VSS-Net: Visual Semantic Self-Mining Network for Video Summarization

Yunzuo Zhang, *Member, IEEE*, Yameng Liu, Weili Kang, and Ran Tao, *Senior Member, IEEE*

***Abstract*— Video summarization, with the target to detect valuable segments given untrimmed videos, is a meaningful yet understudied topic. Previous methods primarily consider inter-frame and inter-shot temporal dependencies, which might be insufficient to pinpoint important content due to limited valuable information that can be learned. To address this limitation, we elaborate on a Visual Semantic Self-mining Network (VSS-Net), a novel summarization framework motivated by the widespread success of cross-modality learning tasks. VSS-Net initially adopts a two-stream structure consisting of a Context Representation Graph (CRG) and a Video Semantics Encoder (VSE). They are jointly exploited to establish the groundwork for further boosting the capability of content awareness. Specifically, CRG is constructed using an edge-set strategy tailored to the hierarchical structure of videos, enriching visual features with local and non-local temporal cues from temporal order and visual relationship perspectives. Meanwhile, by learning visual similarity across features, VSE adaptively acquires an instructive video-level semantic representation of the input video from coarse to fine. Subsequently, the two streams converge in a Context-Semantics Interaction Layer (CSIL) to achieve sophisticated information exchange across frame-level temporal cues and video-level semantic representation, guaranteeing informative representations and boosting the sensitivity to important segments. Eventually, importance scores are predicted utilizing a prediction head, followed by key shot selection. We evaluate the proposed framework and demonstrate its effectiveness and superiority against state-of-the-art methods on the widely used benchmarks.**

***Index Terms*— Video summarization, self-mining, temporal cues, semantic representation, information exchange.**

I. Introduction

OVER the years, explosive growth in the number of videos captured by various devices raises the common need for efficient video browsing [1]. Normally, it requires a complete viewing of untrimmed videos to retrieve meaningful and attractive content, contributing to a time-consuming and resource-shortage process once videos with much redundant information need to be analyzed and stored. Video summarization [2] is a prominent and fundamental task in the field of video understanding, aiming at picking key segments from a video sequence. It has drawn increasing research interest in the face of the looming challenge posed by large-scale videos [3].

Numerous contributions [4], [5], [6], [7], [8], [9] have been proposed over the years, achieving a promising improvement in summarizing videos. The implementation of video summarization has undergone a gradual shift from traditional methods [10], [11], [12], [13] relying on hand-crafted features to modern methods [14], [15], [16] benefiting from the powerful representation capability of deep learning [17]. Specifically, these methods can be categorized into three types: unsupervised methods, weakly supervised methods, and supervised methods. Unsupervised methods [18], [19], [20], [21] are centered around formulating criteria to measure representativeness and diversity. Weakly supervised methods [22], [23] usually summarize videos by utilizing potentially valuable information related to videos. Supervised methods [24], [25], [26], in contrast to them, leverage manual annotations to aid in learning effective representations and often achieve better performance.

Temporal cues are of great essence to representation learning, especially for videos [27], [28]. The majority of methods [29], [30] typically extract frame-level visual features and then adopt Recurrent Neural Networks (RNNs) [31] to gradually model temporal information frame by frame. However, such RNNs-based models encounter problems with gradient computing as well as the decay of temporal cues with duration increasing. To address these issues, Rochan et al. [32] proposed a fully convolutional sequence model to process all frames simultaneously. Although it can alleviate the inherent problems of RNNs, the sophisticated relationships between long-distance frames are difficult to grasp. Popularly applied in video summarization, the attention mechanism [33] is leveraged to enhance modeling capability. These attention-based works [16], [26], [34], [35] are capable of modeling the pairwise temporal relation among frames or shots in one step and dynamically performing feature aggregation, showing considerable improvement. Nevertheless, they only consider inter-frame and inter-shot temporal dependencies, which might be still insufficient to precisely extract important segments owing to limited valuable features that can be learned from them.

In certain cross-modality learning tasks (*e.g.*, video grounding [36] and moment retrieval [37]), in addition to visual

Manuscript received 5 June 2023; revised 5 August 2023; accepted 2 September 2023. Date of publication 5 September 2023; date of current version 5 April 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 61702347 and Grant 62027801, in part by the Natural Science Foundation of Hebei Province under Grant F2022210007 and Grant F2017210161, in part by the Science and Technology Project of Hebei Education Department under Grant ZD2022100 and Grant QN2017132, and in part by the Central Guidance on Local Science and Technology Development Fund under Grant 226Z0501G. This article was recommended by Associate Editor H. Yue. *(Corresponding author: Yunzuo Zhang.)*

Yunzuo Zhang, Yameng Liu, and Weili Kang are with the School of Information Science and Technology, Shijiazhuang Tiedao University, Shijiazhuang 050043, China (e-mail: zhangyunzuo888@sina.com; liuym4647@sina.com; wayleek@sina.com).

Ran Tao is with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China (e-mail: rantao@bit.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2023.3312325.

Digital Object Identifier 10.1109/TCSVT.2023.3312325

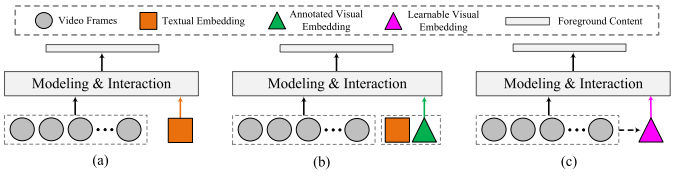

Fig. 1. Illustration of cross-modality learning tasks and our framework. We all aim to extract foreground content within videos. (a) is the general form of the video grounding task and moment retrieval task, which adopts textual embedding to assist in localization. (b) is the procedure of a query-driven summarization method, which support annotated textual or visual embedding to select relevant content. We borrow the idea of cross-modality learning and employ scheme (c) to address generic video summarization using a learnable visual embedding to accurately pinpoint important content.

content understanding through temporal cues, foreground parts are located with the assistance of textual embedding that reflects crucial semantics, as depicted in Fig. 1(a). In order to boost diverse forms for such query-driven tasks, Wu et al. [38] tried to query important segments exploiting snippet-level visual embedding annotated, as depicted in Fig. 1(b), not just limited to textual embedding. These tasks, including video summarization, are essentially aimed at extracting foreground content within videos. Motivated by this finding, this paper goes deeper into devising a Visual Semantic Self-mining Network (VSS-Net), which exploits a learnable visual embedding that can reflect crucial visual content of videos to assist in the network **query** and **locate** important segments by itself, as shown in Fig. 1(c). Compared with previous summarization methods, the proposed method additionally considers a higher-level representation of the input video. By doing this, our method can learn more valuable information from the video sequence and effectively model powerful contextualized representations, unconstrained by the potential risks associated with solely relying on temporal modeling.

Fig. 2 illustrates the overview of our framework, which comprises three dominant components including a Context Representation Graph (CRG), a Video Semantics Encoder (VSE), and a Context-Semantics Interaction Layer (CSIL). More specifically, CRG aims to learn frame-level temporal features for storyline understanding. It models each frame as a node and regards the visual relationship across nodes as edges. Considering the hierarchical structure of videos, we develop three sets of edges to enrich visual features with local and non-local temporal cues from multiple perspectives including temporal order, visual similarity, and dissimilarity. Simultaneously, inspired by the success of the attention mechanism, VSE is devised to aggregate visual features into a concise but high-level visual semantic representation, using a modeling approach from coarse to fine. Finally, CSIL acts as an exchanger to facilitate the mutual exchange of information between aggregated frame-level temporal cues and video-level semantic representation. This is capable of boosting deep content understanding while alleviating the computational burden.

Additionally, the proposed framework is easy to implement in a parallel manner, as it does not contain any recursive structure. We conduct extensive experiments on both the SumMe [39] and TVSum [10] datasets, and VSS-Net demonstrates its effectiveness and superiority by achieving performance on par or better than state-of-the-art methods.

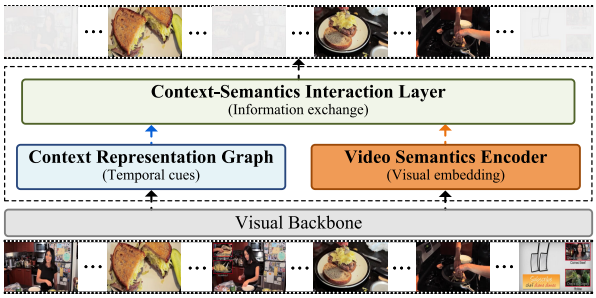

Fig. 2. Overview of the proposed Visual Semantic Self-mining Network (VSS-Net). Our method initially adopts a two-stream structure to separately model frame-level temporal cues and video-level visual semantic embedding for mining more valuable information about the input video. By jointly modeling these features, VSS-Net can learn powerful contextualized representations.

To encapsulate, the technical contributions of this work can be summarized as follows:

- To the best of our knowledge, VSS-Net is the first attempt to adopt the idea of cross-modality learning to address video summarization. It allows for learning more powerful contextualized representations by jointly modeling frame-level and video-level visual representations.
- We design a Context Representation Graph (CRG) to enrich visual features with local and non-local temporal cues. It utilizes the hierarchical structure of videos comprehensively, enabling effective message passing among frames.
- We devise a Video Semantics Encoder (VSE) that dynamically encodes the video into a video-level representation from coarse to fine. This representation semantically represents crucial content in videos, providing more informative guidance features for our model.
- We introduce a Context-Semantics Interaction Layer (CSIL), which can achieve a deep yet economical information exchange across frame-level temporal cues and video-level semantic representation, guaranteeing effective features for precise summarization generation.

The rest of this paper is organized as follows. In Sec. II, we provide a review of the related work. Sec. III details our framework. Experimental results are presented in Sec. IV. Finally, we conclude this paper in Sec. V.

II. Related Work

This section mainly provides a brief review of related topics, including video summarization, graph representation, and two-stream architecture. Each of them is discussed in detail in the following.

A. Video Summarization

The performance of summarizing videos has been significantly improved over the past few years along with numerous related contributions proposed [40], [41], [42]. Existing summarization methods can be cast into three categories: unsupervised methods, weakly supervised methods, and supervised methods. Early unsupervised methods primarily rely on cluster algorithms [11], [43] and dictionary learning [44], [45]. However, these works include weak representation capabilities and

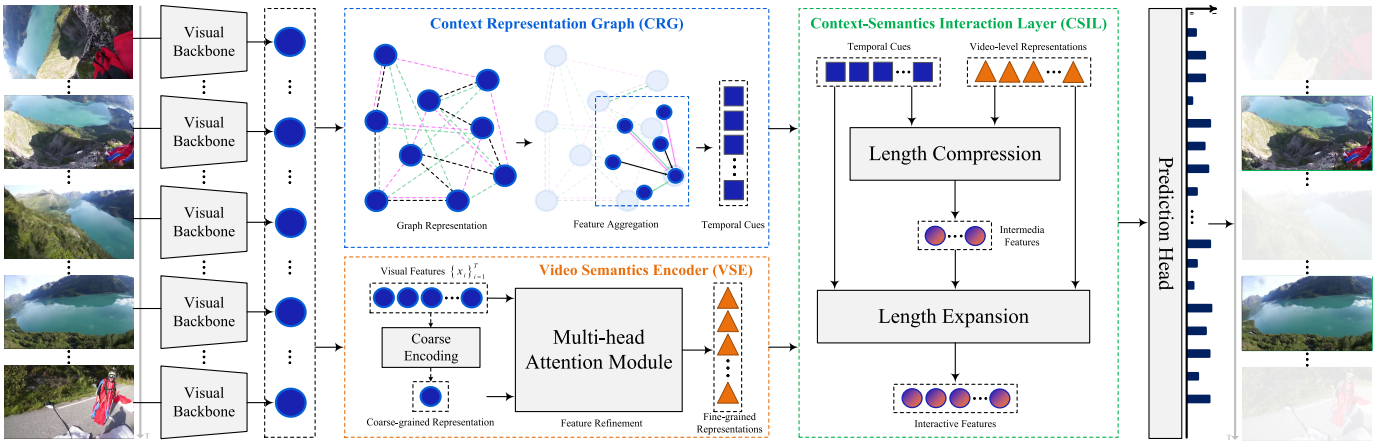

Fig. 3. Overview of the proposed Visual Semantic Self-mining Network (VSS-Net). For an untrimmed video, a visual backbone first extracts the visual feature of each frame. These features are then fed into the Context Representation Graph (CRG) and Video Semantics Encoder (VSE) separately. The former aims to enrich the spatial features with local and non-local contextual information. Lines of different colors represent different types of edges. The VSE module is developed to dynamically learn the video-level semantic representation of the input video. After that, we employ a Context-Semantics Interaction Layer (CSIL) to achieve information exchange. Finally, a prediction head computes frame-level importance scores, which are the basis of generating summarization.

temporal information in the frame sequence. Nowadays, lots of methods explore Generative Adversarial Networks [46], [47] and reinforcement learning [30], [48]. These methods typically generate summaries by formulating fixed properties (*e.g.,* representativeness and diversity), which is mainly reflected in the loss or reward function. Weakly supervised methods utilize auxiliary information. For example, Cai et al. [22] proposed a generative modeling method to learn the latent semantic video representations to bridge the benchmark data and web data. Compared with the unsupervised and weakly supervised methods, supervised methods can generate summaries with higher performance [49] since manual annotations are used to assist the model in learning effective representations. For example, Zhang et al. [50] proposed an LSTM-based network incorporating a determinantal point process (DPP) to boost the diversity in the selected frames. Wei et al. [29] developed a semantic attended summarization network to extract the most semantically relevant video segments by providing textual description supervision. Ji et al. [26] selected summaries by combining self-attention and BiLSTM to mimic the way of selecting the key shots of humans. Zhu et al. [35] modeled both short-range and long-range temporal cues to exploit multiscale information. These methods focus on aggregating temporal cues, especially on improving the capability of modeling long-range context. Different from them, based on exploring temporal features, this paper also deeply considers the higher-level information of the input video itself. By jointly modeling frame-level and video-level visual representations of the input video, our method can learn more valuable information about videos and successfully show effectiveness and superiority against the state-of-the-art methods.

B. Graph Representation

Exploiting a graph to represent a video or a sentence is a natural and effective idea [51]. Benefiting from its powerful representation, graph representation learning has been used to address various tasks [52], [53]. These methods typically regard frames or words as nodes and edges are used to reflect the relationship across them. For example, Wang et al. [54] proposed an edge convolution operation to deal with point cloud data. Li et al. [53] formulated the domain adaptive object detection as a graph-matching problem. Yang et al. [55] constructed a novel graph convolutional network to address the temporal action location task, aiming at enhancing the discriminability of feature representations. In the field of video summarization, Zhao et al. [56] employed a graph to aggregate contextual information across shots, where the dissimilarity is leveraged as aggregation weight to model the entire frame sequence. To alleviate the chaotic message passing, Zhang et al. [57] developed a dissimilarity-guided attention graph by adaptively evaluating the semantic dissimilarity between shots within a video. Motivated by these works, we extend the existing contribution and proposed a novel graph block to achieve a more comprehensive video understanding by aggregating local and non-local temporal cues from multiple perspectives consisting of temporal order, visual similarity, and dissimilarity.

C. Two-Stream Architecture

The two-stream architecture is a profound deep-learning method, specifically designed for action recognition within videos [58]. It features two parallel data streams dedicated to handling spatial and temporal information, respectively. By independently processing spatial and temporal information, the two-stream networks effectively capture crucial action-related features and achieve superior performance in the action recognition task [59]. Recently, numerous contributions still adopt this architecture to model temporal cues and semantic representation. For instance, Zhang et al. [60] proposed a new two-stream architecture to address the video saliency prediction task. Han et al. [61] devised a two-stream network to process RGB information and noise features. Zhao et al. [62] built a novel framework to jointly model appearance and gait information. In this paper, we also adopt

a type of two-stream structure. However, different from the aforementioned methods, these streams in our method are used to model frame-level contextual information for preliminary video understanding and video-level semantic representation for providing more valuable information about the input video, respectively.

III. Methodology

A. Problem Formulation

Given an untrimmed video, the video summarization task aims to select a set of shots that consist of consecutive frames. Suppose $\boldsymbol{F} = \{\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_T\}$ represent a video sequence, where $\boldsymbol{f}_i$ denotes i-th frame and T is the total length. The inputs to VSS-Net are visual features $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T\} \in \mathbb{R}^{T\times d}$ extracted by employing a pre-trained visual backbone, where d is the feature dimension. Our method predicts each frame and generates importance scores $\boldsymbol{S} = \{s_1, s_2, \ldots, s_T\}$, where s_i reveals the importance degree of i-th frame. Additionally, each video has corresponding manual annotation scores $\boldsymbol{O} = \{o_1, o_2, \ldots, o_T\}$, which is used to guide network training by means of a supervised learning paradigm. We specify the duration of selected shots does not exceed $\rho\%$ of the total duration.

B. Overview of VSS-Net Architecture

This section provides an overview of the proposed Visual Semantic Self-mining Network (VSS-Net). Its architecture is illustrated in Fig. 3. First of all, a visual backbone is adopted to extract visual features for all frames. Then, we handle these features through two streams separately. (i) CRG enriches visual features with both local and non-local contextual information for preliminary visual content understanding. (ii) VSE adaptively learns a high-level video-level semantic representation of the input video itself according to these visual features. The two streams converge in CSIL for information exchange across aggregated contextual information and encoded semantic representation. Subsequently, we apply a prediction head to score each frame based on the interactive features. These scores are then post-processed through the knapsack algorithm to generate a summary for the input video. The detail of each module is described in the following.

C. Context Representation Graph

Modeling a video into a graph structure is an effective way to explore the hierarchical information of videos and understand the mutual interaction across frames, which is intractable for attention-based [34], [63] and convolution-based [32], [64] methods. Hence, we adopt the graph modeling approach and propose CRG to aggregate both local and non-local contextual information comprehensively. As depicted in Fig. 4, a video is composed of multiple shots that are visually dissimilar from each other, and each shot consists of visually similar frames. Compared with previous graph-based methods [56], [57] that only consider inter-shot dissimilarity, CRG learns more hierarchical characteristics of videos including the temporal order and inner-shot similarity, allowing for exploring sophisticated feature dependencies across frames and shots.

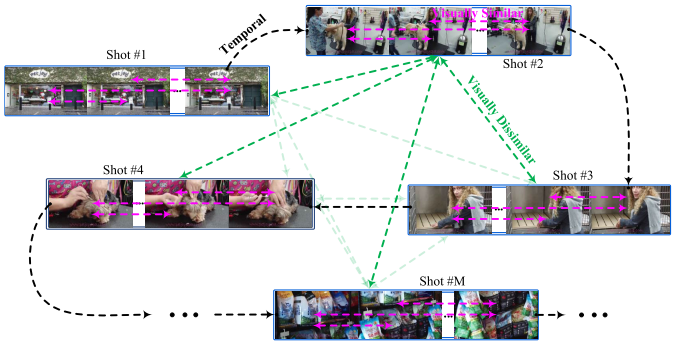

Fig. 4. Illustration of the Context Representation Graph (CRG). Suppose a video can be segmented into M shots. This module includes three types of edges: (i) Temporal Edges (black dashed line) that are used to model the temporal order, (ii) Similar Edges (pink dashed line), and (iii) Dissimilar Edges (green dashed line) that are devised to model feature dependencies across visually similar and dissimilar frames.

Technologically, suppose the tuple $\mathcal{G} = \{\mathcal{V}, \xi\}$ represent a video graph, where $\mathcal{V}$ is the set of unordered frames and ξ is the set of edges indicating the interactive relationship across frames. In this module, three types of edges are devised, including Temporal Edges ξ_t, Similar Edges ξ_s, and Dissimilar Edges ξ_d. Temporal Edges connect the current frame to the next one, targeting to imitate video playback procedure and model temporal order. This set of edges can be represented as:

$$\boldsymbol{\xi}_t = \{< \boldsymbol{x}_i, \boldsymbol{x}_{i+1} > | i = 1, 2, \ldots, T-1\} \tag{1}$$

According to $\boldsymbol{\xi}_t$, we can obtain the temporal adjacency matrix $\boldsymbol{A}_t$. Similar Edges and Dissimilar Edges are constructed by utilizing the k-nearest neighbors algorithm. We first model the pairwise visual similarity relationship $\boldsymbol{\omega}_i^s \in \mathbb{R}^T$ and dissimilarity relationship $\boldsymbol{\omega}_i^d \in \mathbb{R}^T$ across i-th frame and other frames, leveraging the L_2-norm function $\gamma(\boldsymbol{g}_1, \boldsymbol{g}_2) = \|\boldsymbol{g}_1 - \boldsymbol{g}_2\|_2$:

$$\boldsymbol{\omega}_i^s = \left\{\gamma(\boldsymbol{x}_i, \boldsymbol{x}_j) | j = 1, 2, \ldots, T\right\} \tag{2}$$

$$\boldsymbol{\omega}_i^d = \left\{-\gamma(\boldsymbol{x}_i, \boldsymbol{x}_j) | j = 1, 2, \ldots, T\right\} \tag{3}$$

Subsequently, adjacency matrices $\boldsymbol{A}_s$ and $\boldsymbol{A}_d$ for Similar and Dissimilar Edges are established by performing the top-rank operator on these visual relationships of all nodes, formed by $\boldsymbol{I}_{\{s,d\}}(i) = \text{top-rank}(\boldsymbol{\omega}_i^{\{s,d\}}, L)$. Here $\boldsymbol{I}_{\{s,d\}}(i)$ indicates the index set regarding i-th frame and L is a hyperparameter used to control the number of neighbor nodes. Mathematically, we formulate the two sets of edges as:

$$\boldsymbol{\xi}_s = \{< \boldsymbol{x}_i, \boldsymbol{x}_{I_s(i;j)} > \ | i = 1, 2, \ldots, T; \quad j = 1, 2, \ldots, L\} \tag{4}$$

$$\boldsymbol{\xi}_d = \{< \boldsymbol{x}_i, \boldsymbol{x}_{I_d(i;j)} > \ | i = 1, 2, \ldots, T; \quad j = 1, 2, \ldots, L\} \tag{5}$$

where $I_s(i; j)$ and $I_d(i; j)$ are the indexes of j-th nearest and farthest neighbors of i-th frame, respectively. With these edges, our architecture employs edge convolution $h(\cdot)$ [54] to compute the aggregated features $\boldsymbol{C} \in \mathbb{R}^{T\times d}$. Formally, this can be written as follows:

$$\boldsymbol{C} = \sigma\left(h\left(\boldsymbol{X}; \boldsymbol{A}_t\right) + h\left(\boldsymbol{X}; \boldsymbol{A}_s\right) + h\left(\boldsymbol{X}; \boldsymbol{A}_d\right) + \boldsymbol{X}\right) \tag{6}$$

where $\sigma(\cdot)$ denotes the activation function, which is specified as ReLU in this paper. Leveraging this module, our method allows for performing rich message passing through considering the visual relationships within and between shots, offering a preliminary but comprehensive understanding of the video storyline.

D. Video Semantics Encoder

Previous methods predominantly focus on designing networks with the more powerful temporal modeling capability, especially in modeling long-distance sequential data [65]. They might still suffer from limited video understanding owing to the single consideration of inter-frame or inter-shot contextual information. This section introduces the VSE module, which incorporates frame-level visual features into a highly integrated video-level visual embedding that potentially reflects the topic or main content of the input video itself, thus laying the foundation for providing more valuable information for temporal cues. Additionally, with the assistance of this learnable visual embedding, our method holds the promise of more accurately perceiving the crucial segments within videos.

Practically, we have explored two types of video-level embedding, including pooling-based embedding [66] and RNN-based embedding [67]. The pooling-based embedding computes the central representation among visual features by performing a global temporal average pooling (GTAP) along the temporal dimension. However, it might cause much noise to be introduced or many valuable characteristics to be discarded. The RNN-based embedding exploits the hidden state of the final unit as the video-level representation. Its encoding capability may be limited due to its deficiency in dealing with long-range sequential data.

As a result, we propose VSE as a better alternative, which begins with the coarse-grained representation, *i.e.,* pooling-based embedding $\boldsymbol{E} \in \mathbb{R}^{1\times d}$. This module can effectively perform recalibration with the assistance of the attention mechanism without being affected by distance. Its pipeline is illustrated in Fig. 5. Specifically, after performing GTAP over visual features, VSE projects $\boldsymbol{X}$ into $\boldsymbol{K}_i = \boldsymbol{X}\boldsymbol{W}_i^K$ and $\boldsymbol{V}_i = \boldsymbol{X}\boldsymbol{W}_i^V$, respectively, and project $\boldsymbol{E}$ into $\boldsymbol{Q}_i = \boldsymbol{E}\boldsymbol{W}_i^Q$, where i denote the head index. Here $\boldsymbol{X}$ serves as the key (k) and value (v) simultaneously and $\boldsymbol{E}$ acts as the query (q). $\boldsymbol{W}_i^Q$, $\boldsymbol{W}_i^K$, and $\boldsymbol{W}_i^V$ are learnable parameters. Subsequently, we compute the attended features $\boldsymbol{H}_i$ of i-th head using a scaled dot-product attention layer, which can be formulated as follows:

$$\boldsymbol{H}_i(\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i) = \text{softmax}\left(\frac{\boldsymbol{Q}_i\boldsymbol{K}_i^T}{\sqrt{d}}\right)\boldsymbol{V}_i \tag{7}$$

After getting the features of all attention heads, we concatenate them together and feed them into a linear projection layer to form the final refined feature $\boldsymbol{Z} \in \mathbb{R}^{1\times d}$:

$$\boldsymbol{Z} = \text{concat}(\boldsymbol{H}_1, \boldsymbol{H}_2, \ldots, \boldsymbol{H}_n)\boldsymbol{W}^O \tag{8}$$

where n is the number of attention heads. $\boldsymbol{W}^O$ is the parameter that needs to be learned. To simplify notation, we define this whole process as follows:

$$\boldsymbol{Z} = \Gamma(\boldsymbol{E}, \boldsymbol{X}, \boldsymbol{X}) \tag{9}$$

where $\Gamma(\cdot)$ denotes the multi-head attention (MHA) function regarding query, key, and value, respectively. Intuitively, the proposed VSE module can better preserve valuable visual characteristics and suppresses unuseful information within the input video. Also, the attention mechanism effectively ensures parallel computation through matrix operations.

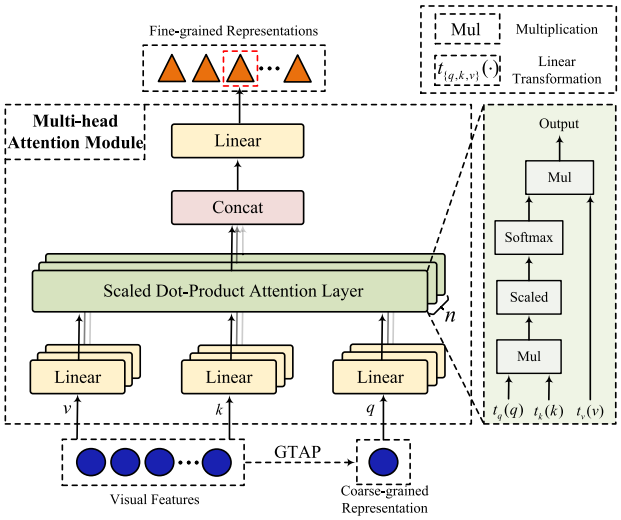

Fig. 5. Pipeline of the Video Semantics Encoder (VSE). Coarse-grained video-level embedding is obtained by performing global temporal average pooling along the temporal dimension. Then, the visual features are set as the key (k) and value (v), respectively, and the coarse-grained video-level representation is set as the query (q) to adaptively compute fine-grained video-level embedding (red dashed line) using the multi-head attention module. The output representation sequence is obtained through replication.

E. Context-Semantics Interaction Layer

After obtaining frame-level temporal cues and video-level semantic representation using the previous modules, we need to embed the latter into the former to form unified feature representations for subsequent importance score prediction. By doing so, the temporal cues are enhanced with the high-level semantic representation that indicates key visual content. This allows for learning more powerful contextualized features for further boosting the awareness capability of the network.

Typically, simple methods are to directly apply summation [68] or concatenation [69]. However, they can be limited in their ability to facilitate adequate message exchange between features, which may result in relatively poor representations. Despite its effective modeling capability, the cross attention (CA) [70] often encounters costly computation. This motivates us to introduce CSIL for summarization, which can perform deep information interaction while achieving economical calculation costs. The pipeline is depicted in Fig. 6. Unlike direct joint modeling features of CA, CSIL adopts bottleneck tokens [71] to indirectly transmit information between features.

Technologically, a set of bottleneck tokens $\boldsymbol{B} = \{\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_m\} \in \mathbb{R}^{m\times d}$ is introduced, where m is the number of tokens. It is responsible for forcing the module to condense

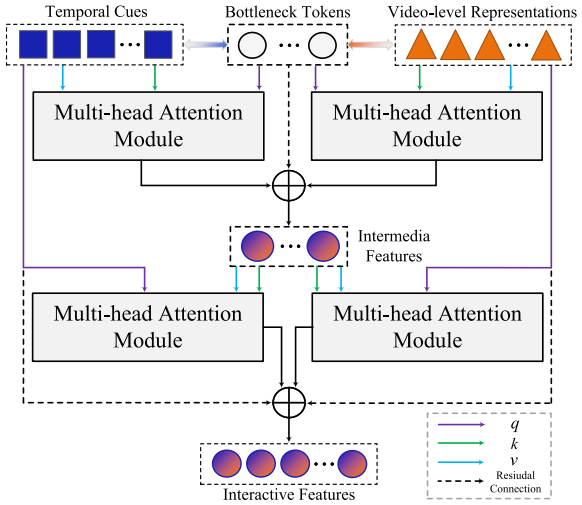

Fig. 6. Pipeline of the Context-Semantics Interaction Layer (CSIL). Firstly, two multi-head attention modules are exploited to share valuable information between bottleneck tokens and temporal cues (video-level representations), obtaining intermedia features with a much smaller length of the sequence than the input video. Then, these intermediate features and temporal cues (video-level representations) participate in the second feature interaction, and the length of the feature sequence is restored to its original size. This module includes several residual connections to guarantee powerful representations.

and deliver valuable information. Then, we set bottleneck tokens $\boldsymbol{B}$ as the query and set the temporal cues $\boldsymbol{C}$ (or semantic representation sequence $\boldsymbol{Z}_s$) as the key and value, respectively. Here, $\boldsymbol{Z}_s \in \mathbb{R}^{T\times d}$ is obtained by $\boldsymbol{Z}$ after T replications along the temporal dimension. Using the MHA module, we let bottleneck tokens guide both temporal cues and semantic representation based on their relationships, thus generating token-guided contextual information and token-guided video-level semantics. The intermediate features $\boldsymbol{B}_c \in \mathbb{R}^{m\times d}$ that have undergone sequence length compression are formed according to a residual connection, which can be written in practice as follows:

$$\boldsymbol{B}_c = \Gamma(\boldsymbol{B}, \boldsymbol{C}, \boldsymbol{C}) + \Gamma(\boldsymbol{B}, \boldsymbol{Z}_s, \boldsymbol{Z}_s) + \boldsymbol{B} \tag{10}$$

where $\boldsymbol{B}_c$ reflects the first interaction between different features with a concise volume. Next, we further impose a second interaction stage, which expands the length of the feature sequence from b to T for the frame-level importance score prediction. Similar but different from the process above, we exchange the attention terms, namely adjusting their query, value, and key. The second interactive feature sequence $\boldsymbol{Z}_e$ is still calculated using residual connections. Formally, the computation is defined as follows:

$$\boldsymbol{B}_e = \Gamma(\boldsymbol{C}, \boldsymbol{B}_c, \boldsymbol{B}_c) + \boldsymbol{C} + \Gamma(\boldsymbol{Z}_s, \boldsymbol{B}_c, \boldsymbol{B}_c) + \boldsymbol{Z}_s \tag{11}$$

We do not use extra feed-forward networks to save parameters. After finishing the second stage, a further feature interaction is achieved, bringing a fine-grained exchange of information. Additionally, with this elaborate architecture, the complexity of computation is reduced to linearity, which significantly alleviates the computational burden caused by the increasing duration of videos.

F. Summarization Generation

The interactive representations, the output of the previous module, are then fed into a prediction head formed by Linear → ReLU → Dropout → LayerNorm → Linear → Sigmoid to regress importance scores. Then, we compute the shot-level importance scores $\boldsymbol{P} = \{p_1, p_2, \ldots, p_M\}$ by taking an average within each subsequence detected by Kernal Temporal Segmentation [72]. Here M denotes the number of detected shots in an input video. A knapsack problem is subsequently created to select the most valuable key shots under the defined duration constraint. To obtain the optimal solution, we adopt the dynamic programming algorithm, which can be mathematically represented by:

$$\max \sum_{i=1}^{M} a_i p_i \qquad s.t. \sum_{i=1}^{M} a_i l_i \leq T \times \rho\% \tag{12}$$

where $a_i \in \{0, 1\}$ stands for whether i-th shot is selected into the final summary. p_i and l_i are the corresponding importance score and duration, respectively. The summaries are generated by concatenating those shots with $a_i = 1$ in chronological order. For training, we supervise this process using the mean squared error (MSE) loss $\mathcal{L}$, which can be represented as follows:

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{T}\sum_{i=1}^{T}(s_i - o_i)^2 \tag{13}$$

where $\boldsymbol{\theta}$ is the parameters in our model.

IV. Experiments

We evaluate the performance of our proposed summarization framework on commonly used benchmarks. This section first introduces datasets, evaluation settings, metrics, and implementation details. Then, we compare our method with other state-of-the-art methods and conduct ablation studies to validate the effectiveness of each module. Next, we provide some visualization results. Finally, the current challenges are discussed.

A. Datasets

We carry out empirical evaluations on two benchmark datasets, including the SumMe [39] dataset and the TVSum [10] dataset. The SumMe dataset is a popular dataset for video summarization, which is a collection of 25 user-generated videos. It covers multiple types of scenes (*e.g.*, cooking), with each video having frame-level importance scores annotated by at least 15 users. The TVSum dataset consists of 50 videos collected from YouTube, covering 10 categories. Each video is annotated by 20 users. Also, we use two additional datasets including OVP [11] and YouTube [11] to augment our training data. The OVP dataset includes 50 videos in total and the YouTube dataset contains 39 videos, both of which are annotated by 5 users. Refer to Table I for details about the datasets.

B. Evaluation Setting

Following previous methods [30], [35], [56], we test our method under three different evaluation settings including Canonical (C), Augmented (A), and Transfer (T). Concretely, the canonical setting is a standard and most widely used

TABLE I
CHARACTERISTICS STATISTICS OF THE SUMME [39], TVSUM [10], OVP [11], AND YOUTUBE [11] DATASETS

Dataset	Number of Videos	Number of Users	Annotation Type	Training-Testing Pair	Duration of Videos		
					Min	Max	Avg
SumMe [39]	25	15-18	importance score	20 for training - 5 for testing	32s	5min 24s	2min 26s
TVSum [10]	50	20	importance score	40 for training - 10 for testing	1min 23s	10min 47s	3min 55s
OVP [11]	50	5	key frame	50 for training	9s	9min 32s	1min 38s
YouTube [11]	39	5	key frame	39 for training	46s	3min 29s	3min 16s

supervised learning setting, where 80% of the videos in used datasets are chosen for training and the rest for testing. Instead of randomly selecting videos (*i.e.*, 5 Random), we employ standard 5-fold cross-validation (*i.e.*, 5 FCV) to ensure that all videos are involved in training and testing. The augmented setting still employs standard 5-fold cross-validation to evaluate our model. All videos in OVP and YouTube are added to the training set for augmenting training data. The transfer setting is that the other three datasets except for the target dataset (*e.g.,* SumMe or TVSum) are regarded as training data, which is exploited to evaluate the transferability of the proposed method.

C. Evaluation Metric

1) F-score Evaluation: Following most state-of-the-art methods, we first report the F-score performance on the SumMe and TVSum datasets. Let $\boldsymbol{F}_p$ and $\boldsymbol{F}_h$ represent the predicted summaries and manual annotations, respectively. The F-score can be computed by:

$$\text{Precision} = \frac{\text{overlapped duration of } \boldsymbol{F}_p \text{ and } \boldsymbol{F}_h}{\text{duration of } \boldsymbol{F}_p} \tag{14}$$

$$\text{Recall} = \frac{\text{overlapped duration of } \boldsymbol{F}_p \text{ and } \boldsymbol{F}_h}{\text{duration of } \boldsymbol{F}_h} \tag{15}$$

$$\text{F-score} = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \times 100\% \tag{16}$$

The larger the F-score, the greater the overlap between the generated summarization and the manual summarization. Also, following the previous evaluation protocol [50], we report the average of F-scores on the TVSum dataset and the largest F-score on the SumMe dataset as the experimental results.

2) Rank-based Evaluation: Moreover, Otani et al. [73] found that F-score is not sensitive enough to the differences in importance score computation. Even randomly generated summaries can also achieve comparable results. Hence, we additionally report correlation coefficients including Kendall's τ and Spearman's ρ for further evaluating our method. They are used to measure the agreement between predicted scores and annotated scores by humans and make the superiority of the proposed method more reliable.

D. Implementation Details

The pre-trained GoogLeNet [80] is selected as our visual backbone and we adopt it to extract 1024-dimensional visual features of all frames from the pool-5 layer. Following previous contributions [30], [79], we downsample the frame sequence to 2 frames per second and specify the duration constraint $\rho\%$ as 15%. The hyperparameter L in CRG is set to 3 for both the SumMe and TVSum datasets. The number of attention heads n in VSE and CSIL are all set to 8. The number of bottleneck tokens m is set to 8. The dropout rate is set to 0.5 for all experiments. We train our model using the Adam optimizer with the learning rate of 1×10^{-5} for both datasets. In addition, the batch size is set to 1. We implement our experiments on the Pytorch platform.

E. Comparisons With State-of-the-Art Methods

1) Baselines: In order to check the effectiveness of the proposed method, we compare the summarization performance of our proposed framework with other state-of-the-art methods on the SumMe and TVSum datasets. Specifically, these methods include traditional methods: TVSum [10], DPP [74], ERSUM [75], MSDS-CC [18], and deep learning-based methods: vsLSTM [50], dppLSTM [50], SUM-GAN [46], DR-DSN [30], SUM-FCN [32], SASUM [29], HSA-RNN [25], ACGAN [76], CSNet [7], VASNet [34], A-AVS [26], M-AVS [26], AVRN [77], RSGN [56], DHAVS [49], LMHA [35], CAAN [64], HMT [78], VJMHT [79], SSPVS [63]. The details of the comparisons and analysis are provided below.

2) Comparisons Under the Standard Setting: Table II reports the experimental results of different methods under the canonical setting on the F-score metric. As can be seen, VSS-Net significantly outperforms traditional methods including TVSum [10], DPP [74], ERSUM [75], and MSDS-CC [18], achieving at least 8.4% and 1.6% absolute gains on the SumMe and TVSum datasets, respectively. This is because these traditional methods lack powerful representation capability, contributing to a limitation in effectively grasping the video storyline. Furthermore, vsLSTM [50], dppLSTM [50], and DR-DSN [30] utilize a single LSTM to aggregate global temporal information, resulting in considerably inferior summarization performance compared to our proposed method. We attribute this to the inherent weakness of RNNs in handling long-range sequential data. Also, local temporal information is also not explored. Compared with SUM-FCN [32] and CAAN [64], our method beats them by at least 0.9% and 1.7% on two datasets. This is because it focuses on modeling local temporal information and cannot effectively explore the sophisticated structural information of videos. Although LMHA [35] that effectively mines local and non-local temporal cues achieves comparable summarization performance with ours on the TVSum dataset, our method differs in the evaluation process, that is, we adopt the standard 5-fold cross-validation to comprehensively test our method using all videos, ensuring fair evaluation across the entire dataset. Compared

TABLE II

COMPARISONS WITH OTHER STATE-OF-THE-ART METHODS ON THE SUMME AND TVSUM DATASETS UNDER THE STANDARD EVALUATION SETTING IN THE F-SCORE (%) EVALUATION. WE ADDITIONALLY PRESENT THE NUMBER OF PARAMETERS AND THE TEST METHOD OF DIFFERENT WORKS. THE BEST RECORDS ARE HIGHLIGHTED IN BOLDFACE. * DENOTES THAT THE RESULTS ARE REPRODUCED BY USING THE SAME SETTING AS OURS FOR A FAIR COMPARISON

Methods	Shot Segmentation	Feature	SumMe ↑	TVSum ↑	Params (M) ↓	Test Method
TVSum [10]	Change-point detection	HoG+GIST+SIFT	-	50.0	-	-
DPP [74]	KTS	AlexNet	40.9	-	-	5 Random
ERSUM [75]	Uniform segmentation	VGGNet-16	43.1	59.4	-	-
MSDS-CC [18]	KTS	GIST+GoogLeNet	40.6	52.3	-	-
vsLSTM [50]	KTS	GoogLeNet	37.6	54.2	2.63	5 Random
dppLSTM [50]	KTS	GoogLeNet	38.6	54.7	2.63	5 Random
SUM-GAN [46]	KTS	GoogLeNet	41.7	56.3	295.86	5 Random
DR-DSN [30]	KTS	GoogLeNet	42.1	58.1	2.63	5 FCV
SUM-FCN [32]	KTS	GoogLeNet	47.5	56.8	36.58	M Random
SASUM [29]	KTS	InceptionV3	45.3	58.2	44.07	10 Random
HSA-RNN [25]	Change-point detection	VGGNet-16	42.3	58.7	4.20	5 Random
ACGAN [76]	KTS	GoogLeNet	47.2	59.4	31.51	5 FCV
CSNet [7]	KTS	GoogLeNet	48.6	58.5	-	5 FCV
VASNet [34]	KTS	GoogLeNet	49.7	**61.4**	7.35	5 Random
VASNet* [34]	KTS	GoogLeNet	49.2	60.7	7.35	5 FCV
A-AVS [26]	KTS	GoogLeNet	43.9	59.4	4.40	5 Random
M-AVS [26]	KTS	GoogLeNet	44.4	61.0	4.40	5 Random
AVRN [77]	KTS	GoogLeNet	44.1	59.7	-	5 Random
RSGN [56]	KTS	GoogLeNet	45.0	60.1	-	5 Random
DHAVS [49]	KTS	3D ResNeXt-101	45.6	60.8	-	5 Random
LMHA [35]	KTS	GoogLeNet	51.1	61.0	-	5 Random
CAAN [64]	KTS	GoogLeNet	50.6	59.3	-	5 FCV
HMT [78]	KTS	GoogLeNet	44.1	60.1	-	5 Random
VJMHT [79]	KTS	GoogLeNet	50.6	60.9	-	5 FCV
SSPVS [63]	KTS	GoogLeNet	48.7	60.3	-	5 FCV
VSS-Net	KTS	GoogLeNet	**51.5**	61.0	23.12	5 FCV

with VASNet* [16], VJMHT [79], and SSPVS [63] that also adopt the 5 FCV test method, VSS-Net still achieves the best summarization performance, which indicates the effectiveness of our elaborate architecture. We believe the reason is that our method can mine more valuable complementary information about videos by jointly modeling frame-level temporal cues and video-level visual semantic representation. By jointly evaluating the performance and the number of parameters of different methods, VSS-Net is proven to achieve a well-performing balance. In brief, compared with other state-of-the-art methods, VSS-Net performs impressively on the benchmark datasets.

3) Comparisons Under the Augmented and Transfer Settings: Table III further explores the performance under the augmented and transfer settings on the SumMe and TVSum datasets along with additional OVP and YouTube datasets. The experimental results under the standard evaluation setting are also provided for an intuitive difference of F-scores. Concretely, we find that the F-scores under the augmented setting are higher than that under the standard setting by 1.3% and 0.4% on the SumMe and TVSum datasets, respectively. This can be explained by the truth that more guidance information is provided with the assistance of adding training data. On the contrary, the summarization performance decays under the transfer setting. Affected by differences in visual content and manual annotation types across datasets, it proves to be a challenging manner in evaluating the transferability of different methods. Nevertheless, our method still outperforms most methods, which reveals the effectiveness of our framework.

TABLE III

COMPARISONS WITH OTHER STATE-OF-THE-ART METHODS ON THE SUMME AND TVSUM DATASETS UNDER DIFFERENT EVALUATION SETTINGS. THE BEST RECORDS ARE HIGHLIGHTED IN BOLDFACE. * DENOTES THAT THE RESULTS ARE REPRODUCED BY USING THE SAME SETTING AS OURS FOR A FAIR COMPARISON

Methods	SumMe ↑			TVSum ↑		
	C	A	T	C	A	T
vsLSTM [50]	37.6	41.6	40.7	54.2	57.9	56.9
dppLSTM [50]	38.6	42.9	41.8	54.7	59.6	58.7
SUM-GAN [46]	41.7	43.6	-	56.3	61.2	-
DR-DSN [30]	42.1	43.9	42.6	58.1	59.8	58.9
SUM-FCN [32]	47.5	51.1	44.1	56.8	59.2	58.2
HSA-RNN [25]	42.3	42.1	-	58.7	59.8	-
CSNet [7]	48.6	48.7	44.1	58.5	57.1	57.4
VASNet [34]	49.7	51.1	-	**61.4**	**62.4**	-
VASNet* [34]	49.2	50.3	44.3	60.7	60.9	57.2
A-AVS [26]	43.9	44.6	-	59.4	60.8	-
M-AVS [26]	44.4	46.1	-	61.0	61.8	-
AVRN [77]	44.1	44.9	43.2	59.7	60.5	58.7
RSGN [56]	45.0	45.7	44.0	60.1	61.1	**60.0**
DHAVS [49]	45.6	46.5	43.5	60.8	61.2	57.5
LMHA [35]	51.1	52.1	45.4	61.0	61.5	55.1
HMT [78]	44.1	44.8	-	60.1	60.3	-
VJMHT [79]	50.6	51.7	46.4	60.9	61.9	58.9
SSPVS [63]	48.7	50.4	45.8	60.3	61.8	57.8
VSS-Net	**51.5**	**52.8**	**48.4**	61.0	61.4	58.5

4) Comparisons of Rank Order Statistics: Table IV presents the experimental results of Kendall's τ and Spearman's ρ. This evaluation metric can reflect the consistency degree across manual annotations and predicted scores. It can be observed

TABLE IV
COMPARISONS WITH OTHER STATE-OF-THE-ART METHODS IN THE RANK-BASED EVALUATION. THIS EXPERIMENT USES THE TVSUM DATASET UNDER THE CANONICAL SETTING. THE BEST RECORDS ARE HIGHLIGHTED IN BOLDFACE

Methods	Kendall's $\tau \uparrow$	Spearman's $\rho \uparrow$
Random [73]	0.000	0.000
Human [73]	0.177	0.204
dppLSTM [50]	0.042	0.055
DR-DSN [30]	0.020	0.026
HSA-RNN [25]	0.082	0.088
AVRN [77]	0.096	0.107
CAAN [64]	0.038	0.050
DHAVS [49]	0.082	0.089
RSGN [56]	0.083	0.090
HMT [78]	0.096	0.107
VJMHT [79]	0.097	0.105
SSPVS [63]	0.177	0.233
SSPVS+Text [63]	0.181	0.238
VSS-Net	**0.190**	**0.249**

that the correlation coefficients given by VSS-Net are higher than human-created summaries and significantly outperform other state-of-the-art methods by a large margin. This can be explained by the following reasons. Firstly, the inconsistent ground truth scores in video summarization can be caused by differences in user preferences and perceptions of what constitutes important content. Secondly, it may benefit from our well-designed VSS-Net that takes into consideration both temporal cues and video-level semantic information of the input video itself. This enables our model to precisely locate valuable parts by dynamically learning the main content as well as performing a high-level semantic interaction across contextual information and the inherent semantic information. It is noteworthy that, with the assistance of textual information, SSPVS [63] has achieved encouraging correlation coefficients, while our appearance-based method still beat it. This observation indicates the absolute superiority of VSS-Net in generating summaries that meet human preferences.

F. Ablation Study

1) Study on Attention Head: To investigate the effectiveness of our multi-head attention module in refined feature learning, we conduct experiments to explore the optimal number of attention heads for the video summarization task. Specifically, we vary the number of attention heads as $\{1, 2, 4, 8, 16\}$, and evaluate the F-score performance on the benchmark datasets. As shown in Fig. 7, the F-score performance generally improves with the number increasing. This is because multi-head attention allows our method to learn more effective feature representations from multiple subspaces, thereby improving the overall quality of summarization. Interestingly, our experiments suggest that the models adopting 8 and 16 attention heads, respectively, showcase a comparable summarization performance. To balance between a high F-score and low parameter complexity, we set the default number of attention heads for both the SumMe and TVSum datasets as 8.

2) Study on Bottleneck Token: Also, Fig. 8 explores the impact of bottleneck tokens on the summarization performance

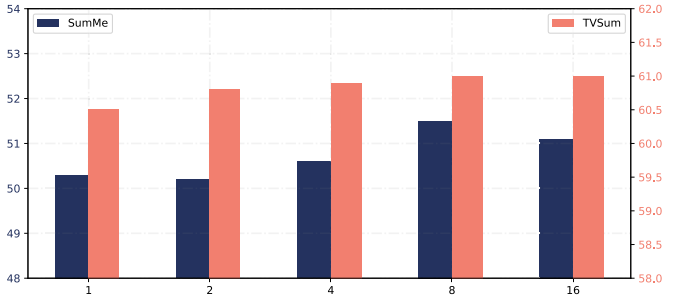

Fig. 7. F-score results (%) of different numbers of attention heads in our framework on the SumMe and TVSum datasets.

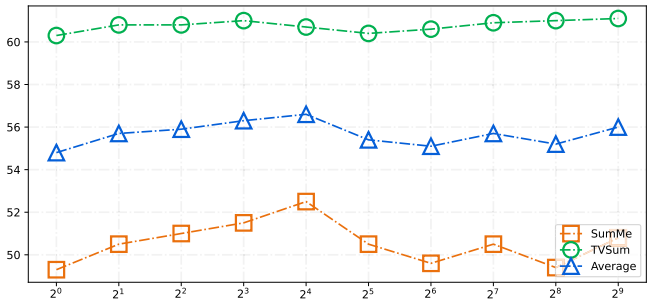

Fig. 8. F-score results (%) of different numbers of bottleneck tokens in CSIL on the SumMe and TVSum datasets.

of our model. We can observe that our model achieves a comparable performance when the length of bottleneck tokens is set to 8, 16, and 512, outperforming other situations. This suggests that even a small number of intermediate transmission tokens can still enable effective information interaction. It is worth noting that the F-score on the SumMe dataset fluctuates more significantly than that on the TVSum dataset, which can be attributed to the different evaluation protocols. In our model, we set the default length of the bottleneck tokens to 8, which strikes a balance between generating high-quality summaries and avoiding expensive computing costs.

3) Study on Dominant Components: The experimental results that can verify the effectiveness of each component are summarized in Table V. Our baseline only consists of a prediction head that is employed to predict scores, without modeling any contextual information within input videos. As expected, it shows the worst summarization performance. Through aggregating contextual information leveraging CRG, the F-scores are significantly improved on two datasets, indicating the effectiveness and necessity of temporal cues to video understanding. By incorporating video-level embedding encoded by VSE into our model, our method can acquire more valuable information regarding videos and generate powerful feature representations, thus further boosting performance remarkably. Here we present three sets of results using concatenation (Concat.), cross attention (CA), and CSIL to achieve information exchange between temporal cues and video-level semantics. We employ a fully connected layer for feature dimensionality reduction after the concatenation. As can be seen, CSIL remarkably surpasses the concatenation-based manner due to the failure of exploring the sophisticated relationship across features. Although the

TABLE V

F-SCORE (%) VIA ABLATION STUDIES ABOUT DOMINANT COMPONENTS ON THE SUMME AND TVSUM DATASETS. ✓ STANDS FOR THE CORRESPONDING MODULE. THE BEST RECORDS ARE HIGHLIGHTED IN BOLDFACE

Dominant Components						SumMe ↑			TVSum ↑		
Baseline	CRG	VSE	Concat.	CA	CSIL	Canonical (C)	Augmented (A)	Transfer (T)	Canonical (C)	Augmented (A)	Transfer (T)
✓						47.2	49.9	44.4	56.0	60.2	56.3
✓	✓					50.0	50.3	44.8	59.4	59.3	56.7
✓	✓	✓	✓			50.1	50.5	46.4	59.8	60.3	57.3
✓	✓	✓		✓		51.2	50.5	**48.7**	**61.1**	60.4	58.2
✓	✓	✓			✓	**51.5**	**52.8**	48.4	61.0	**61.4**	**58.5**

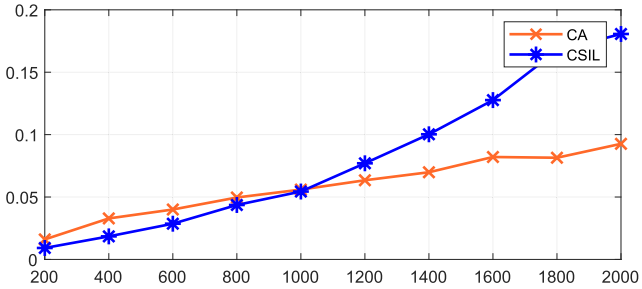

Fig. 9. Runtime analysis about the CSIL and cross attention (CA) module. The x-axis and y-axis denote the runtime (s) and the length of features, respectively.

TABLE VI

F-SCORE (%) VIA ABLATION STUDIES ABOUT DIFFERENT EDGES IN CRG ON THE SUMME AND TVSUM DATASETS, RESPECTIVELY. ✓ AND ✗ STAND FOR WITH OR WITHOUT THE CORRESPONDING EDGE. THE BEST RECORDS ARE HIGHLIGHTED IN BOLDFACE

Exp.	Edge Types			SumMe ↑	TVSum ↑
	Temporal	Similar	Dissimilar		
1	✓	✓	✓	**51.5**	**61.0**
2	✗	✓	✓	50.5	60.7
3	✓	✗	✓	50.2	60.9
4	✓	✓	✗	50.4	60.6

CA-based model is slightly better than the CSIL module under certain situations, the CSIL-based model has an impressive economical calculation cost. Fig. 9 shows the runtime analysis results. Evidently, CA exhibits a shorter runtime until the length increases to a specific threshold since it only incorporates a single multi-head attention module and corresponds to fewer model parameters (16.81M vs. 4.20M). CSIL with linear complexity demonstrates its significant advantage in handling long videos. In short, the above empirical results can well reflect the positive impact of our dominant components.

4) Study on Different Edges: We investigate the contribution of the three sets of edges in CRG by removing them from our full model. The results can be found in Table VI. As previously stated, the Temporal Edges are responsible for modeling the temporal order within videos, which is indispensable to allow for comprehensive content awareness. When removed, they lead to noticeable degradation of the F-score performance of 1.0% and 0.3% on the SumMe and TVSum datasets, respectively. When the Similar Edges and Dissimilar Edges are removed from our default model, the performances are impaired by a drop of 1.3% and 1.1% on the SumMe dataset, 0.1% and 0.4% on the TVSum dataset, showcasing the effectiveness of modeling interaction across frames through considering different visual relationships. These experimental results provide obvious evidence of the impact of each type of edge.

TABLE VII

F-SCORE (%) VIA ABLATION STUDIES ABOUT DIFFERENT AGGREGATION METHODS ON THE SUMME AND TVSUM DATASETS, RESPECTIVELY. THE BEST RECORDS ARE HIGHLIGHTED IN BOLDFACE

Method	SumMe ↑			TVSum ↑		
	C	A	T	C	A	T
BiLSTM	50.9	52.1	47.9	59.8	60.2	57.6
BiGRU	51.3	51.7	47.8	60.0	60.0	57.6
GCN	50.6	51.9	46.5	60.3	60.6	58.0
CRG	**51.5**	**52.8**	**48.4**	**61.0**	**61.4**	**58.5**

TABLE VIII

F-SCORE (%) VIA ABLATION STUDIES ABOUT DIFFERENT VIDEO-LEVEL EMBEDDING ON THE SUMME AND TVSUM DATASETS, RESPECTIVELY. THE BEST RECORDS ARE HIGHLIGHTED IN BOLDFACE

Method	SumMe ↑			TVSum ↑		
	C	A	T	C	A	T
Pooling	48.8	49.1	45.0	60.1	60.8	57.2
BiLSTM	50.3	49.0	45.0	60.7	60.0	57.3
BiGRU	49.5	49.5	47.8	60.5	60.0	57.5
VSE	**51.5**	**52.8**	**48.4**	**61.0**	**61.4**	**58.5**

5) Study on Aggregation Method: We conduct experiments to verify the superiority of the proposed aggregation methods over others including BiLSTM, BiGRU, and GCN. The experimental results are shown in Fig. VII. Following [30], a single-layer RNN is employed in our ablation experiments with only a change in the dimension of hidden states. Our unique aggregation module achieves the best performance, as demonstrated by the results. The summarization performance based on RNNs is relatively poor on the TVSum dataset since its average duration is longer than that on the SumMe dataset. Additionally, VSS-Net outperforms the method that used GCN as the aggregation method, as CRG can effectively explore the hierarchical information within videos by considering the visual relationship across frames.

6) Study on Video-level Embedding Encoder: We conduct an ablation study using three video-level embedding encoders mentioned in Sec. III-D, including pooling-based embedding (Pooling), RNN-based embedding (BiLSTM and BiGRU), and VSE-based embedding (VSE) for dynamically learning important visual content of input videos, which serves as valuable guidance information. The experimental results are presented in Table VIII. As we can see from the results, our default method incorporating VSE achieves the best summarization performance on two benchmark datasets. In contrast, the model combining the pooling-based embedding

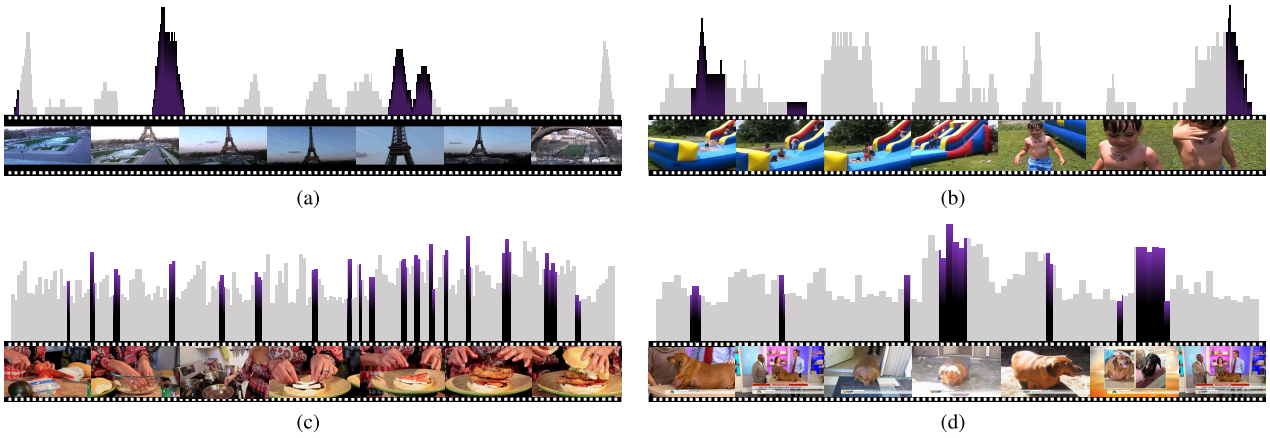

Fig. 10. Visualization results of summarization. Four videos are selected as examples, including 9-th and 16-th video in the SumMe dataset, 16-th and 49-th video in the TVSum dataset. The light gray bars stand for ground truth importance scores and the colored bars are the selected summaries. The x-axis is the frame index. The images below are example frames in generated summaries by our method.

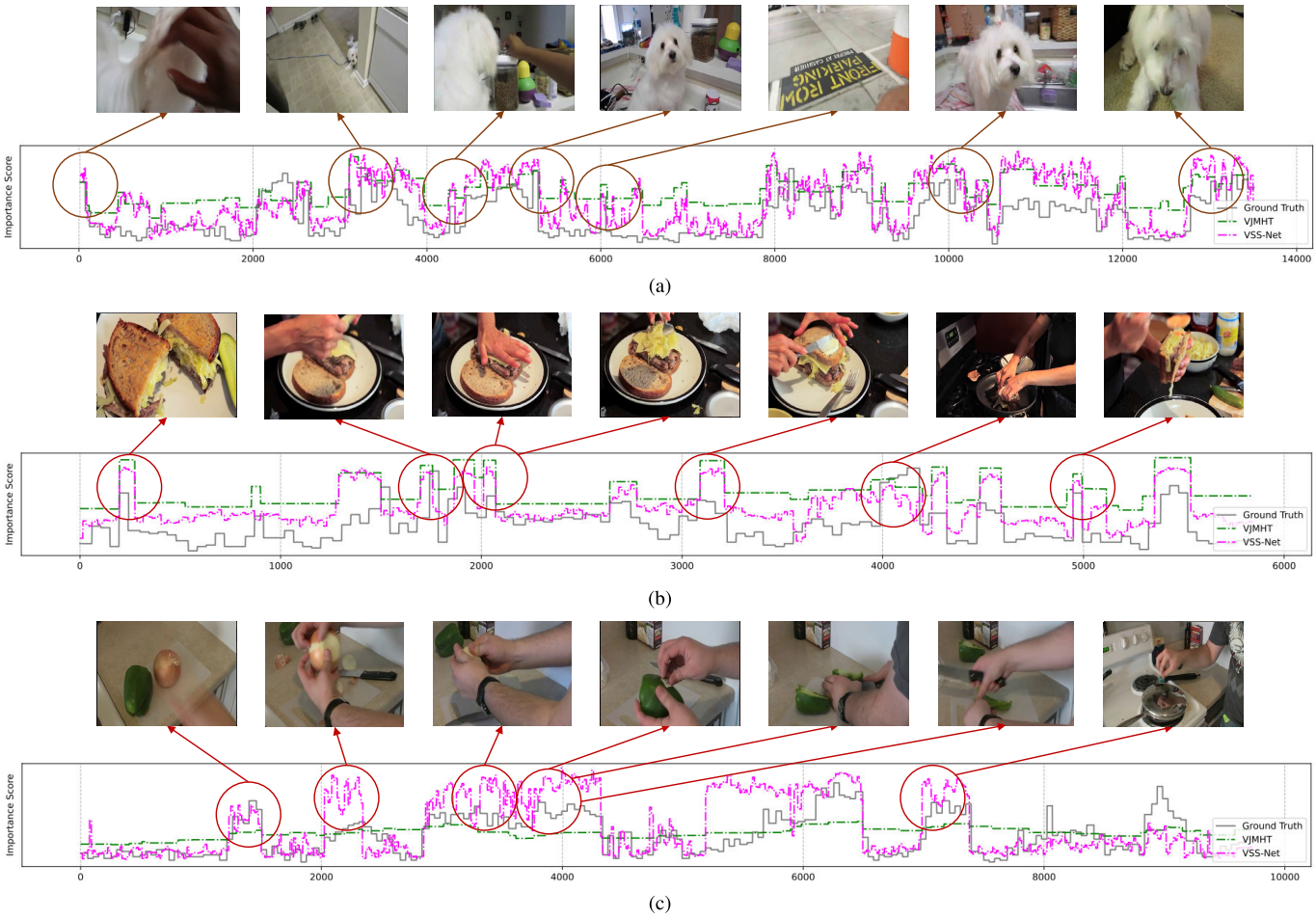

Fig. 11. Visualization results of the correlation. The results of 12-th, 17-th, and 18-th video in the TVSum dataset are presented. The predicted scores and ground truth scores are shown by green and pink/blue lines, respectively. The importance curves of our method are highly consistent with the ground truth importance curves, demonstrating the high sensitivity of our method to important content within videos.

obtains the worst results, likely because rough average pooling inevitably introduces noise and discards valuable information.

G. Visualization Results

To intuitively demonstrate the effectiveness of our proposed network, we select several videos from SumMe and TVSum datasets and present the generated summaries. These videos cover a variety of topics, including buildings, entertainment, food, and pets. Fig. 10 shows the visualization results, showing that VSS-Net can effectively extract key segments from untrimmed videos. Specifically, most peak regions are selected into the final summaries, which enable viewers to quickly

understand the topic and activities in these videos. For example, in Fig. 10(c), the video is titled "Mexican Fried Chicken Sandwich Recipe", and we can see that the generated summary can present the overall process of making sandwiches, which significantly improves the efficiency of video browsing.

Also, in Fig. 11, we visualize the importance curves of the predicted importance scores given by our VSS-Net and VJMHT [79] on several example videos from the TVSum dataset. From the displayed results, we can observe that VJMHT has a considerable performance compared with our method on certain videos. Nevertheless, when dealing with more challenging videos, it might output flat scores, as shown in Fig. 11(c), while VSS-Net can effectively assign higher importance scores to more important segments, and assign lower scores for unmeaningful segments. This demonstrates that our method is more sensitive to the important content within input videos.

H. Discussion on Challenges

Notwithstanding the impressive summarization performance obtained by VSS-Net on benchmark datasets, there are still several challenges that need to be further studied in future research.

(1) Datasets: Existing methods predominantly rely on datasets (*i.e.,* SumMe [39] and TVSum [10]) released in the past few years to test the proposed methods. However, these datasets suffer from a limited amount of data size and inconsistent annotation types, which poses challenges for both training and evaluation. Hence, the proposal of a new large-scale standard dataset is an urgent problem to be addressed.

(2) Change-point Detection: Most contributions, including the proposed VSS-Net, adopt the KTS algorithm [72], which directly affects the performance of video summarization. Intuitively, under the limitation of existing shot boundary detection methods as well as the constraint of the summary length, it is still challenging to include all segments with high-importance scores.

V. Conclusion

This paper has presented a Visual Semantic Self-mining Network (VSS-Net) for video summarization motivated by the recent success of cross-modality tasks. To the best of our knowledge, this is the first attempt to use the idea of cross-modality learning to address video summarization. The proposed method consists of a Context Representation Graph (CRG), a Video Semantics Encoder (VSE), and a Context-Semantics Interaction Layer (CSIL). Specifically, CRG is devised to enrich visual features with local and non-local contextual information through three sets of edges suitable for the hierarchical structure of videos. Meanwhile, VSE can adaptively aggregate the video sequence into an informative video-level semantic representation using a modeling approach from coarse to fine. Finally, deep yet economical information exchange across context and semantic representation is performed with the assistance of CSIL, guaranteeing powerful contextualized representation. Compared with previous works, the proposed framework shows promising experimental results and achieves state-of-the-art performance in the rank-based evaluation. Based on the current challenges, our future research work includes the construction of large-scale datasets and efficient shot boundary detection.

References

[1] Y. Zhang, K. Guo, and R. Tao, "Adaptive spatio-temporal tube for fast motion segments extraction of videos," *IEEE Signal Process. Lett.*, vol. 29, pp. 2308–2312, 2022.

[2] T. Hussain, K. Muhammad, W. Ding, J. Lloret, S. W. Baik, and V. H. C. D. Albuquerque, "A comprehensive survey of multi-view video summarization," *Pattern Recognit.*, vol. 109, Jan. 2021, Art. no. 107567.

[3] S. Zhang, Y. Zhu, and A. K. Roy-Chowdhury, "Context-aware surveillance video summarization," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5469–5478, Nov. 2016.

[4] Y. Zhang, R. Tao, and Y. Wang, "Motion-state-adaptive video summarization via spatiotemporal analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 6, pp. 1340–1352, Jun. 2017.

[5] Y. Yuan, T. Mei, P. Cui, and W. Zhu, "Video summarization by learning deep side semantic embedding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 1, pp. 226–237, Jan. 2019.

[6] M. Ma, S. Mei, S. Wan, Z. Wang, D. D. Feng, and M. Bennamoun, "Similarity based block sparse subset selection for video summarization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 10, pp. 3967–3980, Oct. 2021.

[7] Y. Jung, D. Cho, D. Kim, S. Woo, and I. S. Kweon, "Discriminative feature learning for unsupervised video summarization," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 8537–8544.

[8] E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris, and I. Patras, "AC-SUM-GAN: Connecting actor-critic and generative adversarial networks for unsupervised video summarization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 8, pp. 3278–3292, Aug. 2021.

[9] Y. Zhang, J. Zhang, R. Liu, P. Zhu, and Y. Liu, "Key frame extraction based on quaternion Fourier transform with multiple features fusion," *Expert Syst. Appl.*, vol. 216, Apr. 2023, Art. no. 119467.

[10] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "TVSum: Summarizing web videos using titles," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5179–5187.

[11] S. E. F. de Avila, A. P. B. Lopes, A. D. Luz, and A. D. A. Araújo, "VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method," *Pattern Recognit. Lett.*, vol. 32, no. 1, pp. 56–68, Jan. 2011.

[12] Y. Li and B. Merialdo, "Multi-video summarization based on video-MMR," in *Proc. 11th Int. Workshop Image Anal. Multimedia Interact. Services (WIAMIS)*, Apr. 2010, pp. 1–4.

[13] B. Zhao and E. P. Xing, "Quasi real-time summarization for consumer videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 2513–2520.

[14] L. Yuan, F. E. Tay, P. Li, L. Zhou, and J. Feng, "Cycle-SUM: Cycle-consistent adversarial LSTM networks for unsupervised video summarization," in *Proc. AAAI Conf. Artif. Intell.*, Jul. 2019, vol. 33, no. 1, pp. 9143–9150.

[15] C. Huang and H. Wang, "A novel key-frames selection framework for comprehensive video summarization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 2, pp. 577–589, Feb. 2020.

[16] H. Fu and H. Wang, "Self-attention binary neural tree for video summarization," *Pattern Recognit. Lett.*, vol. 143, pp. 19–26, Mar. 2021.

[17] Y. Zhang, Z. Song, and W. Li, "Enhancement multi-module network for few-shot leaky cable fixture detection in railway tunnel," *Signal Process., Image Commun.*, vol. 113, Apr. 2023, Art. no. 116943.

[18] J. Meng, S. Wang, H. Wang, Y.-P. Tan, and J. Yuan, "Video summarization via multi-view representative selection," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 1189–1198.

[19] R. Panda, N. C. Mithun, and A. K. Roy-Chowdhury, "Diversity-aware multi-video summarization," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4712–4724, Oct. 2017.

[20] Y. Xu, X. Li, L. Pan, W. Sang, P. Wei, and L. Zhu, "Self-supervised adversarial video summarizer with context latent sequence learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 4122–4136, Aug. 2023.

[21] Y. Yuan and J. Zhang, "Unsupervised video summarization via deep reinforcement learning with shot-level semantics," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 1, pp. 445–456, Jan. 2023.

[22] S. Cai, W. Zuo, L. S. Davis, and L. Zhang, "Weakly-supervised video summarization using variational encoder–decoder and web prior," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 184–200.

[23] B. Xiong, Y. Kalantidis, D. Ghadiyaram, and K. Grauman, "Less is more: Learning highlight detection from video duration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 1258–1267.

[24] B. Zhao, X. Li, and X. Lu, "Hierarchical recurrent neural network for video summarization," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 863–871.

[25] B. Zhao, X. Li, and X. Lu, "HSA-RNN: Hierarchical structure-adaptive RNN for video summarization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7405–7414.

[26] Z. Ji, K. Xiong, Y. Pang, and X. Li, "Video summarization with attention-based encoder–decoder networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 6, pp. 1709–1717, Jun. 2020.

[27] Z. Wang et al., "Robust video-based person re-identification by hierarchical mining," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 12, pp. 8179–8191, Dec. 2022.

[28] Z. Tu, Y. Liu, Y. Zhang, Q. Mu, and J. Yuan, "DTCM: Joint optimization of dark enhancement and action recognition in videos," *IEEE Trans. Image Process.*, vol. 32, pp. 3507–3520, 2023.

[29] H. Wei, B. Ni, Y. Yan, H. Yu, X. Yang, and C. Yao, "Video summarization via semantic attended networks," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2018, vol. 32, no. 1, pp. 216–223.

[30] K. Zhou, Y. Qiao, and T. Xiang, "Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2018, vol. 32, no. 1, pp. 7582–7589.

[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[32] M. Rochan, L. Ye, and Y. Wang, "Video summarization using fully convolutional sequence networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 347–363.

[33] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[34] J. Fajtl, H. S. Sokeh, V. Argyriou, D. Monekosso, and P. Remagnino, "Summarizing videos with attention," in *Proc. Asian Conf. Comput. Vis. (ACCV)*, 2019, pp. 39–54.

[35] W. Zhu, J. Lu, Y. Han, and J. Zhou, "Learning multiscale hierarchical attention for video summarization," *Pattern Recognit.*, vol. 122, Feb. 2022, Art. no. 108312.

[36] Z. Tang et al., "Human-centric spatio-temporal video grounding with visual transformers," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 12, pp. 8238–8249, Dec. 2022.

[37] J. Lei, T. L. Berg, and M. Bansal, "Detecting moments and highlights in videos via natural language queries," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 11846–11858.

[38] G. Wu, J. Lin, and C. T. Silva, "IntentVizor: Towards generic query guided interactive video summarization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 10493–10502.

[39] M. Gygli, H. Grabner, H. Riemenschneider, and L. V. Gool, "Creating summaries from user videos," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 505–520.

[40] E. Apostolidis, G. Balaouras, V. Mezaris, and I. Patras, "Combining global and local attention with positional encoding for video summarization," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Nov. 2021, pp. 226–234.

[41] W. Zhu, J. Lu, J. Li, and J. Zhou, "DSNet: A flexible detect-to-summarize network for video summarization," *IEEE Trans. Image Process.*, vol. 30, pp. 948–962, 2021.

[42] J. A. Ghauri, S. Hakimov, and R. Ewerth, "Supervised video summarization via multiple feature sets with parallel attention," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jun. 2021, pp. 1–6s.

[43] J. Wu, S.-H. Zhong, J. Jiang, and Y. Yang, "A novel clustering method for static video summarization," *Multimedia Tools Appl.*, vol. 76, no. 7, pp. 9625–9641, Apr. 2017.

[44] I. Mademlis, A. Tefas, and I. Pitas, "A salient dictionary learning framework for activity video summarization via key-frame extraction," *Inf. Sci.*, vol. 432, pp. 319–331, Mar. 2018.

[45] E. Elhamifar, G. Sapiro, and R. Vidal, "See all by looking at a few: Sparse modeling for finding representative objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1600–1607.

[46] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial LSTM networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2982–2991.

[47] G. Wu, J. Lin, and C. T. Silva, "ERA: Entity relationship aware video summarization with Wasserstein GAN," 2021, *arXiv:2109.02625*.

[48] T. Liu, Q. Meng, J.-J. Huang, A. Vlontzos, D. Rueckert, and B. Kainz, "Video summarization through reinforcement learning with a 3D spatio-temporal U-Net," *IEEE Trans. Image Process.*, vol. 31, pp. 1573–1586, 2022.

[49] J. Lin, S.-H. Zhong, and A. Fares, "Deep hierarchical LSTM networks with attention for video summarization," *Comput. Electr. Eng.*, vol. 97, Jan. 2022, Art. no. 107618.

[50] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 766–782.

[51] M. Soldan, M. Xu, S. Qu, J. Tegner, and B. Ghanem, "VLG-Net: Videolanguage graph matching network for video grounding," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Jun. 2021, pp. 3224–3234.

[52] L. Wang, Y. Huang, Y. Hou, S. Zhang, and J. Shan, "Graph attention convolution for point cloud semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10288–10297.

[53] W. Li, X. Liu, and Y. Yuan, "SIGMA: Semantic-complete graph matching for domain adaptive object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5281–5290.

[54] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, Oct. 2019.

[55] Z. Yang, J. Qin, and D. Huang, "ACGNet: Action complement graph network for weakly-supervised temporal action localization," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2022, vol. 36, no. 3, pp. 3090–3098.

[56] B. Zhao, H. Li, X. Lu, and X. Li, "Reconstructive sequence-graph network for video summarization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 5, pp. 2793–2801, May 2022.

[57] Y. Zhang, Y. Liu, P. Zhu, and W. Kang, "Joint reinforcement and contrastive learning for unsupervised video summarization," *IEEE Signal Process. Lett.*, vol. 29, pp. 2587–2591, 2022.

[58] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[59] Z. Tu, W. Xie, J. Dauwels, B. Li, and J. Yuan, "Semantic cues enhanced multimodality multistream CNN for action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 5, pp. 1423–1437, May 2019.

[60] K. Zhang and Z. Chen, "Video saliency prediction based on spatial–temporal two-stream network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 12, pp. 3544–3557, Dec. 2019.

[61] B. Han, X. Han, H. Zhang, J. Li, and X. Cao, "Fighting fake news: Two stream network for deepfake detection via learnable SRM," *IEEE Trans. Biometrics, Behav., Identity Sci.*, vol. 3, no. 3, pp. 320–331, Jul. 2021.

[62] Y. Zhao, X. Wang, X. Yu, C. Liu, and Y. Gao, "Gait-assisted video person retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 2, pp. 897–908, Feb. 2023.

[63] H. Li, Q. Ke, M. Gong, and T. Drummond, "Progressive video summarization via multimodal self-supervised learning," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2023, pp. 5573–5582.

[64] G. Liang, Y. Lv, S. Li, S. Zhang, and Y. Zhang, "Video summarization with a convolutional attentive adversarial network," *Pattern Recognit.*, vol. 131, Nov. 2022, Art. no. 108840.

[65] E. Apostolidis, G. Balaouras, V. Mezaris, and I. Patras, "Summarizing videos using concentrated attention and considering the uniqueness and diversity of the video frames," in *Proc. Int. Conf. Multimedia Retr.*, Jun. 2022, pp. 407–415.

[66] S. Xiao, Z. Zhao, Z. Zhang, Z. Guan, and D. Cai, "Query-biased self-attentive network for query-focused video summarization," *IEEE Trans. Image Process.*, vol. 29, pp. 5889–5899, 2020.

[67] J. Mun, M. Cho, and B. Han, "Local–global video-text interactions for temporal grounding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10807–10816.

[68] J. Gao, C. Sun, Z. Yang, and R. Nevatia, "TALL: Temporal activity localization via language query," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5277–5285.

[69] Y. Zhang, J. Chen, and D. Huang, "CAT-Det: Contrastively augmented transformer for multimodal 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 898–907.

[70] D. Min, C. Zhang, Y. Lu, K. Fu, and Q. Zhao, "Mutual-guidance transformer-embedding network for video salient object detection," *IEEE Signal Process. Lett.*, vol. 29, pp. 1674–1678, 2022.
[71] A. Nagrani, S. Yang, A. Arnab, A. Jansen, C. Schmid, and C. Sun, "Attention bottlenecks for multimodal fusion," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 14200–14213.
[72] D. Potapov, M. Douze, Z. Harchaoui, and C. Schmid, "Category-specific video summarization," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 540–555.
[73] M. Otani, Y. Nakashima, E. Rahtu, and J. Heikkilä, "Rethinking the evaluation of video summaries," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7588–7596.
[74] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Summary transfer: Exemplar-based subset selection for video summarization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1059–1067.
[75] X. Li, B. Zhao, and X. Lu, "A general framework for edited video and raw video summarization," *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 3652–3664, Aug. 2017.
[76] X. He et al., "Unsupervised video summarization with attentive conditional generative adversarial networks," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 2296–2304.
[77] B. Zhao, M. Gong, and X. Li, "AudioVisual video summarization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 8, pp. 5181–5188, Aug. 2023.
[78] B. Zhao, M. Gong, and X. Li, "Hierarchical multimodal transformer to summarize videos," *Neurocomputing*, vol. 468, pp. 360–369, Jan. 2022.
[79] H. Li, Q. Ke, M. Gong, and R. Zhang, "Video joint modelling based on hierarchical transformer for co-summarization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3904–3917, Mar. 2023.
[80] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

Yunzuo Zhang (Member, IEEE) was born in 1984. He received the Ph.D. degree from the School of Information and Electronics, Beijing Institute of Technology, Beijing, in 2016. In 2018, he was a Visiting Scholar with California State University. He is a Professor with the School of Information Science and Technology, Shijiazhuang Tiedao University, Hebei, Shijiazhuang, China. His research interests include image processing, intelligent video analysis, and big data processing. He is a Senior Member of the China Computer Federation (CCF) and a member of the Chinese–American Engineers and Scientists Association of Southern California (CESASC).

Yameng Liu was born in 1998. He received the B.S. degree from the School of Information Technology, Hebei University of Economics and Business, China, in 2021. He is currently pursuing the master's degree with the School of Information Science and Technology, Shijiazhuang Tiedao University. His research interests include image processing and video summarization.

Weili Kang was born in 1999. She received the B.S. degree in computer science and technology from the Hebei University of Technology, Langfang Branch, in 2021. She is currently a Graduate Research Candidate in electronic information with Shijiazhuang Tiedao University. Her main interests are image processing and person re-identification.

Ran Tao (Senior Member, IEEE) was born in 1964. He received the B.S. degree from the Electronic Engineering Institute of PLA, Hefei, China, in 1985, and the M.S. and Ph.D. degrees from the Harbin Institute of Technology, Harbin, China, in 1990 and 1993, respectively. He is currently a Professor with the School of Information and Electronics, Beijing Institute of Technology, Beijing, China. His current research interests include fractional signal and information processing with applications. He is the Vice Chair of the IEEE China Council and the URSI China Council.