DSNet: A Flexible Detect-to-Summarize Network for Video Summarization

Wencheng Zhu, Jiwen Lu, *Senior Member, IEEE*, Jiahao Li, and Jie Zhou, *Senior Member, IEEE*

***Abstract*— In this paper, we propose a Detect-to-Summarize network (DSNet) framework for supervised video summarization. Our DSNet contains anchor-based and anchor-free counterparts. The anchor-based method generates temporal interest proposals to determine and localize the representative contents of video sequences, while the anchor-free method eliminates the predefined temporal proposals and directly predicts the importance scores and segment locations. Different from existing supervised video summarization methods which formulate video summarization as a regression problem without temporal consistency and integrity constraints, our interest detection framework is the first attempt to leverage temporal consistency via the temporal interest detection formulation. Specifically, in the anchor-based approach, we first provide a dense sampling of temporal interest proposals with multi-scale intervals that accommodate interest variations in length, and then extract their long-range temporal features for interest proposal location regression and importance prediction. Notably, positive and negative segments are both assigned for the correctness and completeness information of the generated summaries. In the anchor-free approach, we alleviate drawbacks of temporal proposals by directly predicting importance scores of video frames and segment locations. Particularly, the interest detection framework can be flexibly plugged into off-the-shelf supervised video summarization methods. We evaluate the anchor-based and anchor-free approaches on the SumMe and TVSum datasets. Experimental results clearly validate the effectiveness of the anchor-based and anchor-free approaches.**

***Index Terms*— Video summarization, interest proposal, anchor-based detection, anchor-free detection, temporal modeling.**

I. Introduction

THE explosive growth of video data has brought an urgency to develop computer vision techniques that can efficiently browse and watch videos [5], [37]. To address this issue, video summarization has drawn increasing attention over the past few years [6], [15], [62], and numerous video summarization methods have been proposed [51], [62], [63]. While considerable progress has been achieved, existing video summarization methods suffer from dynamic visual context and over-fitting problems, easily leading to incorrect and incomplete video summaries. Generally, the objective of video summarization is to generate a more compact version of the original video while preserving its important and relevant contents [47], [68]. Video summarization methods usually proceed in three steps: 1) shot boundary detection, 2) frame-level importance score prediction, and 3) key shot selection.

There has been a rich line of research on video summarization in recent years [3], [15]. Existing video summarization methods can be roughly classified into three categories: 1) unsupervised [4], [14], 2) weakly-supervised [22], [57], and 3) supervised [8], [13]. For the first category, heuristic criteria, such as representativeness [5], [35], diversity [38], [40], and sparsity [41], [62], are exploited to identify important shots. Representative methods include clustering based [36], [77], dictionary learning based [32], [33], [74], subset selection based [5], [6], reinforcement learning based [76], and adversarial learning based [34], [62] approaches. For the second category, some auxiliary information is leveraged, including web priors [1], [3], [21], video titles [64], and video categories [39], [43]. Typical methods include video summarization using web-image priors [21] and category-specific video summarization [43]. While unsupervised and weakly-supervised methods have achieved remarkable performance, they cannot learn from manually created summaries. To handle this problem, supervised methods have been proposed [9], [16], [67], [71]. Representative methods in this category include diverse sequential subset selection for supervised video summarization [11], video summarization with long short-term memory [67], retrospective encoders for video summarization [68], video summarization with attention-based encoder-decoder networks [18], and user-ranking video summarization [16]. However, without the temporal consistency constraint, the predicted scores of video frames in the same semantic segment cannot accurately represent the importance of the corresponding segment.

To address above issues, we present a Detect-to-Summarize network framework called DSNet, which formulates video summarization as a temporal interest detection process and predicts not only temporal locations of segments, but also the corresponding importance scores. Fig. 1 illustrates the main idea of the proposed anchor-based and anchor-free approaches. Specifically, we capture long-range temporal dependencies by using deep convolutional networks for feature extraction,

Manuscript received June 7, 2020; revised October 8, 2020; accepted November 18, 2020. Date of publication December 1, 2020; date of current version December 8, 2020. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802; in part by the National Natural Science Foundation of China under Grant 61822603, Grant U1813218, Grant U1713214, and Grant 61672306; in part by a grant from the Institute for Guo Qiang, Tsinghua University; and in part by the Shenzhen Fundamental Research Fund (Subject Arrangement) under Grant JCYJ20170412170602564. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xiangqian Wu. *(Corresponding author: Jiwen Lu.)*

Wencheng Zhu, Jiwen Lu, and Jiahao Li are with the Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing 100084, China, also with the State Key Laboratory of Intelligent Technologies and Systems, Tsinghua University, Beijing 100084, China, and also with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: zwc17@mails.tsinghua.edu.cn; lujiwen@tsinghua.edu.cn; lijiahao17@mails.tsinghua.edu.cn).

Jie Zhou is with the State Key Laboratory of Intelligent Technologies and Systems, Beijing National Research Center for Information Science and Technology (BNRist), Department of Automation, Tsinghua University, Beijing 100084, China, and also with the Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China (e-mail: jzhou@tsinghua.edu.cn).

Digital Object Identifier 10.1109/TIP.2020.3039886

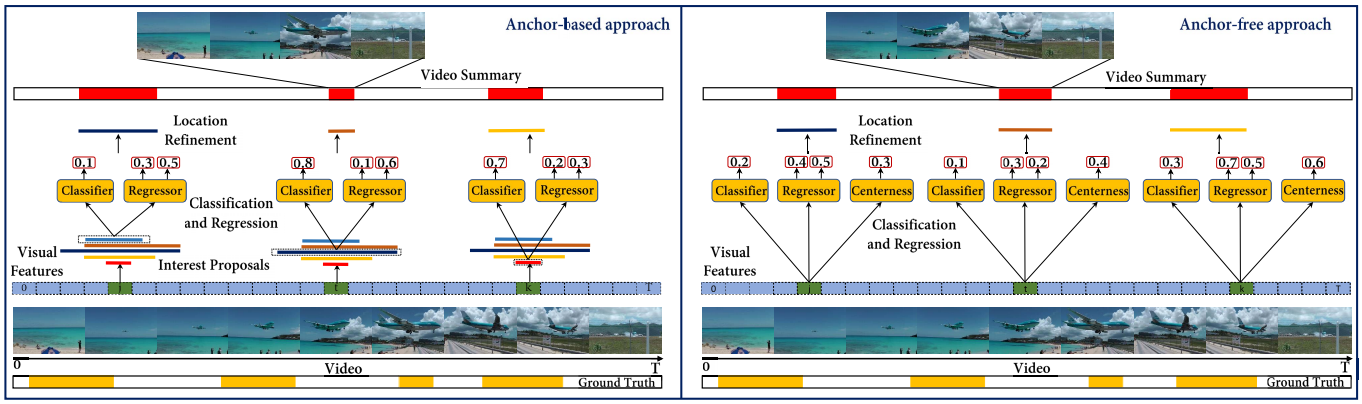


Fig. 1. Illustration of the DSNet framework including the anchor-based and anchor-free approaches. Given a video sequence, we first extract the long-range features. For the anchor-based approach, we produce interest proposals for importance classification and location offset regression, while for the anchor-free approach, we predict the importance score, center-ness score and segment location at each location without proposals. Finally, the video summary is created by using the predicted scores and associated segments.

as video summarization focuses on selecting representatives and thus requires a good understanding of the whole video [8], [67]. For the anchor-based approach, we first produce interest proposals at each temporal location with multi-scale durations, handling the length variations of interest. Due to the bottom-up property, interest proposals can precisely accommodate their boundaries. Then, we perform the interest proposal location regression and the importance prediction. Moreover, to reduce the number of incorrect segments, we minimize the importance scores of negative proposals. Unlike the anchor-based approach which is sensitive to interest proposals and hyper-parameters, we further propose an anchor-free approach to directly predict an importance score, a center-ness score, and a 2D temporal location vector at each temporal location. The anchor-based approach eliminates temporal proposals and avoids complicated computation of temporal Intersection over Union. Finally, we conduct key shot selection, according to the predicted importance scores and segment locations. Experimental results on two standard video summarization datasets, SumMe [12] and TVSum [51], along with two augmented datasets, YouTube [4] and OVP [4], show that the proposed DSNet framework achieves highly competitive performance compared with state-of-the-art methods.

The contributions of this work are summarized as follows:

1) We propose a Detect-to-Summarize network framework, which offers a new perspective of video summarization as a temporal interest detection problem, and simultaneously predicts importance scores and segment locations.
2) We develop the anchor-based approach that generates temporal proposals to handle length variations of interest, and the anchor-free approach that directly learns importance scores and temporal locations, as well as a center-ness score.
3) We conduct extensive experiments on the SumMe and TVSum datasets, and experimental results demonstrate the effectiveness of the proposed approaches.

II. Related Work

In this section, we briefly outline three categories: 1) unsupervised [70], 2) weakly-supervised [22], 3) supervised [48] methods, and the related anchor-based and anchor-free models.

A. Unsupervised Video Summarization

Early unsupervised methods are the clustering-based approaches like k-medoid clustering [14], [77]. These methods mainly leveraged low-level appearance cues and motion information [26], [36]. While good performance has been obtained, they cannot effectively cope with videos with variations in camera motion, illumination conditions and scene clutters [26]. Recently, many unsupervised methods have been proposed and these methods can be roughly grouped into four subcategories: dictionary learning based [32], [33], [74], subset selection based [5], [6] reinforcement learning based [76], and adversarial learning based [9], [62], [69]. Dictionary learning based approaches formulated video summarization as a sparse optimization problem. For example, Elhamifar *et al.* [7] reconstructed the original video by using representative elements in a dictionary. Panda and Roy-Chowdhury [41] developed a sparse representative selection method for multi-video summarization. Subset selection based approaches selected informative subsets of video frames. For example, Elhamifar *et al.* [6] determined representatives by using pairwise dissimilarities between source and target sets. Elhamifar and Kaluza [5] proposed an online subset selection method for video summarization. Reinforcement learning based approaches conducted a discrete sampling of actions to generate summaries. For example, Zhou *et al.* [76] proposed a deep summarization network with a diversity-representativeness reward. Adversarial learning based approaches learned indistinguishable video summaries from ground-truth summaries. For example, Mahasseni *et al.* [34] developed an adversarial LSTM Networks, where the reconstructed videos could not be discriminated by the original videos. Rochan and Wang [46] trained a video summarization model by using unpaired data. Yuan *et al.* [62] leveraged mutual information among summaries and corresponding videos via a cycle-consistent adversarial netowrk.

B. Weakly-Supervised Video Summarization

Weakly-supervised video summarization methods exploited auxiliary information, including web priors [3], [21], video titles [51], and video categories [39], [43]. For example,

Khosla *et al.* [21] leveraged the web-image prior information. Cai *et al.* [1] trained a variational autoencoder (VAE) [23] by using web videos. Song *et al.* [51] selected video shots that are most concerned with visual concepts from title-based image search results. Chu *et al.* [3] developed a video co-summarization method to select shots of the same topics with the frequent visual concepts across videos. Potapov *et al.* [43] proposed a category-specific video summarization method. Panda *et al.* [39] selected video segments by using the derivative of the classification loss.

C. Supervised Video Summarization

Supervised methods have made significant progress, due to human-created summaries. For example, Gygli *et al.* [12] developed a linear model by using information from spatial and temporal salience and landmarks. Gong *et al.* [11] and Sharghi *et al.* [49] both formulated video summarization as a *Determinantal Point Process* [25]. Zhang *et al.* [66] proposed a non-parametric approach that transferred summary structures from training to testing videos. Moreover, deep learning based approaches have been proposed [16], [20]. Among them, RNN based approaches are representative methods. For example, Zhang *et al.* [67] estimated the importance scores of video frames via a bidirectional LSTM. Zhao *et al.* [71], [73] uncovered the underlying hierarchical structure of videos by using a fixed-length hierarchical RNN and a hierarchical structure-adaptive LSTM, respectively. Yao *et al.* [61] built a pairwise deep ranking model to incorporate both spatial and temporal information. Zhang *et al.* [68] formulated video summarization as a sequence-to-sequence learning problem. Huang *et al.* [16] learned multi-stage spatio-temporal representations. Li *et al.* [28] developed a meta learning approach for task-driven video summarization. Zhao *et al.* [72] proposed a property-constrained dual learning approach for video summarization. Hussain *et al.* [17] developed a multi-view video summarization approach by using CNN and Bi-LSTM. Furthermore, attention models [54] have been introduced into recent video summarization methods [55]. For example, Ji *et al.* [18] developed attentive encoder-decoder networks. Fajtl *et al.* [8] conducted video summarization by using a self-attention model. Ji *et al.* [19] learned attentive and distribution consistent video summaries. Xiao *et al.* [56] developed a query-focused approach to learn the semantic information from video descriptions. Besides, Rochan *et al.* [47] applied convolutional sequence networks for video summarization.

D. Anchor-Based and Anchor-Free Models

Due to significant progress in object detection, the idea of region proposal network [45] has been introduced into action localization and visual language grounding tasks. For example, Shou *et al.* [50] proposed a multi-scale Segment-CNN to identify candidate segments. Gao *et al.* [10] simultaneously conducted action proposal generation and temporal coordinate regression. Xu *et al.* [59] predicted variable length proposals by using pre-defined anchors. Chao *et al.* [2] generated multi-scale anchor segments for action localization. Zhao *et al.* [75] applied the structured temporal pyramid to capture the temporal structure of each action instance. Yang *et al.* [60] proposed an anchor-free action localization module to address actions of extremely short or long durations. Zeng *et al.* [65] predicted distances from each frame to the starting and ending frames of segments via a dense regression network.

III. Approach

In this section, we elaborate the proposed Detect-to-Summarize network. We first detail the anchor-based DSNet approach. Then, we describe the anchor-free DSNet approach.

A. Anchor-Based Video Summarization

Fig. 2 depicts the core architecture of the proposed anchor-based approach, which consists of feature extraction, interest proposal generation, interest proposal classification and location regression, and key shot selection steps.

1) Feature Extraction: Capturing long-range temporal information is of great influence for video understanding, especially for video summarization whose goal is to identify the most representative video frames, therefore, the high-level understanding of the whole video is indispensable [8], [67]. Moreover, long-range representations are beneficial to location regression of event boundaries with more context information. In our method, we extract frame-level features and apply a temporal modeling layer to capture long-range representations.

Specifically, given a video sequence $\boldsymbol{V}$ of T frames, we first employ GoogLeNet [52] without the last three layers to extract feature vectors $\boldsymbol{v}_j, j \in \{i, \ldots, T\}$. As the attention mechanism [54] has been proved to be effective to capture long-range dependencies [8], [18], we adopt the self-attention mechanism [54] to extract long-range representations $\{\boldsymbol{w}_j\}_1^T$ in default. Moreover, we investigate the effect of other long-range sequence modeling layers including LSTM, Bi-LSTM, and graph convolution in experiments. To integrate the long-range representation $\boldsymbol{w}_j$ with the original spatial feature $\boldsymbol{v}_j$, we use the summation over $\boldsymbol{w}_j$ and $\boldsymbol{v}_j$. Namely, the final representations are computed as $\boldsymbol{x}_j = \boldsymbol{w}_j + \boldsymbol{v}_j$.

2) Temporal Interest Proposals: Variable-length durations of video interests pose unique challenges for video summarization. However, most existing supervised video summarization methods do not take this problem into account, thereby leading to the incomplete segment selection and unequal importance scores within the same semantic segment. To tackle this issue, we adopt the temporal proposal generation strategy for supervised video summarization, which is inspired by the recent success in region proposal networks [45] and action localization methods [2], [58].

We generate temporal interest proposals with the pre-defined multi-scale intervals. Mathematically, at the t^{th} temporal location, K interest proposals are appointed with the fixed range $[t - l_k/2, t + l_k/2)$, $k = 1, 2, \cdots, K$, where l_k represents the duration of the k^{th} interest proposal. Therefore, $K \times T$ interest proposals are totally produced in a video sequence with T frames. Since which frames will be selected as representatives is unknown beforehand, it is better to keep the same

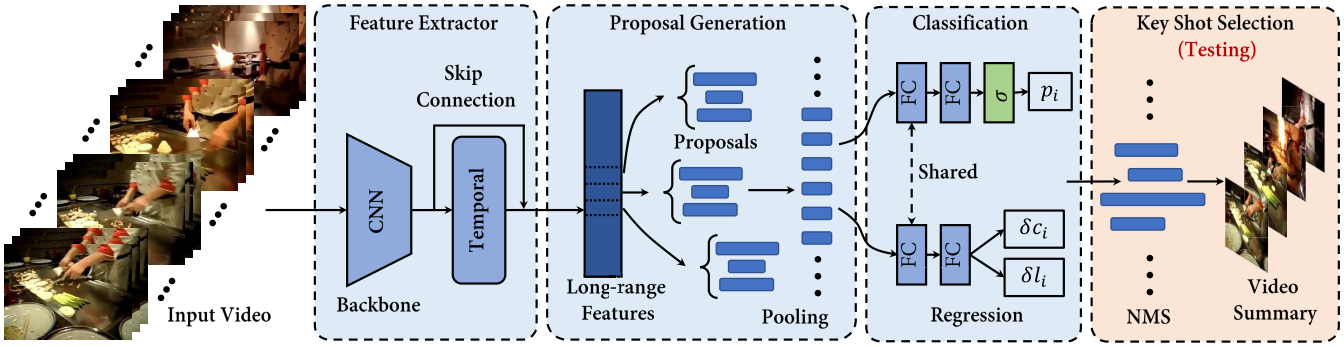


Fig. 2. The architecture of the proposed anchor-based DSNet. We first apply deep convolutional networks and a sequence modeling layer to extract long-range feature vectors. Then, the proposals are generated and pooled for importance classification and location offset regression. For testing, segments are refined by using the predicted offsets, and are further filtered with the non-maximum suppression technique. Finally, the video summary is created.

probability for every input frame, which makes the proposal generation strategy to be temporally invariant.

In the training phase, we assign binary class labels, i.e., positive or negative, to interest proposals. To alleviate the class imbalance problem, we sample the positive and negative proposals in the ratio of 1 : 3. To be more specific, we consider a proposal to be positive when its temporal Intersection over Union (tIoU) with any ground truth segment is higher than 0.6, while we assign a negative label to an interest proposal with tIoU $= 0$ for its unimportance, or with $0 <$ tIoU < 0.3 for its incompleteness. The unimportant and incomplete interest proposals occupy 2/3 and 1/3 of negative samples, respectively. In addition, we observe that assigning negative proposals with a higher tIoU, for example $0.3 <$ tIoU < 0.6, does harm to summary performance. This may be caused by the confusion of the overlap between positive and negative interest proposals.

Generally, there are several benefits for the label assignment of interest proposals in the anchor-based approach. On one hand, by using positive and negative proposals, our method tends to select consecutive frames of a high tIoU with ground truth segments and meanwhile reduce the number of irrelevant segments. On the other hand, our method can cope with incomplete segments by regarding proposals with $0 <$ tIoU $<$ 0.3 as negative examples. Besides, by separating the proposal completeness subtask from classification, our method disentangles two different objectives and prevents the completeness objective from confusing the classifier.

We perform statistics about the durations of ground truth segments on the widely-used SumMe and TVSum benchmarks. The durations of ground truth segments on both datasets range from 1 to 44. In our experiments, we set the tIoU threshold of positive proposals as 0.6. According to *Theorem 1*, we know that $\ell_1/\ell_2 < \zeta^2$, when $\ell_1 = 1$, $\ell_2 = 44$, and $\zeta = 0.6$. Therefore, multi-scale proposals should be specified as Eq. (1). For simplicity, we choose proposals with scales of 1, 2, 4, 8, 16, and 32. To balance the efficiency and effectiveness, we merely specify 4 proposals whose durations are 4, 8, 16, and 32, covering most durations of ground truth segments.

Theorem 1: Denote $[\ell_1, \ell_2]$ as the range of the segment distribution and ζ as the threshold of positive segments. The multi-scale proposals with durations l_k, $k \in \{1, \dots K\}$ in an increasing order are assigned: If $\ell_1/\ell_2 \geq \zeta^2$, $K = 1$ and $l \leq \ell_1/\zeta, l \geq \ell_2 \times \zeta$. Otherwise $\ell_1/\ell_2 < \zeta^2$, and multi-scale interest proposals are appointed as,

$$\begin{cases} l_k/l_{k+1} \geq \zeta^2, \\ l_K \geq \zeta \times \ell_2, \quad l_1 \leq \ell_1/\zeta. \end{cases} \tag{1}$$

Proof: For a proposal with the length of l_k, $k \in \{1, \dots K\}$, it can generate positive segments within the range of $[l_k \times \zeta, l_k/\zeta]$ when the threshold of positive segments is ζ and $0 < \zeta < 1$. Therefore, for K interest proposals at each temporal location, positive segments are produced within the range of $\bigcup_{k=1}^{K} [l_k \times \zeta, l_k/\zeta]$. In order to produce ground truth segments with high probabilities, the specified proposals should meet the constraint that $[\ell_1, \ell_2] \in \bigcup_{k=1}^{K} [l_k \times \zeta, l_k/\zeta]$. Generally, there are two different conditions: 1) $[\ell_1, \ell_2] \in [l_k \times \zeta, l_k/\zeta]$, indicating that one proposal ($K = 1$) is enough to cover the segment distribution. Formally, the length l of this proposal is bounded as,

$$l \times \zeta \leq \ell_1, \quad l/\zeta \geq \ell_2 \Rightarrow l \leq \ell_1/\zeta, \quad l \geq \ell_2 \times \zeta; \tag{2}$$

2) $[\ell_1, \ell_2] \notin [l_k \times \zeta, l_k/\zeta]$ for any k. Then, we achieve the following constraint that,

$$l_1 \times \zeta \leq \ell_1, \quad l_K/\zeta \geq \ell_2 \Rightarrow l_1 \leq \ell_1/\zeta, \quad l_K \geq \ell_2 \times \zeta, \tag{3}$$

where $l_1 \times \zeta$ and l_K/ζ are the minimum and maximum lengths of positive segments. But, there may be interspaces between adjacent segments. Hence, we obtain the second constraint as,

$$l_k/\zeta \geq l_{k+1} \times \zeta \Rightarrow l_k/l_{k+1} \geq \zeta^2 \tag{4}$$

3) Proposal Classification and Regression: Since the fully-connected layer requires fixed-length inputs, we exploit a temporal average pooling layer to pool features of arbitrarily sized proposals, avoiding temporal warping or cropping. Then, the pooled features are fed into the classification and regression module shown in Fig. 3. The module is composed of a shared fully connected layer followed by tanh, dropout (0.5), and layer-normalization layers and two sibling output branches. The first branch outputs importance scores of proposals and the second branch outputs the associated center and

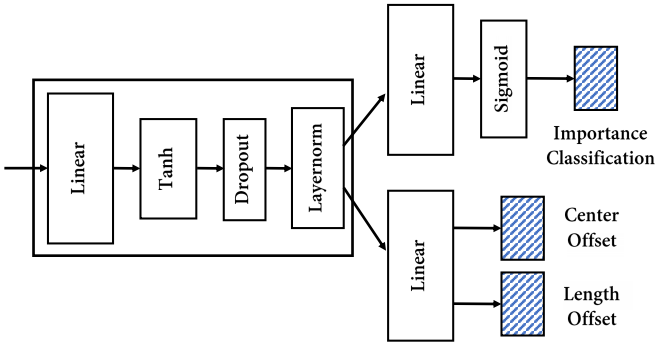


Fig. 3. The detailed components of the classification and regression module.

segment length offsets. We adopt a multi-task loss $\mathcal{L}$ to jointly train our network. Formally, the objective function is written as,

$$\mathcal{L}(\boldsymbol{p}, \boldsymbol{p}^*, \boldsymbol{t}, \boldsymbol{t}^*) = \frac{1}{N}\sum_i \mathcal{L}_{cls}(p_i, p_i^*) + \frac{\lambda}{N_{pos}}\sum_i p_i^* \mathcal{L}_{reg} \times \left(\boldsymbol{t}_i, \boldsymbol{t}_i^*\right), \quad (5)$$

where the hyper-parameter λ balances the classification loss and the location regression loss, N_{pos} represents the number of positive proposals, and N is the number of both positive and negative proposals. p_i (p_i^*) and $\boldsymbol{t}_i$ ($\boldsymbol{t}_i^*$) are the predicted (ground truth) importance score and location offset for the i^{th} proposal, respectively. $\mathcal{L}_{cls}$ represents the cross-entropy loss, and $\mathcal{L}_{reg}$ is the smooth$_{\mathcal{L}_1}$ loss for location offset regression,

$$\mathcal{L}_{reg}\left(\boldsymbol{t}_i, \boldsymbol{t}_i^*\right) = \frac{1}{Q}\sum_{q=1}^{Q} \text{smooth}_{\mathcal{L}_1}\left(t_{iq} - t_{iq}^*\right), \quad (6)$$

where t_{iq} is the q^{th} element of $\boldsymbol{t}_i$. Formally, the smooth$_{\mathcal{L}_1}$ loss is defined as,

$$\text{smooth}_{\mathcal{L}_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1, \\ |x| - 0.5 & \text{otherwise.} \end{cases} \quad (7)$$

The predicted location offset $\boldsymbol{t}_i = (\delta c_i, \delta l_i)$ contains center position and length offsets between generated segments and pre-defined proposals. The ground truth location offset $\boldsymbol{t}_i^* = (\delta c_i^*, \delta l_i^*)$ is computed as follows,

$$\delta c_i^* = \left(c_i^* - c_i\right)/l_i, \quad \delta l_i^* = \ln\left(l_i^*/l_i\right), \quad (8)$$

where c_i^* and l_i^* are the center location and the length of the ground truth segment, and c_i and l_i are the center location and the length of the i^{th} proposals.

Algorithm 1 summarizes the training procedure, performing alternate training for M epochs with several video sequences. For training, we only optimize our network by using the multi-task loss in Eq. (5). However, for testing, we further conduct key shot selection to produce the final video summaries after obtaining the refined proposals.

4) Key Shot Selection: In the testing phase, we generate the refined segments by using the predicted offsets, which is analogous to the training stage. However, many segments are of low confidence and have high overlaps with each other.

Algorithm 1 Training Process of Anchor-Based DSNet

Input: (videos $\{\boldsymbol{V}_i\}$, frame-level annotations $\{\boldsymbol{u}_i^*\}$)
Output: Importance scores $\boldsymbol{p}$ and location offsets $\boldsymbol{t} = (\boldsymbol{\delta c}, \boldsymbol{\delta l})$ of segments; parameters $\boldsymbol{\theta}$ of the anchor-based model

```
θ ← Parameters(DSNet)
for epoch ∈ {1, 2, ..., M} do
    for video V ∈ {V_i} and annotation u* ∈ {u_i*} do
        % Apply KTS and knapsack
        s* ← ToShotLevelAnnotation(u*)
        for frame I_j ∈ V do
            v_j ← GoogLeNet(I_j)
            for k ∈ {1, 2, ..., K} do
                b_jk ← [j − l_k/2, j + l_k/2)  % proposals
                p*_jk ← AssignLabel(s*, b_jk)  % tIoU
                t*_jk ← ComputeOffset(s*, b_jk)  % Eq. (8)
            end
        end
        w ← ExtractTemporalFeature(v)
        x ← w + v
        x ← TemporalPooling(x)
        p ← ClassificationBranch(x)  % importance scores
        t ← RegressionBranch(x)  % location offsets
        L ← L(p, p*, t, t*)  % the loss in Eq. (5)
        θ ← θ − η ∂L/∂θ
    end
end
```

Therefore, we perform the non-maximum suppression (NMS), a separate post-processing technique, on these refined proposals to remove the redundant and low-quality segments [59].

To further generate video summaries, we need to segment video sequences into shots and estimate the shot-level importance scores. Firstly, we follow previous works [68], [76] and apply kernel temporal segmentation (KTS) [43], a fast and accurate shot detection approach, to segment video sequences into video shots. Secondly, for testing video sequences, we employ our training model to predict segment boundaries and their importance scores. According to the information, we design a simple strategy to provide the final frame-level importance scores: assign the maximum value of predicted segments at the t^{th} temporal location as the t^{th} frame-level importance score. Once the frame-level importance scores are obtained, we compute the shot-level importance score y_h by averaging the frame-level importance scores inside the same shot,

$$y_h = \frac{1}{n_h}\sum_{r=1}^{n_h} s_{h,r}, \quad (9)$$

where n_h is the length of the h^{th} shot, $s_{h,r}$ is the r^{th} frame-level importance score in the h^{th} shot. This strategy has been widely used in video summarization methods for converting between different formats of ground truth annotations [67]. Finally, for a fair comparison with previous methods, we also produce the summaries under the constraint that the total length of selected shots is no more than 15% of the original video length. Formally, we formalize this problem as,

$$\max \sum_{h=1}^{c} u_h y_h, \quad \text{s.t.} \sum_{h=1}^{c} u_h n_h \leq 15\% \times T, \quad (10)$$

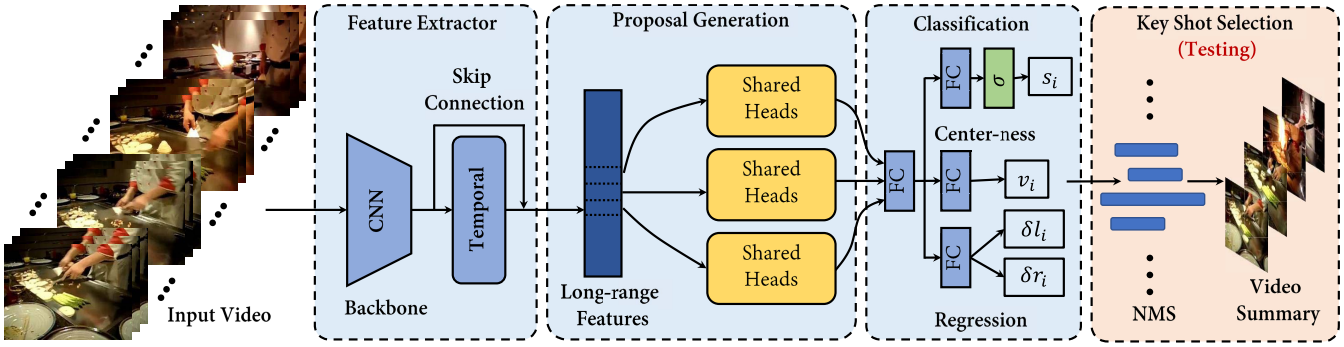


Fig. 4. The architecture of the proposed anchor-free DSNet. We first employ deep convolutional networks and a sequence modeling layer to extract long-range features. Then, we apply a shared classification and regression module to predict the importance score, center-ness score, and segment boundaries at each temporal location, respectively. For testing, segments are refined by using the predicted locations, and further filtered with non-maximum suppression. Finally, the video summary is generated by using a dynamic programming algorithm.

where $u_h \in \{0, 1\}$ indicates whether the h^{th} shot is selected in summaries, c is the number of shots, and T is the length of the video. Eq. (10) is the classic 0/1 knapsack problem. We apply a dynamic programming approach to solve this maximization problem. The final summaries are created by selecting shots with $u_h = 1$.

Discussions: We did not use the predicted segments to generate summaries. Since we set the tIoU threshold of positive proposals as 0.6, there must be many predicted segments of high overlaps. Direct voting of such relevant segments induces performance degradation as many similar segments will be selected. Therefore, we need consider the predicted scores and segment boundaries together. However, the predicted segment boundaries from location regression are less accurate than results produced by KTS, which are taken as ground truth locations. Hence, we adopt the key shot selection strategy to create summaries by using the refined shot boundaries and the predicted shot-level importance scores. The strategy meets the 15% budget constraint.

B. Anchor-Free Video Summarization

While the anchor-based DSNet approach is developed for temporal interest proposals to address the incorrect and incomplete segment issue, this method can be further extended to a more simple and flexible anchor-free framework. In this subsection, we elaborate the anchor-free DSNet approach.

The anchor-based framework has been applied into object detection [45], semantic segmentation [42], and action detection [27], [59]. However, there are several drawbacks for video summarization by using interest proposals. First, to obtain a high recall of ground truth segments, our anchor-based DSNet approach is required to densely sample interest proposals at each temporal location. But, most proposals are assigned as negative examples, leading to a severe class imbalance problem. Second, although multi-scale interest proposals cover most ground truth segments, the pre-defined nature is not suitable to handle complex and dynamic scenes. Third, labeling the positive and negative samples requires costly tIoU computation. Finally, the anchor-based approach needs subtle tuning for hyper-parameters relevant to temporal interest proposals, including the ratio of positive and negative examples, intervals of proposals, and the NMS threshold.

Inspired by prevalent spatial anchor-free works [44], [53], we propose the anchor-free DSNet. Fig. 4 presents the main architecture of the anchor-free DSNet approach. Specifically, for each temporal location, we avoid producing multi-scale temporal proposals for their importance score and location offset predictions. Instead, we directly predict the importance score, segment boundaries, and the center-ness score of each video frame after the long-range feature extraction. Our anchor-free approach includes feature extraction, segment prediction, and key shot selection steps. The feature extraction and key shot selection steps are the same as the anchor-based approach. Likewise, the annotated frame-level importance scores are converted to shot-level importance scores. The ground truth summaries are created via kernel temporal segmentation and the 0/1 knapsack algorithm by using the generated segments $C = \{(t_o^s, t_o^e, c_o)\}_1^{n_c}$, where t_o^s, t_o^e, and c_o are the starting time, the ending time, and the importance score of the o^{th} segment $\boldsymbol{C}_o$ and n_c is the number of segments in video summaries.

1) Feature Extraction: Given the i^{th} video sequence $\boldsymbol{V}_i$, we follow our anchor-based approach and extract representations $\{\boldsymbol{v}_j\}_1^{T_i}$ by applying GoogLeNet [52]. Then, we exploit a self-attention layer to learn long-range representations $\{\boldsymbol{w}_j\}_1^{T_i}$. The final representations $\{\boldsymbol{x}_j\}_1^{T_i}$ are obtained via the summation over $\boldsymbol{v}_j$ and $\boldsymbol{w}_j$, i.e., $\boldsymbol{x}_j = \boldsymbol{v}_j + \boldsymbol{w}_j$.

2) Segment Prediction: Unlike our anchor-based approach that predicts the offsets of pre-defined proposals at each temporal location, the anchor-free approach aims to directly learn the segment location and the importance score of each video frame. Specifically, for training, we view the j^{th} frame as a positive class when the j^{th} frame is selected in ground truth summaries. Otherwise, we assign a negative label to this frame. Since each frame only falls into a certain segment, the label assignment of video frames is unambiguous. Furthermore, for each positive frame, the anchor-free approach learns a ground truth 2D vector $\boldsymbol{\delta t}^* = (\delta l^*, \delta r^*)$, where δl^* and δr^* are the intervals between the current location and left and right

TABLE I
DESCRIPTIONS OF VIDEO SUMMARIZATION DATASETS USED IN OUR EXPERIMENTS

Dataset	#Number of videos	#User number	Content	Annotation type	Duration (Min,Max,Avg)
SumMe [12]	25	15-18	User generated videos	frame-level score	32s, 324s, 146s
TVSum [51]	50	20	Web videos	frame-level score	83s, 647s, 235s
OVP [4]	50	5	Various genre videos	key frames	83s, 647s, 235s
YouTube [4]	39	5	Web videos	key frames	83s, 647s, 235s

Algorithm 2 Training Process of Anchor-Free DSNet

Input: (videos $\{\boldsymbol{V}_i\}$, frame-level annotations $\{\boldsymbol{u}_i^*\}$)
Output: Importance scores $\boldsymbol{s}$, left and right boundaries $\boldsymbol{\delta t} = (\boldsymbol{\delta l}, \boldsymbol{\delta r})$, and the center-ness scores $\boldsymbol{v}$ of video frames; parameters $\boldsymbol{\theta}$ of the anchor-based model
$\boldsymbol{\theta} \leftarrow$ Parameters(DSNet)
for *epoch* $\in \{1, 2, ..., M\}$ **do**
  for *video* $\boldsymbol{V} \in \{\boldsymbol{V}_i\}$ *and annotation* $\boldsymbol{u}^* \in \{\boldsymbol{u}_i^*\}$ **do**
    % Apply KTS and knapsack
    $\boldsymbol{s}^* \leftarrow$ ToShotLevelAnnotation($\boldsymbol{u}^*$)
    for *frame* $\boldsymbol{I}_j \in \boldsymbol{V}$ **do**
      $\boldsymbol{v}_j \leftarrow$ GoogLeNet($\boldsymbol{I}_j$)
      $s_j^* \leftarrow$ AssignLabel(j,$\boldsymbol{s}^*$)
      $\boldsymbol{\delta t}_j^* \leftarrow$ ComputeBoundary(j,$\boldsymbol{s}^*$) *% Eq. (11)*
      $v_e^* \leftarrow$ ComputeCenterness($\boldsymbol{\delta t}_j^*$) *% Eq. (13)*
    end
    $\boldsymbol{w} \leftarrow$ ExtractTemporalFeature($\boldsymbol{v}$)
    $\boldsymbol{x} \leftarrow \boldsymbol{w} + \boldsymbol{v}$
    $\boldsymbol{s} \leftarrow$ ClassificationBranch($\boldsymbol{x}$) *% importance scores*
    $\boldsymbol{v} \leftarrow$ CenternessBranch($\boldsymbol{x}$) *% center-ness scores*
    $\boldsymbol{\delta t} \leftarrow$ RegressionBranch($\boldsymbol{x}$) *% intervals of boundary*
    $\mathcal{L} \leftarrow \mathcal{L}(\boldsymbol{s}, \boldsymbol{s}^*, \boldsymbol{\delta t}, \boldsymbol{\delta t}^*, \boldsymbol{v}, \boldsymbol{v}^*)$ *% the loss in Eq. (14)*
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta\, \partial\mathcal{L}/\partial\boldsymbol{\theta}$
  end
end

boundaries of the associated segment $\boldsymbol{C}_o$, respectively, i.e.,

$$\delta l^* = j - t_o^s, \quad \delta r^* = t_o^e - j. \tag{11}$$

We utilize the exp(·) function to guarantee predicted results to be positive. Different from the anchor-based approach that only leverages interest proposals of high tIoU, our anchor-free approach is capable of optimizing our network by using all positive locations inside the summary.

We apply the focal loss $\mathcal{L}_{cls}$ [31] for importance classification, which handles the class imbalance issue by down-weighting losses for well-classified samples, and we exploit the tIoU loss $\mathcal{L}_{reg}$ for location regression, which is robust to temporal interests of varied intervals. Formally, the training loss is defined as,

$$\mathcal{L} = \frac{1}{N_{\text{pos}}} \sum_j \mathcal{L}_{cls}(s_j, s_j^*) + \frac{\lambda}{N_{\text{pos}}} \sum_e \mathcal{L}_{reg}(\boldsymbol{\delta t}_e, \boldsymbol{\delta t}_e^*), \tag{12}$$

where s_j and s_j^* are the j^{th} predicted and ground truth frame-level scores, $\boldsymbol{\delta t}_e$ and $\boldsymbol{\delta t}_e^*$ represent the predicted and ground truth locations of the e^{th} positive sample. λ balances the classification and regression losses.

Since many positive temporal locations are close to boundaries of the corresponding ground truth segments, our method will generate many low-quality segments. To address this issue, we utilize a center-ness constraint to make sure that the temporal location is close to the center of the predicted segment. The ground truth center-ness score is defined as,

$$v_e^* = \frac{\min(\delta l^*, \delta r^*)}{\max(\delta l^*, \delta r^*)}. \tag{13}$$

We utilize the binary cross entropy (BCE) loss $\mathcal{L}_{center}$ with the balanced weight μ for center-ness scores. Finally, we sum three losses together. The objective is formally written as,

$$\mathcal{L}^* = \mathcal{L} + \frac{\mu}{N_{\text{pos}}} \sum_e \mathcal{L}_{center}(v_e, v_e^*). \tag{14}$$

Similar to the anchor-based classification and regression module that is shown in Fig. 3, the anchor-free approach also contains a shared fully connected layer and two separate branches for importance classification and location regression. Besides, the center-ness branch is added for the center-ness score regression, which has the same structure as the regression branch expect the output dimension. We also summarize the training procedure of the anchor-free approach in **Algorithm 2**. In the training stage, we optimize parameters of the anchor-free model by using the multi- task loss in Eq. (14).

3) Key Shot Selection: For testing, we first obtain the importance score s_j, the location prediction δl and δr, and the center-ness score v_j of each temporal location by employing the training model. Then, we compute the starting and ending times t_o^s and t_o^e of each predicted segment as follows,

$$t_o^s = j - \delta l, \quad t_o^e = j + \delta r, \tag{15}$$

where j represents the temporal index of the video frame. and its confidence score is computed as $c_o = s_j \times v_j$, which indicates that a good segment should have a high importance score and meanwhile the associated location should be at the central portion of the segment. Due to high overlaps and low confidence of predicted segments, we filter out redundant and low-quality segments via the non-maximum suppression algorithm. Afterwards, we adopt the frame-level importance score assignment strategy that is the same as the anchor-based method and we convert frame-level importance scores to shot-level importance scores. Finally, the 0/1 knapsack algorithm is applied to select video shots.

IV. EXPERIMENTS

A. Datasets and Protocols

1) Datasets: We evaluated the performance of our anchor-based and anchor-free approaches on the SumMe [12] dataset and the TVSum [51] dataset. The SumMe dataset totally consists of 25 video sequences, covering various genres such

as holidays, cooking and sports. The TVSum dataset is composed of 50 video sequences downloaded from YouTube with 10 categories, including changing vehicle tire, parade, and dog show. Both datasets provided multiple user annotations. We used another two datasets, i.e., OVP [4] and YouTube [4], to augment the training datasets. The OVP dataset has 50 video sequences and the YouTube dataset consists of 39 video sequences excluding the cartoon videos. Specifically, we followed previous works and downsampled all videos, originally captured at 30 fps, to 2 fps to handle temporal redundancy and reduce computation. Table I showcases detailed descriptions of four video summarization datasets.

For the proposal classification and location regression, the ground truth segments are required. However, both SumMe and TVSum datasets only provide frame-level importance scores. Hence, we followed conventional methods and applied KTS to segment videos into serveral shots, where the shot-level importance scores were computed by Eq. (9). Then, the knapsack algorithm was applied to produce the key shot based summaries. For the additional OVP and YouTube datasets, a collection of keyframes was provided. We first converted their keyframes to key shot candidates when a segment contains a keyframe, and then the knapsack algorithm was applied to meet the constraint of 15% of the original video length [67].

We exploited three evaluation settings to assess the performance of the proposed method, i.e., canonical, augmented, and transfer settings. For canonical and augmented settings, we randomly divided the dataset into 5 splits. In the canonical setting, 80% of the dataset was used for training, and the remaining 20% was used for evaluation. However, in the augmented setting, 80% of the dataset augmented with another three datasets was used for training. In the transfer setting, three datasets were used for training and the remaining one dataset was used for evaluation. In experiments, we set the canonical setting as the default setting. We ran our method five times for each setting and reported the average performance of these five runs.

2) Evaluation Metrics: We exploited the F_β-measure to evaluate the agreement between the generated summaries and the human-created summaries. For the i^{th} generated summary $\boldsymbol{gs}_i$ and corresponding annotated summary $\boldsymbol{gt}_i$, the precision p_i and recall r_i were computed as,

$$p_i = \frac{\text{length}\ \left(\boldsymbol{gs}_i \cap \boldsymbol{gt}_i\right)}{\text{length}\left(\boldsymbol{gs}_i\right)}, \quad r_i = \frac{\text{length}\left(\boldsymbol{gs}_i \cap \boldsymbol{gt}_i\right)}{\text{length}\left(\boldsymbol{gt}_i\right)}. \tag{16}$$

Specifically, the F_β-measure was computed as follows,

$$F_\beta = \frac{\left(1+\beta^2\right) \times p_i \times r_i}{\left(\beta^2 \times p_i\right) + r_i}, \tag{17}$$

we adopted the harmonic mean F_1-measure ($\beta = 1$) as the default F-score result in our experiments. Following the evaluation protocol of SumMe and TVSum [46], [67], we evaluated the quality of a predicted summary by computing the F-score between the generated summary and its corresponding summaries created by multiple users for each video.

TABLE II

COMPARISONS OF F-SCORE (%) AND PARAMETERS (MILLION) WITH STATE-OF-THE-ART VIDEO SUMMARIZATION METHODS ON THE SUMME AND TVSUM DATASETS UNDER THE CANONICAL SETTING

Method	SumMe	TVSum	Params
Video MMR [30]	26.6	-	-
LiveLight [74]	-	46.0	-
ERSUM [29]	43.1	59.4	-
MSDS-CC [35]	40.6	52.3	-
vsLSTM [67]	37.6	54.2	2.63
dppLSTM [67]	38.6	54.7	2.63
SUM-GAN$_{dpp}$ [34]	39.1	51.7	295.86
SUM-GAN$_{sup}$ [34]	41.7	56.3	295.86
A-AVS [18]	43.9	59.4	4.40
M-AVS [18]	44.4	61.0	4.40
SASUM [55]	40.6	53.9	44.07
SASUM$_{sup}$ [55]	45.3	58.2	44.07
DR-DSN [76]	41.4	57.6	2.63
DR-DSN$_{sup}$ [76]	42.1	58.1	2.63
TS-STN [16]	46.1	60.0	16.18
FCSN [47]	48.8	58.4	-
VASNet [8]	49.7	61.4	7.35
Ours	**50.2**	**62.1**	**8.53**

B. Experiments on Anchor-Based DSNet

In this subsection, we describe implementation details and present experimental results and analyses of the anchor-based DSNet approach on the SumMe and TVSum datasets.

1) Implementation Details: For visual representations, we extracted 1024 dimensional features from outputs of the pool5 layer in GoogLeNet, pre-trained on ImageNet. In addition, the anchor-based model contains a multi-head self-attention layer [54] with 8 heads, a layer normalization, a fully-connected layer with a dropout layer (0.5) and a tanh activation function, followed by two output fully-connected layers. Details of the multi-head self-attention layer can be found in [54]. The dimensions of hidden states in the self-attention layer and the first fully-connected layer are 1024 and 128, respectively. For parameters, the value of hyper-parameter λ in Eq. (5) was set as 1, and the threshold of non-maximum suppression was set as 0.5. We trained our anchor-based model over 300 epochs by using Adam optimizer with a base learning rate of 5×10^{-5} and a weight decay of 10^{-5}. We conducted our experiments on a Nvidia GTX 1080Ti GPU and implemented our method by using PyTorch.

2) Comparison With State-of-the-Art Methods: We compared the anchor-based DSNet approach with state-of-the-art video summarization methods on SumMe and TVSum. Comparison methods can be classified into two categories: 1) conventional methods including Video MMR [30], LiveLight [74], ERSUM [29], MSDS-CC [35], and 2) deep learning based methods including vsLSTM [67], dppLSTM [67], SUM-GAN [34], AVS [18], SASUM [55], DR-DSN [76], TS-STN [16], FCN [47], and VASNet [8].

Table II tabulates experimental results and the parameter number of different video summarization methods under the

TABLE III

COMPARISONS OF F-SCORE (%) WITH STATE-OF-THE-ART VIDEO SUMMARIZATION METHODS ON THE SUMME AND TVSUM DATASETS UNDER THE CANONICAL (C), AUGMENTED (A) AND TRANSFER (T) SETTINGS, RESPECTIVELY

Method	SumMe			TVSum		
	C	A	T	C	A	T
vsLSTM [67]	37.6	41.6	40.7	54.2	57.9	56.9
dppLSTM [67]	38.6	42.9	41.8	54.7	59.6	58.7
SUM-GAN [34]	41.7	43.6	-	56.3	61.2	-
DR-DSN [76]	42.1	43.9	42.6	58.1	59.8	58.9
A-AVS [18]	43.9	44.6	-	59.4	60.8	-
M-AVS [18]	44.4	46.1	-	61.0	61.8	-
FCSN [47]	48.8	50.2	45.0	58.4	59.1	57.4
Ours	**50.2**	**50.7**	**46.5**	**62.1**	**63.9**	**59.4**

TABLE IV

COMPARISONS OF F-SCORE (%) BY USING DIFFERENT TEMPORAL MODELING LAYERS ON THE SUMME AND TVSUM DATASETS

Method	SumMe			TVSum		
	C	A	T	C	A	T
LSTM	48.5	49.6	45.4	60.6	60.5	56.8
Bi-LSTM	48.5	50.5	45.6	60.8	60.5	56.6
GCN [24]	**50.5**	50.2	**46.6**	61.7	61.3	57.9
Attention [54]	50.2	**50.7**	46.5	**62.1**	**63.9**	**59.4**

canonical setting. We clearly observe that our method yields the best performance, surpassing current state-of-the-art methods by at least 0.5% on both the SumMe and TVSum datasets. Moreover, we observe that our method achieves a good balance between F-score and parameters. While vsLSTM, dppLSTM, and DR-DSN contain the least number of parameters by only employing a bidirectional LSTM, their performance is at least 4% behind that of our method on SumMe and TVSum.

a) Augmentation and transfer experiments: To further validate the effectiveness of the proposed method, we conducted experiments under the augmented and transfer settings on the SumMe and TVSum datasets along with the additional OVP and YouTube datasets. Detailed descriptions about these two settings are described in the datasets subsection.

Table III presents experimental results of state-of-the-art methods and our method. We clearly observe that our method consistently achieves the best performance against the other methods under the augmented and transfer settings. Given that the lack of annotated video data may impair the capacity of supervised methods, we compared different methods under the augmented setting to alleviate the influence of overfitting. We see that all methods under this setting achieve a better performance compared with the performance under the canonical and transfer settings, which indicates that more training videos are beneficial to performance improvements. We also conducted experiments by using the more challenging transfer setting. We see that our method outperforms the previous state-of-the-art methods, proving the effectiveness of the proposed method across videos from different datasets.

TABLE V

COMPARISONS OF F-SCORE (%) WITH (✓) OR WITHOUT (✗) TEMPORAL AVERAGE POOLING LAYER ON THE SUMME AND TVSUM DATASETS

Pooling	SumMe			TVSum		
	C	A	T	C	A	T
✗	**51.4**	**52.3**	44.9	61.2	61.9	56.7
✓	50.2	50.7	**46.5**	**62.1**	**63.9**	**59.4**

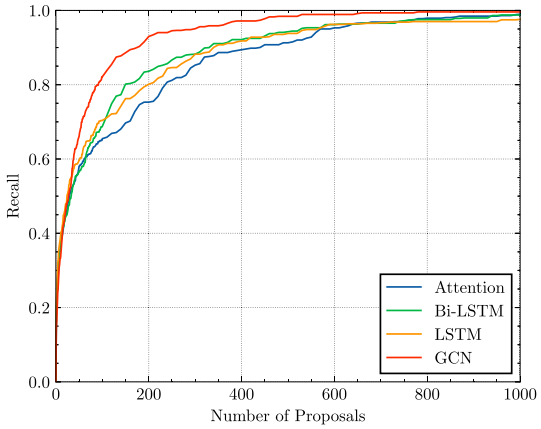


Fig. 5. The recall curve for selected proposals on the TVSum dataset.

b) Evaluation of long-range features: To analyze the effect of long-range features with different feature extraction layers, we replaced the default self-attention layer with a LSTM layer, a bidirectional LSTM layer, or a graph convolutional layer [24], respectively. Following the self-attention layer setting, we set the dimension of outputs in different layers as 1024. Since we concatenated the hidden states of both directions in Bi-LSTM, its hidden dimension was set as 512.

Table IV tabulates the performance of the self-attention layer with different feature extraction layers on the SumMe and TVSum datasets. We clearly observe that our models with different feature extraction layers achieve very competitive results against state-of-the-art methods.

c) Evaluation of temporal pooling layers: Since we applied a temporal average pooling layer on proposal features of arbitrary length, we investigated the influence of the temporal average pooling layer in the classification and regression module. Table V shows experimental results of our method with or without the pooling layer on the SumMe and TVSum datasets. We see that the temporal pooling mechanism improves the performance on both datasets especially under the transfer setting, and it can enhance the robustness of our model, due to the effective utilization of temporal local information.

d) NMS threshold study: We conducted ablation experiments on the SumMe and TVSum datasets to analyze the effect of the NMS threshold. Since a high threshold may filter out high-quality segments while a low threshold may introduce low-quality segments, the NMS threshold significantly affects the performance of our method. Fig. 6 and Fig. 7 present

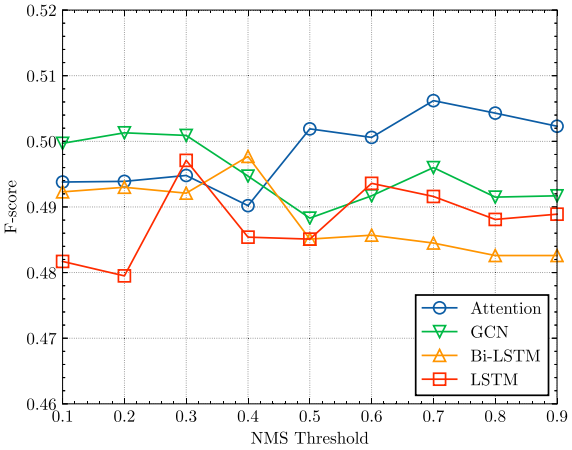


Fig. 6. Parameter analysis of the NMS threshold on the SumMe dataset in the anchor-based approach. We set the default value as 0.5.

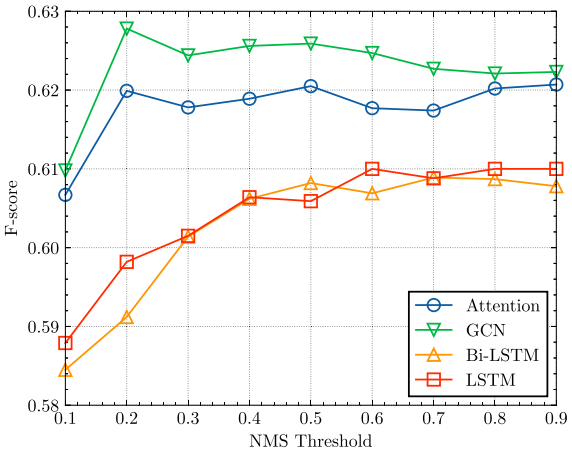


Fig. 7. Parameter analysis of the NMS threshold on the TVSum dataset in the anchor-based approach. We set the default value as 0.5.

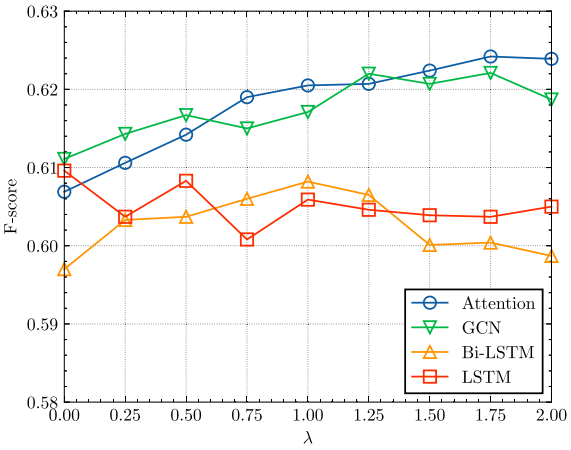


Fig. 8. Parameter analysis of λ on the TVSum dataset in the anchor-based approach. We set the default value as 1.00.

TABLE VI

COMPARISONS OF THE F-SCORE (%) WITH STATE-OF-THE-ART VIDEO SUMMARIZATION METHODS ON THE SUMME AND TVSUM DATASETS UNDER THE CANONICAL (C), AUGMENTED (A) AND TRANSFER (T) SETTINGS

Method	SumMe			TVSum		
	C	A	T	C	A	T
vsLSTM [67]	37.6	41.6	40.7	54.2	57.9	56.9
dppLSTM [67]	38.6	42.9	41.8	54.7	59.6	58.7
SUM-GAN [34]	41.7	43.6	-	56.3	61.2	-
DR-DSN [76]	42.1	43.9	42.6	58.1	59.8	**58.9**
A-AVS [18]	43.9	44.6	-	59.4	60.8	-
M-AVS [18]	44.4	46.1	-	61.0	61.8	-
FCSN [47]	48.8	50.2	45.0	58.4	59.1	57.4
Ours	**51.2**	**53.3**	**47.6**	**61.9**	**62.2**	58.0

experimental results of the anchor-based method with different NMS thresholds. We observe that the performance of our method on the TVSum dataset becomes stable with a threshold value around 0.5. To simultaneously guarantee a high recall and a high accuracy, we both set the NMS thresholds on the SumMe and TVSum datasets as 0.5.

e) Parameter analysis: We also analyzed the effect of the parameter λ in Eq. (5), which trades off the classification and regression losses. It is noteworthy that if the parameter $\lambda = 0$, the regression item is removed from the loss function. Fig. 8 presents experimental results on the TVSum dataset with different λ. We see that our method obtains an inferior performance when $\lambda = 0$, which suggests that the regression item is beneficial to improve the summary performance by providing refined proposals. We set the value of λ as 1.0 for equal importance of the classification and regression branches.

f) Recall analysis: To guarantee a high recall of ground truth segments, we depicted the recall curve about refined proposals. Fig. 5 presents the recall results with different number of proposals. Table I shows that videos in the TVSum dataset consist of at less 83 seconds and 166 frames. Each temporal location generates 4 proposals and totally 664 proposals. Therefore, our method can achieve a recall over 95% of ground truth segments with different long-range temporal layers.

C. Experiments on Anchor-Free DSNet

1) Implementation Details: We extracted 1024 dimensional features from GoogLeNet and the number of heads in self-attention layer was set as 8. In Eq. (14), the balance parameters λ and μ were set as 1. For the focal loss, α and γ were set as 0.25 and 2, respectively. Moreover, we trained our anchor-free model over 300 epochs by using the Adam optimizer with a base learning rate of 5×10^{-5} and a weight decay of 1×10^{-5}. The NMS threshold was set as 0.4.

2) Comparison With State-of-the-Art Methods: To validate the effectiveness of the proposed anchor-free method, we compared our method with state-of-the-art methods under three dataset settings including canonical, augmented and transfer settings. Table VI presents experimental results by using different video summarization methods on the SumMe and TVSum datasets. We clearly observe that our anchor-free method outperforms the other state-of-the-art methods

TABLE VII

EFFECT OF THE CENTER-NESS LOSS AND EXP EMBEDDING ON SUMME AND TVSUM WITH F-SCORE (F), PRECISION (P) AND RECALL (R)

Method	SumMe			TVSum		
	F	P	R	F	P	R
w/o exp	51.6	51.3	52.0	61.8	61.8	61.7
w/o center-ness	49.0	48.5	49.8	61.5	61.5	61.4
ours	51.2	50.8	51.9	61.9	61.9	61.9

TABLE VIII

EFFECT OF THE DIFFERENT LOSS FUNCTIONS ON SUMME AND TVSUM INCLUDING CROSS-ENTROPY (CROSS-ENT.), FOCAL, TIOU, AND SMOOTH$_{\mathcal{L}_1}$ LOSSES

Cross-Ent.	*Focal*	*tIoU*	*smooth*$_{\mathcal{L}_1}$	SumMe	TvSum
✓	✗	✗	✓	48.3	60.7
✗	✓	✗	✓	48.3	61.1
✓	✗	✓	✗	49.1	60.5
✗	✓	✓	✗	51.2	61.9

on both datasets except DR-DSN on the TVSum dataset under the transfer setting. Compared with our anchor-based method in Table III, our anchor-free method obtains a better performance on SumMe. However, the F-score of the anchor-based method on TVSum is higher than that of the anchor-free method. The reason may be that durations of ground truth segments on TVSum tend to be longer than those on SumMe, and it is much more easier for the anchor-based method to deal with long segments with pre-defined multi-scale proposals.

a) Evaluation of the center-ness loss and exp embedding: To analyze the effect of the center-ness branch and exp embedding in the regression branch, we conducted experiments without (w/o) the center-ness loss or exp$(\cdot)$ embedding. Table VII shows experimental results including F-score, precision, and recall by using these two settings. We observe that without the center-ness branch, our anchor-free approach induces performance degradation on the SumMe and TVSum datasets. A comparable performance is observed without the exp mapping that promises outputs in the regression branch to be positive.

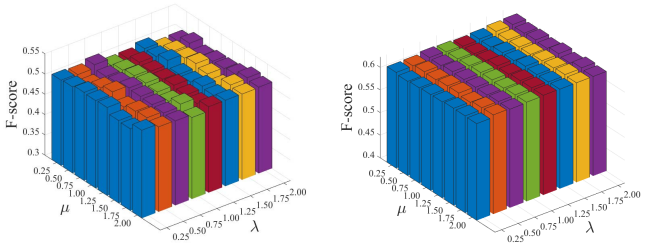
b) Comparisons of the classification and regression losses: The loss functions play a crucial role in training our model and we compared the focal loss with the cross-entropy loss in the classification branch and the tIou loss with the smooth$_{\mathcal{L}_1}$ loss in the regression branch. Table VIII tabulates experimental results by using different loss functions. We observe that our method with the focal and tIoU losses outperforms the other alternative settings on both datasets. A possible reason is that the focal loss addresses the class imbalance issue by automatically decreasing the contributions of well-classified examples compared with the cross-entropy loss, and the tIoU loss is more robust than the smooth$_{\mathcal{L}_1}$ loss.

c) Evaluation of long-range features: Similar to the anchor-based approach, we investigate the effect of long-range features by using different feature extraction layers. Table IX presents F-score, precision, and recall results with different

TABLE IX

EFFECT OF LONG-RANGE FEATURES BY USING DIFFERENT TEMPORAL MODELING LAYERS ON THE SUMME AND TVSUM DATASETS

Method	SumMe			TVSum		
	F	P	R	F	P	R
LSTM	49.5	48.7	51.2	59.8	59.8	59.8
Bi-LSTM	48.6	48.4	48.8	60.5	60.4	60.5
GCN [24]	50.5	50.0	51.3	59.8	59.8	59.8
Attention [54]	**51.2**	**50.8**	**51.9**	**61.9**	**61.9**	**61.9**



(a) SumMe (b) TVSum

Fig. 9. F-score results of different λ and μ values on the SumMe and TVSum datasets in the anchor-free approach.

long-range features. We observe that our anchor-free method with the attention layer obtains the best performance against the other temporal layers on both datasets.

d) Parameter analysis: We also provided comparisons by using different parameter values to evaluate parameter sensitivity. Note that we adopted a simple grid search strategy to tune parameters λ and μ in the range of [0.25, 2.00] with an interval of 0.25. The grid search strategy uniformly samples parameter λ and μ values within the specified value range. Therefore, the value sets of parameters λ and μ are {0.25, 0.5, 0.75, 1.00, 1.25, 1.50, 1.75, 2.00} and {0.25, 0.5, 0.75, 1.00, 1.25, 1.50, 1.75, 2.00}, respectively. Trials are formed by assembling every possible combination of these values and thus there are 64 trials in total. Fig. 9 visualizes experimental results with different λ and μ on the SumMe (Fig. 9(a)) and TVSum (Fig. 9(b)) datasets. We observe that our method is not sensitive to parameters λ and μ within the range. For simplicity, we both set λ and μ values as 1.00 in experiments.

e) NMS threshold study: Since NMS filters out low-quality and redundant segments and is very important to final results, we further conducted analyses of the NMS threshold on the SumMe and TVSum datasets. Fig. 11 shows experimental results by using different NMS thresholds. We see that when the NMS threshold is 0.4, our anchor-free method achieves the best performance. Therefore, we set the default value of the NMS threshold as 0.4 in our experiments.

f) Effect of temporal continuity and integrity constraints: We conducted experiments to demonstrate the effectiveness of temporal continuity and integrity constraints. For the baseline, we removed the interest proposal formulation and only applied a self-attention layer to predict the importance scores. Table X presents experimental results by using the self-attention layer on SumMe and TVSum. Specifically, Anchor-based$_{\lambda=0}$ indicates that the parameter λ in Eq. (5) is set as 0 and the

Fig. 10. Qualitative results of different video summarization methods, including the ground truth, our anchor-based method, our anchor-free method, DR-DSN [76], dppLSTM [67], and VASNet [8]. The x-axis represents the frame index and the line segments denote the selected segments. Frames are uniformly sampled and shown below the summary results and the selected frames are highlighted via red boundary boxes.

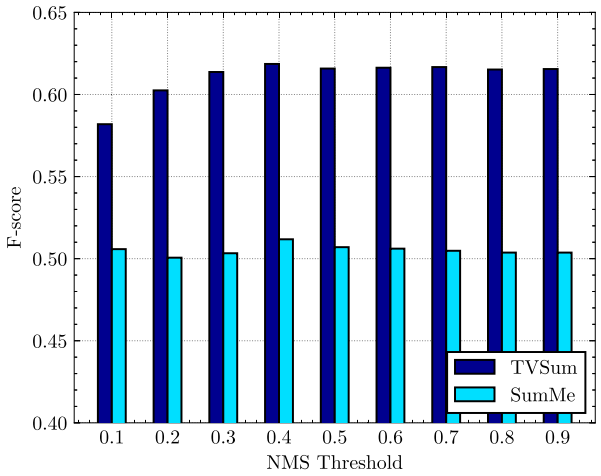


Fig. 11. Parameter analysis of the NMS threshold in the anchor-free approach.

anchor-based method produces interest proposals but does not refine these proposals. Anchor-free$_{\lambda=0,\mu=0}$ adopts the same setting without refined proposals. We observe that only producing interest proposals without refining these proposals, our Anchor-based$_{\lambda=0}$ and Anchor-free$_{\lambda=0,\mu=0}$ approaches provide the inferior performance on SumMe. Moreover, our anchor-based and anchor-free methods with temporal continuity and integrity constraints achieve the superior performance, which

TABLE X

COMPARISONS OF DIFFERENT SETTINGS WITH (✓) OR WITHOUT (✗) THE INTEREST PROPOSALS (I) AND REFINED PROPOSALS (R) CONSTRAINTS BY USING THE SELF-ATTENTION LAYER

Method	I	R	SumMe	TVSum
Baseline	✗	✗	48.8	59.6
Anchor-based$_{\lambda=0}$	✓	✗	47.4	60.7
Anchor-based	✓	✓	50.2	62.1
Anchor-free$_{\lambda=0,\mu=0}$	✓	✗	46.5	61.7
Anchor-free	✓	✓	51.2	61.9

are 1.4% and 2.4% relative improvements on SumMe, 2.5% and 2.3% relative improvements on TVSum over the baseline.

g) Diversity analysis: Diversity is a key property of video summarizes. We followed experimental settings in [76], and used its diversity metric to evaluate the diversity of generated summaries on the SumMe and TVSum datasets. A more diverse summary corresponds to a higher diversity score. Table XI shows the diversity scores by using different methods. Both dppLSTM and DR-DSN exploited the diversity constraints. We observe that our anchor-based and anchor-free approaches obtain higher scores compared with dppLSTM and DR-DSN.

h) Runtime analysis: Compared with the anchor-based method, our anchor-free method avoids generating interest proposals and saves more running time. To validate the efficiency of our anchor-based method, we recorded the inference

TABLE XI

THE DIVERSITY SCORES OF GENERATED SUMMARIES ON THE SUMME AND TVSUM DATASETS

Dataset	dppLSTM	DR-DSN	Anchor-based	Anchor-free
SumMe	0.591	0.594	0.642	0.664
TVSum	0.463	0.464	0.476	0.477

TABLE XII

THE AVERAGE INFERENCE TIME (ms) AND AVERAGE FRAMES (×15) FOR PER VIDEO BY USING OUR ANCHOR-BASED AND ANCHOR-FREE METHODS

	Method	SumMe	TVSum
average frames	-	293	470
average time	Anchor-based	17.25	31.18
	Anchor-free	6.13	6.73

times of our anchor-based and anchor-free methods after feature extraction by using GoogLeNet. Table XII presents the average inference time on the SumMe and TVSum datasets. We observe that our anchor-free method is more efficient than our anchor-based method, speeding up the inference stage more than 2.8× on SumMe and 4.6× on TVSum.

i) Qualitative results: We also provided qualitative results to intuitively evaluate the performance of different video summarization methods on the 25^{th} video of the SumMe dataset and on the 24^{th} and 42^{nd} videos of the TVSum dataset. Fig. 10 visualizes the example summaries generated by different video summarization methods as well as human-created summaries. The first video (Fig. 10(a)) is about playing a ball, the second video (Fig. 10(b)) is related to parkour, and the third video (Fig. 10(c)) shows an interview with a motorcyclist and the riding show. We clearly observe that both the anchor-based and anchor-free methods produce segments that have high overlaps with ground truth segments. Moreover, the anchor-free approach compares favorably with the anchor-based approach due to the interest proposal formulation. Besides, although the video on SumMe has few representative segments and many segments are chosen from videos on TVSum, our anchor-based and anchor-free methods avoid selecting unrepresentative segments on SumMe and excluding key segments on TVSum. The generated summaries are perfectly consistent with the ground truth summaries. The qualitative results intuitively demonstrate the effectiveness of the proposed methods.

D. Analysis

According to the above experimental results, several key observations are summarized:

1) Different from existing supervised video summarization methods without temporal continuity and integrity constraints, the anchor-based DSNet regards video summarization as an interest detection problem, and achieves very promising performance on two widely-used datasets.
2) The anchor-based DSNet approach conducts dense sampling of multi-scale interest proposals to accommodate various durations of segments and exploits negative samples to deal with incorrect and incomplete segments.
3) Based on the anchor-based DSNet, we further propose the anchor-free DSNet, which eliminates the predefined interest proposals in the anchor-based DSNet by directly learning importance scores, segment locations, and center-ness scores.
4) Moreover, the anchor-free approach can be viewed as a special anchor-based approach. Differently, there is only one single 'anchor' at each temporal location in the anchor-free approach that is of flexible length and is unsymmetrical, meaning that the left and right boundaries of an 'anchor' are unfixed and can dynamically change. Obviously, we need another a center-ness constraint to restrict the flexibility of these 'anchors'.
5) Our anchor-free DSNet achieves comparable performance against the anchor-based DSNet due to the similar interest proposal formulation.

V. CONCLUSION

In this paper, we have proposed a Detect-to-Summarize network framework for video summarization, including anchor-based and anchor-free approaches. Unlike existing supervised methods which only learn the importance score of each frame, our anchor-based DSNet approach formulates video summarization as an interest detection problem and simultaneously learns importance scores and location offsets of generated interest proposals, handling incorrect and incomplete segments. To eliminate the drawbacks of interest proposals, we further propose the anchor-free DSNet approach to directly predict the importance scores and segment boundaries. The proposed anchor-based and anchor-free DSNet approaches outperform most state-of-the-art supervised methods on the widely-used SumMe and TVSum datasets. In the future, we will attempt to incorporate key shot selection into a unified framework.

REFERENCES

[1] S. Cai, W. Zuo, L. S. Davis, and L. Zhang, "Weakly-supervised video summarization using variational encoder-decoder and Web prior," in *Proc. ECCV*, Sep. 2018, pp. 184–200.
[2] Y.-W. Chao, S. Vijayanarasimhan, B. Seybold, D. A. Ross, J. Deng, and R. Sukthankar, "Rethinking the faster R-CNN architecture for temporal action localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1130–1139.
[3] W.-S. Chu, Y. Song, and A. Jaimes, "Video co-summarization: Video summarization by visual co-occurrence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3584–3592.
[4] S. E. F. de Avila, A. P. B. Lopes, A. da Luz, and A. de Albuquerque 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.
[5] E. Elhamifar and M. C. De Paolis Kaluza, "Online summarization via submodular and convex optimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1818–1826.
[6] E. Elhamifar, G. Sapiro, and S. S. Sastry, "Dissimilarity-based sparse subset selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 11, pp. 2182–2197, Nov. 2016.
[7] 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.
[8] J. Fajtl, H. S. Sokeh, V. Argyriou, D. Monekosso, and P. Remagnino, "Summarizing videos with attention," in *Proc. ACCV*, 2018, pp. 39–54.

[9] T.-J. Fu, S.-H. Tai, and H.-T. Chen, "Attentive and adversarial learning for video summarization," in *Proc. WACV*, Jan. 2019, pp. 1579–1587.

[10] J. Gao, Z. Yang, K. Chen, C. Sun, and R. Nevatia, "Turn tap: Temporal unit regression network for temporal action proposals," in *ICCV*, Oct. 2017, pp. 3628–3636.

[11] B. Gong, W.-L. Chao, K. L. Grauman, and F. Sha, "Diverse sequential subset selection for supervised video summarization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 3, 2014, pp. 2069–2077.

[12] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *Proc. ECCV*, 2014, pp. 505–520.

[13] M. Gygli, H. Grabner, and L. Van Gool, "Video summarization by learning submodular mixtures of objectives," in *Proc. CVPR*, Jun. 2015, pp. 3090–3098.

[14] Y. Hadi, F. Essannouni, and R. O. H. Thami, "Video summarization by k-medoid clustering," in *Proc. SAC*, 2006, pp. 1400–1401.

[15] C. Huang and H. Wang, "Novel key-frames selection framework for comprehensive video summarization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 2, pp. 577–589, Feb. 2019.

[16] S. Huang, X. Li, Z. Zhang, F. Wu, and J. Han, "User-ranking video summarization with multi-stage spatio–temporal representation," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2654–2664, Jun. 2019.

[17] T. Hussain, K. Muhammad, A. Ullah, Z. Cao, S. W. Baik, and V. H. C. de Albuquerque, "Cloud-assisted multiview video summarization using CNN and bidirectional LSTM," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 77–86, Jan. 2020.

[18] 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.

[19] Z. Ji, Y. Zhao, Y. Pang, X. Li, and J. Han, "Deep attentive video summarization with distribution consistency learning," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, May 11, 2020, doi: 10.1109/TNNLS.2020.2991083.

[20] A. Kanehira, L. Van Gool, Y. Ushiku, and T. Harada, "Viewpoint-aware video summarization," in *Proc. CVPR*, Jun. 2018, pp. 7435–7444.

[21] A. Khosla, R. Hamid, C. Lin, and N. Sundaresan, "Large-scale video summarization using Web-image priors," in *Proc. CVPR*, Jun. 2013, pp. 2698–2705.

[22] G. Kim and E. P. Xing, "Reconstructing storyline graphs for image recommendation from Web community photos," in *Proc. CVPR*, Jun. 2014, pp. 3882–3889.

[23] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. ICLR*, 2014, pp. 1–14.

[24] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. ICLR*, 2017, pp. 1–13.

[25] A. Kulesza, "Determinantal point processes for machine learning," *Found. Trends Mach. Learn.*, vol. 5, nos. 2–3, pp. 123–286, 2012.

[26] Y. Jae Lee, J. Ghosh, and K. Grauman, "Discovering important people and objects for egocentric video summarization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1346–1353.

[27] Y. J. Lee and K. Grauman, "Predicting important objects for egocentric video summarization," *Int. J. Comput. Vis.*, vol. 114, no. 1, pp. 38–55, Aug. 2015.

[28] X. Li, H. Li, and Y. Dong, "Meta learning for task-driven video summarization," *IEEE Trans. Ind. Electron.*, vol. 67, no. 7, pp. 5778–5786, Jul. 2020.

[29] 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.

[30] Y. Li and B. Merialdo, "Multi-video summarization based on video-MMR," in *Proc. WIAMIS*, Apr. 2010, pp. 1–4.

[31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[32] S. Lu, Z. Wang, T. Mei, G. Guan, and D. D. Feng, "A bag-of-importance model with locality-constrained coding based feature learning for video summarization," *IEEE Trans. Multimedia*, vol. 16, no. 6, pp. 1497–1509, Oct. 2014.

[33] Q. Luan, M. Song, C. Y. Liau, J. Bu, Z. Liu, and M.-T. Sun, "Video summarization based on nonnegative linear reconstruction," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2014, pp. 1–6.

[34] 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. 202–211.

[35] 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.

[36] P. Mundur, Y. Rao, and Y. Yesha, "Keyframe-based video summarization using delaunay clustering," *Int. J. Digit. Libraries*, vol. 6, no. 2, pp. 219–232, 2006.

[37] M. Otani, Y. Nakashima, E. Rahtu, and J. Heikkila, "Rethinking the evaluation of video summaries," in *Proc. CVPR*, Jun. 2019, pp. 7596–7604.

[38] 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.

[39] R. Panda, A. Das, Z. Wu, J. Ernst, and A. K. Roy-Chowdhury, "Weakly supervised summarization of Web videos," in *Proc. ICCV*, Oct. 2017, pp. 3677–3686.

[40] R. Panda and A. K. Roy-Chowdhury, "Collaborative summarization of topic-related videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4274–4283.

[41] R. Panda and A. Roy-Chowdhury, "Multi-view surveillance video summarization via joint embedding and sparse optimization," *IEEE Trans. Multimedia*, vol. 19, no. 9, pp. 2010–2021, Sep. 2017.

[42] J. Pont-Tuset, P. Arbeláez, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping for image segmentation and object proposal generation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 1, pp. 128–140, Jan. 2017.

[43] D. Potapov, M. Douze, Z. Harchaoui, and C. Schmid, "Category-specific video summarization," in *Proc. ECCV*, 2014, pp. 540–555.

[44] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[45] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[46] M. Rochan and Y. Wang, "Video summarization by learning from unpaired data," in *Proc. CVPR*, Jun. 2019, pp. 7902–7911.

[47] M. Rochan, L. Ye, and Y. Wang, "Video summarization using fully convolutional sequence networks," in *Proc. ECCV*, Sep. 2018, pp. 347–363.

[48] A. Sharghi, A. Borji, C. Li, T. Yang, and B. Gong, "Improving sequential determinantal point processes for supervised video summarization," in *Proc. ECCV*, Sep. 2018, pp. 517–533.

[49] A. Sharghi, B. Gong, and M. Shah, "Query-focused extractive video summarization," in *Proc. ECCV*, 2016, pp. 3–19.

[50] Z. Shou, D. Wang, and S.-F. Chang, "Temporal action localization in untrimmed videos via multi-stage CNNs," in *Proc. CVPR*, Jun. 2016, pp. 1049–1058.

[51] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "TVSum: Summarizing Web videos using titles," in *Proc. CVPR*, Jun. 2015, pp. 5179–5187.

[52] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[53] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9627–9636.

[54] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[55] H. Wei, B. Ni, Y. Yan, H. Yu, X. Yang, and C. Yao, "Video summarization via semantic attended networks," in *Proc. AAAI*, 2018, pp. 216–223.

[56] 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.

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

[58] Y. Xiong, Y. Zhao, L. Wang, D. Lin, and X. Tang, "A pursuit of temporal accuracy in general activity detection," 2017, *arXiv:1703.02716*. [Online]. Available: http://arxiv.org/abs/1703.02716

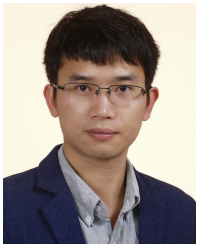
[59] H. Xu, A. Das, and K. Saenko, "Two-stream region convolutional 3D network for temporal activity detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2319–2332, Oct. 2019.

[60] L. Yang, H. Peng, D. Zhang, J. Fu, and J. Han, "Revisiting anchor mechanisms for temporal action localization," *IEEE Trans. Image Process.*, vol. 29, pp. 8535–8548, 2020.

[61] T. Yao, T. Mei, and Y. Rui, "Highlight detection with pairwise deep ranking for first-person video summarization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 982–990.

[62] L. Yuan, P. Li, J. Feng, L. Zhou, and F. E. Tay, "Cycle-sum: Cycle-consistent adversarial LSTM networks for unsupervised video summarization," in *Proc. AAAI*, vol. 2019, pp. 9143–9150.

[63] 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.
[64] K.-H. Zeng, T.-H. Chen, J. C. Niebles, and M. Sun, "Title generation for user generated videos," in *Proc. ECCV*, 2016, pp. 609–625.
[65] R. Zeng, H. Xu, W. Huang, P. Chen, M. Tan, and C. Gan, "Dense regression network for video grounding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10287–10296.
[66] 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.
[67] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in *Proc. ECCV*, 2016, pp. 766–782.
[68] K. Zhang, K. Grauman, and F. Sha, "Retrospective encoders for video summarization," in *Proc. ECCV*, Sep. 2018, pp. 383–399.
[69] Y. Zhang, M. Kampffmeyer, X. Zhao, and M. Tan, "DTR-GAN: Dilated temporal relational adversarial network for video summarization," in *Proc. ACM Turing Celebration Conf. China - ACM TURC*, 2019, pp. 1–6.
[70] Y. Zhang, X. Liang, D. Zhang, M. Tan, and E. P. Xing, "Unsupervised object-level video summarization with online motion auto-encoder," *Pattern Recognit. Lett.*, vol. 130, pp. 376–385, Feb. 2020.
[71] B. Zhao, X. Li, and X. Lu, "Hierarchical recurrent neural network for video summarization," in *Proc. ACM Multimedia Conf. - MM*, 2017, pp. 863–871.
[72] B. Zhao, X. Li, and X. Lu, "Property-constrained dual learning for video summarization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 10, pp. 3989–4000, Oct. 2020.
[73] B. Zhao, X. Li, and X. Lu, "TTH-RNN: Tensor-train hierarchical recurrent neural network for video summarization," *IEEE Trans. Ind. Electron.*, early access, Mar. 16, 2020, doi: 10.1109/TIE.2020.2979573.
[74] B. Zhao and E. P. Xing, "Quasi real-time summarization for consumer videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2513–2520.
[75] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, X. Tang, and D. Lin, "Temporal action detection with structured segment networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2914–2923.
[76] K. Zhou, Y. Qiao, and T. Xiang, "Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward," in *Proc. AAAI*, 2018, pp. 7582–7589.
[77] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra, "Adaptive key frame extraction using unsupervised clustering," in *Proc. Int. Conf. Image Process. ICIP*, Oct. 1998, pp. 866–870.



Wencheng Zhu received the B.Eng. and M.Eng. degrees from the School of Computer Science and Technology, Tianjin University, China, in 2014 and 2017, respectively. He is currently pursuing the Ph.D. degree with the Department of Automation, Tsinghua University, China. His research interests include video summarization and video object segmentation.

Jiwen Lu (Senior Member, IEEE) received the B.Eng. degree in mechanical engineering and the M.Eng. degree in electrical engineering from the Xi'an University of Technology, Xi'an, China, in 2003 and 2006, respectively, and the Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore, in 2012. He is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include computer vision, pattern recognition, and machine learning. He has authored/coauthored more than 250 scientific papers in these areas, where more than 80 of them are the IEEE TRANSACTIONS papers and more than 70 of them are CVPR/ICCV/ECCV papers. He is a Fellow of IAPR. He was/is a member of the Image, Video, and Multidimensional Signal Processing Technical Committee, the Multimedia Signal Processing Technical Committee, and the Information Forensics and Security Technical Committee of the IEEE Signal Processing Society; and a member of the Multimedia Systems and Applications Technical Committee and the Visual Signal Processing and Communications Technical Committee of the IEEE Circuits and Systems Society. He serves as the Co-Editor-in-Chief for the *Pattern Recognition Letters* and an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON BIOMETRICS, BEHAVIOR, AND IDENTITY SCIENCE, and *Pattern Recognition*.

Jiahao Li is currently pursuing the bachelor's degree with Tsinghua University, Beijing, China, advised by Dr. Jiwen Lu at the Department of Automation, Tsinghua University. His current research interests include computer vision and deep learning.

Jie Zhou (Senior Member, IEEE) received the B.S. and M.S. degrees from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the Ph.D. degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology (HUST), Wuhan, China, in 1995. From then to 1997, he has served as a Postdoctoral Fellow with the Department of Automation, Tsinghua University, Beijing, China, where he has been a Full Professor since 2003. His research interests include computer vision, pattern recognition, and image processing. In recent years, he has authored more than 100 papers in peer-reviewed journals and conferences. Among them, more than 30 papers have been published in top journals and conferences, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON IMAGE PROCESSING, and CVPR. He received the National Outstanding Youth Foundation of China Award. He is an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and two other journals.