# Transductive Video Segmentation on Tree-Structured Model

Botao Wang, *Student Member, IEEE*, Zhihui Fu, Hongkai Xiong, *Senior Member, IEEE*, and Yuan F. Zheng, *Fellow, IEEE*

***Abstract*— This paper presents a transductive multicomponent video segmentation algorithm, which is capable of segmenting the predefined object of interest in the frames of a video sequence. To ensure temporal consistency, a temporal coherent parametric min-cut algorithm is developed to generate segmentation hypotheses based on visual cues and motion cues. Furthermore, each hypothesis is evaluated by an energy function from foreground resemblance, foreground/background divergence, boundary strength, and visual saliency. In particular, the state-of-the-art R-convolutional neural network descriptor is leveraged to encode the visual appearance of the foreground object. Finally, the optimal segmentation of the frame can be attained by assembling the segmentation hypotheses through the Monte Carlo approximation. In particular, multiple foreground components are built to capture the variances of the foreground object in shapes and poses. To group the frames into different components, a tree-structured graphical model named temporal tree is designed, where visually similar and temporally coherent frames are arranged in branches. The temporal tree can be constructed by iteratively adding frames to the active nodes by probabilistic clustering. In addition, each component, consisting of frames in the same branch, is characterized by a support vector machine classifier, which is learned in a transductive fashion by jointly maximizing the margin over the labeled frames and the unlabeled frames. As the frames from the same video sequence follow the same distribution, the transductive classifiers achieve stronger generalization capability than inductive ones. Experimental results on the public benchmarks demonstrate the effectiveness of the proposed method in comparison with other state-of-the-art supervised and unsupervised video segmentation methods.**

***Index Terms*— Monte Carlo approximation, parametric min-cut, temporal tree, transductive learning, video segmentation.**

## I. Introduction

VIDEO segmentation has many important applications in computer vision, such as object detection, visual tracking, and action recognition. Current video segmentation algorithms can be broadly divided into two approaches: unsupervised and supervised.

Unsupervised video segmentation [1]–[6] has received more attention in recent years than its supervised counterpart due to its automatic capability in the era of big data. The goal of the unsupervised video segmentation is to automatically pop up the primary object in the video [1]–[4] or group the pixels into spatiotemporal supervoxels based on the visual and motion cues [5]. However, unsupervised methods always discover the most motionally distinct and visually salient object in the video, which may not be the object that people are actually interested in. On the other hand, oversegmentation is also a serious problem for motion segmentation techniques [5].

Supervised video segmentation [7]–[10], also known as interactive video segmentation, also plays an important role in many applications, which unsupervised approaches cannot handle properly. The goal of the supervised video segmentation is to segment the object specified by the user, and the problem is more well defined, i.e., to separate the foreground object of interest from the background in the frames of the video. Many graph-based supervised video segmentation methods [8], [9], [11] use a single model to characterize the foreground object in the video. However, their performance is compromised if the object exhibits significant variations in visual appearance across frames. Tracking-based methods [3], [12] associate the object proposals across frames through visual tracking. However, tracking techniques are not always reliable, especially for objects with a significant visual variation in the video. Fortunately, provided with the labeled key frames distributed across the frames, the proposed method is capable of capturing different visual appearances of the object more accurately.

The proposed algorithm belongs to the category of the unsupervised video segmentation, and aims at segmenting the predefined object in video so that it is more well suited to high-level computer vision tasks, such as image retrieval, object detection, and video summarization. In addition, the proposed method is effective for static objects, provided with some partially labeled frames. The proposed method constructs multiple foreground models to represent the foreground object, which is more robust against variations in poses, viewpoint, and so on.

The task of segmenting common object in multiple images gives rise to image cosegmentation, which was first introduced in [13]. The main challenge of adopting image cosegmentation for video segmentation is that the background of the frames in the same video sequence is also highly correlated,

Manuscript received October 16, 2015; revised November 30, 2015; accepted January 16, 2016. Date of publication February 8, 2016; date of current version May 3, 2017. This work was supported in part by the National Science Foundation of China under Grant 61425011, Grant U1201255, Grant 61271218, Grant 61529101, Grant 61472234, and Grant 61271211, and in part by the Shu Guang Project under Grant 13SG13. This paper was recommended by Associate Editor P. Salembier.

B. Wang, Z. Fu, and H. Xiong are with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: botaowang@sjtu.edu.cn; yukin@sjtu.edu.cn; xionghongkai@sjtu.edu.cn).

Y. F. Zheng is with the Department of Electrical and Computer Engineering, Ohio State University, Columbus, OH 43210 USA (e-mail: zheng@ece.osu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2016.2527378

so it is not possible to distinguish between the foreground region and the background region from visual correlation. A couple of methods [14]–[19] adopt image cosegmentation techniques for video segmentation. However, those methods segment common foreground object in multiple videos with diverse background. In contrast, this paper is dedicated to segmenting the foreground object in the frames of a single video sequence, which distinguishes our work from other related studies. To disambiguate the foreground and the background in a video sequence, certain prior knowledge will be desired, which is provided by some prelabeled frames. To obtain accurate segmentation in the unlabeled frames, the proposed method jointly minimizes the prediction error in the labeled frames and the unlabeled frames through transductive learning, which achieves stronger generalization capability than inductive methods.

This paper makes two major technical contributions. First, a multicomponent temporal-coherent video segmentation algorithm is proposed. Motivated by image cosegmentation, the proposed algorithm segments the foreground object in multiple frames of a video sequence simultaneously by maximizing the inter-frame similarity of the foreground region and the intra-frame foreground/background divergence. In particular, instead of treating the frames in a video sequence as a collection of independent images, the proposed algorithm pursues temporal consistency in segmentation so that the segmentation results can be both visually consistent and temporally coherent. To be concrete, the temporal consistency is enforced by generating segmentation hypotheses with not only the low-level visual cues from the current frame but also the motion cues from consecutive frames. In addition, an energy function is carefully designed to evaluate the segmentation hypotheses from four aspects: the resemblance to the foreground model, the foreground/background divergence, the boundary strength, and the visual saliency. Finally, the optimal segmentations of the frames can be obtained by assembling hypotheses weighted by their energies through Monte Carlo approximation.

The second contribution of this paper is to present a transductive learning algorithm to estimate the hyperparameters of the multicomponent foreground model. Rather than using only one model to encode the foreground object, we construct a multicomponent model to characterize the variation of the object across frames. Each component captures a unique appearance of the object so that the proposed model is more robust for videos in which the objects exhibit significant changes in visual appearance. In particular, a temporal tree is constructed to group the frames into different components by probabilistic clustering. Consequently, each branch in the temporal tree is composed of visually and temporally consistent frames, which corresponds to a component. The visual appearance of each component is characterized by an support vector machine (SVM) classifier, which is trained in a transductive manner by maximizing the margin of the labeled frames and the unlabeled frames. As the labeled frames and the unlabeled frames in the same video sequence are generated from the same distribution, the transductive learning algorithm provides stronger generalization capability than inductive approaches because of the cluster assumption. The experimental results demonstrate that the proposed method outperforms many state-of-the-art video segmentation algorithms in public benchmarks.

The rest of this paper is organized as follows: Section II reviews some related work. Section III briefly introduces the overview of the proposed algorithm. Section IV describes the proposed temporal coherent video segmentation algorithm. Section V presents the transductive learning algorithm to attain the hyperparameters of the multicomponent model. Section VI gives the experimental results. Finally, the conclusion is drawn in Section VII.

## II. Related Work

### *A. Unsupervised Video Segmentation*

The unsupervised video segmentation [1]–[6] is targeted at grouping the pixels in the video into supervoxels, which are both photometrically and temporally consistent. Ma and Latecki [2] formulated the problem of video object segmentation as finding the maximum clique in a weighted region graph. To ensure the reliability of the potentials, two types of mutex constraints are designed, which can be expressed in a single quadratic form. Zhang *et al.* [1] built a layered directed acyclic graph over the object proposals and formulated the problem of primary object extraction as finding the longest path in the graph. Likewise, Cao *et al.* [4] and Zhang *et al.* [6] also developed a graphical model over the segmentation candidates and selected the most corresponding object region in each frame by finding the shortest path in the graph. Inspired by video retargeting, Ramakanth and Babu [3] used seams to propagate temporal labels across the frames to segment object in videos. Wang *et al.* [20] introduced an unsupervised, geodesic distance based, salient video object segmentation method, which extracts objects from the saliency of the geodesic distance to the spatial edges [21] and motion boundaries. Luo *et al.* [22] divided a long video sequence into consistent shot cuts and segmented objects in relative video shots. Focused on a graph construction for video segmentation, Khoreva *et al.* [23] used the calibrated classifier outputs as edge weights and defined the graph topology by edge selection.

Tracking-based approaches segment videos by tracking interest points [5] or regions [12], [24] followed by merging or clustering. Ochs *et al.* [5] established long-term motion cues that span the whole video shot, so that static objects and occlusions can be well handled. Li *et al.* [24] performed video segmentation by simultaneously tracking multiple holistic figure-ground segments (FGSs), and incrementally training online nonlocal appearance models for each track using a multi-output regularized least squares formulation. Varas and Marques [12] extended particle filtering for video object segmentation and segmented the current frame by coclustering with the object partition of the previous frame. In general, interest points are more distinctive and robust for tracking, whereas regions are the characteristic of the spatial coverage of objects. As unsupervised video segmentation techniques are generally bottom-up, they often produce over-fragmented results and lack semantic interpretation of the segments.

### B. Supervised Video Segmentation

On the contrary, the supervised video segmentation [7]–[11] discloses the objects of interest that the users identify, and thus the segments are more semantically meaningful. Boykov and Funka-Lea [11] made use of graph cuts whose boundary regularization and color model are based on the user's strokes, to segment video sequence as a spatiotemporal volume. Price *et al.* [8] also adopted graph cut optimization to combine various features weighted by their estimated accuracy based on the user guidance. Bai *et al.* [7] leveraged multiple overlapping localized classifiers, each of which segments a piece of the foreground boundary to extract foreground objects in videos. Grundmann *et al.* [9] designed a hierarchical region graph to represent the segmentation hierarchy of the oversegmented space-time regions, and leveraged dense optical flow to enforce temporal consistency to the segmentation. Huang *et al.* [10] presented a hypergraph to represent the complex spatiotemporal neighborhood relationship of the oversegmented image patches, and solved video segmentation with a hypergraph cut algorithm. Chien *et al.* [25] proposed a robust threshold decision algorithm for video object segmentation with a multibackground model. However, we do not explicitly model the background but design a multicomponent foreground model, which is more consistent across the frames.

### C. Video Cosegmentation

Inspired by image cosegmentation [13], video cosegmentation algorithms [14]–[16] have been advocated to extract common object from videos. Lou and Gevers [16] established a probabilistic graphical model across a set of videos to learn the primary object, considering the appearance, spatial, and temporal consistency. Zhang *et al.* [15] cosegmented objects in arbitrary videos by sampling, tracking, and matching object proposals through a regulated maximum weight clique (MWC) extraction scheme, which is capable of handling multiple objects, temporary occlusions, and objects going in and out of view. Wang *et al.* [17] performed subspace clustering on the temporal superpixels to segment the videos into consistent spatiotemporal regions and then used the quadratic pseudoboolean optimization to minimize the MRF energy of video cosegmentation. Fu *et al.* [18], [19] proposed a multistate selection graph model to segment multiple common foreground objects in the videos based on the category-independent object proposals. Wang *et al.* [14] incorporated intra-frame saliency, inter-frame consistency, and across-video similarity into an energy optimization framework for video object cosegmentation, and introduced a spatiotemporal SIFT descriptor to integrate across-video correspondence into inter-frame motion flow.

## III. System Overview

The task of the proposed algorithm is to separate the foreground object, which is specified in several key frames, from the background in the frames of a video sequence. To begin with, some notations will be introduced. The frames of the video sequence are denoted by $\{I_t\}_{t=1}^{N}$, where $N$ is the number of frames and $I_t$ is the $t$th frame. The height and width of the frames are denoted by $H$ and $W$, respectively. Thus, the foreground mask of a frame is represented by a binary matrix $x \in \mathcal{X}$, where $\mathcal{X} = \{0,1\}^{H\times W}$ is the space of all possible binary segmentations. As the proposed method is supervised, the foreground masks of some key frames will be provided. Let the frame indices of the labeled frames be denoted by $\pi_L$, and the ones of the unlabeled frames $\pi_U$. Therefore, the foreground masks of the labeled frames are denoted by $X_L = \{x_i | i \in \pi_L\}$, and the goal of the proposed method is to compute the masks of the unlabeled frames, i.e., $X_U = \{x_i | i \in \pi_U\}$.

The general framework of the proposed algorithm is demonstrated in Fig. 1. It contains two subroutines, namely, temporal-coherent video segmentation and transductive learning of model parameters. First, the initial foreground model is learned from the labeled frames in an inductive manner by training a traditional SVM classifier. With the foreground model being fixed, the first step computes the optimal foreground masks of the unlabeled frames. For each unlabeled frame, a set of segmentation hypotheses $\mathcal{H} = \{h_i\}_{i=1}^{|\mathcal{H}|}$ are generated by temporal coherent parametric min-cut, combining both visual cues and motion cues, where $h_i \in \mathcal{X}$ is the $i$th hypothesis. An energy function is designed to evaluate each hypothesis from two aspects: the foreground fidelity and the segmentation quality. The foreground fidelity is measured by the similarity of the foreground region in the hypothesis with the foreground model, while the segmentation quality offers a low-level evaluation of the segmentation hypothesis based on the foreground/background divergence, the boundary strength, and the visual saliency. Finally, the optimal segmentation of the frame is approximated by Monte Carlo sampling of the segmentation hypotheses weighted by their energies. The second step, i.e., the transductive learning of model parameters, groups the frames into multiple components and computes the optimal parameters with the masks of the unlabeled frames being fixed. Specifically, frames are organized into a tree-structured graphical model named temporal tree via temporal coherent probabilistic clustering, where visually and temporally consistent frames are organized together as branches. To obtain a robust representation of the image, the state-of-the-art convolutional neural network (CNN) is utilized to extract features from the foreground region. In each branch of the temporal tree, a foreground component can be trained by fitting a transductive SVM classifier over the CNN descriptors by jointly maximizing the margin of the labeled frames and the unlabeled ones. Eventually, these two steps iterate until convergence. In practice, the algorithm terminates if the update of the foreground masks of the unlabeled frames is $\sim$2%. The procedure of the proposed algorithm is summarized in Algorithm 1. The details about the temporal-coherent video segmentation and transductive learning of model parameters will be elaborated in Sections IV and V, respectively.

## IV. Temporal Coherent Video Segmentation

Given the foreground model, the proposed temporal coherent video segmentation algorithm computes the optimal segmentations of the frames by assembling segmentation hypotheses generated by a temporal coherent parametric

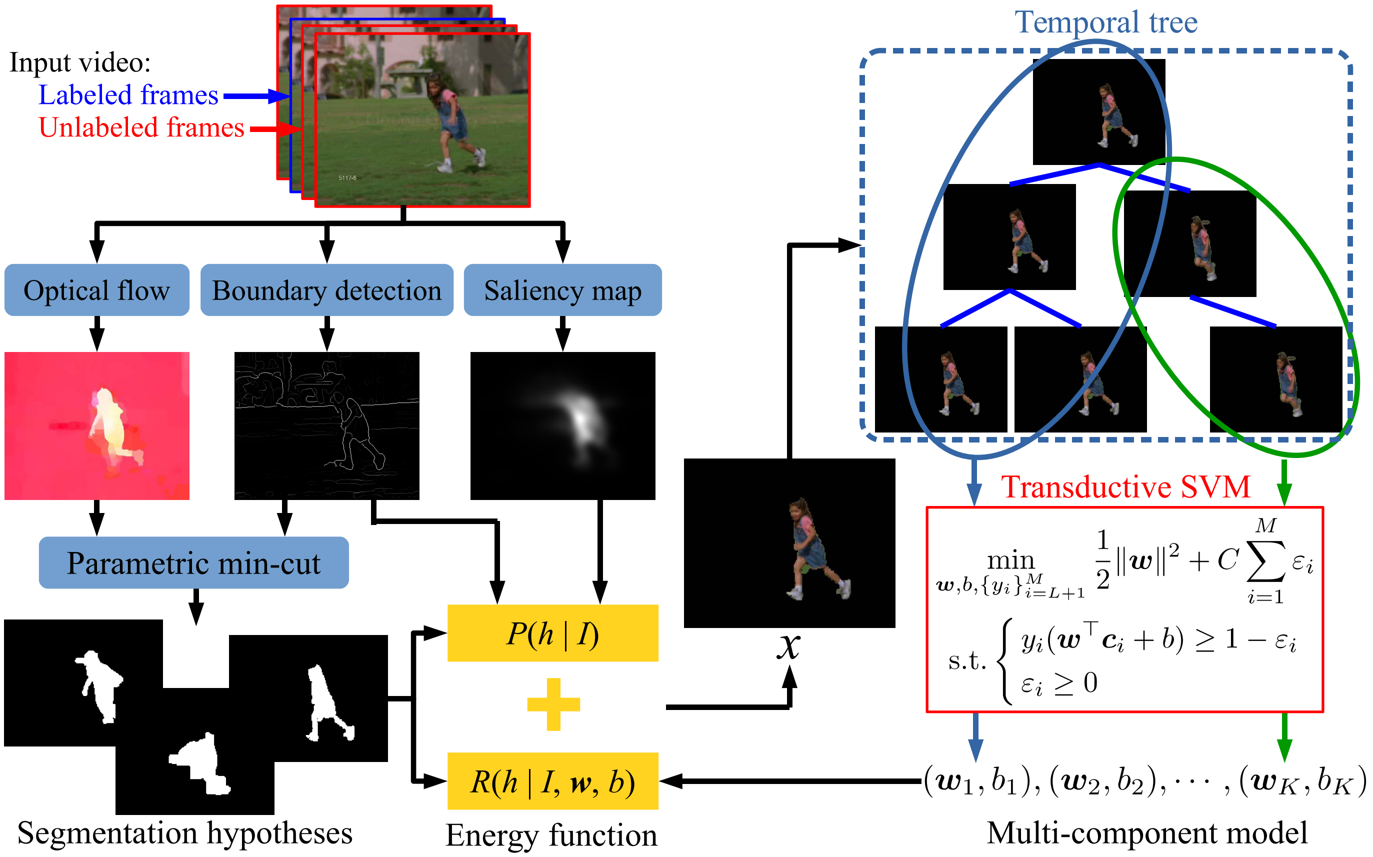

Input video:
Labeled frames
Unlabeled frames
Optical flow
Boundary detection
Saliency map
Parametric min-cut
$P(h \mid I)$
+
$R(h \mid I, \boldsymbol{w}, b)$
Segmentation hypotheses
Energy function
$\mathcal{X}$
Temporal tree
Transductive SVM
$\min_{\boldsymbol{w},b,\{y_i\}_{i=L+1}^{M}} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{M}\varepsilon_i$
s.t. $y_i(\boldsymbol{w}^\top \boldsymbol{c}_i + b) \geq 1 - \varepsilon_i$, $\varepsilon_i \geq 0$
$(\boldsymbol{w}_1, b_1), (\boldsymbol{w}_2, b_2), \cdots, (\boldsymbol{w}_K, b_K)$
Multi-component model

Fig. 1. Framework of the proposed multicomponent video segmentation algorithm. It consists of two steps: temporal-coherent video segmentation and transductive learning of model parameters. The first step forms the optimal foreground masks of the unlabeled frames, given the foreground model. The second step groups the frames into multiple components, and estimates the hyperparameter of each component by fitting a transductive SVM classifier over the temporal tree.

min-cut algorithm. In addition, the segmentation hypotheses are weighted by an energy function, which measures fidelity of the hypotheses with respect to the foreground model and the segmentation quality based on the low-level image features. We first introduce the generation of the segmentation hypotheses in Section IV-A and then present the energy function in Section IV-B.

### *A. Temporal Coherent Parametric Min-Cut*

A parametric min-cut algorithm [26] has been used for video segmentation by many approaches [6], [24]. However, to generate more effective segments for the frames in a video sequence, a temporal coherent parametric min-cut algorithm is devised, which preserves the motion consistency in the segmentation.

Specifically, an image $I$ is represented by a graph $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of pixels and $\mathcal{E}$ is the set of edges that link neighboring pixels. Upon this graph, a set of foreground seeds $\mathcal{V}_f \in \mathcal{V}$ and background seeds $\mathcal{V}_b \in \mathcal{V}$ are sampled from the foreground and background regions, respectively. Finally, a segmentation hypothesis can be computed by optimizing the following function through constrained parametric min-cut (CPMC):

$$\min_h \sum_{u\in\mathcal{V}} D_\phi(h(u)) + \sum_{(u,v)\in\mathcal{E}} V(h(u), h(v)) \qquad (1)$$

where $D_\phi$ is the unary term parameterized by $\phi$, and $V$ is the pairwise term that penalizes the assignment of different labels to neighboring pixels.

The unary term in (1) is defined as

$$D_\phi(h(u)) = \begin{cases} \infty, & \text{if } h(u) = 1, \quad u \in \mathcal{V}_b \\ \infty, & \text{if } h(u) = 0, \quad u \in \mathcal{V}_f \\ 0, & \text{if } h(u) = 1, \quad u \notin \mathcal{V}_b \\ f(h(u)) + \phi, & \text{if } h(u) = 0, \quad u \notin \mathcal{V}_f. \end{cases} \qquad (2)$$

The first two cases in (2) ensure that the labels of the seed nodes are fixed. The last case in (2) is a foreground bias, which consists of a pixel-dependent function $f(h(u))$ and a uniform offset $\phi$ that controls the scale of the foreground region. The larger the $\phi$ is, the smaller the foreground region will be. Specifically, $f(h(u))$ is the similarity of the current pixel with the foreground seeds, which is more effective if the foreground object has distinctive color with respect to the background. The technical details about the CPMC algorithm can be found in [26].

The binary term encourages the labeling of the neighboring pixels to be spatially and temporally homogeneous, which is defined as

$$V(h(u), h(v)) = \begin{cases} 0, & \text{if } h(u) = h(v) \\ g(u, v), & \text{if } h(u) \neq h(v) \end{cases} \qquad (3)$$

**Algorithm 1** Multicomponent Transductive Video Segmentation Algorithm

```
Input: Frames $\{I_t\}_{t=1}^{N}$, masks of the labeled frames $X_L = \{x_i | i \in \pi_L\}$
Output: Masks of the unlabeled frames $X_U = \{x_i | i \in \pi_U\}$

# Initialization;
Train an inductive SVM classifier $(\boldsymbol{w}, b)$ using $X_L$;
while unconverged do
    # Temporal coherent video segmentation;
    for each unlabeled frame $t$ in $X_U$ do
        Generate segmentation hypotheses $\mathcal{H}_t = \{h_i^t\}_{i=1}^{|\mathcal{H}_t|}$ via parametric min-cut;
        for each hypothesis $i$ do
            Compute the discriminant term $R(h_i^t | I_t, w, b)$ based on the foreground model;
            Compute the prior term $P(h_i^t | I_t)$ based on the low-level image features;
            Compute the weight of the hypothesis $\beta_i^t$;
        end
        Compute soft segmentation $x_t' = \sum_{i=1}^{|\mathcal{H}_t|} \beta_i^t h_i^t$ via Monte Carlo approximation;
        Threshold $x_t'$ to generate the optimal segmentation $x_t'$;
    end

    # Transductive learning of model parameters;
    for labeled frames followed by unlabeled frames do
        Randomly pick a frame;
        Compute the assignment probability $p$ of the chosen frame with the active nodes;
        Add the chosen frame as the child of an active node by drawing $p$;
        Train transductive SVM classifiers;
    end
    Collect branches from the temporal tree and re-estimate the parameters;
end
```

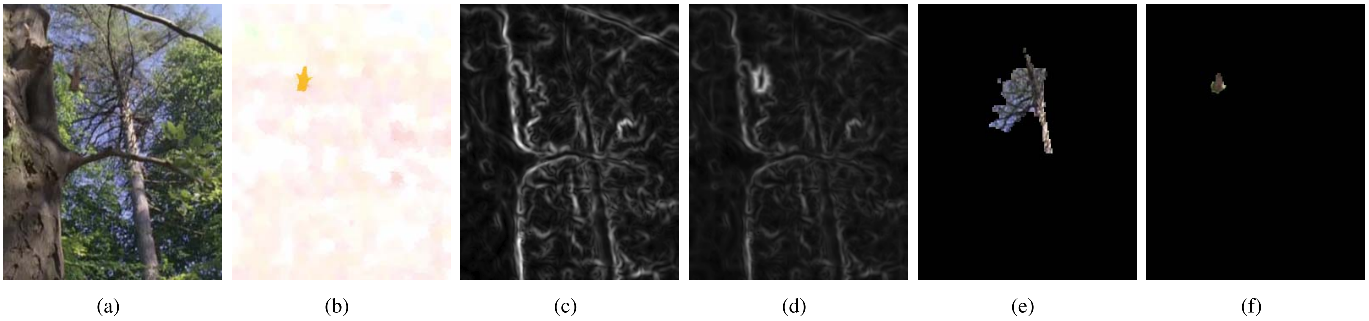

(a) (b) (c) (d) (e) (f)

Fig. 2. (a) Input frame. (b) Optical flow field. (c) Spatial boundary. (d) Spatiotemporal boundary. (e) Segmentation hypothesis computed from spatial boundary. (f) Segmentation hypothesis computed from spatiotemporal boundary.

where

$$g(u, v) = \exp\left\{-\frac{\max(r(u), r(v))}{\sigma^2}\right\}. \quad (4)$$

In (4), $r(u)$ is the boundary strength of pixel $u$, and $\sigma$ is the boundary sharpness parameter controlling the smoothness of the pairwise term. The boundary strength is computed by the generalized boundary detection [27] over the original frame and the optical flow field, because pixels with strong texture boundary or motion boundary are likely to be in the same object. As shown in Fig. 2, the moving object can be well captured, when the motion cues are considered.

In practice, we randomly sample $\phi$ and $\sigma$ from $[0, 1]$ to produce segmentation hypotheses of various scales, and 200 segmentation hypotheses are generated for each frame, which generally cover the foreground object.

### *B. Energy Function of Segmentation Hypotheses*

An energy function is devised to evaluate the segmentation hypothesis $h$, which is defined as

$$E(h|I, \boldsymbol{w}, b) = R(h|I, \boldsymbol{w}, b) + P(h|I) \quad (5)$$

where $\boldsymbol{w}$ and $b$ are the parameters of the foreground model. The energy function is comprised of two terms: the discriminant term $R(h|I, \boldsymbol{w}, b)$ and the prior term $P(h|I)$. The discriminant term measures the resemblance of the foreground region in $h$ with the foreground model, while the prior

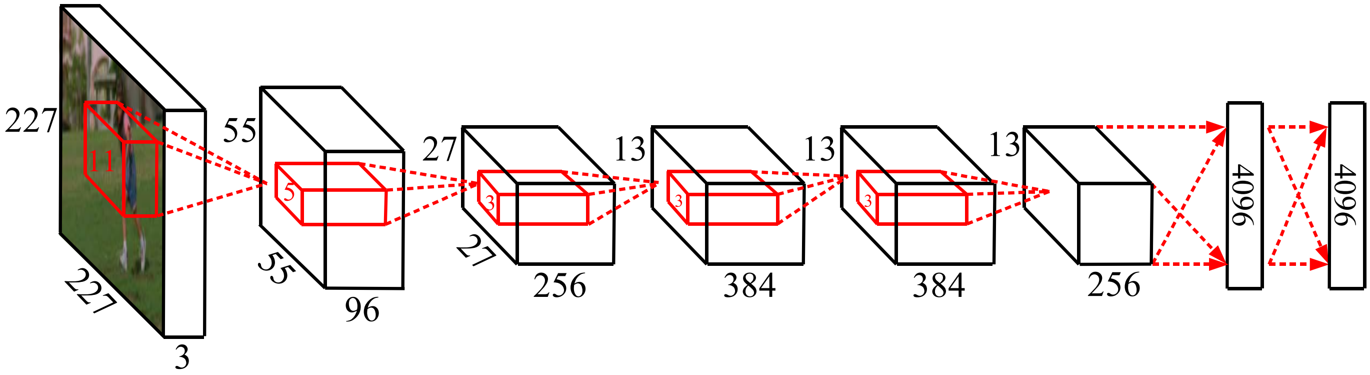


Fig. 3. CNN is composed of the five convolutional layers and two fully connected layers.

term evaluates the quality of the segmentation based on the low-level image features.

To be specific, the discriminant term is defined as

$$R(h|I, \boldsymbol{w}, b) = \boldsymbol{w}^\top \boldsymbol{c} + b \tag{6}$$

where $\boldsymbol{c} \in \mathbb{R}^d$ is the $d$-dimensional vector encoding the visual feature of the foreground region in $h$. Here, the foreground model is represented by a linear discriminative function, which is parameterized by $\boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$. To obtain a robust representation of image features, R-CNN descriptor [28] is adopted to encode the visual appearance of the foreground region in the segmentation hypothesis. To be concrete, the topology of the CNN structure is shown in Fig. 3, which consists of five convolutional layers and two fully connected layers. The first convolutional layer convolutes the 227 × 227 × 3 rescaled image patch with 96 kernels of size 11 × 11 × 3 at a stride of 4 pixels. Then, the second convolutional layer filters the output of the first layer with 256 kernels of size 5 × 5 × 96. In the same way, the third and the fourth convolutional layers both have 384 kernels of size 3 × 3 × 256 and 3 × 3 × 384, respectively. Finally, the fifth convolutional layer has 256 kernels of size 3 × 3 × 384, and each of the two fully connected layers has 4096 units, which is the dimension of the final R-CNN descriptor.

The prior term $P(h|I)$ measures the quality of the segmentation hypothesis $h$ based on the low-level image features, including the color distribution, the boundary strength, and the visual saliency, which is defined as

$$\begin{aligned} P(h|I) = \alpha_1 \|\text{hist}(h|I) - \text{hist}(1-h|I)\|_1 \\ + \alpha_2 B(h|I) + \alpha_3 S(h|I). \end{aligned} \tag{7}$$

The first term in Eq. (7) measures the divergence of the foreground region and the background region of $h$ in color distribution, where hist$(h|I)$ is the color histogram of the foreground pixels and hist$(1-h|I)$ is the one of the background pixels in $h$. The second term $B(h|I)$ measures the mean boundary strength of the foreground region, which can be computed by averaging the pixelwise boundary strength $r(u)$ in (4) along the boundary of $h$. The basic intuition is that, for a good segmentation hypothesis, the foreground and the background should be separated by strong object boundaries, which have large boundary strength. Finally, the third term $S(h|I)$ measures the visual saliency [29] of the foreground region in $h$, because the foreground object is supposed to be salient compared with the background region. In addition, ($\alpha_1, \alpha_2$, and $\alpha_3$) are the weights of the foreground/background divergence, the boundary strength, and the visual saliency, respectively.

Up to now, we have a collection of segmentation hypotheses $\mathcal{H} = \{h_i\}_{i=1}^{|\mathcal{H}|}$ sampled from the segmentation space $\mathcal{X}$ based on the visual and temporal constraints, and an energy function to measure the confidence of each hypothesis. With this sample-based representation, we leverage the sequential Monte Carlo approximation to estimate the optimal segmentation of the frame. Naturally, the segmentation hypotheses act as particles in the Monte Carlo framework, and the normalized weight of the $i$th sample is

$$\beta_i = \frac{\exp(E(h_i|I, \boldsymbol{w}, b))}{\sum_{i=1}^{|\mathcal{H}|} \exp(E(h_i|I, \boldsymbol{w}, b))}. \tag{8}$$

Therefore, the maximum-*a-posteriori* estimation of the segmentation is approximated by the weighted combination of the hypotheses

$$x' = \sum_{i=1}^{|\mathcal{H}|} \beta_i h_i. \tag{9}$$

As $x'$ is real-valued, the binary segmentation of the frame $x$ can be obtained by thresholding $x'$

$$x(u) = \begin{cases} 1, & \text{if } x'(u) \geq T \\ 0, & \text{if } x'(u) < T \end{cases} \tag{10}$$

where $T$ is the threshold, which can be determined by minimizing the prediction error over the labeled frames. It is worth mentioning that to ensure the accuracy of the proposed algorithm, the segmentation hypotheses will be regenerated by resampling the foreground seeds in the predicted foreground region during the update of the model parameters.

## V. Transductive Learning on the Temporal Trees

Given the video sequence with some labeled frames, the initial foreground model is learned from the labeled frames by training an inductive SVM classifier. To be concrete, positive features are extracted from the foreground region in the labeled

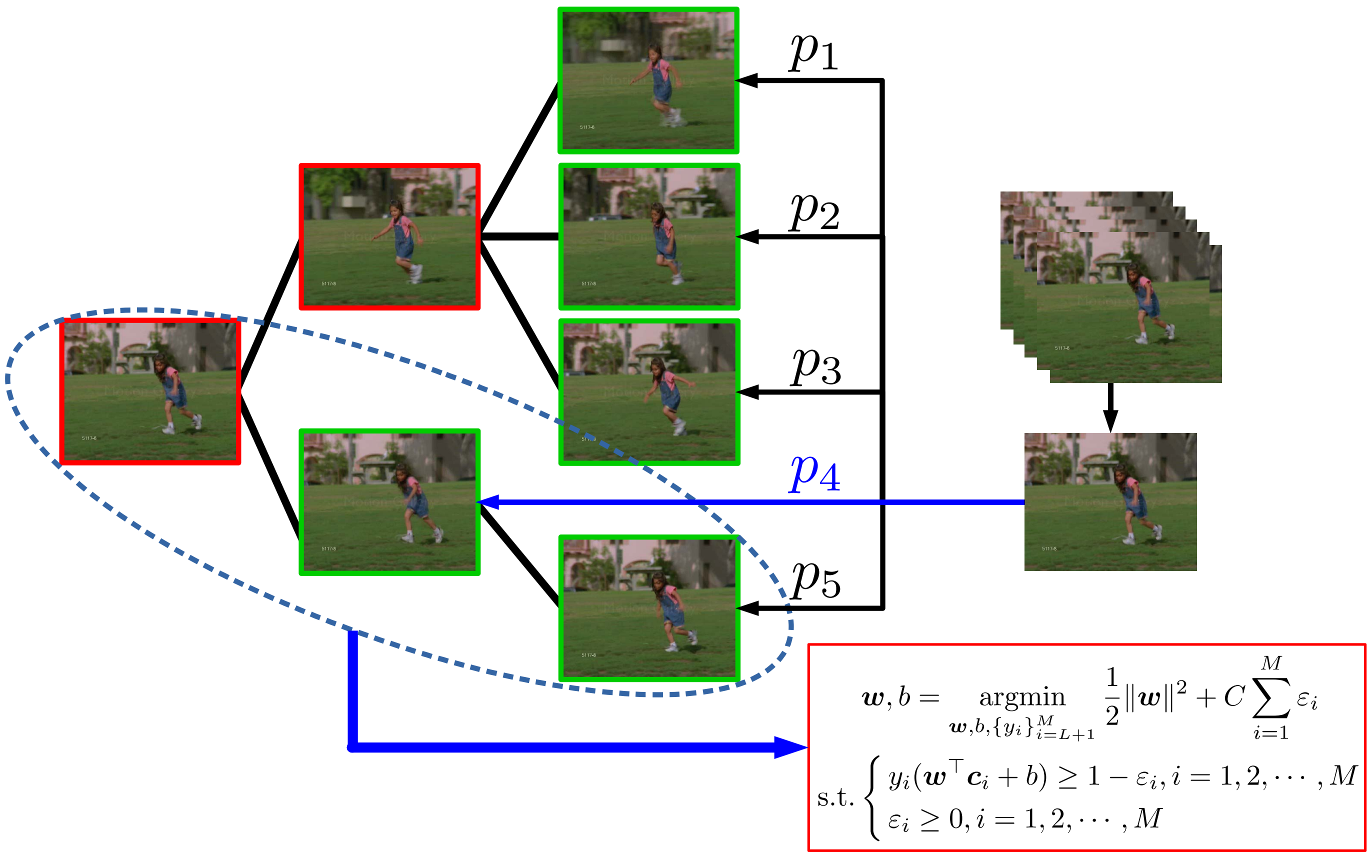

$p_1$
$p_2$
$p_3$
$p_4$
$p_5$
$\boldsymbol{w}, b = \underset{\boldsymbol{w}, b, \{y_i\}_{i=L+1}^{M}}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{M}\varepsilon_i$
s.t. $\begin{cases} y_i(\boldsymbol{w}^\top \boldsymbol{c}_i + b) \geq 1 - \varepsilon_i, i = 1, 2, \cdots, M \\ \varepsilon_i \geq 0, i = 1, 2, \cdots, M \end{cases}$

Fig. 4. Illustration of the construction of the temporal tree ($D = 3$). Red frames: inactive nodes. Green frames: active nodes. The rest of the frames are added to the temporal tree by sampling (11). Each branch in the tree corresponds to a foreground component, whose parameters can be determined by training a transductive SVM classifier.

frames, and negative features are computed from patches that are randomly sampled from the background region of the labeled frames. Consequently, a one-component foreground model can be obtained by training a traditional linear SVM classifier over the labeled features.

On the other hand, provided that the foreground masks of the unlabeled frames are available, the parameters of the multicomponent foreground model can be learned by the proposed transductive learning algorithm, using the labeled frames and the unlabeled frames to acquire stronger generalization capability. In general, the algorithm consists of two steps: 1) temporal coherent probabilistic clustering and 2) transductive learning of SVM classifiers.

### A. Probabilistic Clustering on the Temporal Tree

Given the estimated foreground masks of the unlabeled frames, as illustrated in Fig. 4, a temporal tree is established by organizing the frames into a graphical structure based on their visual similarity and temporal coherence. Each node in the temporal tree is a frame with a specific foreground mask, and each branch, consisting of visually similar and temporally coherent frames, defines a foreground component in the model. To constrain the complexity of the tree, we define the maximum number of children for a node to be $D$. Consequently, a node is active if the number of its children is smaller than $D$; otherwise, it is inactive.

The labeled frames will be added to the temporal tree before the unlabeled frames in order to provide a good initialization. Let $K$ be the number of active nodes in the current tree, and $F_k$ be the frame index of the $k$th active node. We uniformly sample one frame from the rest of the frames, and append it to one of the active nodes by drawing

$$p_k \propto \frac{1}{1 + \exp\left(\frac{|F - F_k| - Q}{\tau}\right)} \exp(\boldsymbol{w}_k^\top \boldsymbol{c} + b_k), \quad 1 \leq k \leq K \tag{11}$$

where $F$ is the frame index of the chosen frame, $(\boldsymbol{w}_k, b_k)$ are the parameters of the $k$th component, $Q$ is the maximum frame interval, and $\tau$ is the scaling factor. Here, $p_k$ is the probability that the chosen frame is added as the child of the $k$th active node. The first part in (11), that is

$$\frac{1}{1 + \exp\left(\frac{|F - F_k| - Q}{\tau}\right)} \tag{12}$$

measures the temporal coherence of two frames. A symmetric logistic function is utilized to penalize the interval of two frames, which is shown in Fig. 5. Equation (12) rapidly increases to 1 when the frame interval is smaller than the threshold $Q$, and decreases to 0 otherwise. The transition width is determined by the scaling parameter $\tau$, which is set to 0.5 in experiments. Therefore, a pair of parent–child nodes in the

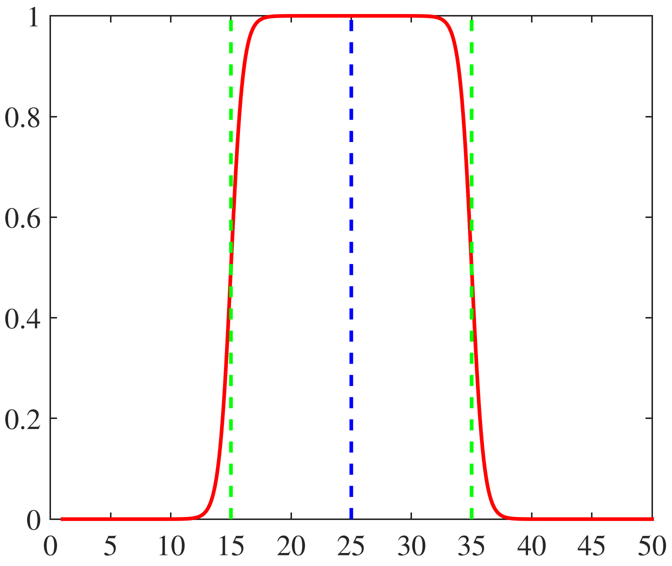


Fig. 5. Demonstration of the symmetric logistic function, where $Q = 10$, $F_k = 25$, and $\tau = 0.5$. It rapidly increases 1 when the frame interval is smaller than the threshold, and decreases to 0 otherwise.

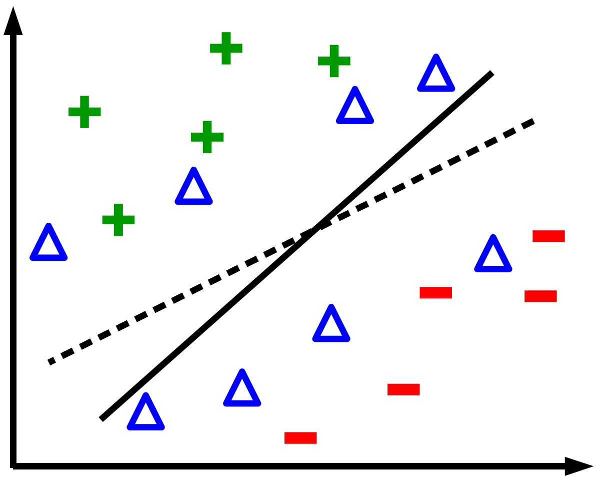


Fig. 6. Inductive learning versus transductive learning. The green crosses, red bars, and blue triangles represent the positive, negative, and unlabeled samples, respectively. Solid line: inductive separating hyperplane computed from the labeled samples. Dashed line: transductive separating hyperplane, which has a stronger generalization capability by incorporating the unlabeled samples.

tree are required to be temporal neighbors to preserve temporal consistency of the components. The second part in (11), i.e., $\exp(\boldsymbol{w}_k^\top \boldsymbol{c} + b_k)$, is the exponent of the discriminant score by the $k$th foreground component. The larger the discriminant score is, the more likely that the frame is assigned to that component.

In this way, all the frames can be added to the temporal tree one by one, and the final components can be determined by repeatedly collecting the longest branches from the temporal tree. Finally, the component parameters will be retrained with the frames in each component.

### *B. Transductive SVM Classifier*

The frames in a branch that contain both labeled and unlabeled ones compose a foreground component, and an SVM classifier is then trained to characterize the visual appearance of the component. Inductive approaches use only the labeled samples to train the classifier, which may result in overfitting because the labeled samples are scarce. However, as the labeled frames and the unlabeled frames in the same video sequence are highly correlated, maximizing the margin in the presence of the unlabeled frames can improve the generalization capability of the classifier based on the cluster assumption [30], which is shown in Fig. 6. Therefore, we take advantage of both labeled and unlabeled frames to train the classifier in a transductive manner.

The proposed algorithm of learning transductive SVM classifiers is shown in Fig. 7. Positive and negative features are both generated from the labeled frames, whereas features are generated from the unlabeled samples. To be concrete, the positive features are computed from the (ground truth) foreground regions in the labeled frames. The negative features are randomly sampled by a pool of rectangular patches from the labeled frames whose intersection-over-union (IoU) ratio with the foreground must be smaller than 0.5. In experiments, ten negative patches are sampled from each labeled frame. On the other hand, the unlabeled features are computed from the estimated foreground regions in the unlabeled frames, so they can potentially be both positive and negative.

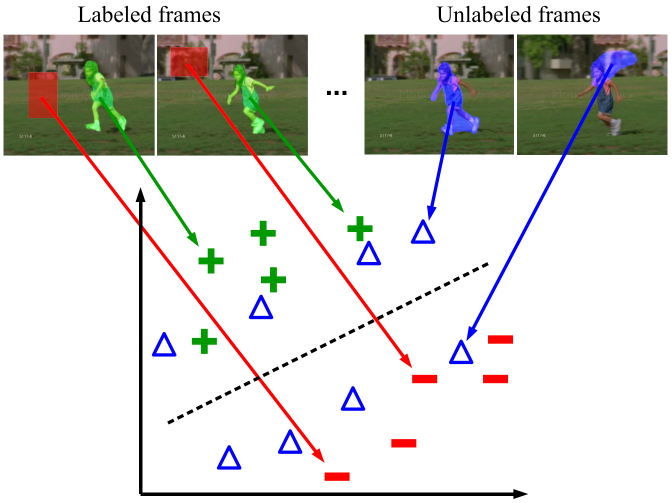


Fig. 7. Learning of transductive SVM classifier. The positive features are extracted from the foreground region of the labeled frames. The negative features are extracted from the background region of the labeled frames. The unlabeled features are extracted from the foreground region of the unlabeled frames.

Let $\{\boldsymbol{c}_i\}_{i=1}^M$ be the features extracted from the frames, where $M$ is the number of features. Without loss of generality, we assume that the first $L$ features are labeled, whose labels are denoted by $\{y_i\}_{i=1}^L$, where $y_i \in \{-1, +1\}$ for $i = 1, \ldots, L$. Consequently, the parameters of the transductive SVM classifier can be computed by

$$\min_{\boldsymbol{w}, b, \{y_i\}_{i=L+1}^M} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^M \varepsilon_i$$
$$\text{s.t.} \begin{cases} y_i(\boldsymbol{w}^\top \boldsymbol{c}_i + b) \geq 1 - \varepsilon_i, & i = 1, \ldots, M \\ \varepsilon_i \geq 0, & i = 1, \ldots, M. \end{cases} \tag{13}$$

Obviously, (13) is not convex, and many approximation solutions [31] have been proposed to solve this combinatorial optimization problem. However, these methods generally suffer from high computational complexity and are susceptible to local optima. To obtain an efficient solution, we adopt the convex relaxation formulation [32] by approximating the nonconvex optimization problem by its dual problem. Compared with semidefinite relaxation, this formulation provides a tighter convex relaxation and also has fewer free parameters. The solution is briefly described as follows, and more details can be found in [32].

Equation (13) can be transformed into the following form according to the Lagrange Theorem:

$$\begin{aligned} \min_{\boldsymbol{\eta}, \{y_i\}_{i=L+1}^{M}, \boldsymbol{\delta}, \lambda} \quad & C \sum_{i=1}^{M} \delta_i^2 \\ & + \frac{1}{2}(\mathbf{1} + \boldsymbol{\eta} - \boldsymbol{\delta} + \lambda \boldsymbol{y})^\top \mathcal{D} K^{-1} \mathcal{D} (\mathbf{1} + \boldsymbol{\eta} - \boldsymbol{\delta} + \lambda \boldsymbol{y}) \\ \text{s.t.} \quad & \eta_i \geq 0, \delta_i \geq 0, \text{ for } i = 1, \ldots, M \end{aligned} \tag{14}$$

where $\boldsymbol{\eta} \in \mathbb{R}^M$, $\boldsymbol{\delta} \in \mathbb{R}^M$, and $\lambda$ are dual variables. $\mathbf{K}$ is the kernel matrix, where $\mathbf{K}_{ij} = \langle \boldsymbol{c}_i, \boldsymbol{c}_j \rangle$ in our case. $\mathcal{D}$ is a diagonal matrix whose diagonal entries are $\{y_i\}_{i=1}^M$. The solution of (14) can be acquired by solving

$$\begin{aligned} \max_{\boldsymbol{\gamma}, t, \alpha, \beta} \quad & -\frac{1}{4} t + \sum_{i=1}^{M} \gamma_i - \epsilon(\alpha + \beta) \\ \text{s.t.} \quad & \begin{bmatrix} \boldsymbol{A} - \mathcal{D}(\gamma \circ \boldsymbol{b}) & \boldsymbol{\gamma} \circ \boldsymbol{a} - (\alpha - \beta)\boldsymbol{c} \\ (\boldsymbol{\gamma} \circ \boldsymbol{a} - (\alpha - \beta)\boldsymbol{c})^\top & t \end{bmatrix} \geq 0 \\ & 0 \leq \gamma_i \leq C, \quad i = 1, \ldots, M \\ & \alpha \geq 0, \quad \beta \geq 0 \end{aligned} \tag{15}$$

where

$$\boldsymbol{a} = (y_1, \ldots, y_L, \mathbf{0}_{M-L+1}) \in \mathbb{R}^{M+1} \tag{16}$$

$$\boldsymbol{b} = (\mathbf{0}_L, \mathbf{1}_{M-L}, 0) \in \mathbb{R}^{M+1} \tag{17}$$

$$\boldsymbol{c} = \left( \frac{1}{L}\mathbf{1}_L, -\frac{1}{M-L}\mathbf{1}_{M-L}, 0 \right) \in \mathbb{R}^{M+1} \tag{18}$$

$$\boldsymbol{A} = (\boldsymbol{I}_M, \mathbf{1}_M)^\top \mathbf{K}^{-1} (\boldsymbol{I}_M, \mathbf{1}_M)^\top \tag{19}$$

$\epsilon \geq 0$ is a constant, and $\circ$ represents the elementwise product. Finally, (15) is a semidefinite programming problem, which can be solved effectively, e.g., by the interior-point method.

### C. Model Validation and Convergence

The proposed multicomponent transductive learning algorithm prevents the solution to be stuck in local extrema via Bayesian sampling in the construction of the temporal tree. However, this scheme brings a side effect that the temporal tree may be falsely structured by grouping the visually dissimilar frames into the same branch. To handle this situation, model validation will be performed after a new model is learned, and the new model will be accepted only if it passes the model validation; otherwise, the temporal tree is rebuilt.

To be concrete, the model validation is conducted with the labeled frames. Let the segmentation result of the labeled frames by the new model be denoted by $\{x_t\}_{t=1}^L$, and the ground truth foreground masks be denoted by $\{x_t^{GT}\}_{t=1}^L$. The segmentation accuracy is measured by the IoU ratio, which is defined as

$$\text{IoU} = \frac{1}{L} \sum_{t=1}^{L} \frac{\|x_t \cap x_t^{GT}\|_0}{\|x_t \cup x_t^{GT}\|_0} \tag{20}$$

where $\|x\|_0$ is the $l_0$ norm of matrix $x$, i.e., the number of nonzero elements in $x$. Naturally, the new model can be considered as effective only if it is capable of segmenting the labeled frames correctly. In practice, the criterion of model validation is that the IoU of the labeled frames is smaller than 0.3.

On the other hand, the criteria of convergence of the proposed algorithm are based on the increment of the average IoU of the labeled frames. In practice, the algorithm is considered to be converged if the increment is $\sim$0.01.

## VI. Experiments

The experiments are conducted on the SegTrack data set [33], which contains video sequences of large (*cheetah* and *monkeydog*), medium (*girl* and *parachute*), and small (*penguin* and *birdfall*) variations in visual appearance.

### A. Comparison With State-of-the-Art Methods

As far as we know, there are few studies on video segmentation that exactly share the same setting as the proposed method in which a key frame with the ground truth foreground mask is available every few frames. Therefore, we compare the performance of the proposed method with conventional video segmentation methods, including the following methods.

*Unsupervised Methods:*

1) hierarchical graph-based segmentation [9];
2) key segments for video segmentation (KeySeg) [34];
3) fast object segmentation (FOS) [35];
4) primary object regions [1];
5) saliency-aware geodesic segmentation [20];
6) tracking FGSs [24];
7) MWCs [2];
8) salient segment chain composition [36];
9) SeamSeg [3];
10) shortest path algorithm [4];
11) region-based particle filter [12].

*Supervised Methods:*

1) adaptive fragments-based tracking [37];
2) motion coherent tracking [33];
3) tree-structured graphical models [38];
4) video SnapCut (SnapCut) [7].

The unsupervised methods do not require the ground truth foreground masks, whereas the supervised methods require the ground truth foreground mask of the first frame for each video sequence for initialization. In particular, SnapCut is implemented in Adobe After Effects as "Roto Brush Tool," and requires manual reinitialization 2$\sim$3 times after losing the target for each video.

The parameters of the proposed method are configured as follows. A key frame with the ground truth foreground mask will be available in every ten frames, and the accuracy is measured only on the unlabeled frames. The weights $(\alpha_1, \alpha_2, \alpha_2)$

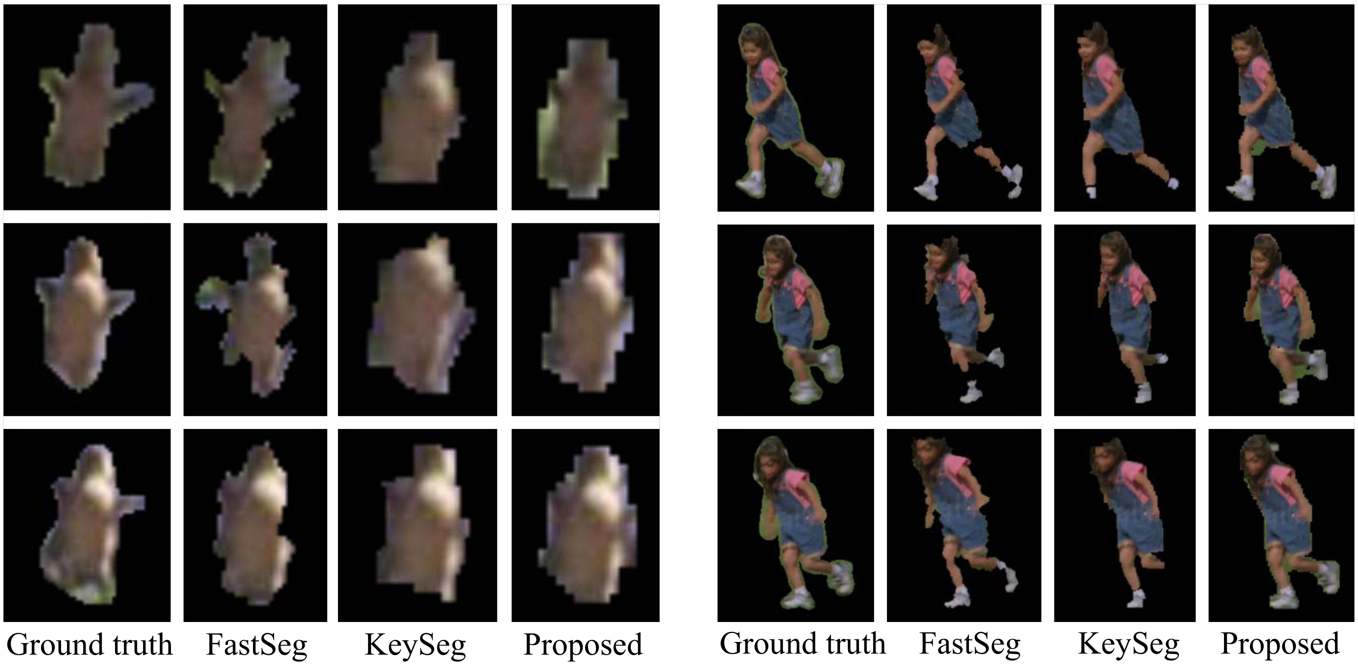


Fig. 8. Segmentation result of *birdfall* and *girl* sequences in the SegTrack data set.

TABLE I
MPE ON THE SEGTRACK DATA SET

| | *birdfall* | *cheetah* | *girl* | *monkeydog* | *parachute* | *penguin* |
|---|---|---|---|---|---|---|
| AFT | 454 | 1217 | 1755 | 683 | 502 | 6627 |
| HGS | 305 | 1219 | 5777 | 493 | 1202 | 2116 |
| KeySeg | 288 | 905 | 1785 | 521 | 201 | 136285 |
| MCT | 252 | 1142 | 1304 | 563 | 253 | 1705 |
| FOS | 272 | 1050 | 4241 | 401 | 405 | 23504 |
| POR | **155** | 633 | 1488 | 365 | 220 | N/A |
| TSGM | 259 | 923 | 820 | 589 | 258 | 21141 |
| SAG | 209 | 796 | 1040 | 562 | 207 | N/A |
| FGS | 242 | 1156 | 1564 | 483 | 328 | 5026 |
| MWC | 189 | 806 | 1698 | 472 | 221 | N/A |
| SSC | 166 | 661 | 1214 | 394 | 218 | N/A |
| SeamSeg | 186 | 535 | **761** | 358 | 249 | **335** |
| SPA | 267 | 799 | 1582 | 398 | 197 | N/A |
| RPF | 243 | **391** | 1935 | 497 | **187** | 903 |
| SnapCut | 354 | 1139 | 5805 | 440 | 434 | 444 |
| hypo. | 308 | 1398 | 1762 | 632 | 304 | 713 |
| Proposed | 163 | 688 | 1186 | **354** | 209 | 456 |

TABLE II
MPE RANKINGS ON THE SEGTRACK DATA SET

| | *birdfall* | *cheetah* | *girl* | *monkeydog* | *parachute* | average |
|---|---|---|---|---|---|---|
| AFT | 16 | 15 | 11 | 16 | 15 | 14.6 |
| HGS | 14 | 16 | 15 | 10 | 16 | 14.2 |
| KeySeg | 13 | 9 | 12 | 12 | 3 | 9.8 |
| MCT | 9 | 13 | 6 | 14 | 10 | 10.4 |
| FOS | 12 | 11 | 14 | 6 | 13 | 11.2 |
| POR | 1 | 3 | 7 | 3 | 7 | 4.2 |
| TSGM | 10 | 10 | 2 | 15 | 11 | 9.6 |
| SAG | 6 | 6 | 3 | 13 | 4 | 6.4 |
| FGS | 7 | 14 | 8 | 9 | 12 | 10.0 |
| MWC | 5 | 8 | 10 | 8 | 8 | 7.8 |
| SSC | 3 | 4 | 5 | 4 | 6 | 4.4 |
| SeamSeg | 4 | 2 | 1 | 2 | 9 | 3.6 |
| SPA | 11 | 7 | 9 | 5 | 2 | 6.8 |
| RPF | 8 | 1 | 13 | 11 | 1 | 6.8 |
| SnapCut | 15 | 12 | 16 | 7 | 14 | 12.8 |
| Proposed | 2 | 5 | 4 | 1 | 5 | 3.4 |

of the prior term in (7) are determined empirically. Specifically, since we consider the distinctiveness of the foreground region and the boundary strength more important in defining an object, we set $\alpha_1 = 0.4$, $\alpha_2 = 0.4$, and $\alpha_3 = 0.2$. Since most of the test sequences are short, we set $D = 3$ to prevent the branches not having enough frames to train a valid model.

Since most of the approaches listed above report their results by the mean pixel error (MPE), i.e., the average number of wrongly labeled pixels in each frame, we also use this metric to evaluate the proposed method. The result is shown in Table I with some examples shown in Figs. 8 and 9. In addition, the MPE of the segmentation hypotheses generated by CPMC is also listed, which is denoted by hypo.

Compared with the segmentation hypotheses, the proposed method effectively assembles coarse hypotheses into accurate segmentations based on the multicomponent foreground model and the temporal coherent video segmentation algorithm. However, it is hard to compare the overall performance of the proposed method with other methods based on Table I, because simply averaging the MPE over all sequence is meaningless since MPE is dependent on the size of the videos and the size of the foreground objects. Therefore, we compute the MPE rankings of these methods for the first five sequences, which is shown in Table II, since many methods do not report their performance of the *penguin* sequence.

It is clearly shown in Table II that the proposed method obtains the highest average rank among all the methods, which validates the effectiveness of the proposed method. The proposed method achieves top five in all test sequences,

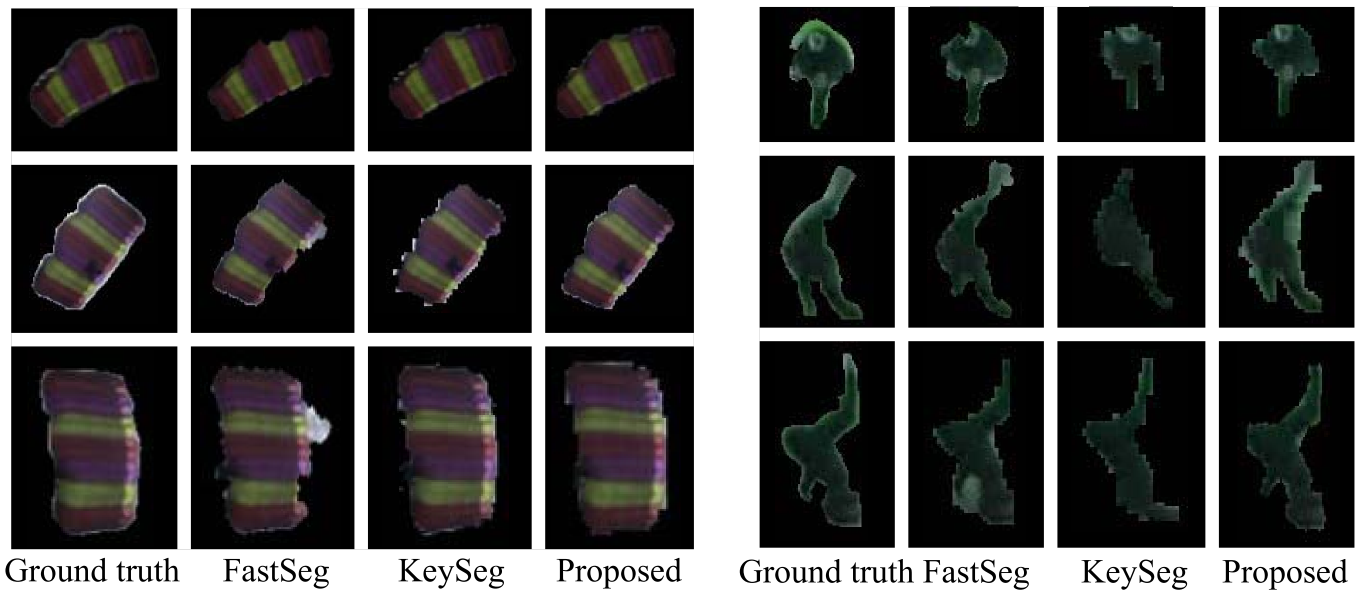


Fig. 9. Segmentation result of *parachute* and *monkeydog* sequences in the SegTrack data set.

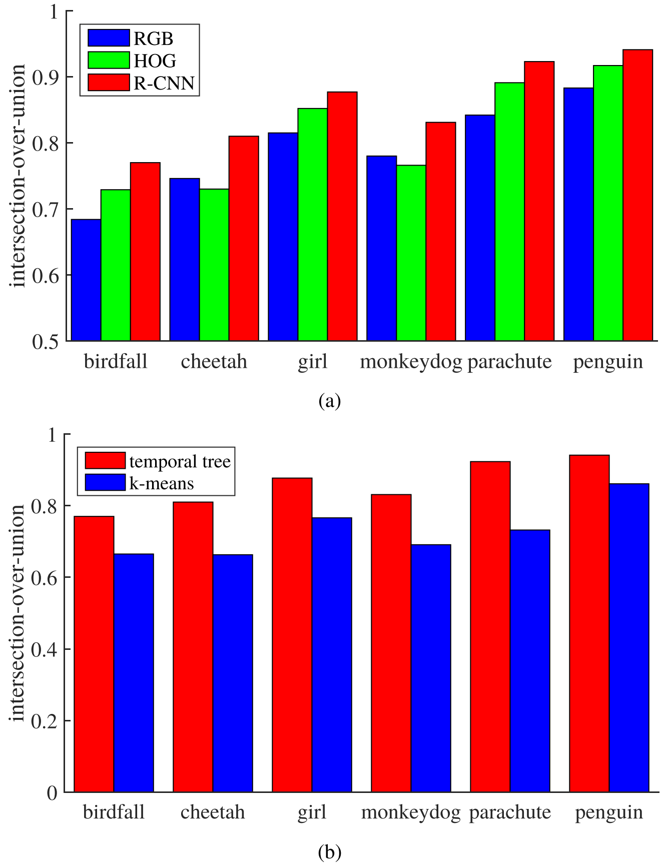
demonstrating that it is consistent and robust for different types of videos.

Moreover, we elaborately evaluate the impact of several aspects to the performance of the proposed method, including the features, the clustering schemes, the key frame interval, and the convergence rate. Since the drawback of the MPE metric has been mentioned above, we use the IoU ratio, which is defined in (20), to measure of the performance of segmentation in the following evaluation.

### B. Impact of Image Feature

Although the R-CNN descriptor is used as the feature of the foreground region in the experiments, we also test the performance of the proposed method, which incorporates other features, including red-green-blue (RGB) histogram and histogram of oriented gradients (HOG). For the RGB histogram, a codebook of 256 codewords is learned over the 3D RGB vectors of the pixels in a video sequence, and the foreground region is represented by the 256D codeword histogram accordingly. For the HOG, the foreground region is resized to the average size of the object in the labeled frames and then an HOG descriptor can be computed. The segmentation accuracy of the RGB histogram, HOG, and R-CNN is shown in Fig. 10(a).

In general, two conclusions can be drawn from Fig. 10(a). First, the proposed method is quite robust with different features, because the IoU does not change significantly for different features. Second, among the three features, R-CNN is the most discriminant one due to the strong representation capability of deep neural networks, and the RGB histogram has the worse performance because it does not depict the shape and texture of the object, which are important cues for video segmentation.



(a)

(b)

Fig. 10. (a) Video segmentation performance of different features. (b) Video segmentation performance of temporal tree and $k$-means.

### C. Temporal Tree Versus k-Means

To validate the effectiveness of the proposed temporal tree algorithm, we compare it with the widely used $k$-means algorithm. Specifically, for the $k$-means, all the frames are clustered into three groups, each of which corresponds to a foreground component. The result is shown in Fig. 10(b), where temporal tree achieves better performance than $k$-means, because it regularizes the relation of parent–child pairs with temporal constraint to avoid outliers.

TABLE III
MPE ON THE ADDITIONAL SEQUENCES IN THE SEGTRACK-V2 DATA SET

| | *bird of paradise* | *bmx* | *drift_1* | *drift_2* | *frog* | *hummingbird_1* | *hummingbird_2* | *monkey* | *soldier* | *worm* |
|---|---|---|---|---|---|---|---|---|---|---|
| FOS | 6754 | 11449 | 12656 | 15834 | 3115 | 14936 | 9316 | 2476 | 4270 | 1460 |
| KeySeg | 34916 | 6274 | 5449 | 7382 | 3560 | 19838 | 17799 | 2597 | 2054 | 815 |
| SnapCut | 3379 | 1292 | 2160 | 3697 | 1889 | 6337 | 7785 | 1901 | 1302 | 767 |
| Proposed | **1506** | **1248** | **717** | **652** | **1179** | **924** | **1114** | **686** | **1116** | **441** |

TABLE IV
MPE ON THE SFO DATA SET

| | *Garden* | *Hall* | *Calendar* | *Stefan* | *Table Tennis* |
|---|---|---|---|---|---|
| FOS | 13275 | 1586 | 30033 | 3258 | 11749 |
| KeySeg | 48706 | **712** | 48164 | 68659 | 3193 |
| SnapCut | 5112 | 1967 | **3889** | 3994 | 3172 |
| Proposed | **2532** | 1673 | 10054 | **2362** | **2507** |

### *D. Impact of Key Frame Interval*

Since the number of labeled frames is also important to the performance of the proposed method, we further evaluate the impact of the key frame interval to the segmentation accuracy. Specifically, we evaluate the proposed method with a key frame provided in every 5, 10, and 15 frames, and the result is shown in Fig. 11(a). As expected, the performance of the proposed method improves with the number of labeled frames, and the gain is trivial when the key frame interval is smaller than 10, which we use for the experiment.

### *E. Convergence Rate*

Furthermore, we also demonstrate the evolution of the proposed model in each iteration, which is shown in Fig. 11(b). Basically, the algorithm converges after five iterations for the test sequences. The initial model after the first iteration is based on inductive SVM, which is described in Section V. As a result, the proposed transductive learning model is advantageous over the inductive one, which can be concluded by comparing the accuracy of the first iteration and the fifth iteration.

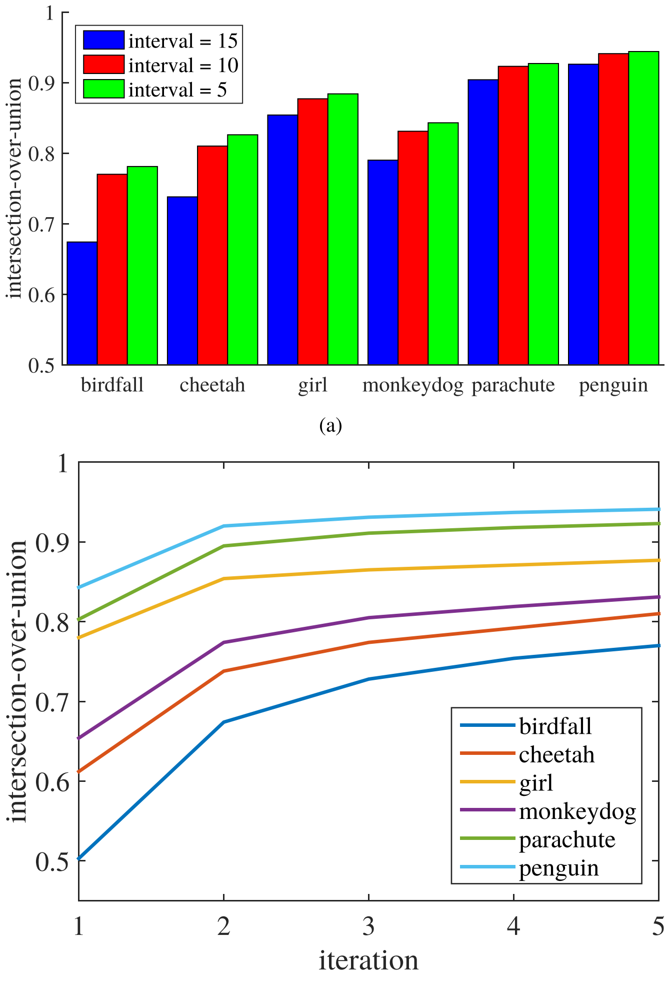
### *F. Evaluation on Additional Data Sets*

In addition to the original SegTrack data set, we also evaluate the proposed method on the SegTrack-v2 data set [24] and the segmented foreground objects (SFOs) data set [39]. Specifically, the SegTrack-v2 data set enhances the original SegTrack data set with eight additional sequences, and the SFO data set consists of five video sequences. The proposed method is compared with FOS, KeySeg, and SnapCut, whose source codes are publicly available. The MPEs are shown in Tables III and IV.

The experimental result shows that the proposed method outperforms FOS, KeySeg, and SnapCut in all additional sequences in the SegTrack-v2 data set, and achieves the lowest MPE in three out of five sequences in the SFO data set.

### *G. Computational Complexity*

Finally, the computational complexity of the proposed method will be discussed. The experiments are conducted on a computer with Intel Xeon E5520 CPU, 4 GB RAM, and Ubuntu 14.04 LTS operating system, and the algorithm is implemented with MATLAB. In general, the execution time of the proposed algorithm varies with the size of the video and the size of the foreground object. On average, for the SegTrack data set, it takes about 4 min to process each frame. In particular, 40% of the runtime is spent on computing the segmentation hypotheses, 30% of the runtime is spent on training the transductive SVM classifiers, and the other operations occupy the rest of the 30% runtime altogether.



(a)

(b)

Fig. 11. (a) Impact of the key frame interval to the accuracy. (b) Segmentation accuracy after each iteration.

## VII. CONCLUSION

In this paper, a transductive multicomponent video segmentation algorithm is proposed, which is capable of segmenting the object of interest in the frames of the video clip while preserving the temporal consistency. In particular, the proposed method uses the multiple foreground model to capture the variances in visual appearance of the objects. Moreover, an energy function is introduced to evaluate the quality of segmentation based on the foreground prior, the low-level features, and the temporal consistency. To estimate the parameters of the foreground models, a transductive learning algorithm is proposed to jointly minimize the prediction error of the labeled frames and the unlabeled frames. Specifically, frames are organized into a tree-structured graphical model named temporal tree, where visually similar and temporally coherent frames are grouped together as branches. Experimental results show that the proposed method outperforms many state-of-the-art video segmentation methods in the public benchmark.

## References

[1] D. Zhang, O. Javed, and M. Shah, "Video object segmentation through spatially accurate and temporally dense extraction of primary object regions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Portland, OR, USA, Jun. 2013, pp. 628–635.

[2] T. Ma and L. J. Latecki, "Maximum weight cliques with mutex constraints for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Providence, RI, USA, Jun. 2012, pp. 670–677.

[3] S. A. Ramakanth and R. V. Babu, "SeamSeg: Video object segmentation using patch seams," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Columbus, OH, USA, Jun. 2014, pp. 376–383.

[4] X. Cao, F. Wang, B. Zhang, H. Fu, and C. Li, "Unsupervised pixel-level video foreground object segmentation via shortest path algorithm," *Neurocomputing*, vol. 172, pp. 235–243, Jan. 2015.

[5] P. Ochs, J. Malik, and T. Brox, "Segmentation of moving objects by long term video analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 6, pp. 1187–1200, Jun. 2014.

[6] B. Zhang, H. Zhao, and X. Cao, "Video object segmentation with shortest path," in *Proc. 20th ACM Int. Conf. Multimedia (ACMMM)*, Nara, Japan, Oct. 2012, pp. 801–804.

[7] X. Bai, J. Wang, D. Simons, and G. Sapiro, "Video SnapCut: Robust video object cutout using localized classifiers," *ACM Trans. Graph.*, vol. 28, no. 3, Aug. 2009, Art. no. 70.

[8] B. L. Price, B. S. Morse, and S. Cohen, "LIVEcut: Learning-based interactive video segmentation by evaluation of multiple propagated cues," in *Proc. IEEE 12th Int. Conf. Comput. Vis. (ICCV)*, Kyoto, Japan, Sep./Oct. 2009, pp. 779–786.

[9] M. Grundmann, V. Kwatra, M. Han, and I. Essa, "Efficient hierarchical graph-based video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, San Francisco, CA, USA, Jun. 2010, pp. 2141–2148.

[10] Y. Huang, Q. Liu, and D. Metaxas, "Video object segmentation by hypergraph cut," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Miami, FL, USA, Jun. 2009, pp. 1738–1745.

[11] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient N-D image segmentation," *Int. J. Comput. Vis.*, vol. 70, no. 2, pp. 109–131, Nov. 2006.

[12] D. Varas and F. Marques, "Region-based particle filter for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Columbus, OH, USA, Jun. 2014, pp. 3470–3477.

[13] C. Rother, T. Minka, A. Blake, and V. Kolmogorov, "Cosegmentation of image pairs by histogram matching—Incorporating a global constraint into MRFs," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog. (CVPR)*, New York, NY, USA, Jun. 2006, pp. 993–1000.

[14] W. Wang, J. Shen, X. Li, and F. Porikli, "Robust video object cosegmentation," *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 3137–3148, Oct. 2015.

[15] D. Zhang, O. Javed, and M. Shah, "Video object co-segmentation by regulated maximum weight cliques," in *Proc. 13th Eur. Conf. Comput. Vis. (ECCV)*, Zürich, Switzerland, Sep. 2014, pp. 551–566.

[16] Z. Lou and T. Gevers, "Extracting primary objects by video co-segmentation," *IEEE Trans. Multimedia*, vol. 16, no. 8, pp. 2110–2117, Dec. 2014.

[17] C. Wang, Y. Guo, J. Zhu, L. Wang, and W. Wang, "Video object co-segmentation via subspace clustering and quadratic pseudo-Boolean optimization in an MRF framework," *IEEE Trans. Multimedia*, vol. 16, no. 4, pp. 903–916, Jun. 2014.

[18] H. Fu, D. Xu, B. Zhang, and S. Lin, "Object-based multiple foreground video co-segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Columbus, OH, USA, Jun. 2014, pp. 3166–3173.

[19] H. Fu, D. Xu, B. Zhang, S. Lin, and R. K. Ward, "Object-based multiple foreground video co-segmentation via multi-state selection graph," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3415–3424, Nov. 2015.

[20] W. Wang, J. Shen, and F. Porikli, "Saliency-aware geodesic video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 3395–3402.

[21] W. Wang, J. Shen, and L. Shao, "Consistent video saliency using local gradient flow optimization and global refinement," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4185–4196, Nov. 2015.

[22] B. Luo, H. Li, T. Song, and C. Huang, "Object segmentation from long video sequences," in *Proc. 23rd ACM Int. Conf. Multimedia (ACMMM)*, Brisbane, QLD, Australia, Oct. 2015, pp. 1187–1190.

[23] A. Khoreva, F. Galasso, M. Hein, and B. Schiele, "Classifier based graph construction for video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 951–960.

[24] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg, "Video segmentation by tracking many figure-ground segments," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Sydney, NSW, Australia, Dec. 2013, pp. 2192–2199.

[25] 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 Trans. Circuits Syst. Video Technol.*, vol. 23, no. 6, pp. 921–934, Jun. 2013.

[26] J. Carreira and C. Sminchisescu, "CPMC: Automatic object segmentation using constrained parametric min-cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1312–1328, Jul. 2012.

[27] M. Leordeanu, R. Sukthankar, and C. Sminchisescu, "Generalized boundaries from multiple image interpretations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1312–1324, Jul. 2014.

[28] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Columbus, OH, USA, Jun. 2014, pp. 580–587.

[29] E. Erdem and A. Erdem, "Visual saliency estimation by nonlinearly integrating features using region covariances," *J. Vis.*, vol. 13, no. 4, p. 11, Mar. 2013.

[30] V. N. Vapnik, *Statistical Learning Theory*. Hoboken, NJ, USA: Wiley, 1998.

[31] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. 16th Int. Conf. Mach. Learn. (ICML)*, Bled, Slovenia, Jun. 1999, pp. 200–209.

[32] Z. Xu, R. Jin, J. Zhu, I. King, and M. Lyu, "Efficient convex relaxation for transductive support vector machine," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Vancouver, BC, Canada, Dec. 2008, pp. 1641–1648.

[33] D. Tsai, M. Flagg, A. Nakazawa, and J. M. Rehg, "Motion coherent tracking using multi-label MRF optimization," *Int. J. Comput. Vis.*, vol. 100, no. 2, pp. 190–220, Nov. 2012.

[34] Y. J. Lee, J. Kim, and K. Grauman, "Key-segments for video object segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Vancouver, BC, Canada, Nov. 2011, pp. 1995–2002.

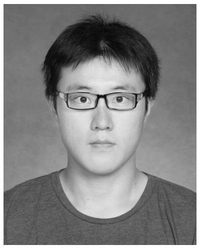
[35] A. Papazoglou and V. Ferrari, "Fast object segmentation in unconstrained video," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Sydney, NSW, Australia, Dec. 2013, pp. 1777–1784.

[36] D. Banica, A. Agape, A. Ion, and C. Sminchisescu, "Video object segmentation by salient segment chain composition," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Sydney, NSW, Australia, Dec. 2013, pp. 283–290.

[37] P. Chockalingam, N. Pradeep, and S. Birchfield, "Adaptive fragments-based tracking of non-rigid objects using level sets," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Kyoto, Japan, Sep. 2009, pp. 1530–1537.

[38] V. Badrinarayanan, I. Budvytis, and R. Cipolla, "Semi-supervised video segmentation using tree structured graphical models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2751–2764, Nov. 2013.

[39] Y.-M. Chen, I. V. Bajic, and P. Saeedi, "Coarse-to-fine moving region segmentation in compressed video," in *Proc. IEEE 10th Workshop Image Anal. Multimedia Interact. Services (WIAMIS)*, London, U.K., May 2009, pp. 45–48.



**Botao Wang** (S'15) received the B.S. degree in electronics engineering from Shanghai Jiao Tong University, Shanghai, China, in 2010, where he is currently pursuing the Ph.D. degree.

His research interests include object detection, scene classification, and image understanding.

**Zhihui Fu** received the B.S. and M.S. degrees in electronics engineering from Shanghai Jiao Tong University, Shanghai, China, in 2012 and 2015, respectively.

**Hongkai Xiong** (M'01–SM'10) received the Ph.D. degree in communication and information systems from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2003.

He was with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA, as a Research Scholar, from 2007 to 2008. From 2011 to 2012, he was a Scientist with the Division of Biomedical Informatics, University of California at San Diego, San Diego, CA, USA. He is currently a Distinguished Professor with the Department of Electronic Engineering, SJTU. He has authored over 140 refereed journal/conference papers. His research interests include source coding/network information theory, signal processing, computer vision, and machine learning.

Dr. Xiong has been a member of the Innovative Research Group of the National Natural Science Foundation since 2012. He received the Best Student Paper Award at the 2014 IEEE Visual Communication and Image Processing, the best paper award at the 2013 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, and the Top 10% Paper Award at the 2011 IEEE International Workshop on Multimedia Signal Processing. He also received the New Century Excellent Talents in University Award from the Ministry of Education of China, in 2009, and the Shanghai Shu Guang Scholar Award in 2013. In 2011, he received the First Prize in the Shanghai Technological Innovation Award for Network-Oriented Video Processing and Dissemination: Theory and Technology. In 2010 and 2013, he received the SMC-A Excellent Young Faculty Award of SJTU. He served as a Technical Program Committee Member for prestigious conferences, such as ACM Multimedia, the International Conference on Image Processing, the International Conference on Multimedia and Expo, and the International Symposium on Circuits and Systems.

**Yuan F. Zheng** (F'97) received the bachelor's degree from Tsinghua University, Beijing, China, in 1970, and the M.S. and Ph.D. degrees in electrical engineering from Ohio State University, Columbus, OH, USA, in 1980 and 1984, respectively.

He was with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC, USA, from 1984 to 1989. Since 1989, he has been with Ohio State University, where he is currently a Professor and was the Chairman of the Department of Electrical and Computer Engineering from 1993 to 2004. From 2004 to 2005, he spent a sabbatical year with Shanghai Jiao Tong University, Shanghai, China, where he continued as the Dean of the School of Electronic, Information and Electrical Engineering until 2008. His research interests include wavelet transform for image and video, and object classification and tracking, and robotics, which includes robotics for life science applications, multiple robots coordination, legged walking robots, and service robots.

Dr. Zheng received the Presidential Young Investigator Award from Ronald Reagan in 1986, and the Research Awards from the College of Engineering, Ohio State University, in 1993, 1997, and 2007. Along with his students, he received the Best Conference and Best Student Paper Award in 2000, 2002, and 2006, and also received the Fred Diamond for Best Technical Paper Award from the Air Force Research Laboratory, Rome, NY, USA, in 2006. He has been on the Editorial Board of five international journals. In 2004, he was appointed to the International Robotics Assessment Panel by the National Science Foundation, NASA, and the National Institutes of Health to assess the robotics technologies worldwide in 2004 and 2005.