This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology

Separable Structure Modeling for Semi-supervised Video Object Segmentation

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

***Abstract*—In this paper, we propose a separable structure modeling approach for semi-supervised video object segmentation. Unlike most existing methods which preclude the semantically structural information of target objects, our method not only captures pixel-level similarity relationships between the reference and target frames but also reveals the separable structure of the specified objects in target frames. Specifically, we first compute a pixel-wise similarity matrix by using representations of reference and target pixels and then select top rank reference pixels for target pixel classification. According to the prior knowledge from these top-rank reference pixels, we further appoint the representative target pixels for object structure modeling. Particularly, in the structure modeling branch, we extract the shared and individual features that can well represent the whole object and its components, respectively. Moreover, the proposed method is a fast algorithm without online fine-tuning and any post-processing. We conduct extensive experiments and ablation studies on the DAVIS-16, DAVIS-17, and YouTube-VOS datasets, and experimental results on three widely-used datasets demonstrate that our method achieves a superior performance, compared with state-of-the-art semi-supervised video object segmentation approaches in terms of speed and accuracy.**

***Index Terms*—Video object segmentation, feature matching, separable structure modeling, individual and shared components, semi-supervised learning**

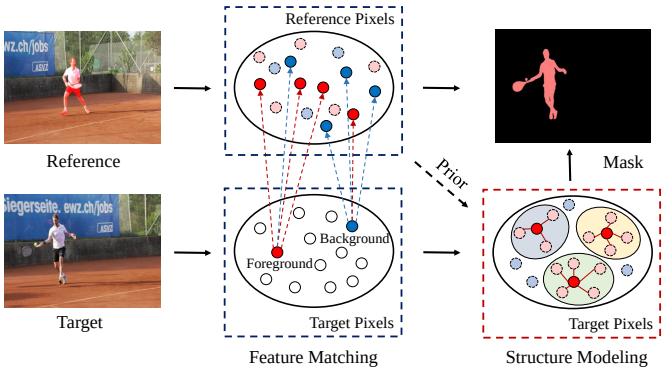


Fig. 1. The overview of the separable structure modeling approach. Our method is composed of two branches, one of which is a feature matching branch (blue dashed box) and another one is the separable structure modeling branch (red dashed box). The feature matching branch regards reference and target frames as pixel sets and utilizes the representative reference pixels (solid circle) to vote target pixels as foreground (red circle) or background (blue circle). The separable structure modeling branch leverages the prior information of object components from the feature matching branch, and models the individual and shared components of target objects. Finally, we integrate the information from these two branches for the final target object segmentation.

I. Introduction

VIDEO object segmentation is a fundamental and important task in computer vision [18], [47], [69], and has been applied into many practical applications including action recognition [46], [68], object tracking [30], [53], video editing [2], [6]. In recent years, great efforts have been devoted to develop fast and accurate methods [17], [27], [65]. Generally, video object segmentation contains unsupervised [28], [42], [44], [54], [63], semi-supervised [33], [40], [52], and interactive [34], [35], [39] tasks. We address the semi-supervised task in this paper. Semi-supervised video object segmentation aims to segment target objects along a video sequence with the initial masks provided [10], [15], [43]. While encouraging performance has been achieved, semi-supervised video object segmentation is faced with complex visual appearance, fast motion, and background clutter problems [61].

Recently, numerous semi-supervised video object segmentation methods have been proposed [16], [45], and these methods can be mainly categorized into four classes: 1) online learning based [4], [22]; 2) mask propagation based [48], [56], [59]; 3) feature matching based [7], [20], [55]; and 4) tracking based [50] approaches. For the first class, online learning based methods [9], [24] first train a segmentation network and then online fine-tune the network by augmenting the annotated frame. Representative methods include OSVOS [4], OSVOS-S [32], and OnAVOS [51]. While these methods have reported impressive performance, online learning procedure entails data augmentation that is computationally expensive. For the second class, mask propagation based methods [1], [37], [62] capture the temporal consistency by using the previous frame and the predicted mask. Typical methods include MSK [37], OSMN [62], and RGMP [56]. Mask propagation based methods need no time consuming online fine-tuning, but they highly

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, and Grant U1713214, in part by Beijing Academy of Artificial Intelligence (BAAI), in part by a grant from the Institute for Guo Qiang, Tsinghua University, and in part by in part by the Shenzhen Fundamental Research Fund (Subject Arrangement) under Grant JCYJ20170412170602564. *(Corresponding author: Jiwen Lu).*

Wencheng Zhu, Jiahao Li, and Jiwen Lu are with the Beijing National Research Center for Information Science and Technology (BNRist), and the Department of Automation, Tsinghua University, Beijing, 100084, China. Email: zwc17@mails.tsinghua.edu.cn; lijiahao17@mails.tsinghua.edu.cn; lujiwen@tsinghua.edu.cn.

Jie Zhou is with the Beijing National Research Center for Information Science and Technology (BNRist), Department of Automation, Tsinghua University, and the Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China. Email: jzhou@tsinghua.edu.cn.

rely on the predicted results of pervious frames and suffer from occlusion and drifting problems. For the third class, feature matching methods [7], [55] embed both reference and target frames into a common representation space and exploit pixel-level similarities for the label assignment. Representative methods include PML [7], VideoMatch [20], and RANet [55]. Feature matching based methods neglect both the object structure and appearance information due to unordered pixels. Moreover, they are easily affected by noises and outliers. For the fourth class, tracking based methods [50], [53], [64] first track candidate regions and then accurately segment each local region. Typical methods include SiamMask [53] and FAVOS [8]. Unlike feature matching methods, tracking based methods leverage the object appearance information and maintain the fast merit of object tracking. However, these methods view object tracking and segmentation as two separate steps and the segmentation performance critically hinges on tracking results.

To address the above-mentioned time consuming and structure modeling problems, we propose a feature matching based approach for fast semi-supervised video object segmentation. Notably, Fig. 1 illustrates the basic idea of our method. We clearly observe that our method contains a pixel-wise feature matching branch and a separable structure modeling branch. The feature matching branch learns pixel-level similarities between the reference and target frames, while the separable structure modeling branch extracts the structure information of target objects. To be specific, we leverage information from these two branches to guide video object segmentation in two aspects. On one hand, since video object segmentation is formulated as a pixel labeling problem, the feature matching branch learns pixel-wise correspondences for fine-grained segmentation. However, this branch lacks the cognitive ability of object structures. On the other hand, the separable structure modeling is beneficial to reveal subtle object patterns and coarsely localize objects. Particularly, the branch is inspired by the observations that the appearance of a target object may be dynamically changing throughout a video sequence but its components tend to be consistent and are more discriminative than the whole object. Hence, we learn individual features of object components and the shared feature of the whole object.

Unlike conventional matching based methods [7], [20], [55] which merely exploit pixel-wise relationships between the reference and target frames to classify target pixels, our method not only learns pixel-level similarities, but also extracts the semantically structural information of target objects for spatial details and localization. Notably, our method provides a strong baseline for matching based methods in terms of both accuracy and speed. Furthermore, there are two key differences with prevalent structure modeling methods [14], [67]. One is that previous structure modeling methods either introduce additional image data with the same labels for localizing objects [67] or pose structural constraints on parsing results via a loss function [14]. Differently, the separable structure modeling branch utilizes pseudo labels of target pixels from the feature matching branch without extra data and the loss function introduced. Another one is that our method learns the individual and shared representations for modeling the local and global internal structures of objects, respectively.

Importantly, our method takes the structure modeling of the target frame into consideration. We further conduct comprehensive experiments to evaluate the effectiveness of the proposed method, and experimental results on the DAVIS-16, DAVIS-17, and YouTube-VOS datasets clearly demonstrate that our method achieves very competitive performance compared with state-of-the-art methods on the tradeoff between accuracy and speed.

The main contributions of the proposed method are summarized as follows:

1) We propose a separable structure modeling approach for fast semi-supervised video object segmentation without online fine-tuning and post-processing.
2) Our method leverages not only pixel-wise similarities between reference and target frames but also the structure information of target objects.
3) We learn the shared and individual representations for the structure modeling, where the individual representations model object components and the shared representation encodes information of the whole object.
4) We conduct extensive experiments on the DAVIS-16, DAVIS-17, and YouTube-VOS datasets, and experimental results validate the effectiveness and efficiency of the proposed method.

The remainder of this paper is organized as follows: Section II reviews the related work. Section III details the proposed separable structure modeling approach. Section IV describes experimental settings, results and analyses, and visualizations. Section V concludes this paper.

II. Related Work

Existing semi-supervised methods can be roughly classified into four categories, i.e., online learning based [4], [22], mask propagation based [48], [56], [59], feature matching based [7], [20], [55], and tracking based [50] methods. Next, we briefly introduce these four categories.

A. Online Learning

Online learning based methods fine-tune a pre-trained segmentation network by using the first annotated frame in testing videos. For example, Caelles *et al.* [4] extracted the target-specific appearance information at test time by fine-tuning a pre-trained network on the first frame. Paul *et al.* [51] extended the previous method and selected confident regions for online adaptation. Maninis *et al.* [32] combined the instance-level semantic information by using instance proposals to improve segmentation performance. Khoreva *et al.* [24] conducted data augmentation to generate training data for the proposed Lucid tracker. Cheng *et al.* [9] proposed a SegFlow architecture that jointly learned object segmentation and optical flow. Andreas *et al.* [40] proposed a novel segmentation framework that was composed of two network components, where the target appearance model was updated online and the segmentation model was trained offline, respectively. Xiao *et al.* [58] learned a meta-learner of a base segmentation model for online adaptation. Generally, many existing semi-supervised video object segmentation methods [37], [56] regarded online fine-tuning

as a post-processing step, which has been proven to certainly promote segmentation performance. However, online learning based methods are time-consuming for practical applications due to data augmentation in the test phase.

B. Mask Propagation

Mask propagation based methods promise to maintain the spatio-temporal consistency of segmentation results by leveraging the initial and previous frame information. For example, Perazzi *et al.* [37] concatenated both the predicted mask of the previous frame with the current frame as an input of MaskTrack, which produced a refined mask for the current frame. Seoung *et al.* [56] proposed a Siamese encoder-decoder network that also inputted the target frame and the mask of the previous frame. Yang *et al.* [62] adapted a segmentation network to target-specific objects by using conditional batch normalization [12]. Xu *et al.* [60] proposed a sequence-to-sequence network to capture long-term spatial-temporal information among videos. Carles *et al.* [48] proposed a recurrent spatial and temporal architecture to cope with multi-object segmentation. Lin *et al.* [26] designed the instance-agnostic and instance-specific modules for multiple object segmentation. Joakim *et al.* [23] developed a generative appearance model that provided both foreground and background feature distributions in a single forward pass.

C. Feature Matching

Feature matching based methods compute a similarity matrix by using representations of reference and target pixels and further assign binary labels to these target pixels based on the learned similarity relationships. For example, Yoon *et al.* [41] encoded multi-scale pixel-level similarities from different depth layers. Chen *et al.* [7] formulated semi-supervised video object segmentation as a pixel-level retrieval problem and adopted metric learning to learn pixel-wise correspondences. Hu *et al.* [20] matched foreground and background features of the reference frame with target features, simultaneously. Ci *et al.* [11] learned location-sensitive embeddings for foreground prediction. Paul *et al.* [49] adapted both the global and local information from the first and previous frames to the current frame. Wang *et al.* [55] combined feature matching and mask propagation into an encoder-decoder framework. Behl *et al.* [3] developed a meta-learning approach that represented target objects by using visual words. Li *et al.* [25] developed a video object segmentation approach that first tracked and segmented objects, and then re-identified these objects. Seoung *et al.* [36] proposed the space-time memory networks for spatial and temporal matching. Kevin *et al.* [13] proposed a capsule-based approach for capsule matching between video clips and the reference frame. Zhang *et al.* [66] developed a transductive approach for video object segmentation. Lu *et al.* [29] performed message passing and memory updating via a graph memory network. Lu *et al.* [30] leveraged multi-granularity information including frame, short-term, long-term, and video granularities for both zero-shot and one-shot video object segmentation.

D. Tracking

Tracking based methods usually track objects or object parts by using object appearance information. For example, Hu *et al.* [19] proposed an instance-level segmentation framework that tracked and segmented individual objects. Cheng *et al.* [8] developed a part tracker to track representative object parts in the initial frame. Luiten *et al.* [31] proposed a proposal generation, refinement, and merging approach for video object segmentation. Wang *et al.* [53] unified object tracking and object segmentation into a framework. Zeng *et al.* [64] proposed a differentiable matching layer to merge object proposals. Chen *et al.* [5] designed a state-aware tracker that iteratively updated the cropping strategy of tracklet and state estimator. Huang *et al.* [21] proposed a temporal aggregation network and a template matching mechanism to integrate segmentation and tracking. Paul *et al.* [50] conducted multi-object detection, tracking, and segmentation in a unified network.

III. Approach

In this section, we first provide an overview of the proposed method. Then, we elaborate on the feature matching branch and the separable structure modeling branch, respectively.

A. Overview

Given a video sequence $\{\mathbf{I}_i\}_{i=1}^{T}$ with T frames and the ground truth segmentation $\mathbf{Y}_1 = \{\mathbf{y}_1^{(j)}\}_{j=1}^{N}$ of the first frame $\mathbf{I}_1$ with N target objects, the objective of semi-supervised video object segmentation is to produce the segmentation masks $\{\mathbf{Y}_i\}_{i=2}^{T}$ of subsequent video frames $\{\mathbf{I}_i\}_{i=2}^{T}$ [20]. Specifically, N represents the number of objects to be segmented along a video sequence. To be simple, we only take one object situation $N = 1$ for an example, and the multiple object extension is described at last.

Fig. 2 illustrates the pipeline of our method, which mainly consists of two branches: 1) the feature matching branch and 2) the separable structure modeling branch. The feature matching branch provides fine-grained information about foreground and background. However, due to the lack of structure information of target objects, pixel-wise matching easily suffers from drifting and discontinuity problems [20]. To alleviate these issues, we further propose a separable structure modeling branch. While visual patterns of object appearances can be also used for object localization, separable structure modeling identifies more discriminative regions subject to occlusions and rotations since object components may keep consistent throughout video sequences [14], [67]. Besides, the structure modeling branch leverages information from the feature matching branch without much computation. Finally, we incorporate both pixel-wise similarity and structure information for object segmentation.

Generally, our network follows the encoder-decoder architecture, where the encoder is used for feature extraction and the decoder is used for feature fusion and object segmentation. For the reference frame and the target frame, we first employ a Siamese network [56] with shared parameters to transform them into a common representation space and extract their features. Then, the similarity relationships between target pixels with foreground and background pixels of the reference frame

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology

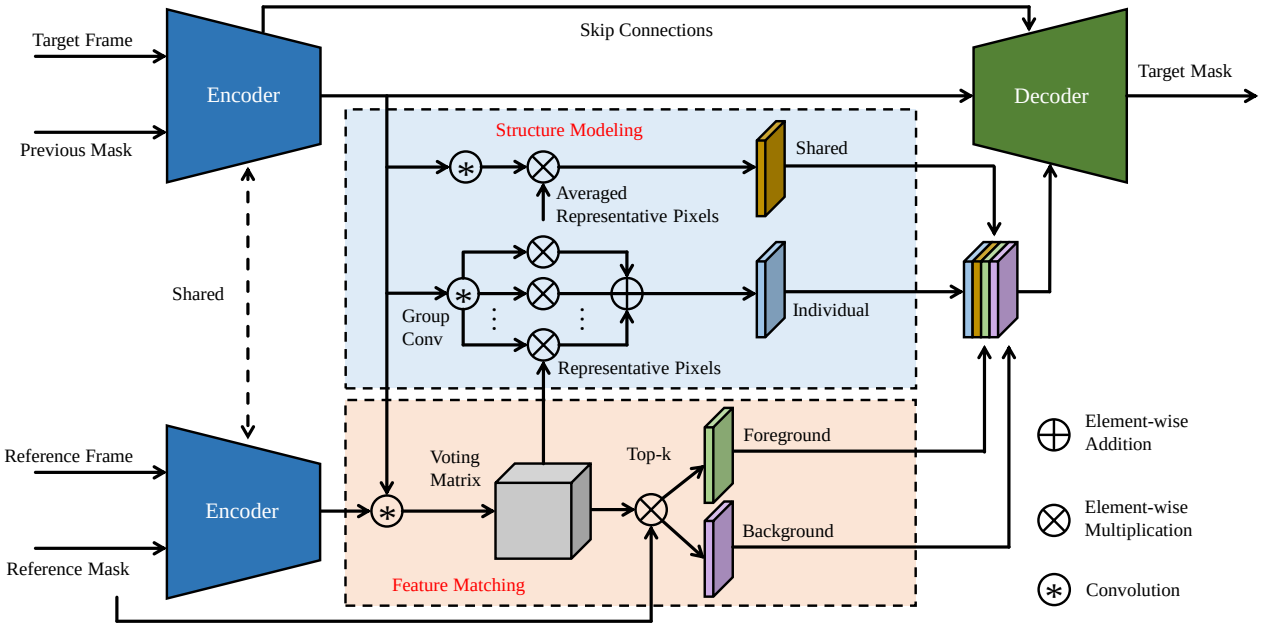


Fig. 2. The architecture of the separable structure modeling approach. Our network is an encoder-decoder framework with a frame and mask pair as an input. The feature matching branch computes a voting matrix by using pixel representations of reference and target frames, and the top rank reference pixels are used to classify foreground and background. The structure modeling branch computes attentive maps between all target pixels with representative target pixels that are specified by using the voting matrix. Moreover, the individual and shared features are learned with these attentive maps. Finally, the skip connections from the encoder, high-level semantic features, voting representations, and structure representations are concatenated and decoded for segmentation.

are captured. According to the prior information concerned with target objects, we also select representative target pixels to model objects in the target frame. Particularly, the individual features of the selected pixels are extracted and the shared feature of the target object is also learned, representing object components and the whole object, respectively. In the same way as the feature matching branch, we compute similarity relationships between the individual and shared features with target features. Lastly, our network decodes the merged information from these two branches. Next, we describe the feature matching and separable structure modeling branches in detail.

B. Feature Matching Branch

The feature matching branch is inspired by the assumption that pixels of the same category are close to each other in the same embedding space [7], [20], [41]. We follow this assumption and vote target pixels by using annotated reference pixels. Specifically, the feature matching branch regards both the reference frame and the target frame as pixel sets. By using the binary mask provided in the reference frame, we can obtain foreground and background pixels of reference objects. Then, the foreground and background voting of target pixels is performed based on similarities with foreground and background pixels in the reference frame.

Formally, we denote $\mathbf{X}_r \in \mathbb{R}^{H\times W\times C}$ and $\mathbf{X}_t \in \mathbb{R}^{H\times W\times C}$ as feature vectors of reference $\mathbf{I}_r$ and target $\mathbf{I}_t$ frames, where H, W, and C are the height, width and the channel number of feature maps. We denote the spatial domain of the feature map as $\Omega = \{(h,w) \mid h \le H, w \le W, h, w \in \mathbb{N}_+\}$. As the reference mask $\mathbf{Y}_r \in \{0,1\}^{H\times W}$ is given, we define sets of foreground and background features $\mathbf{r}_f \in \mathbb{R}^{N_f\times C}$ and $\mathbf{r}_b \in \mathbb{R}^{N_b\times C}$ as,

$$\begin{aligned}\mathbf{r}_f &= \{\mathbf{X}_r(p) \mid \mathbf{Y}_r(p) = 1, p \in \Omega\},\\ \mathbf{r}_b &= \{\mathbf{X}_r(p) \mid \mathbf{Y}_r(p) = 0, p \in \Omega\},\end{aligned} \tag{1}$$

where $\mathbf{Y}_r(p)$ and $\mathbf{X}_r(p)$ are the element and feature of $\mathbf{Y}_r$ and $\mathbf{X}_r$ at the spatial position p, respectively. N_f and N_b are the numbers of foreground and background features.

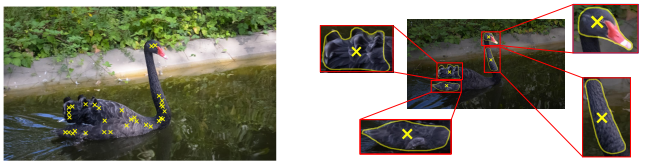
Having obtaining the foreground features $\mathbf{r}_f$ and background features $\mathbf{r}_b$, we perform a voting for target features. We first flatten the target feature $\bar{\mathbf{X}}_t \in \mathbb{R}^{HW\times C}$, and then compute the scoring matrices $\mathbf{S}_f \in \mathbb{R}^{HW\times N_f}$ and $\mathbf{S}_b \in \mathbb{R}^{HW\times N_b}$ between target features with foreground and background features as,

$$\begin{aligned}\mathbf{S}_f &= [\mathbf{r}_f^1 * \bar{\mathbf{X}}_t; \ldots; \mathbf{r}_f^{N_f} * \bar{\mathbf{X}}_t]\\ \mathbf{S}_b &= [\mathbf{r}_b^1 * \bar{\mathbf{X}}_t; \ldots; \mathbf{r}_b^{N_b} * \bar{\mathbf{X}}_t],\end{aligned} \tag{2}$$

where $\mathbf{r}_f^i \in \mathbb{R}^C$ and $\mathbf{r}_b^i \in \mathbb{R}^C$ represent the i^{th} elements of feature sets $\mathbf{r}_f$ and $\mathbf{r}_b$, respectively. $*$ denotes the convolution operation between a foreground or background feature and the target feature.

To eliminate the influence of noises and outliers, we only take top rank similarities among foreground and background pixels with target pixels into account. Specifically, we adopt this top rank strategy based on the following insights. Firstly, it is quite unreasonable for foreground reference pixels to vote for target background pixels or for background reference pixels to vote for target foreground pixels. Secondly, one object is usually composed of several different components, but there exists a considerably large intra-class difference among each

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology



(a) Representative target pixels (b) Selected pixel and region examples

Fig. 3. Visualizations of top rank target pixels and the concerned regions from the blackswan video on DAVIS-17. We only show four representative target pixels (yellow $\times$) for an example, and we select the concerned coarse regions (yellow curve regions) that have high responses with the corresponding target pixels by using heat maps of 32 individual features. Zoom in for details.

component, indicating that one target pixel can be only well represented by reference pixels within a similar local region. The top rank strategy selects representative pixels with high probabilities to filter out unrelated foreground or background pixels. Intuitively, we can identify foreground objects either by using foreground information or by removing the background. Particularly, background regions may contain similar objects with foreground, which confuses the foreground pixel selection. This finding also drives us to equally treat foreground and background pixels.

To be specific, we first compute ranking scores $\mathbf{rs}_f \in \mathbb{R}^{N_f}$ and $\mathbf{rs}_b \in \mathbb{R}^{N_b}$ by averaging columns of the scoring matrices $\mathbf{S}_f$ and $\mathbf{S}_b$. Then, we select top K foreground and K background pixels according to ranking scores $\mathbf{rs}_f$ and $\mathbf{rs}_b$. Afterwards, we obtain the voting matrices $\mathbf{V}_f \in \mathbb{R}^{HW\times K}$ and $\mathbf{V}_b \in \mathbb{R}^{HW\times K}$ by selecting columns in $\mathbf{S}_f$ and $\mathbf{S}_b$ that are concerned with top K foreground and K background pixels. We also analyze different voting strategies including maximum and averaged voting matrices and different numbers of selected pixels. Finally, we achieve the voting feature $\mathbf{V} \in \mathbb{R}^{H\times W\times 2K}$ by reshaping the concatenated voting matrix $[\mathbf{V}_f; \mathbf{V}_b]$.

Our feature matching branch exploits pixel-level matching between reference and target pixels, and provides more fine-grained information than propagation and tracking based methods. However, this branch neglects structure information.

C. Separable Structure Modeling Branch

To address the above-mentioned issue, we propose a separable structure modeling branch. On one hand, the separable structure modeling has been proved to be beneficial to identify discriminative object regions [67]. On the other hand, regarding an object as a whole severely suffers from many challenges including viewpoint variations and deformation [3], but the separable structure modeling of target objects is more robust to intra-object variations. Besides, object segmentation requires a full understanding of the composition of segmented objects [14]. However, due to the absence of class and object information of the target frame, many video object segmentation methods adopt matching and tracking mechanisms to capture correspondences between the reference and target frames. Our structure modeling branch is the first attempt to model target objects. This branch is motivated by the observation that since objects have internal structures, many unsupervised methods [30], [42] succeed in detecting salient objects without the label information. Our method aims to provide this intrinsic object information as well as inter-frame correspondences.

Existing prevalent object structure modeling approaches [14], [67] either exploit semantic joints in person or extract attentive regions of objects in an unsupervised manner. Differently, our separable structure modeling branch reveals object components by using the prior information from the feature matching branch. Specifically, in the feature matching branch, target pixels are utilized to select top rank foreground pixels. On the contrary, these foreground pixels are informative and can be used to vote for target representative pixels. Therefore, we reuse the information of the foreground voting matrix to choose representative target pixels. Fig. 3 visualizes top rank target pixels, example pixels and their concerned regions with high responses. The selected pixels can well represent object patterns, such as the head, tail, and neck of the black swan.

Unlike the feature matching branch, we compute the ranking score by averaging rows of the voting matrix $\mathbf{V}_f$, and select top rank K target features. Formally, we obtain K individual attentive matrices $\{\mathbf{M}_k \in \mathbb{R}^{H\times W}\}_{k=1}^{K}$ between representative target features $\{\mathbf{x}_t^k \in \mathbb{R}^C\}_{k=1}^{K}$ and $\mathbf{X}_t$ as,

$$\mathbf{M}_k = \mathbf{x}_t^k * \mathbf{X}_t, \tag{3}$$

where $\mathbf{x}_t^k$ and $\mathbf{M}_k$ represent the k^{th} representative feature and attentive matrix. Each attentive matrix focuses on a local region of objects. Then, we partition the target features into K individual features denoted as $\mathbf{X}_t = [\mathbf{X}_t^1; \mathbf{X}_t^2; \ldots; \mathbf{X}_t^K]$, $\mathbf{X}_t^k \in \mathbb{R}^{H\times W\times \frac{C}{K}}$, and we conduct element-wise multiplication $\odot$ on the k^{th} individual feature and attentive matrix $\mathbf{M}_k$ to learn the k^{th} individual feature $\mathbf{Z}_k \in \mathbb{R}^{H\times W\times \frac{C}{K}}$ as,

$$\mathbf{Z}_k = \mathbf{X}_t^k \odot \mathbf{M}_k. \tag{4}$$

Afterwards, K individual features are totally concatenated for the final individual feature. Furthermore, we average the selected features of representative target features and compute the shared attentive matrix $\mathbf{M}_s \in \mathbb{R}^{H\times W}$ as,

$$\mathbf{M}_s = (\frac{1}{K}\sum\nolimits_k \mathbf{x}_t^k) * \mathbf{X}_t. \tag{5}$$

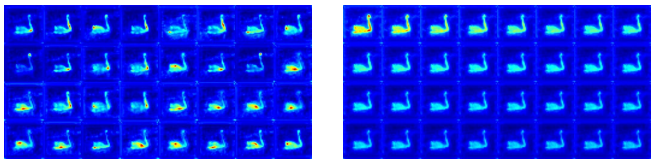
Then, we conduct dimension reduction on the input feature with a 1×1 convolutional layer, and the shared feature $\mathbf{Z}_s$ is generated by applying element-wise product on the shared attentive matrix and the target feature,

$$\mathbf{Z}_s = \mathbf{X}_t \odot \mathbf{M}_s. \tag{6}$$

We concatenate the individual and shared attentive features as the output of the separable structure modeling branch. Finally, we integrate the information from both the feature matching branch and the separable structure modeling branch. Specifically, the inputs to our decoder are features from skip connections and high-level sematic information from encoder together with these combined features. Likewise, we adopt the cross-entropy loss as the loss function to optimize model parameters. The training procedure of the proposed method is summarized in **Algorithm** 1.

Discussion: Due to different context information among reference and target frames, it is more appropriate to capture target

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology



(a) Heat maps of individual features (b) Heat maps of voting feature

Fig. 4. Visualizations of 32 heat maps for each individual feature $\mathbf{Z}_k$ and 32 heat maps for the voting feature $\mathbf{V}$ from the blackswan video on DAVIS-17. Zoom in for more details.

object details via target pixels instead of reference pixels. Fig. 4 visualizes the heat maps of individual features and the voting feature. For each heat map in individual features, the pixel with the highest response represents a selected pixel shown in Fig. 3(a). We observe that heat maps from individual features are attentive to local regions around representative target pixels and spatial details can be easily extracted. However, selected reference pixels tend to have similar responses to pixels of the whole target object, which is insensitive to object structures.

D. Extension to Multiple Object Segmentation

Many semi-supervised video object segmentation methods [53], [56] handle the multiple object issue by separating a multi-object mask into multiple masks with one single object and segmenting each object one by one. However, this strategy is very time-consuming. Differently, we adopt a flexible strategy to share both encoder features and voting matrices for all objects in the same frame. By using this strategy, our method can alleviate repetitive computation and harvest time-saving.

Algorithm 1: Training procedure of the proposed method

Input: A video sequence $\{\mathbf{I}_i\}_{i=1}^{T}$ and mask $\{\mathbf{Y}_i\}_{i=1}^{T}$;
Output: The parameters θ of the proposed method;
$\theta \leftarrow$ InitializeParameter(θ);
for *frame* $\mathbf{I}_r \in \{\mathbf{I}_i\}_{i=1}^{T}$ **do**
  $[\mathbf{I}_r;\mathbf{Y}_r],[\mathbf{I}_t;\mathbf{Y}_{t-1}] \leftarrow$ SamplingPairs($\{\mathbf{I}_i\}_{i=1}^{T},\{\mathbf{Y}_i\}_{i=1}^{T}$);
  $\mathbf{X}_r,\mathbf{X}_t \leftarrow$ ResNet50($[\mathbf{I}_r;\mathbf{Y}_r],[\mathbf{I}_t;\mathbf{Y}_{t-1}]$);
  % The feature matching branch
  $\mathbf{r}_f,\mathbf{r}_b \leftarrow$ ObtainFBgroundFeatures($\mathbf{X}_r,\mathbf{Y}_r$) via Eq.(1)
  $\mathbf{S}_f,\mathbf{S}_b \leftarrow$ ComputeScoring($\mathbf{X}_t,\mathbf{r}_f,\mathbf{r}_b$) via Eq. (2);
  $\mathbf{V}_f,\mathbf{V}_b \leftarrow$ObtainVotingMatrices($\mathbf{S}_f,\mathbf{S}_b$);
  $\mathbf{V} \leftarrow[\mathbf{V}_f;\mathbf{V}_b]$;
  % The separable structure modeling branch
  $\{\mathbf{x}_t^k\}_{k=1}^{K} \leftarrow$ SelectRepresentativeTargetPixels($\mathbf{X}_t,\mathbf{V}_f$);
  for $\mathbf{x}_t^k \in \{\mathbf{x}_t^k\}_{k=1}^{K}$ **do**
    $\mathbf{M}_k \leftarrow$ComputeIndividualMatrices($\mathbf{x}_t^k,\mathbf{X}_t$) via Eq. (3);
  end
  $\mathbf{M}_s \leftarrow$ComputeSharedMatrix($\{\mathbf{x}_t^k\}_{k=1}^{K},\mathbf{X}_t$) via Eq. (5);
  $\mathbf{Z}_k,\mathbf{Z}_s \leftarrow$ ObtainAttentiveFeatures($\mathbf{X}_t,\mathbf{M}_k,\mathbf{M}_s$);
  % Predict segmentation result
  $\mathbf{Y}_\mathbf{t}^* \leftarrow$ Decoder($[\mathbf{X}_t,\mathbf{X}_r,\mathbf{Z}_k,\mathbf{Z}_s,\mathbf{V}]$);
  % Back propagation
  $\mathbf{L} \leftarrow$ ComputeCrossEntropy($\mathbf{Y}_\mathbf{t}^*,\mathbf{Y}_\mathbf{t}$)
  $\theta \leftarrow \theta - \eta\ \partial\mathbf{L}/\partial\theta$
end

IV. Experiments

In this section, we first detail experimental settings including datasets, evaluation metrics, the model architecture, implementation details, and baselines. Then, we compare our method with state-of-the-art methods and conduct ablation studies. Finally, we visualize some experimental results.

A. Experimental Setup

1) Datasets: We evaluated the proposed method on three standard video object segmentation datasets, including the DAVIS-16 [38], DAVIS-17 [39], and YouTube-VOS datasets [60]. Specifically, DAVIS-16 consists of 50 video sequences totally with 3,455 annotated frames, and each video sequence was captured at a 24 frame per second and a 1080p spatial resolution. The temporal extent of these video sequences is about 2-4 seconds. In DAVIS-16, the segmented target contains one single object or a combination of multiple objects. The DAVIS-17 dataset is an extension for the DAVIS-16 dataset. In DAVIS-17, there are 150 video sequences and 10,459 frames, and these video sequences are divided into the training set (60), the validation set (30), and the test set (60). Each video sequence contains 3 objects on average. DAVIS-17 is a multi-object segmentation dataset. The YouTube-VOS dataset is a large-scale video object segmentation dataset and has 197,272 annotations and 4,453 YouTube video clips of which 3471 for training, 474 for validation, and the rest 508 for testing. The YouTube-VOS dataset covers many various challenges including occlusions, fast object motions, and change of appearances. Moreover, YouTube-VOS contains 91 object categories including 65 seen object categories and 26 unseen object categories in the validation dataset, which measures the generalization performance of video object segmentation methods. Besides, YouTube-VOS is also a multi-object segmentation dataset.

2) Evaluation Metrics: Following previous methods [38], [39], we adopted three conventional evaluation metrics, i.e., region similarity $\mathcal{J}$, contour accuracy $\mathcal{F}$, and their average $\mathcal{G}$, to measure the performance of each video frame. Region similarity $\mathcal{J}$ is defined as the intersection-over-union of the predicted segmentation and the ground truth mask. Contour accuracy $\mathcal{F}$ reflects contour-based precision and recall, and is computed by using the F-measure of contour points between the predicted segmentation and the ground truth mask. $\mathcal{G}$ is defined as the average of $\mathcal{J}$ and $\mathcal{F}$. For evaluation of video sequences, we adopted the $\mathcal{J}$ ($\mathcal{F}$) mean, $\mathcal{J}$ ($\mathcal{F}$) recall, and $\mathcal{J}$ ($\mathcal{F}$) decay, representing the mean, recall, and decay measures throughout video sequences. For YouTube-VOS, $\mathcal{J}$ ($\mathcal{F}$) seen and $\mathcal{J}$ ($\mathcal{F}$) unseen metrics were adopted to report segmentation performance when seen and unseen object categories exist in the training stage.

3) Model Architecture: Our model followed an encoder-decoder architecture, where the encoder was a pyramid-like ResNet-50 network pre-trained on ImageNet. In the feature matching branch, we only computed voting matrices by using the Res4 feature maps with the dot product operator. We selected both top 32 representative foreground and background pixels in the reference frame to vote for each target pixel. For the separable structure modeling branch, we also selected the top 32 target pixels with the highest matching scores as

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology

TABLE I
QUANTITATIVE COMPARISONS WITH STATE-OF-THE-ART ONLINE AND OFFLINE LEARNING BASED METHODS ON THE DAVIS-16 VALIDATION DATASET BY USING $\mathcal{J}$, $\mathcal{F}$, $\mathcal{G}$ METRICS, AND AVERAGE PER-FRAME RUNTIME. KEY PROPERTIES OF THESE METHODS ARE SUMMARIZED, I.E., FINE-TUNING (FT), OPTICAL FLOW (OF), POST-POSTING (PP), AND PRE-TRAINED DATASETS INCLUDING PASCAL VOC (P), MS-COCO (C) AND SYNTHETIC DATA (S).

Method	FT	OF	PP	PD			DAVIS-16							Runtime
				P	C	S	$\mathcal{G}\uparrow$	$\mathcal{J}_{mean}$	$\mathcal{J}_{recall}\uparrow$	$\mathcal{J}_{decay}\downarrow$	$\mathcal{F}_{mean}$	$\mathcal{F}_{recall}\uparrow$	$\mathcal{F}_{decay}\downarrow$	
Online methods														
OSVOS [4]	✓	✗	✓	✗	✗	✗	80.2	79.8	93.6	14.9	80.6	92.6	15.0	10.0s
OSVOS-S [32]	✓	✗	✓	✗	✓	✗	86.6	85.6	96.8	5.5	87.5	95.9	8.2	4.5s
OnAVOS [51]	✓	✗	✓	✓	✓	✓	85.5	86.1	96.1	5.2	84.9	89.7	5.8	13.0s
CINM [1]	✓	✓	✗	✓	✗	✓	84.2	83.4	94.9	12.3	85.0	92.1	14.7	$>$ 120s
MSK [37]	✓	✓	✓	✓	✗	✗	77.6	79.7	93.1	8.9	75.4	87.1	9.0	12.0s
Lucid [24]	✓	✓	✓	✓	✗	✓	83.6	84.8	-	-	82.3	-	-	40.0s
MoNet [57]	✓	✓	✓	✓	✗	✗	84.7	84.7	96.8	6.4	84.8	94.7	8.6	14.1s
SFL [9]	✓	✓	✗	✗	✗	✓	76.1	76.1	90.6	12.1	76.0	85.5	10.4	7.90s
PReMVOS [31]	✓	✓	✗	✗	✗	✓	86.8	84.9	96.1	8.8	88.6	94.7	9.8	38.0s
MVOS [58]	✓	✗	✗	✓	✗	✗	83.7	83.3	-	-	84.1	-	-	0.43s
Offline methods														
RGMP [56]	✗	✗	✗	✓	✗	✓	81.8	81.5	91.7	10.9	82.0	90.8	10.1	0.13s
FAVOS [8]	✗	✗	✓	✗	✗	✗	81.0	82.4	96.5	4.5	79.5	89.4	5.5	1.80s
PML [41]	✗	✗	✗	✗	✗	✗	77.4	75.5	89.6	8.5	79.3	93.4	7.8	0.28s
OSMN [62]	✗	✗	✗	✗	✓	✗	73.5	74.0	87.6	9.0	72.9	84.0	10.6	0.14s
VideoMatch [20]	✗	✗	✓	✗	✗	✗	-	81.0	-	-	-	-	-	0.23s
RANet [55]	✗	✗	✗	✗	✗	✓	85.5	85.5	97.2	6.2	85.4	94.9	5.1	0.033s
SiamMask [53]	✗	✗	✗	✗	✓	✓	70.0	71.7	86.8	3.0	67.8	79.8	2.1	0.028s
FEELVOS [49]	✗	✗	✗	✗	✓	✗	81.7	81.1	90.5	13.7	82.2	86.6	14.1	0.51s
STM [36]	✗	✗	✗	✗	✗	✓	89.3	88.7	97.2	5.0	89.9	95.4	4.3	0.16s
Proposed method														
Ours	✗	✗	✗	✗	✗	✗	85.9	86.2	97.1	5.3	85.6	92.3	5.6	0.027s

attentive pixels. The input feature maps were separated into 32 groups in the group convolution layer, and each group was weighted by an attentive map to focus on an individual component of target objects. All groups were further concatenated and channel-wisely summed up as individual features in the final. Furthermore, we employed another 1×1 convolutional layer on input feature maps, and then conducted element-wise multiplication on the averaged attentive maps to extract shared features. Afterwards, the outputs of two branches were merged by concatenation, and were fed into a 3-layer decoder.

4) Implementation Details: For training, we randomly selected two frames from a video sequence, of which one frame and its annotated mask served as the reference input, while another one frame and its blurred previous mask were viewed as the target input. Notably, a Gaussian kernel was employed to blur the previous mask. We applied random flipping, scaling, translation, rotation, cropping to augment the training data. Besides, we cropped a frame-mask pair around object locations identified by using the predicted mask of the previous frame. This crop strategy ensures that an object randomly appears at any position of the cropping window. For testing, we also took the first frame of a video sequence and its mask as the reference input. Moreover, we cropped the target frame according to the estimated mask of the previous frame to keep target objects at the center of cropping windows. Note that our model predicted results of subsequent frames in a video sequence without any data augmentation and post-processing.

We iteratively trained our network for 20 epochs on the YouTube-VOS training set with Adam optimizer, whose batch size and initial learning rate were set as 16 and 10^{-5}, respectively. The initial learning rate decayed 10% after every epoch. Our model was implemented by using PyTorch and was trained on 4 TITAN Xp GPUs. We evaluated the proposed method on the YouTube-VOS validation set and test set with only 1 GPU. Having pre-trained on the YouTube-VOS dataset, we further fine-tuned our model on DAVIS-16 and DAVIS-17 by using the same strategy. The learning rate was initialized as 10^{-6} and decayed every 5 epochs. Following fine-tuning, we evaluated our model on the DAVIS-16 and DAVIS-17 validation set. The fine-tuned model with the best validation performance was selected to test on the DAVIS-17 test set.

5) Baselines: In this paper, we compared the proposed method with state-of-the-art semi-supervised video object segmentation methods. These methods are one-shot video object segmentation (OSVOS) [4], video object segmentation without temporal information (OSVOS-S) [32], online adaptation of convolutional neural networks for video object segmentation (OnAVOS [51]), online meta adaptation for fast video object segmentation (MVOS) [58], deep motion exploitation for video object segmentation (MoNet) [57]. learning video object segmentation from static images (MSK) [37], proposal-generation, refinement and merging for video object segmentation (PReMVOS) [31], joint learning for video object segmentation and optical flow (SFL) [9], video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf (CINM) [1], fast video object segmentation by reference-guided mask propagation (RGMP) [56], fast and accurate online video object segmentation via tracking parts (FAVOS) [8], blazingly fast video object segmentation with pixel-wise metric learning (PML) [7], efficient video object segmentation via network modulation (OSMN) [62], matching based video object segmentation (VideoMatch) [20], end-to-end recurrent

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology

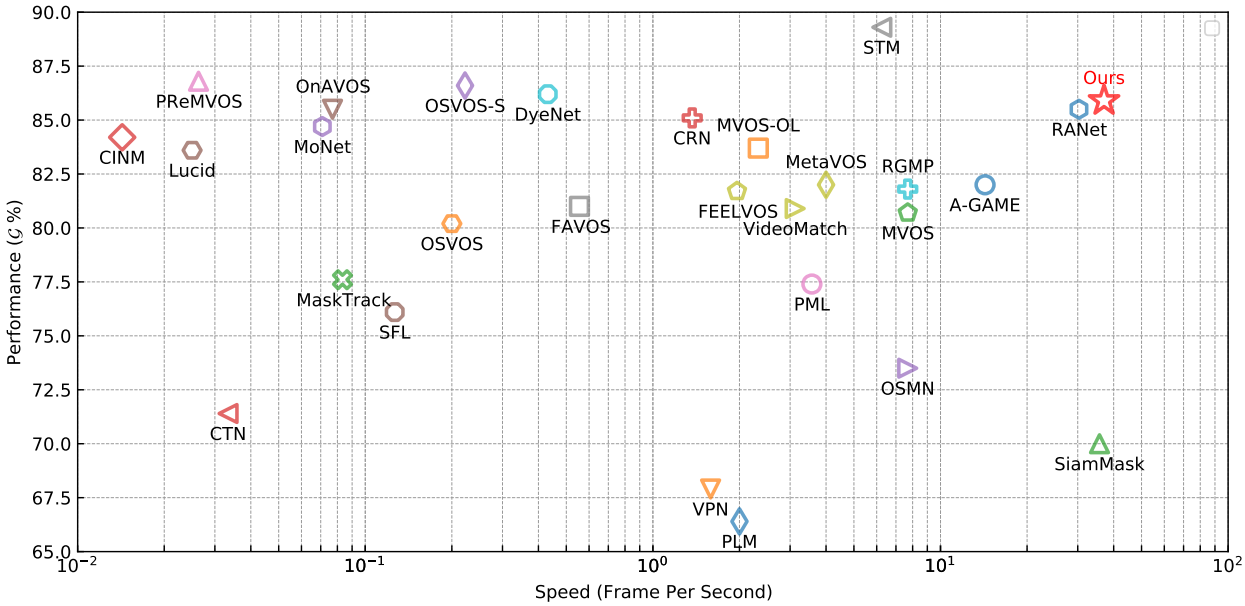


Fig. 5. Visualizations of performance and speed comparisons between our method and state-of-the-art methods on the DAVIS-16 validation dataset.

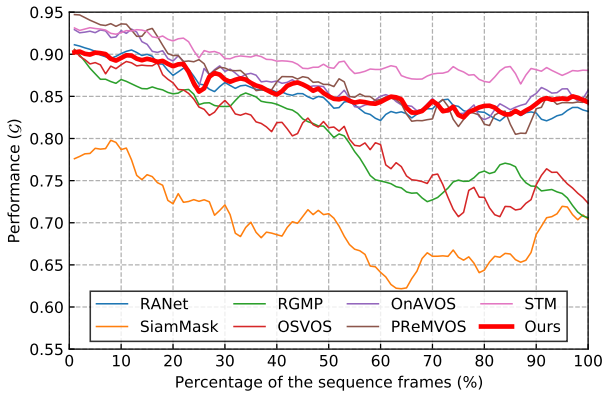


Fig. 6. Performance $\mathcal{G}$ under different percentages of the sequence lengths on the DAVIS-16 validation dataset.

network for video object segmentation (RVOS) [48], ranking attention network for fast video object segmentation (RANet) [55], fast end-to-end embedding learning for video object segmentation (FEELVOS) [49], fast online object tracking and segmentation (SiaMask) [53], state-aware tracker for real-time video object segmentation (SAT) [5], a generative appearance model for end-to-end video object segmentation (AGAME) [23], video object segmentation using space-time memory networks (STM) [36].

B. Comparison with State-of-the-Arts

1) DAVIS-16: To evaluate the effectiveness of our method, we provided comprehensive comparisons between our method with state-of-the-art ten online and nine offline methods on the DAVIS-16 validation set. Online methods are OSVOS [4], OSVOS-S [32], OnAVOS [51], CINM [1], MSK [37], Lucid [24], MoNet [57], SFL [9], PReMVOS [31], and MVOS [58]. Offline methods are RGMP [56], FAVOS [8], PML [41], OSMN [62], VideoMatch [20], RANet [55], SiamMask [53], FEELVOS [49], and STM [36]. We detailed key properties of each method including online fine-tuning, optical flow, and pre-trained datasets.

Table I presents experimental results of different semi-supervised video object segmentation methods. We clearly observe that our method achieves the best speed and accuracy trade-off without any fine-tuning, optical flow, post-posting, and data augmentation strategies. While STM obtains the best performance, this method highly relies on additional datasets including salient object detection and semantic segmentation datasets for pre-training. Without pre-training, the performance of STM drops sharply. Besides, this method requires more training time. SiamMask is one of the fastest approaches but its performance is unsatisfactory. Both VideoMatch and RANet are representative feature matching methods and exhibit the superior performance. However, VideoMatch conducts post-processing for outlier removal and online update, and RANet also needs many static image datasets for pre-training. Our method promotes the performance of feature matching methods by capturing the separable structure in the target frame.

To intuitively compare different semi-supervised video object segmentation methods, we provided the performance ($\mathcal{G}$) and speed (frame per second) comparisons between the proposed method and state-of-the-art methods on the DAVIS-16 validation dataset. Our method ran on a TITAN Xp GPU and the average time for processing a video frame was reported. Fig. 5 depicts performance ($\mathcal{G}$) and speed (frame per second) comparisons. We observe that our method achieves the best balance in terms of performance and speed. While online fine-tuning based methods including PReMVOS, OnAVOS, and OSVOS-S tend to have a high performance, their speeds are far from the real-time requirement. STM achieves the best

TABLE II
ATTRIBUTE-BASED ANALYSIS ON THE DAVIS-16 VALIDATION DATASET. THERE ARE FIFTEEN ATTRIBUTES, INCLUDING APPEARANCE CHANGES (AC), BACKGROUND CLUTTER (BC), CAMERA SHAKE (CS), DYNAMIC BACKGROUND (DB), NON-LINEAR DEFORMATION (DEF), EDGE AMBIGUITY (EA), FAST-MOTION (FM), HETEROGENEUS OBJECT (HO), INTERACTING OBJECTS (IO), LOW RESOLUTION (LR), MOTION BLUR (MB), OCCLUSIONS (OCC), OUT-OF-VIEW (OV), SHAPE COMPLEXITY (SC), SCALE VARIATION (SV) [38]. WE REPORTED THE AVERAGE $\mathcal{G}$ (%) OF VIDEOS WITH ONE ATTRIBUTE.

Metric	AC	BC	CS	DB	DEF	EA	FM	HO	IO	LR	MB	OCC	OV	SC	SV
$\mathcal{G}\uparrow$	89.2	89.0	87.9	78.5	84.4	81.8	86.7	82.3	79.4	90.2	82.0	84.6	82.8	74.7	83.4

TABLE III
QUANTITATIVE COMPARISON WITH STATE-OF-THE-ART ONLINE AND OFFLINE LEARNING BASED METHODS ON THE DAVIS-17 VALIDATION DATASET BY USING $\mathcal{J}$, $\mathcal{F}$, $\mathcal{G}$ METRICS.

Method	$\mathcal{G}\uparrow$	$\mathcal{J}_{mean}$	$\mathcal{J}_{decay}\downarrow$	$\mathcal{F}_{mean}$	$\mathcal{F}_{decay}\downarrow$
Online methods					
OSVOS [4]	60.3	56.6	26.1	63.9	27.0
OSVOS-S [32]	68.0	64.7	15.1	71.3	18.5
OnAVOS [51]	65.4	61.6	27.9	69.1	26.6
CINM [1]	70.6	67.1	24.6	74.1	26.2
MSK [37]	54.3	51.2	28.3	57.3	29.1
MVOS [58]	59.2	56.3	-	62.1	-
MoNet [57]	58.8	55.4	-	66.2	-
PReMVOS [31]	77.8	73.9	16.2	81.8	19.5
Offline methods					
RGMP [56]	66.7	64.8	18.9	68.6	19.6
FAVOS [8]	58.2	54.6	14.1	61.8	18.0
OSMN [62]	54.8	52.5	21.5	57.1	24.3
RVOS [48]	60.6	57.5	24.9	63.6	28.2
RANet [55]	65.7	63.2	18.6	68.2	19.7
Siam RCNN [52]	70.6	66.1	15.8	75.0	16.2
SiamMask [53]	53.1	51.1	-1.1	55.0	1.9
FEELVOS [49]	71.5	69.1	17.5	74.0	20.1
SAT [5]	72.3	68.6	13.6	76.0	-
AGAME [23]	71.0	68.5	14.0	73.6	18.5
STM [36]	81.7	79.2	8.0	84.3	10.5
Proposed method					
Ours	77.6	75.3	11.7	79.9	15.3

performance but its speed is relatively low compared with the latest methods. For feature matching methods, RANet and our method obtain a high speed and a comparable accuracy on the DAVIS-16 validation dataset. However, RANet carries a risk of model over-fitting as the pre-trained datasets are static images with only one single object.

Since semi-supervised video object segmentation only provides object information in the first frame, their segmentation performance decreases throughout video sequences. To analyze performance decays of different methods, we conducted experiments on the DAVIS-16 validation dataset. As different video sequences contain different numbers of frames, we first normalized the length of each sequence into $[0, 1]$ by dividing its sequence length, and then computed the percentage of the sequence length for each frame. We averaged all video sequences on the DAVIS-16 dataset and reported the mean $\mathcal{G}$ result under different percentages of the sequence length. Fig. 6 presents experimental results of different methods along the temporal domain. We clearly observe that the $\mathcal{G}$ performance for all methods drops over time. The reason is that due to appearance changes along video sequences, target objects are more dissimilar from the specified objects in the first frame. However, we see that the performance of our method decreases smoothly, validating the robustness of our method throughout video sequences. While PReMVOS and OnAVOS attain a better performance than our method before 20% of the sequence length, our method reports a comparable performance with these methods in the following frames. SiamMask, RGMP, and OSVOS are sensitive to appearance changes and a sharp performance decay is witnessed.

Furthermore, to evaluate the effectiveness of our method under different challenging factors, we conducted an attribute-based analysis on the DAVIS-16 validation dataset. Table II presents experimental results with different attributes such as appearance change, background clutter, and camera shake. We observe that our method achieves the best performance with low resolution and the worst performance with shape complexity. Moreover, dynamic background and edge ambiguity are also intractable.

2) DAVIS-17: We further conducted experiments on the DAVIS-17 validation set to verify the effectiveness of the proposed method for multiple object segmentation. Specifically, comparison methods consist of online fine-tuning based methods including OSVOS [4], OSVOS-S [32], OnAVOS [51], CINM [1], MSK [37], MoNet [57], MVOS [58], PReMVOS [31], and offline learning based methods including RGMP [56], FAVOS [8], OSMN [62], RVOS [48] RANet [55], AGAME [23], Siam RCNN [52], SiamMask [53], FEELVOS [49], SAT [5], STM [36].

Table III tabulates experimental results of different methods on the DAVIS-17 validation dataset. We see that our method achieves superior performance in comparison with state-of-the-art methods. Moreover, our method has the lowest decay rate for both the region similarity and contour accuracy except SiamMask. Due to difficulties in multiple object segmentation, the performance on the DAVIS-17 validation dataset is much lower than that on the DAVIS-16 validation dataset with the same method. While RANet achieves a comparable performance with our method on the DAVIS-16 validation dataset, it exhibits a worse result than our method on the DAVIS-17 dataset, proving that our method can effectively cope with multiple objects in videos. STM consistently exhibits the best performance by leveraging long-term information. PReMVOS provides a similar result, but this method is too complex due to the combination of segmentation, optical flow and Re-ID, and it is far away from the real-time requirement in practical applications. Methods including RANet, MVOS, MSK, OnAVOS, OSVOS, OSVOS-S exploit static image datasets for pre-training. These methods witness a performance drop on the DAVIS-17 validation dataset. The reason is that static image datasets only contain one single object in each frame and the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology

TABLE IV
QUANTITATIVE COMPARISON WITH STATE-OF-THE-ART ONLINE AND OFFLINE LEARNING BASED METHODS ON THE DAVIS-17 TEST-DEV DATASET BY USING $\mathcal{J}$, $\mathcal{F}$, $\mathcal{G}$ METRICS.

Method	$\mathcal{G}\uparrow$	$\mathcal{J}_{mean}$	$\mathcal{J}_{decay}\downarrow$	$\mathcal{F}_{mean}$	$\mathcal{F}_{decay}\downarrow$
Online methods					
OSVOS [4]	50.9	47.0	19.2	54.8	19.8
OSVOS-S [32]	57.5	52.9	24.1	62.1	21.9
OnAVOS [51]	52.8	49.9	23.0	55.7	23.4
CINM [1]	67.5	64.5	20.0	70.5	20.0
PReMVOS [31]	71.6	67.5	21.7	75.8	20.6
Offline methods					
RGMP [56]	52.8	51.3	34.3	54.4	37.2
FAVOS [8]	43.6	42.9	18.1	44.2	19.8
RVOS [48]	50.3	47.9	35.7	52.6	36.7
AGAME [23]	52.3	49.2	28.9	55.3	27.6
RANet [55]	55.4	53.4	21.9	57.3	22.1
Siam RCNN [52]	53.3	48.0	21.8	58.6	20.2
SiamMask [53]	43.2	40.6	21.9	45.8	22.4
FEELVOS [49]	57.8	55.1	29.8	60.4	33.5
AGAME [23]	52.3	49.2	28.9	55.3	27.6
STM [36]	72.2	69.3	16.9	75.2	17.5
Proposed method					
Ours	62.0	60.2	23.5	63.8	25.3

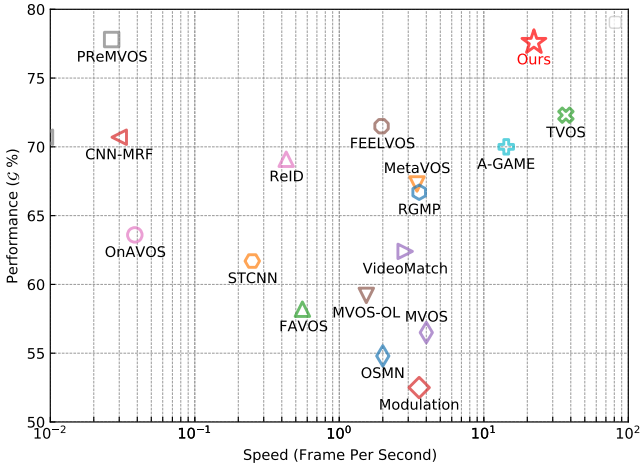


Fig. 7. Visualizations of performance and speed comparisons between our method and state-of-the-art methods on the DAVIS-17 validation dataset.

pre-training on these datasets easily leads to over-fitting.

Moreover, we provided quantitative comparisons with state-of-the-art methods. Table IV presents experimental results on the DAVIS-17 test-dev dataset by using different semi-supervised video object segmentation methods. We clearly see that our method consistently achieves promising performance on the DAVIS-17 test-dev dataset, and our method largely improves the segmentation performance of offline methods except STM. Since the predicted segmentation masks on the DAVIS-17 test-dev dataset are online submitted and evaluated, the ranking score of our method can be found on the website of the semi-supervised DAVIS challenge.

We also visualized performance and speed comparisons on the DAVIS-17 validation dataset to evaluate the effectiveness and efficiency of different methods for multiple object segmentation. In practice, most multiple object segmentation methods require a longer time than single object segmentation methods. As some methods only reported their runtime on the DAVIS-16 dataset, we did not show their results. The averaged time for processing one video frame was recorded with a TITAN Xp GPU. Fig. 7 depicts performance and speed comparisons on the DAVIS-17 validation dataset. We observe that our method consistently has a good balance between accuracy and speed, compared with state-of-the-art methods.

TABLE V
QUANTITATIVE COMPARISON WITH STATE-OF-THE-ART ONLINE AND OFFLINE LEARNING BASED METHODS ON THE YOUTUBE-VOS VALIDATION DATASET BY USING $\mathcal{J}$, $\mathcal{F}$ METRICS. *Seen* INDICATES THE OBJECT CATEGORIES ARE AVAILABLE IN THE TRAINING STAGE, WHILE *UnSeen* INDICATES THE OBJECT CATEGORIES ARE UNAVAILABLE IN THE TRAINING STAGE. OVERALL IS THE AVERAGED SCORE OF *Seen* AND *UnSeen* RESULTS.

Method	Overall	*Seen*		*UnSeen*	
		$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
Online methods					
OSVOS [4]	58.8	59.8	60.5	54.2	60.7
OnAVOS [51]	55.2	60.1	62.7	46.6	51.4
MSK [37]	53.1	59.9	59.5	45.0	47.9
PReMVOS [31]	66.9	71.4	75.9	56.5	63.7
Offline methods					
RGMP [56]	53.8	59.5	-	45.2	-
AGAME [23]	66.1	67.8	-	60.8	-
OSMN [62]	51.2	60.0	60.1	40.6	44.0
RVOS [48]	56.8	63.6	67.2	45.5	51.0
S2S (offline) [60]	57.6	66.7	48.2	65.5	50.3
CapsuleVOS [13]	62.3	67.3	53.7	68.1	59.9
SAT [5]	63.6	67.1	55.3	70.2	61.7
STM [36]	79.4	79.7	84.2	72.8	80.9
Proposed method					
Ours	66.5	72.3	57.8	73.3	62.6

TABLE VI
ABLATION STUDY ABOUT THE SEPARABLE STRUCTURE MODELING BRANCH. EXPERIMENTS WERE CONDUCTED ON THE DAVIS-16 AND DAVIS-17 DATASETS WITH (✓) OR WITHOUT (✗) THIS BRANCH.

Metric	DAVIS-16		DAVIS-17	
	✓	✗	✓	✗
$\mathcal{G}\uparrow$	85.9	84.5	77.6	76.0
$\mathcal{J}_{mean}\uparrow$	86.2	85.1	75.3	73.6
$\mathcal{J}_{recall}\uparrow$	97.1	96.8	85.7	82.9
$\mathcal{J}_{decay}\downarrow$	5.3	5.7	11.7	17.6
$\mathcal{F}_{mean}\uparrow$	85.6	83.9	79.9	78.4
$\mathcal{F}_{recall}\uparrow$	92.3	89.6	88.0	85.8
$\mathcal{F}_{decay}\downarrow$	5.6	7.2	15.3	21.0

3) YouTube-VOS: To evaluate the effectiveness and generalization performance of our method on a large-scale dataset, we conducted experiments on the YouTube-VOS dataset.

Table V presents experimental results on the YouTube-VOS dataset by using different methods. We clearly observe that our method attains very competitive performance on a large-scale dataset over state-of-the-art methods except STM. Our method also has a better generalization ability than most methods as the result of unseen object segmentation is higher than that of seen object segmentation. The reason may be that our feature matching branch and separable structure modeling branch are robust to objects. Notably, online fine-tuning based methods

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology

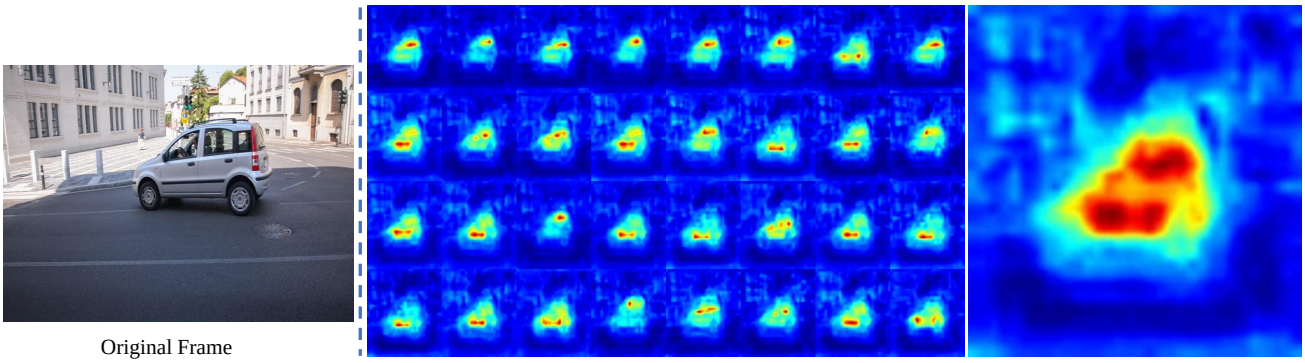


Fig. 8. Visualization of heat maps of individual and shared features from the separable structure modeling branch on the DAVIS-16 dataset. The original frame is shown on the left, and heat maps of individual features and shared feature are shown on the right. Best viewed in color.

TABLE VII
ABLATION STUDY ABOUT THE SEPARABLE STRUCTURE MODELING BRANCH (SEP.). EXPERIMENTS WERE CONDUCTED ON THE YOUTUBE-VOS DATASET WITH (✓) OR WITHOUT (✗) THIS BRANCH.

Dataset	Sep.	$\mathcal{G}$	$\mathcal{J}$ seen	$\mathcal{J}$ unseen	$\mathcal{F}$ seen	$\mathcal{F}$ unseen
YouTube-VOS	✓	66.5	72.3	57.8	73.3	62.6
	✗	65.0	71.1	56.6	71.7	60.5

TABLE VIII
ABLATION STUDY ABOUT DIFFERENT VOTING STRATEGIES. EXPERIMENTS WERE CONDUCTED ON THE DAVIS-16 AND DAVIS-17 DATASETS BY USING MAXIMUM (MAX.), AVERAGE (AVG.), AND TOP RANK (TOP.), RESPECTIVELY.

Metric	DAVIS-16			DAVIS-17		
	Max.	Avg.	Top.	Max.	Avg.	Top.
$\mathcal{G}\uparrow$	84.4	81.9	85.9	76.5	74.1	77.6
$\mathcal{J}_{mean}\uparrow$	85.1	81.5	86.2	74.5	72.3	75.3
$\mathcal{J}_{recall}\uparrow$	97.0	89.7	97.1	84.2	82.9	85.7
$\mathcal{J}_{decay}\downarrow$	5.1	11.0	5.3	13.2	14.4	11.7
$\mathcal{F}_{mean}\uparrow$	83.8	82.4	85.6	78.5	75.8	79.9
$\mathcal{F}_{recall}\uparrow$	89.2	89.3	92.3	86.9	84.8	88.0
$\mathcal{F}_{decay}\downarrow$	6.9	11.6	5.6	16.3	18.1	15.3

TABLE IX
ABLATION STUDY ABOUT DIFFERENT VOTING STRATEGIES (VOT.). EXPERIMENTS WERE CONDUCTED ON THE YOUTUBE-VOS DATASET BY USING MAXIMUM (MAX.), AVERAGE (AVG.), AND TOP RANK (TOP.).

Dataset	Vot.	$\mathcal{G}$	$\mathcal{J}$ seen	$\mathcal{J}$ unseen	$\mathcal{F}$ seen	$\mathcal{F}$ unseen
YouTube-VOS	Max.	66.1	71.8	57.8	72.8	62.1
	Avg.	64.5	70.9	55.9	71.6	59.5
	Top.	66.5	72.3	57.8	73.3	62.6

TABLE X
ABLATION STUDY ABOUT THE SPECIFIED NUMBER OF REPRESENTATIVE REFERENCE PIXELS. EXPERIMENTS WERE CONDUCTED ON THE DAVIS-16 AND DAVIS-17 DATASETS WITH 16, 32, AND 64 REFERENCE PIXELS, RESPECTIVELY.

Metric	DAVIS-16			DAVIS-17		
	16	32	64	16	32	64
$\mathcal{G}\uparrow$	84.2	85.9	83.9	76.0	77.6	75.7
$\mathcal{J}_{mean}\uparrow$	84.4	86.2	84.5	74.0	75.3	73.5
$\mathcal{J}_{recall}\uparrow$	95.3	97.1	96.7	84.3	85.7	83.6
$\mathcal{J}_{decay}\downarrow$	6.8	5.3	6.3	14.2	11.7	16.6
$\mathcal{F}_{mean}\uparrow$	84.1	85.6	83.2	78.0	79.9	77.9
$\mathcal{F}_{recall}\uparrow$	90.8	92.3	89.2	86.0	88.0	85.9
$\mathcal{F}_{decay}\downarrow$	6.1	5.6	8.0	18.4	15.3	20.2

usually have a poor generalization ability due to over-fitting. For STM and PReMVOS, the performance of unseen object segmentation drops sharply about 5.1 and 13.6. OSMN and RVOS have the worst generalization at a performance drop of 17.8 and 17.2, respectively.

C. Ablation Studies

In this subsection, we conducted ablation studies to investigate effects of the proposed structure modeling branch, voting strategies, the number of selected reference pixels, shared and individual features, and also provided time analyses.

1) Effect of Separable Structure Modeling Branch: Our method consists of the feature matching branch and the separable structure modeling branch. The separable structure modeling branch leverages target object information from the feature matching branch. To demonstrate the contribution of the separable structure modeling, we conducted experiments on the DAVIS-16, DAVIS-17, and YouTube-VOS validation datasets with (✓) or without (✗) this branch. Table VI and Table VII tabulate experimental results under different settings. We see that our method with the separable structure modeling branch obtains better performance than that without this branch on the DAVIS-16, DAVIS-17, and YouTube-VOS validation datasets, which confirms that the separable structure modeling branch contributes to performance improvement.

2) Effect of Voting Strategies: In the feature matching branch, we selected top rank reference pixels of high confidence to classify target pixels. We analyzed the influence of different voting strategies including average and maximum of all foreground or background pixels. Table VIII and Table IX show experimental results on the DAVIS-16 DAVIS-17, and YouTube-VOS datasets. We observe that the top rank strategy achieves the best performance and the average strategy obtains the worst result. The reason may be that it is unreasonable to match all foreground or background pixels in the reference

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology

TABLE XI
ABLATION STUDY ABOUT THE SPECIFIED NUMBER OF REPRESENTATIVE REFERENCE PIXELS. EXPERIMENTS WERE CONDUCTED ON THE YOUTUBE-VOS DATASET WITH 16, 32, AND 64 REFERENCE PIXELS.

Dataset	#Pixel	$\mathcal{G}$	$\mathcal{J}$ seen	$\mathcal{J}$ unseen	$\mathcal{F}$ seen	$\mathcal{F}$ unseen
YouTube-VOS	16	65.7	71.9	56.8	72.9	61.2
	32	66.5	72.3	57.8	73.3	62.6
	64	64.9	71.5	55.7	72.3	60.0

TABLE XII
ABLATION STUDY ABOUT INDIVIDUAL AND SHARED FEATURES IN THE SEPARABLE STRUCTURE MODELING BRANCH. EXPERIMENTS WERE CONDUCTED ON THE DAVIS-16 AND DAVIS-17 DATASETS WITHOUT INDIVIDUAL FEATURES (I) OR WITHOUT (S) SHARED FEATURE.

Metric	DAVIS-16		DAVIS-17	
	w/o S	w/o I	w/o S	w/o I
$\mathcal{G}\uparrow$	85.1	84.2	77.3	76.2
$\mathcal{J}_{mean}\uparrow$	85.6	84.4	74.9	73.9
$\mathcal{J}_{recall}\uparrow$	96.4	95.3	85.1	83.0
$\mathcal{J}_{decay}\downarrow$	6.3	6.8	12.6	14.8
$\mathcal{F}_{mean}\uparrow$	84.7	84.1	79.6	78.5
$\mathcal{F}_{recall}\uparrow$	92.0	90.8	87.6	86.0
$\mathcal{F}_{decay}\downarrow$	6.3	6.1	15.7	17.9

frame to a target pixel and target pixels should match local regions in the reference frame. Therefore, the average strategy attains the worst performance. Moreover, for the maximum strategy, the reference pixel with the maximal ranking score is easily affected by outliers and noisy pixels.

3) Effect of the Selected Number of Reference Pixels: For the top rank voting strategy, the number of selected pixels has a great influence on the feature matching and structure modeling branches. To evaluate the effect of this parameter, we conducted ablation studies with 16, 32, and 64 representative reference pixels. Table X and Table XI present experimental results on the DAVIS-16, DAVIS-17, and YouTube-VOS datasets with different numbers of selected reference pixels. We clearly observe that our method achieves the best and worst performance when the numbers of selected pixels are set as 32 and 64, respectively. Notably, selecting too many reference pixels may lead to a wrong selection, which is a much severer issue. When the pixel number is set as 16, we observe that our method still obtains inferior results as they cannot fully represent target objects.

4) Effect of Individual and Shared Features: To analyze the influence of individual and shared features, we further conducted ablation studies without either individual or shared features. Table XII and Table XIII present experimental results on the DAVIS-16, DAVIS-17, and YouTube-VOS validation datasets. We observe that without either individual features or the shared feature, the performance of our method drops. Individual features play a more important role than the shared feature because individual features are attentive to detailed information about object components.

5) Time Analyses: Moreover, we provided time analyses about the multiple object segmentation extension. Table XIV tabulates the average inference time for processing one frame on DAVIS-16, DAVIS-17, and YouTube-VOS. We clearly observe that our multiple object extension strategy is much more time-efficient than the one-by-one strategy on the DAVIS-17 and YouTube-VOS datasets. Since DAVIS-16 is a single object segmentation dataset, the average inference time for the one-by-one strategy and our strategy is the same. DAVIS-17 and YouTube-VOS are multiple object segmentation datasets and our method takes less time to process one frame in inference.

TABLE XIII
ABLATION STUDY ABOUT INDIVIDUAL AND SHARED FEATURES IN THE SEPARABLE STRUCTURE MODELING BRANCH. EXPERIMENTS WERE CONDUCTED ON THE YOUTUBE-VOS DATASET WITH (✓) OR WITHOUT (✗) INDIVIDUAL (I) AND SHARED (S) FEATURES.

Dataset	S	I	$\mathcal{G}$	$\mathcal{J}$ seen	$\mathcal{J}$ unseen	$\mathcal{F}$ seen	$\mathcal{F}$ unseen
YouTube-VOS	✗	✓	65.7	71.6	57.4	72.5	61.3
	✓	✗	65.0	71.5	56.2	72.0	60.1
	✓	✓	66.5	72.3	57.8	73.3	62.6

TABLE XIV
THE AVERAGED INFERENCE TIME (MILLISECOND) FOR PROCESSING EACH VIDEO FRAME ON THE DAVIS-16, DAVIS-17, AND YOUTUBE-VOS DATASETS, RESPECTIVELY.

Dataset	Ours	One-by-one
DAVIS-16	27.4	27.4
DAVIS-17	44.9	55.2
YouTube-VOS	41.3	47.6

D. Visualization and Qualitative Results

In this subsection, we first performed feature visualizations for a better understanding of the structure modeling branch and then visualized segmentation results of example videos.

1) Visualizations of Separable Structure Modeling Branch: In the separable structure modeling branch, we selected 32 target pixels to represent object components, where individual features and the shared feature were learned by using attentive maps. We computed heat maps by averaging these features in the dimension direction. Fig. 8 depicts heat maps of individual features and the shared feature on the DAVIS-16 dataset. We clearly observe that heat maps of individual features have high responses on object components, while that of the shared feature has a high response on the whole object. Visualizations validate that our individual and shared features can well focus on object components and the whole object.

2) Visualizations of Example Videos: We further presented some example segmentations to intuitively evaluate qualitative results of the proposed method. Fig. 9 depicts the predicted segmentation masks on the DAVIS-16, DAVIS-17, and YouTube-VOS datasets. Example video sequences consist of several challenges including background clutter (the first example), interacting objects (the second, third, fourth, and fifth examples), fast motion (the second and fourth examples), occlusion (the third and fourth examples), and non-linear deformation (the first, second, and fifth examples). According to Fig. 9 and Table II, we clearly observe that our method produces accurate segmentation results on the first, second, and third videos, which mainly contain non-linear deformation, fast motion, and background clutter. Furthermore, we

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3060015, IEEE Transactions on Circuits and Systems for Video Technology

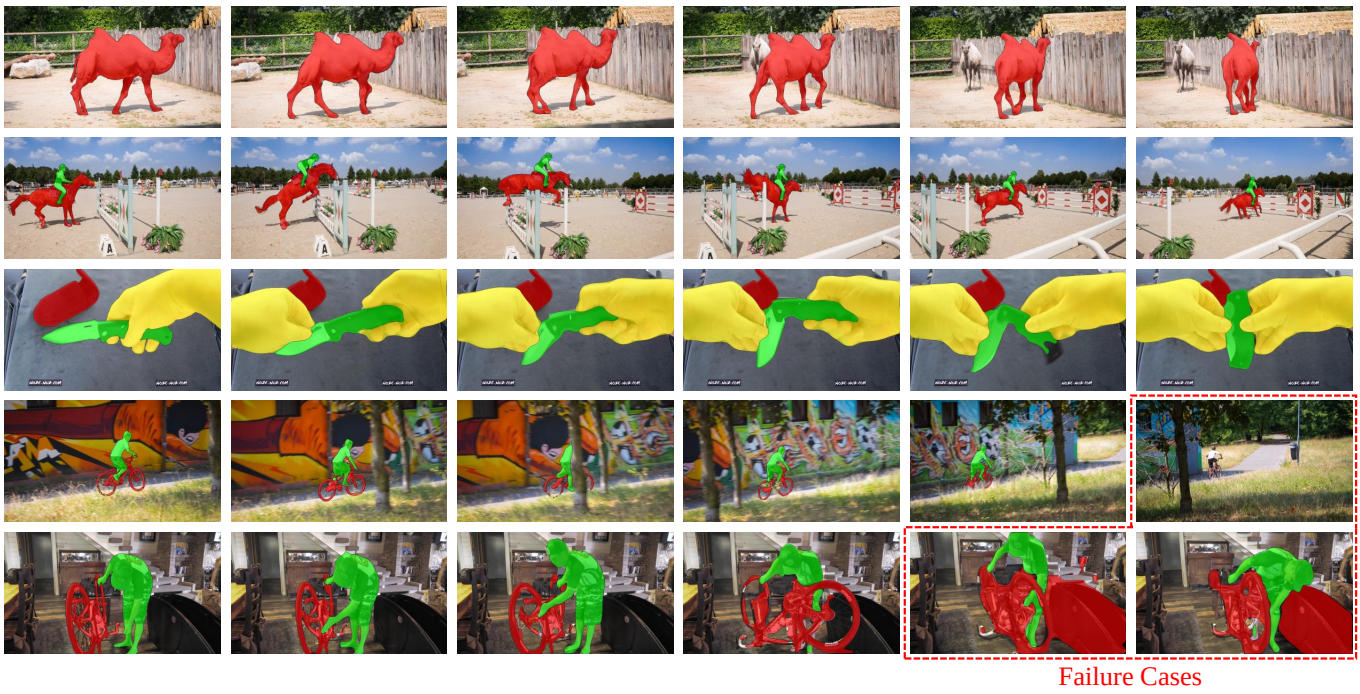


Fig. 9. Visualizations of segmentation results on the DAVIS-16, DAVIS-17, and YouTube-VOS datasets. Failure cases on the DAVIS-17 dataset are also depicted in the last two rows. Best viewed in color.

also provided failure cases on the DAVIS-17 dataset in the last two rows. In the fourth video, our method fails in passing through a tree when occlusion happens. Moreover, in the fifth video, the proposed method unfortunately mistakes pixels from the closest object. These two visualization results indicate that interacting objects and occlusion cause confusion to our method.

V. Conclusion

In this paper, we have proposed a fast semi-supervised video object segmentation method. Our method consists of the feature matching branch and the separable structure modeling branch. The feature matching branch matches annotated reference pixels to target pixels, while the separable structure modeling branch leverages object prior information from the feature matching branch, and models components of target objects by using the learned individual and shared representations. Finally, representations from these two branches are concatenated and are further fed into the decoder for object segmentation. Experimental results on three standard datasets have verified the effectiveness and efficiency of our method. In the future, we will attempt to incorporate long-range temporal information into our framework.

References

[1] L. Bao, B. Wu, and W. Liu, "Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5977–5986. 1, 7, 8, 9, 10

[2] D. Bau, J.-Y. Zhu, H. Strobelt, Z. Bolei, J. B. Tenenbaum, W. T. Freeman, and A. Torralba, "Gan dissection: Visualizing and understanding generative adversarial networks," in *International Conference on Learning Representations*, 2019. 1

[3] H. S. Behl, M. Najafi, and P. H. Torr, "Meta learning deep visual words for fast video object segmentation," *arXiv*, 2018. 3, 5

[4] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool, "One-shot video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 221–230. 1, 2, 7, 8, 9, 10

[5] X. Chen, Z. Li, Y. Yuan, G. Yu, J. Shen, and D. Qi, "State-aware tracker for real-time video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9384–9393. 3, 8, 9, 10

[6] Y.-W. Chen, Y.-H. Tsai, Y.-Y. Lin, and M.-H. Yang, "Vostr: Video object segmentation via transferable representations," *International Journal of Computer Vision*, vol. 128, no. 4, pp. 931–949, 2020. 1

[7] Y. Chen, J. Pont-Tuset, A. Montes, and L. Van Gool, "Blazingly fast video object segmentation with pixel-wise metric learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1189–1198. 1, 2, 3, 4, 7

[8] J. Cheng, Y.-H. Tsai, W.-C. Hung, S. Wang, and M.-H. Yang, "Fast and accurate online video object segmentation via tracking parts," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7415–7424. 2, 3, 7, 8, 9, 10

[9] J. Cheng, Y.-H. Tsai, S. Wang, and M.-H. Yang, "Segflow: Joint learning for video object segmentation and optical flow," in *IEEE International Conference on Computer Vision*, 2017, pp. 686–695. 1, 2, 7, 8

[10] S.-Y. Chien, W.-K. Chan, Y.-H. Tseng, and H.-Y. Chen, "Video object segmentation and tracking framework with improved threshold decision and diffusion distance," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 6, pp. 921–934, 2013. 1

[11] H. Ci, C. Wang, and Y. Wang, "Video object segmentation by learning location-sensitive embeddings," in *European Conference on Computer Vision*, 2018, pp. 501–516. 3

[12] H. De Vries, F. Strub, J. Mary, H. Larochelle, O. Pietquin, and A. C. Courville, "Modulating early visual processing by language," in *Advances in Neural Information Processing Systems*, 2017, pp. 6594–6604. 3

[13] K. Duarte, Y. S. Rawat, and M. Shah, "Capsulevos: Semi-supervised video object segmentation using capsule routing," in *IEEE International Conference on Computer Vision*, 2019, pp. 8480–8489. 3, 10

[14] K. Gong, X. Liang, D. Zhang, X. Shen, and L. Lin, "Look into person: Self-supervised structure-sensitive learning and a new benchmark for human parsing," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 932–940. 2, 3, 5

[15] B. A. Griffin and J. J. Corso, "Bubblenets: Learning to select the guidance frame in video object segmentation by deep sorting frames," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8914–8923. 1

[16] Y. Gui, Y. Tian, D.-J. Zeng, Z.-F. Xie, and Y.-Y. Cai, "Reliable and dynamic appearance modeling and label consistency enforcing for fast and coherent video object segmentation with the bilateral grid," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 12, pp. 4781–4795, 2019. 1

[17] P. Hu, G. Wang, X. Kong, J. Kuen, and Y.-P. Tan, "Motion-guided cascaded refinement network for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1400–1409. 1

[18] Y.-T. Hu, H.-S. Chen, K. Hui, J.-B. Huang, and A. G. Schwing, "Sail-vos: Semantic amodal instance level video object segmentation-a synthetic dataset and baselines," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3105–3115. 1

[19] Y.-T. Hu, J.-B. Huang, and A. Schwing, "Maskrnn: Instance level video object segmentation," in *Advances in Neural Information Processing Systems*, 2017, pp. 325–334. 3

[20] Y.-T. Hu, J.-B. Huang, and A. G. Schwing, "Videomatch: Matching based video object segmentation," in *European Conference on Computer Vision*, 2018, pp. 54–70. 1, 2, 3, 4, 7, 8

[21] X. Huang, J. Xu, Y.-W. Tai, and C.-K. Tang, "Fast video object segmentation with temporal aggregation network and dynamic template matching," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8879–8889. 3

[22] W.-D. Jang and C.-S. Kim, "Online video object segmentation via convolutional trident network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5849–5858. 1, 2

[23] J. Johnander, M. Danelljan, E. Brissman, F. S. Khan, and M. Felsberg, "A generative appearance model for end-to-end video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8953–8962. 3, 8, 9, 10

[24] A. Khoreva, R. Benenson, E. Ilg, T. Brox, and B. Schiele, "Lucid data dreaming for video object segmentation," *International Journal of Computer Vision*, vol. 127, no. 9, pp. 1175–1197, 2019. 1, 2, 7, 8

[25] X. Li and C. Change Loy, "Video object segmentation with joint re-identification and attention-aware mask propagation," in *European Conference on Computer Vision*, 2018, pp. 90–105. 3

[26] H. Lin, X. Qi, and J. Jia, "Agss-vos: Attention guided single-shot video object segmentation," in *IEEE International Conference on Computer Vision*, 2019, pp. 3949–3957. 3

[27] W. Liu, G. Lin, T. Zhang, and Z. Liu, "Guided co-segmentation network for fast video object segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2020. 1

[28] X. Lu, W. Wang, C. Ma, J. Shen, L. Shao, and F. Porikli, "See more, know more: Unsupervised video object segmentation with co-attention siamese networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3623–3632. 1

[29] X. Lu, W. Wang, D. Martin, T. Zhou, J. Shen, and V. G. Luc, "Video object segmentation with episodic graph memory networks," in *European conference on computer vision*, 2020, pp. 661–679. 3

[30] X. Lu, W. Wang, J. Shen, Y.-W. Tai, D. J. Crandall, and S. C. Hoi, "Learning video object segmentation from unlabeled videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8960–8970. 1, 3, 5

[31] J. Luiten, P. Voigtlaender, and B. Leibe, "Premvos: Proposal-generation, refinement and merging for video object segmentation," in *Asian Conference on Computer Vision*, 2018, pp. 565–580. 3, 7, 8, 9, 10

[32] K.-K. Maninis, S. Caelles, Y. Chen, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool, "Video object segmentation without temporal information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 6, pp. 1515–1530, 2018. 1, 2, 7, 8, 9, 10

[33] N. Märki, F. Perazzi, O. Wang, and A. Sorkine-Hornung, "Bilateral space video segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 743–751. 1

[34] J. Miao, Y. Wei, and Y. Yang, "Memory aggregation networks for efficient interactive video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 366–10 375. 1

[35] S. W. Oh, J.-Y. Lee, N. Xu, and S. J. Kim, "Fast user-guided video object segmentation by interaction-and-propagation networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5247–5256. 1

[36] S. W. Oh, J.-Y. Lee, N. Xu, and S. J. Kim, "Video object segmentation using space-time memory networks," in *IEEE International Conference on Computer Vision*, 2019, pp. 9226–9235. 3, 7, 8, 9, 10

[37] F. Perazzi, A. Khoreva, R. Benenson, B. Schiele, and A. Sorkine-Hornung, "Learning video object segmentation from static images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2663–2672. 1, 2, 3, 7, 8, 9, 10

[38] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 724–732. 6, 9

[39] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool, "The 2017 davis challenge on video object segmentation," *arXiv*, 2017. 1, 6

[40] A. Robinson, F. J. Lawin, M. Danelljan, F. S. Khan, and M. Felsberg, "Learning fast and robust target models for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7406–7415. 1, 2

[41] J. Shin Yoon, F. Rameau, J. Kim, S. Lee, S. Shin, and I. So Kweon, "Pixel-level matching for video object segmentation using convolutional neural networks," in *IEEE International Conference on Computer Vision*, 2017, pp. 2167–2176. 3, 4, 7, 8

[42] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam, "Pyramid dilated deeper convlstm for video salient object detection," in *European conference on computer vision*, 2018, pp. 715–731. 1, 5

[43] Z. Tan, B. Liu, Q. Chu, H. Zhong, Y. Wu, W. Li, and N. Yu, "Real time video object segmentation in compressed domain," *IEEE Transactions on Circuits and Systems for Video Technology*, 2020. 1

[44] P. Tokmakov, C. Schmid, and K. Alahari, "Learning to segment moving objects," *International Journal of Computer Vision*, vol. 127, no. 3, pp. 282–301, 2019. 1

[45] Y.-H. Tsai, M.-H. Yang, and M. J. Black, "Video segmentation via object flow," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3899–3908. 1

[46] Z. Tu, W. Xie, J. Dauwels, B. Li, and J. Yuan, "Semantic cues enhanced multimodality multistream cnn for action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 5, pp. 1423–1437, 2018. 1

[47] G. Vecchio, S. Palazzo, D. Giordano, F. Rundo, and C. Spampinato, "Mask-rl: Multiagent video object segmentation framework through reinforcement learning," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2020. 1

[48] C. Ventura, M. Bellver, A. Girbau, A. Salvador, F. Marques, and X. Giro-i Nieto, "Rvos: End-to-end recurrent network for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5277–5286. 1, 2, 3, 8, 9, 10

[49] P. Voigtlaender, Y. Chai, F. Schroff, H. Adam, B. Leibe, and L.-C. Chen, "Feelvos: Fast end-to-end embedding learning for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9481–9490. 3, 7, 8, 9, 10

[50] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, B. B. G. Sekar, A. Geiger, and B. Leibe, "Mots: Multi-object tracking and segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7942–7951. 1, 2, 3

[51] P. Voigtlaender and B. Leibe, "Online adaptation of convolutional neural networks for video object segmentation," *arXiv*, 2017. 1, 2, 7, 8, 9, 10

[52] P. Voigtlaender, J. Luiten, P. H. Torr, and B. Leibe, "Siam r-cnn: Visual tracking by re-detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6578–6588. 1, 9, 10

[53] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. Torr, "Fast online object tracking and segmentation: A unifying approach," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1328–1338. 1, 2, 3, 6, 7, 8, 9, 10

[54] W. Wang, H. Song, S. Zhao, J. Shen, S. Zhao, S. C. Hoi, and H. Ling, "Learning unsupervised video object segmentation through visual attention," in *IEEE conference on computer vision and pattern recognition*, 2019, pp. 3064–3074. 1

[55] Z. Wang, J. Xu, L. Liu, F. Zhu, and L. Shao, "Ranet: Ranking attention network for fast video object segmentation," in *IEEE International Conference on Computer Vision*, 2019, pp. 3978–3987. 1, 2, 3, 7, 8, 9, 10

[56] S. Wug Oh, J.-Y. Lee, K. Sunkavalli, and S. Joo Kim, "Fast video object segmentation by reference-guided mask propagation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7376–7385. 1, 2, 3, 6, 7, 8, 9, 10

[57] H. Xiao, J. Feng, G. Lin, Y. Liu, and M. Zhang, "Monet: Deep motion exploitation for video object segmentation," in *IEEE Conference on*

Computer Vision and Pattern Recognition, 2018, pp. 1140–1148. 7, 8, 9

[58] H. Xiao, B. Kang, Y. Liu, M. Zhang, and J. Feng, "Online meta adaptation for fast video object segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 5, pp. 1205–1217, 2019. 2, 7, 8, 9

[59] K. Xu, L. Wen, G. Li, L. Bo, and Q. Huang, "Spatiotemporal cnn for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1379–1388. 1, 2

[60] N. Xu, L. Yang, Y. Fan, J. Yang, D. Yue, Y. Liang, B. Price, S. Cohen, and T. Huang, "Youtube-vos: Sequence-to-sequence video object segmentation," in *European Conference on Computer Vision*, 2018, pp. 585–601. 3, 6, 10

[61] S. Xu, D. Liu, L. Bao, W. Liu, and P. Zhou, "Mhp-vos: Multiple hypotheses propagation for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 314–323. 1

[62] L. Yang, Y. Wang, X. Xiong, J. Yang, and A. K. Katsaggelos, "Efficient video object segmentation via network modulation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6499–6507. 1, 3, 7, 8, 9, 10

[63] Z. Yang, Q. Wang, L. Bertinetto, W. Hu, S. Bai, and P. H. Torr, "Anchor diffusion for unsupervised video object segmentation," in *IEEE International Conference on Computer Vision*, 2019, pp. 931–940. 1

[64] X. Zeng, R. Liao, L. Gu, Y. Xiong, S. Fidler, and R. Urtasun, "Dmm-net: Differentiable mask-matching network for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3929–3938. 2, 3

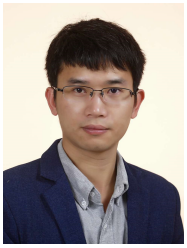
[65] L. Zhang, Z. Lin, J. Zhang, H. Lu, and Y. He, "Fast video object segmentation via dynamic targeting network," in *IEEE International Conference on Computer Vision*, 2019, pp. 5582–5591. 1

[66] Y. Zhang, Z. Wu, H. Peng, and S. Lin, "A transductive approach for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6949–6958. 3

[67] M. Zhou, Y. Bai, W. Zhang, T. Zhao, and T. Mei, "Look-into-object: Self-supervised structure modeling for object recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 774–11 783. 2, 3, 5

[68] T. Zhou, S. Wang, Y. Zhou, Y. Yao, J. Li, and L. Shao, "Motion-attentive transition for zero-shot video object segmentation." in *AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, 2020, pp. 13 066–13 073. 1

[69] T. Zhuo, Z. Cheng, P. Zhang, Y. Wong, and M. Kankanhalli, "Unsupervised online video object segmentation with motion property understanding," *IEEE Transactions on Image Processing*, vol. 29, pp. 237–249, 2019. 1



Wencheng Zhu received the BS and MS degree both in school of computer science and technology from Tianjin University, China, in 2014 and 2017, respectively. He is currently working toward the Ph.D. degree in the Department of Automation, Tsinghua University, China. His research interests include video summarization and video object segmentation. He serves as a regular reviewer member for the IEEE Transactions on Image Processing, IEEE Transactions on Circuits and Systems for Video Technology, Pattern Recognition.

Jiahao Li is a senior undergraduate student at Tsinghua University, Beijing, China, advised by Dr. Jiwen Lu at the Department of Automation of Tsinghua University. His current research interests include computer vision and deep learning.

Jiwen Lu (M11-SM15) received the B.Eng. degree in mechanical engineering and the M.Eng. degree in electrical engineering from the Xian University of Technology, Xian, 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 intelligent robotics, where he has authored/co-authored over 270 scientific papers in these areas. He serves the Co-Editor-of-Chief of the Pattern Recognition Letters, an Associate Editor of 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 the Pattern Recognition journal. He also serves as the General Co-Chair of IEEE ICME'2022, and the Program Co-Chair of IEEE FG'2023, IEEE VCIP'2022, IEEE AVSS'2021 and IEEE ICME'2020. He is an IAPR Fellow.

Jie Zhou (M01-SM04) received the BS and MS degrees both from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the PhD 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 served as a postdoctoral fellow in the Department of Automation, Tsinghua University, Beijing, China. Since 2003, he has been a full professor in the Department of Automation, Tsinghua University. 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 is an associate editor for the IEEE Transactions on Pattern Analysis and Machine Intelligence and two other journals. He received the National Outstanding Youth Foundation of China Award. He is an IAPR Fellow.