Looking inside the level module F in detail, we omit the level index i to be uncluttered. It takes two inputs: the frame representation $\mathbf{Z}^{(i)} \in \mathbb{R}^{N \times d_2}$ learned at the previous higher level and the semantic hierarchy structure $P^{(i)}$ of the video (marked in dotted blue line in Figure 2), represented as a set of semantic boundaries to the next lower level. Formally, $P^{(i)} = \{p_j^{(i)}\}$ for $j = 1, ..., M^{(i)}$, where $M^{(i)}$ is the number of boundaries at the i-th semantic level and $p_j^{(i)}$ is a set of frames that belong to the j-th semantic unit. In Fig. 2, for example, $P = \{(1, 2, 3), (4), (5, 6, 7), (8, 9)\}$ and $p_3 = (5, 6, 7)$, indicating that the semantic unit p_3 is composed of 3 frames, $\mathbf{z}_{5:7}$. Each F produces two outputs: the encoded semantic feature $\mathbf{H}^{(i)} \in \mathbb{R}^{N \times d_2}$ and estimated semantic level score $\mathbf{s}^{(i)} \in \mathbb{R}^N$:

$$\mathbf{H}^{(i)}, \mathbf{s}^{(i)} = F\left(\mathbf{Z}^{(i)}, P^{(i)}\right). \tag{1}$$

Specifically, among the input frame features $\mathbf{Z}^{(i)}$, those belonging to the same semantic unit are first averaged to form a i-th level embedding by

$$\tilde{\mathbf{z}}_j^{(i)} = \frac{1}{|p_j^{(i)}|} \sum_{t \in p_j^{(i)}} \mathbf{z}_t^{(i)} \in \mathbb{R}^{d_2}, \tag{2}$$

where $t = 1, ..., N$ and $j = 1, ..., M^{(i)}$. The tilde notation ($\tilde{\mathbf{z}}$) refers to averaged one-level higher semantic feature. $\tilde{\mathbf{z}}_j^{(i)}$ is now fed into an encoder $E : \mathbb{R}^{d_2} \to \mathbb{R}^{d_2}$ to learn more appropriate hidden representation $\tilde{\mathbf{h}}_j^{(i)} \in \mathbb{R}^{d_2}$ of the semanic unit j at level i:

$$\tilde{\mathbf{h}}_j^{(i)} = E\left(\tilde{\mathbf{z}}_j^{(i)}\right). \tag{3}$$

Note that the output $\tilde{\mathbf{h}}_j$ should be in the same dimensionality with the input to make the residual connection compatible. We use a Transformer for E, but any dimensionality-preserving layers may be used.

This semantic feature $\tilde{\mathbf{h}}_j^{(i)}$ is used to estimate the semantic level score $\mathbf{s}_j^{(i)} \in \mathbb{R}^{M^{(i)}}$ and is conveyed to the next lower level as a feature representing the current semantic level. However, at the next level there are different number ($M^{(i+1)}$) of boundaries (*e.g.*, $M^{(2)} = 6$ while $M^{(1)} = 4$ in Fig. 2). To enable operations between sequences of different length, we stretch $\tilde{\mathbf{h}}_j^{(i)}$ into $\mathbf{H}^{(i)} = \{\mathbf{h}_1^{(i)}, ..., \mathbf{h}_N^{(i)}\}$ of length N by repeating same feature inside each semantic boundary $|p_j^{(i)}|$ times. Formally, the stretched feature $\mathbf{h}_t^{(i)}$ can be written as follows:

$$\mathbf{h}_t^{(i)} = \tilde{\mathbf{h}}_j^{(i)} \quad \text{s.t} \quad t \in p_j^{(i)} \tag{4}$$

In this way, the output vectors are always length of N, and thus the score $\mathbf{s}^{(i)}$ can be compared with the ground truth regardless of the level i. Finally, $\mathbf{H}^{(i)}$ is passed to the next lower level after adding through the residual path

$$\mathbf{Z}^{(i+1)} = \mathbf{H}^{(i)} + \mathbf{Z}^{(1)} \tag{5}$$

for $i = 1, 2, ..., L$. The main purpose of this residual path is to convey fine-grained frame-level information to upper levels.

Semantic level scores $\tilde{\mathbf{s}}^{(i)} \in [0,1]^{M^{(i)}}$ are estimated by a regressor R, from $\tilde{\mathbf{h}}^{(i)} \in \mathbb{R}^{M^{(i)} \times d_2}$. R is composed of MLP layers and sigmoid activation at the end to predict importance scores $\tilde{s}_j^{(i)} \in [0, 1]$:

$$\tilde{s}_j^{(i)} = R\left(\tilde{\mathbf{h}}_j^{(i)}\right). \tag{6}$$

By following the same procedure of Eq. (4), $\tilde{s}_j^{(i)}$ is stretched to the frame level as:

$$\hat{s}_t^{(i)} = \tilde{s}_j^{(i)} \quad \text{s.t.} \quad t \in p_j^{(i)} \tag{7}$$

The overall loss $\mathcal{L}$ is the weighted sum of mean squared error between the predicted scores $\hat{\mathbf{s}}^{(i)} = \{\hat{s}_1^{(i)}, ..., \hat{s}_N^{(i)}\}$ and ground truth frame importance scores $\mathbf{s} \in [0, 1]^N$, the average of binary labels over multiple raters, at each level:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{L} \lambda_i \|\mathbf{s}^{(i)} - \mathbf{s}\|^2, \tag{8}$$

where λ_i controls relative importance among different levels.

Inference. Given a video $\mathbf{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$, the trained model outputs a score vector $\hat{\mathbf{s}} \in [0, 1]^N$. This score vector is converted to a binary vector $\hat{\mathbf{y}} \in \{0, 1\}^N$ by selecting up to K frames, where K is the summarization budget. We formulate this as a 0/1 knapsack problem and solve with a dynamic programming, following Song et al. (2015).

4 Experimental settings

Datasets. To evaluate our method, we conduct experiments on two standard public benchmarks: SumMe (Gygli et al., 2014) and TVSum (Song et al., 2015). The SumMe dataset is composed of 25 videos of 30–360 second-long, covering various contents(*e.g.*, events, sports, holidays). The TVSum dataset is composed of 50 videos which are a part of 10 categories in the TRECVid Multimedia Event Detection (MED) dataset (Smeaton et al., 2006). Both datasets provide summaries by multiple (18–20) raters per video. Following previous works (Zhang et al., 2016a; Zhou et al., 2018; Wei et al., 2018; Huang et al., 2018; Jung et al., 2020), we randomly split the data into training and test set with 4:1 ratio.

Evaluation Protocol. For evaluation metrics, following the standard benchmark, we use $F_1@\rho$ between the ground-truth $\mathbf{y} \in \{0, 1\}^N$ and the predicted summaries $\hat{\mathbf{y}} \in \{0, 1\}^N$, where $\rho = K/N$ is the summary budget, or the ratio of frames to select as a summary. For both datasets, we use the common setting of $\rho = 0.15$. The precision and recall for each video are given by $\text{Prec} = |\mathbf{y} \cap \hat{\mathbf{y}}|/|\hat{\mathbf{y}}|$, $\text{Recall} = |\mathbf{y} \cap \hat{\mathbf{y}}|/|\mathbf{y}|$, respectively, and $F_1 = \frac{2 \times \text{Prec} \times \text{Recall}}{\text{Prec} + \text{Recall}}$. Since there are multiple user-annotated summaries per video, we compute the F_1-score for each annotation. The final performance per video is computed by the average over multiple annotations for TVSum, while by the maximum over them for SumMe, following the original design of each dataset. All scores are averaged over test videos, and we use 5-fold cross-validation for all experiments.

Implementation Details. Following the common practice of previous video summarization models (Fajtl et al., 2018; Feng et al., 2018; Apostolidis et al., 2021b; Jiang & Mu, 2022; Zhang et al., 2016a), we extract visual features from the *pool5* layer ($d_1 = 1024$) of GoogLeNet (Szegedy et al., 2015), pre-trained on ImageNet. For the detailed architecture of our model, we use $n = 4$ fully-connected layers to map $\mathbf{X}$ to $\mathbf{Z}^{(1)}$. We use a standard Transformer for the encoder E in F. The regressor R is implemented with 4 fully-connected layers followed by a sigmoid activation.

We train our model over 300 epochs using AdamW optimizer (Loshchilov & Hutter, 2017) with learning rate 1×10^{-7} (SumMe) and 5×10^{-7} (TVSum), and a weight decay of 1×10^{-4}. Our implementation is based on PyTorch Lightning[1] and we conduct experiments on Nvidia Tesla-V100 with Cuda 11.3. More details are available in Appendix A.1–A.2. We plan to release our code upon acceptance.

Baseline Models. We compare HiSum with several best-performing video summarization models: VASNet (Fajtl et al., 2018)[2], MSVA (Ghauri et al., 2021)[3], PGL-SUM (Apostolidis et al., 2021b)[4], iPTNet (Jiang & Mu, 2022)[5], and SHTVS (An & Zhao, 2022). To reproduce the reported scores by baselines, we use the publicly available code released by the original authors, unless noted otherwise.

5 Results and Discussion

5.1 Comparison with Baselines

We compare our proposed model with previous methods on the benchmark datasets, following the standard evaluation protocol used in Zhang et al. (2016a); Zhou et al. (2018); Wei et al. (2018); Huang et al. (2018); Jung et al. (2020); that is, 80% of the data is for training and 20% of the data is for evaluation. The performance is compared in the first and second columns in Table 1. At a glance, we may observe that HiSum performs comparably among the state-of-the-art methods, no significant improvement over them.

However, during the meticulous examination on the released code by previous works (Fajtl et al., 2018; Apostolidis et al., 2021b; Ghauri et al., 2021; Jiang & Mu, 2022), we realize that there was no set-aside validation set, while the actual test set was being used for model selection. To be specific, all of the above have evaluated on the test set at every epoch and have reported the highest score among them as their performance. Presumably, this way of model selection is due to a lack of training data. Nonetheless, this may induce severe overfitting on the test set and would lead to poor generalization on other unseen examples. For this reason, the result in the first and second columns of Table 1 is not trustworthy, where higher score here does not necessarily indicate actually better performance on unseen data.

Invalidating the first two columns of Table 1, we reevaluate all models including ours on a fair playground, getting rid of the concern. Specifically, we split the dataset into 7:1:2 for training, validation, and test set, respectively.[6] The validation set is used for model selection; that is, the model parameters are learned on the training set, and at the end of each epoch, the model is evaluated on the set-aside validation set. We select the model with highest F_1 score on this validation set, and evaluate it on the test set. We maintain the 5-fold cross-validation to fully utilize the videos in the datasets.

The last column of Table 1 shows the revised results. First of all, the performance metrics for all models significantly drop from the reported and reproduced scores in the first and second columns of Table 1, respectively. This confirms our concern that the previous state-of-the-art performance was a result of severe overfitting on the test set. (Partly, this is also because of the reduced training set size by 10%.) Second, HiSum achieves the best performance on both SumMe and TVSum. Considering relatively lower scores in the second column of Table 1, HiSum tends to slightly more robust on overfitting than baselines.

In addition, to better understand these scores, we provide the human score from Fajtl et al. (2018) and random summary score from Apostolidis et al. (2021a), where each can be considered as the up-

[1] `https://github.com/Lightning-AI/lightning`

[2] VASNet code: `https://github.com/ok1zjf/VASNet`

[3] MSVA code: `https://github.com/TIBHannover/MSVA`

[4] PGL-SUM, code:`https://github.com/e-apostolidis/PGL-SUM`

[5] iPTNet, code: `https://code-website.wixsite.com/iptnet`

[6] SumMe has only 25 videos, so we use 17 for training (68%), 3 for validation (12%), and 5 for test (20%).

Method	Reported in paper		Our Reproduction		Corrected	
Dataset	SumMe	TVSum	SumMe	TVSum	SumMe	TVSum
Human (Fajtl et al., 2018)	64.2	63.7	–	–	–	–
Random (Apostolidis et al., 2021a)	40.2	54.4	–	–	–	–
VASNET (Fajtl et al., 2018)	49.7	61.4	46.9	60.6	42.2	58.6
MSVA (Ghauri et al., 2021)	54.5	62.8	52.8	62.3	41.2	58.7
PGL-SUM (Apostolidis et al., 2021b)	55.6	61.0	54.7	62.2	41.2	57.9
iPTNet (Jiang & Mu, 2022)	54.5	63.4	53.6	63.4	42.5	57.1
SHTVS An & Zhao (2022)	52.3	61.4	54.7	61.8	40.4	57.5
HiSum (Ours)	–	–	51.7	63.2	**42.9**	**58.8**

Table 1: F_1-scores (%) by competing models. The scores in the first column are from each paper. The second column reports our reproduction result under the same protocol (80% for training and 20% for test). This result is NOT valid, as model selection is performed on the test set. (See Sec. 5.1.) The last column is corrected result, where model selection is performed on a set-aside validation set.

Event-level	Shot-level	Frame-level	SumMe
		✓	40.6
	✓	✓	41.6
✓		✓	42.3
✓	✓	✓	42.9

(a) Ablations on semantic hierarchy levels

Method	SumMe
Uniform boundaries (Avg)	41.9
Uniform boundaries (Short)	41.6
Uniform boundaries (Long)	41.6
Semantic boundaries (Ours)	42.9

(b) Ablation on semantic boundary discovery

Table 2: F_1-scores (%) for ablation studies

per and lower bound, respectively. The random summary is generated by 0/1 knapsack on randomly assigned frame scores. Surprisingly, the best performing models including ours are much closer to the random score, while leaving a huge headroom for further improvement to the human level. Considering the small gap between the ML model performance and the random score, the performance gap between HiSum and others can be acknowledged as significant. Another interesting observation is that the human score is also far from 100% (although it is significantly higher than the current best models'), disclosing innate subjectivity of the summarization task. That is, unlike objective tasks such as action recognition, there is no single certain answer for what is a good summary.

5.2 Ablation Study

Effect of Semantic Hierarchy. Depending on the complexity of a video, it is important to construct appropriate levels of semantic hierarchy. Table 2a shows the effect of each semantic hierarchy level. Solely with the frame-level representations (without semantic hierarchy), it performs the worst. When there is an additional level such as event-level or shot-level, we observe that performance gets better. In particular, the result indicates that using the event-level is more effective than using the shot-level when we adopt the semantic hierarchy of two levels. Above all, the three-level hierarchy, which consists of event, shot, and frame level, shows the best performance.

Effect of Semantic Boundaries. We demonstrate the effectiveness of the semantic boundaries by comparing them to the boundaries uniformly selected by a fixed length. For a fair comparison, we use the three-level hierarchy for both settings and repeat experiments five times and report the averaged scores. As baselines, we try three uniform boundary lengths. First, uniform boundaries are selected for every 40 frames (event-level) and 10 frames (shot-level), which is the average length of semantic boundaries (Avg) discovered by the event boundary detector we use in HiSum. Considering the number of boundaries is not known in most cases, we also select boundaries with a shorter length as 20 frames for the event-level and 10 frames for the shot-level (Short), and 60 frames for the event-level and 15 frames for the shot-level (Long).

Table 2b reports the performance with the semantic boundary, compared to the three uniform boundary settings. We observe that the semantic boundaries generated by the event boundary detection help our model to construct a hierarchical structure of the video well. Also, uniform boundaries with an indirect hint about the average length shows slightly better performance than shorter or longer length of uniform boundaries. This conclude that the semantic boundary detection task is beneficial to construct a semantic hierarchy, which eventually leads to better performance in video summarization.

Dataset	SumMe			TVSum		
Method	DB	DU	IU	DB	DU	IU
VASNET	31.7	32.8	35.8	53.9	54.3	**47.6**
PGL-SUM	31.7	37.1	36.9	54.0	54.5	47.1
HiSum (Ours)	**32.5**	**38.0**	**38.6**	**54.8**	**54.6**	**47.6**

Table 3: F_1-scores (%) on Diversity and Importance datasets. HiSum outperforms baselines.

6 TACKLING SUBJECTIVITY

Regarding the unavoidable subjectivity in video summarization, we pose both covering diverse topics within a video and reflecting relative importance as two criteria. As a first step towards building a model that take these into account, we first propose to measure how much each model can capture the tendency of annotations, focusing more on diversity *vs.* importance.

Unfortunately, current benchmarks provide just multiple annotations, without a specific guidance to the raters regarding this diversity *vs.* importance trade-off. Thus, we design three new benchmarks assigning the summary labels more clearly inclined to either diversity or importance, by reusing videos and labels in TVSum and SumMe.

- **Diversity-Balanced (DB)** consists of videos with three sub-clips concatenated, sampled from three different videos in the same dataset, where each clip is in similar length. Each clip is considered equally important in this setting, so the original label for each frame is used as is. This setting is intended to measure if a summarizer can understand this balanced structure and give equal importance in the summary.
- **Diversity-Unbalanced (DU)** represents a case where a video is composed of multiple sub-topics with significantly different lengths, while we still want to have a summary reflecting each subtopic equally. We concatenate three clips from three different videos such that the number of positive labels is balanced, while each clip length can be significantly unbalanced.
- **Importance-Unbalanced (IU)** dataset is created by concatenating one long video and multiple short (1/10 of the long one) clips sampled from other videos. The long video becomes the important part of the concatenated video, which makes this part of the summary dominant.
- Note that Importance-Balanced is hard to create without manual labeling.

Subjectivity in summarization comes from conflicting two criteria of diversity and importance. These datasets try to break away from this situation by providing videos clearly inclined to one criterion, reducing subjectivity on summaries. Our intention in these datasets is to measure each summarization model's capability to learn each criterion underlying in the data. More details about the dataset design are described in Appendix C. We will release these datasets upon acceptance.

Table 3 compares the performance of our model and two baselines on these new datasets. HiSum outperforms on all three settings, indicating that utilizing semantic hierarchy helps the model to better understand both diversity and importance aspect of the summary.

This experimental analysis is an initial step of tackling subjectivity in summarization. That is, we show that a structure-aware model like ours can learn the philosophy of the summary (diversity or importance), if the data is clearly inclined to one. Videos and people in the real world are not polarized like this artificial setting. An ideal summarizer may need to decide how to summarize purely from the nature of video contents, and this will be an interesting future work.

7 CONCLUSION

Video summarization task is a challenging task since both "video" and "summarization" are not trivial to model. Leveraging semantic hierarchy constructed by using an event boundary detector, our HiSum model outperforms other video summarization models. Depending on the video contents and the annotator, summarization is inherently subjective, either focusing more on certain important parts or balancing diverse parts more equally. As a first step towards tackling subjectivity of summarization, we propose diversity and importance datasets by concatenating videos in existing benchmarks. More systematic design of subjectivity dataset and experimentation on it should be done and we leave this as a promising future work. Last but not least, previous video summarization models selected their best performing model based on the test set without set-aside validation set. We discover this indeed led to serious overfitting on the test set, losing generalizability on unseen videos. We fix this by providing corrected benchmark results on these baselines and our model.

REFERENCES

Yubo An and Shenghui Zhao. SHTVS: Shot-level based hierarchical transformer for video summarization. In *Proc. of the International Conference on Image and Graphics Processing (ICIGP)*, 2022.

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*, 31 (8):3278–3292, 2020a.

Evlampios Apostolidis, Eleni Adamantidou, Alexandros I Metsai, Vasileios Mezaris, and Ioannis Patras. Unsupervised video summarization via attention-driven adversarial learning. In *Proc. of the International Conference on Multimedia Modeling*, 2020b.

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

Evlampios Apostolidis, Georgios Balaouras, Vasileios Mezaris, and Ioannis Patras. Combining global and local attention with positional encoding for video summarization. In *Proc. of the IEEE International Symposium on Multimedia (ISM)*, 2021b.

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. ViViT: A video vision transformer. In *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proc. of the International Conference on Machine Learning (ICML)*, 2021.

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *Proc. of the International Conference on Machine Learning (ICML)*, 2020.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2021.

Ehsan Elhamifar and M Clara De Paolis Kaluza. Online summarization via submodular and convex optimization. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. Summarizing videos with attention. In *Proc. of the Asian Conference on Computer Vision (ACCV)*, 2018.

Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Litong Feng, Ziyin Li, Zhanghui Kuang, and Wei Zhang. Extractive video summarizer with memory augmented neural networks. In *Proc. of the ACM international conference on Multimedia (MM)*, 2018.

Tsu-Jui Fu, Shao-Heng Tai, and Hwann-Tzong Chen. Attentive and adversarial learning for video summarization. In *Proc. of the IEEE Winter Conference on applications of computer vision (WACV)*. IEEE, 2019.

Junaid Ahmed Ghauri, Sherzod Hakimov, and Ralph Ewerth. Supervised video summarization via multiple feature sets with parallel attention. In *Proc. of the IEEE International Conference on Multimedia and Expo (ICME)*, 2021.

Meng-Hao Guo, Tian-Xing Xu, Jiang-Jiang Liu, Zheng-Ning Liu, Peng-Tao Jiang, Tai-Jiang Mu, Song-Hai Zhang, Ralph R Martin, Ming-Ming Cheng, and Shi-Min Hu. Attention mechanisms in computer vision: A survey. *Computational Visual Media*, pp. 1–38, 2022.

Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos. In *Proc. of the European conference on computer vision (ECCV)*, 2014.

Siyu Huang, Xi Li, Zhongfei Zhang, Fei Wu, and Junwei Han. User-ranking video summarization with multi-stage spatio–temporal representation. *Transactions on Image Processing*, 28(6):2654–2664, 2018.

Tanveer Hussain, Khan Muhammad, Amin Ullah, Zehong Cao, Sung Wook Baik, and Victor Hugo C de Albuquerque. Cloud-assisted multiview video summarization using cnn and bidirectional LSTM. *Transactions on Industrial Informatics*, 16(1):77–86, 2019.

Tanveer Hussain, Khan Muhammad, Weiping Ding, Jaime Lloret, Sung Wook Baik, and Victor Hugo C de Albuquerque. A comprehensive survey of multi-view video summarization. *Pattern Recognition*, 109:107567, 2021.

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

Hao Jiang and Yadong Mu. Joint video summarization and moment localization by cross-task sample transfer. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Yunjae Jung, Donghyeon Cho, Sanghyun Woo, and In So Kweon. Global-and-local relative position embedding for unsupervised video summarization. In *Proc. of the European Conference on Computer Vision (ECCV)*, 2020.

Hyolim Kang, Jinwoo Kim, Taehyun Kim, and Seon Joo Kim. Uboco: Unsupervised boundary contrastive learning for generic event boundary detection. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Abdullah Aman Khan and Jie Shao. SPNet: A deep network for broadcast sports video highlight generation. *Computers and Electrical Engineering*, 99:107779, 2022.

Maheshkumar H Kolekar and Somnath Sengupta. Event-importance based customized and automatic cricket highlight generation. In *IEEE International Conference on Multimedia and Expo*, 2006.

Younghyun Lee, Hyunjo Jung, Cheoljong Yang, and Joonsoo Lee. Highlight-video generation system for baseball games. In *IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, 2020.

Jie Lei, Qiao Luan, Xinhui Song, Xiao Liu, Dapeng Tao, and Mingli Song. Action parsing-driven video summarization based on reinforcement learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(7):2126–2137, 2018.

Congcong Li, Xinyao Wang, Longyin Wen, Dexiang Hong, Tiejian Luo, and Libo Zhang. End-to-end compressed video representation learning for generic event boundary detection. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Yen-Ting Liu, Yu-Jhe Li, Fu-En Yang, Shang-Fu Chen, and Yu-Chiang Frank Wang. Learning hierarchical self-attention for video summarization. In *Proc. of the IEEE international conference on image processing (ICIP)*, 2019.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv:1711.05101*, 2017.

Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. Unsupervised video summarization with adversarial LSTM networks. In *Proc. of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Khan Muhammad, Tanveer Hussain, and Sung Wook Baik. Efficient cnn based summarization of surveillance videos for resource-constrained devices. *Pattern Recognition Letters*, 130:370–375, 2020.

Jonghwan Mun, Minchul Shin, Gunsoo Han, Sangho Lee, Seongsu Ha, Joonseok Lee, and Eun-Sol Kim. Boundary-aware self-supervised learning for video scene segmentation. In *Proc. of the Asian conference on computer vision (ACCV)*, 2022.

Rameswar Panda and Amit K Roy-Chowdhury. Multi-view surveillance video summarization via joint embedding and sparse optimization. *Transactions on Multimedia*, 19(9):2010–2021, 2017.

Rameswar Panda, Niluthpol Chowdhury Mithun, and Amit K Roy-Chowdhury. Diversity-aware multi-video summarization. *IEEE Transactions on Image Processing*, 26(10):4712–4724, 2017.

Danila Potapov, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. Category-specific video summarization. In *Proc. of the European Conference on Computer Vision (ECCV)*, 2014.

Ayush K Rai, Tarun Krishna, Julia Dietlmeier, Kevin McGuinness, Alan F Smeaton, and Noel E O'Connor. Discerning generic event boundaries in long-form wild videos. *arXiv:2106.10090*, 2021.

Anyi Rao, Linning Xu, Yu Xiong, Guodong Xu, Qingqiu Huang, Bolei Zhou, and Dahua Lin. A local-to-global approach to multi-modal movie scene segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10146–10155, 2020.

Mrigank Rochan and Yang Wang. Video summarization by learning from unpaired data. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Mike Zheng Shou, Stan Weixian Lei, Weiyao Wang, Deepti Ghadiyaram, and Matt Feiszli. Generic event boundary detection: A benchmark for event segmentation. In *Proc. of the IEEE/CVF International Conference on Computer Vision*, 2021.

Alan F Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and TRECVid. In *Proc. of the ACM International Workshop on Multimedia Information Retrieval*, 2006.

Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. TVSum: Summarizing web videos using titles. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2015.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2015.

Jiaqi Tang, Zhaoyang Liu, Chen Qian, Wayne Wu, and Limin Wang. Progressive attention on multi-level dense difference maps for generic event boundary detection. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems (NIPS)*, 30, 2017.

Junbo Wang, Wei Wang, Zhiyong Wang, Liang Wang, Dagan Feng, and Tieniu Tan. Stacked memory network for video summarization. In *Proc. of the ACM International Conference on Multimedia (MM)*, 2019.

Huawei Wei, Bingbing Ni, Yichao Yan, Huanyu Yu, Xiaokang Yang, and Chen Yao. Video summarization via semantic attended networks. In *Proc. of the AAAI conference on Artificial Intelligence*, 2018.

Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22–31, 2021.

Ting Yao, Tao Mei, and Yong Rui. Highlight detection with pairwise deep ranking for first-person video summarization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 982–990, 2016.

Li Yuan, Francis EH Tay, Ping Li, Li Zhou, and Jiashi Feng. Cycle-sum: Cycle-consistent adversarial lstm networks for unsupervised video summarization. In *Proc. of the AAAI Conference on Artificial Intelligence*, 2019a.

Yitian Yuan, Tao Mei, Peng Cui, and Wenwu Zhu. Video summarization by learning deep side semantic embedding. *Transactions on Circuits and Systems for Video Technology*, 29(1):226–237, 2017.

Yuan Yuan, Haopeng Li, and Qi Wang. Spatiotemporal modeling for video summarization using convolutional recurrent neural network. *IEEE Access*, 7:64676–64685, 2019b.

Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *Proc. of the european conference on computer vision (ECCV)*, 2018a.

Bowen Zhang, Hexiang Hu, Joonseok Lee, Ming Zhao, Sheide Chammas, Vihan Jain, Eugene Ie, and Fei Sha. A hierarchical multi-modal encoder for moment localization in video corpus. *arXiv:2011.09046*, 2020.

Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In *Proc. of the European conference on computer vision (ECCV)*, 2016a.

Ke Zhang, Kristen Grauman, and Fei Sha. Retrospective encoders for video summarization. In *Proc. of the European conference on computer vision (ECCV)*, 2018b.

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

Yujia Zhang, Michael Kampffmeyer, Xiaoguang Zhao, and Min Tan. DTR-GAN: Dilated temporal relational adversarial network for video summarization. In *Proc. of the ACM Turing Celebration Conference - China*, 2019.

Bin Zhao, Xuelong Li, and Xiaoqiang Lu. Hierarchical recurrent neural network for video summarization. In *Proc. of the ACM international conference on Multimedia (MM)*, 2017.

Bin Zhao, Xuelong Li, and Xiaoqiang Lu. TTH-RNN: Tensor-train hierarchical recurrent neural network for video summarization. *IEEE Transactions on Industrial Electronics*, 68(4):3629–3637, 2020.

Kaiyang Zhou, Yu Qiao, and Tao Xiang. Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. In *Proc. of the AAAI Conference on Artificial Intelligence*, 2018.