Saliency-aware Video Object Segmentation

Wenguan Wang, Jianbing Shen, *Senior Member, IEEE*, Ruigang Yang, *Senior Member, IEEE*, and Fatih Porikli, *Fellow, IEEE*

Abstract—Video saliency, aiming for estimation of a single dominant object in a sequence, offers strong object-level cues for unsupervised video object segmentation. In this paper, we present a geodesic distance based technique that provides reliable and temporally consistent saliency measurement of superpixels as a prior for pixel-wise labeling. Using undirected intra-frame and inter-frame graphs constructed from spatiotemporal edges or appearance and motion, and a skeleton abstraction step to further enhance saliency estimates, our method formulates the pixel-wise segmentation task as an energy minimization problem on a function that consists of unary terms of global foreground and background models, dynamic location models, and pairwise terms of label smoothness potentials. We perform extensive quantitative and qualitative experiments on benchmark datasets. Our method achieves superior performance in comparison to the current state-of-the-art in terms of accuracy and speed.

Index Terms—Video saliency, video object segmentation, geodesic distance, spatiotemporal object prior.

1 Introduction

Unsupervised video object segmentation, a key challenge in computer vision, aims at partitioning multiple video frames into objects and background regions. Such an automatic segmentation has been shown to benefit a variety of applications such as video summarization, video compression, content based video retrieval and human-computer interaction, to name a few.

Traditionally, video object segmentation task is performed with motion and appearance information represented by motion vectors, feature point trajectories, color descriptors, and boundary indicators. Depending on the availability and quality of these inputs, object regions are usually obtained after complicated and fragile inference procedures often with preset assumptions of object and camera motion. In simple scenarios where the foreground object moves distinctly from its background, grouping of motion vectors and feature point trajectories generates semantically meaningful segments. Several works [1], [2], [3] analyzed point trajectories to leverage the motion information. But, what about if a part of the object remains static? In typical complex videos, the assumption of motion consistency may result in oversegmentation, thus failing to extract entire object regions. Utilizing both motion and appearance cues seems to be a better choice as it was adopted by many methods [4], [5], [6], [7], [8], [9]. Specially, [4], [5], [6] generate a large number of object proposals [10], [11], [12] in every frame using these cues, and cast the task of video object segmentation as the problem of inferring and selecting the most relevant object proposal.

- *This work was supported in part by the National Basic Research Program of China (973 Program) (No. 2013CB328805), the National Natural Science Foundation of China (No. 61272359), the Australian Research Council's Discovery Projects funding scheme (DP150104645), and the Fok Ying-Tong Education Foundation for Young Teachers. Specialized Fund for Joint Building Program of Beijing Municipal Education Commission. (Corresponding author: Jianbing Shen).*
- *W. Wang and J. Shen are with Beijing Laboratory of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, (email: shenjianbing@bit.edu.cn)*
- *R. Yang is with the University of Kentucky, Lexington, KY 40507. (email: ryang@cs.uky.edu)*
- *F. Porikli is with the Research School of Engineering, Australian National University, and NICTA. (email: fatih.porikli@anu.edu.au)*

However, all these approaches still face many difficulties. On one hand, they all require complicated object inference techniques, which comes with a high computational expense. On the other hand, they impose heuristically chosen cues which may not be the right choice for a general class of objects. Besides, proposal based methods sustain the disadvantage that correct proposals are often few or do not exist at all when the foreground object is small or similar to the background. We can ask whether there is any reliable object descriptor that can be employed for a general class of video objects. We address this challenge by giving emphasis to the value of video saliency to automatically identify visually prominent object regions in dynamic scenes. Our intuition is that potentially discriminative yet confined motion and appearance cues should be combined with more comprehensive spatiotemporal saliency cues in order to generate reliable object prior. Once a reliable saliency prior is built, estimating refined appearance models and then in turn generating accurate object segments becomes feasible. This motivates us to decompose the automatic segmentation problem into two stages: video saliency detection and video object partitioning.

For an effective solution to unsupervised video segmentation, we need the capability to detect salient regions in a video. While salient object detection in still images has been exploited in the past, computing spatiotemporal saliency in videos is still an active area of research since extending image based algorithms to video is nontrivial. Temporal coherence yields significant information, nevertheless, it is inevitably susceptible to noise due to nonuniform background motions and well-known motion estimation errors. Moreover, most video saliency methods simply treat the motion feature as another cue within their image saliency models [13], [14], [15], lacking an elegant framework to incorporate intra-frame and inter-frame information in a unified fashion.

In this paper, we aim to partition the foreground objects from their backgrounds in all frames of a given video sequence without any user assistance or contextual assumptions. To this end, we propose a video object segmentation method that consists of a superpixel based spatiotemporal saliency prior detection stage and pixel based binary labeling stage that runs in a recursive fashion. Our proposed video segmentation framework is depicted

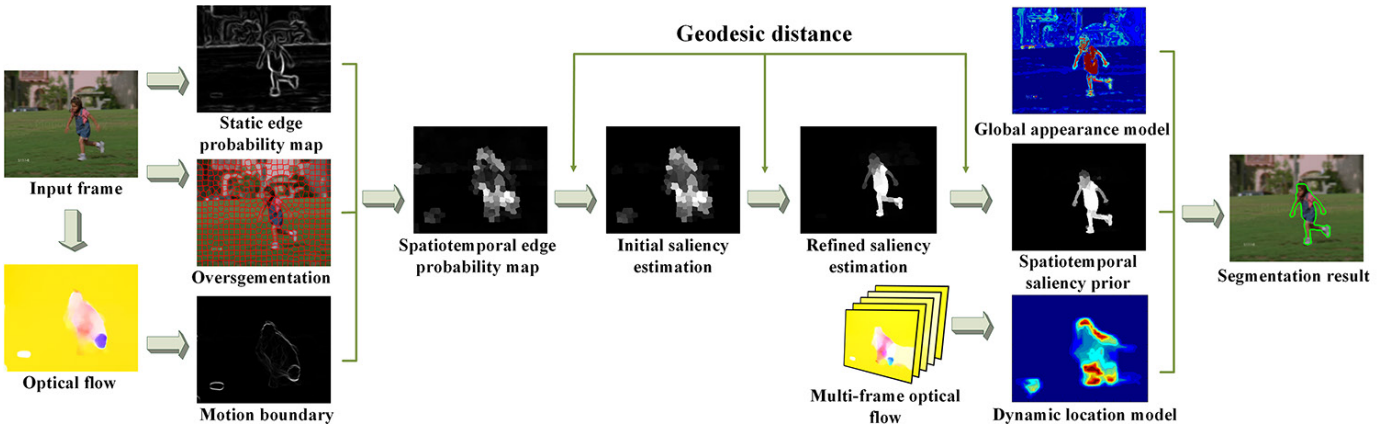

Fig. 1. Overview of our video object segmentation framework. Input frame is over-segmented into superpixels and a spatiotemporal edge map is produced by the combination of static edge probability and optical flow gradient magnitude. For each superpixel, we compute its object-probability and the refined saliency estimate via intra-frame graph and inter-frame graph, respectively. An object skeleton abstraction method is further derived for obtaining final saliency estimates via biasing the central skeleton regions with higher saliency values. Finally, spatiotemporal saliency priors, global appearance models and dynamic location models are combined for producing correct video segmentation.

in Fig. 1. We first introduce a spatiotemporal saliency prior that incorporates spatial and temporal stimulus and temporal coherence into a unified, geodesic distance based model. The geodesic distance, which has been shown to be effective in many interactive computer vision applications [16], [17], [18], [19], has the power of abstracting object structure to efficiently determine its central regions by assigning higher saliency values to more representative regions. Saliency of a region is measured by its shortest geodesic distance to background regions in inter-frame and intra-frame graphs. Hence, we design a skeleton abstraction method that explicitly incorporates weak object structure and emphasizes the saliency values of the central skeleton regions based on geodesic distances. After obtaining video saliency, we integrate saliency prior, dynamic location models as well as global appearance models into an energy minimization that is optimized via graph-cuts to generate final video object segments. This paper builds upon and extends our recent work in [20] with a more in depth discussion of the algorithm and expanded evaluations. We further introduce a new skeleton regions abstraction method that regularizes the original regions of object with higher saliency. Our source code and supplemental materials will be available at [1].

To summarize, our main contributions are:

- A unified framework that incorporates video saliency for unsupervised pixel-wise labeling of foreground objects using an energy function that contains three unary and two pair-wise terms.
- A new formulation for video saliency by exploiting intra-frame and inter-frame relevancy via undirected graphs. For the intra-frame stimulus, we employ geodesic distance on spatiotemporal edges within a single frame. We construct the inter-frame graph for temporal coherence between consecutive frames.
- A geodesic distance based weighting of intra-frame and inter-frame graphs based on the observation that salient regions have higher geodesic distances to background regions.
- A greedy skeleton abstraction scheme for iteratively selecting confident foreground regions.

1. http://github.com/shenjianbing/saliencysegment

2 Related Work

In this section, we give a brief overview of recent works in unsupervised video segmentation and saliency detection.

2.1 Unsupervised Video Segmentation

A variety of techniques have been proposed for unsupervised video segmentation in the past decade. Most approaches are based on bottom-up models using low-level features such as motion, color, and edge orientation. In particular, the importance of the motion information was emphasized in many works [1], [2], [3], [21], [22], [23], [24]. While the use of short duration motion boundaries in pairs of subsequent frames is not uncommon [22], several methods [1], [2], [3], [21], [23] argued that motion should be analyzed over longer periods, as such long term analysis is able to decrease the intra-object variance of motion relative to the inter-object variance and propagate motion information to frames in which the object remains static. For this, [2] grouped pixels with coherent motion computed via long-range motion vectors from the past and future frames. Similarly, the work in [1] offered a framework for trajectory-based video segmentation through building affinity matrix between pairs of trajectories. In [3], discontinuities of embedding density between spatially neighboring trajectories were detected. Incorporating higher order motion models, a clustering method for point tracks was proposed in [23]. In general, motion based methods suffer difficulties when different parts of an object exhibit nonhomogeneous motion patterns. This problem is exacerbated further with the absence of a strong prior for object. Moreover, these approaches require careful selection of a suitable model especially for the trajectory clustering process, which often comes with a high computation complexity, as [7] pointed out.

There were previous efforts [4], [5], [6], [25], [26], [27] that presented optimization frameworks for bottom-up segmentation employing both appearance and motion cues. Several methods [7], [8], [9], [28], [29] proposed to select primary object regions in object proposal domain based on the notion of what a generic object looks like. These approaches benefit from the work of object hypotheses proposals [10], [11], [12] that offer a large number of object candidates in every frame. Therefore, segmenting video object is transformed into an object region selection problem. In the selection process, both motion and appearance cues are

used to measure the *objectness* of a proposal. More specifically, a clustering process was introduced for finding objects by [7], a constrained maximum weight cliques technique to model the selection process was imposed [8], and a layered directed acyclic graph based framework was presented by [9]. Work of [28] segmented video objects by ranking spatiotemporal segment proposals with moving objectness detector trained on image and motion fields. In [29], tracking and segmentation were integrated to detect the primary object proposal and handle the video segmentation task. The main drawbacks of the proposal based algorithms are their high computational cost [30] associated with proposal generation and complicated object inference schemes.

2.2 Saliency Detection for Image and Video

Saliency detection [31] is originally a task of simulating the human visual system for predicting scene locations where a human observer may fixate. Recent research has shown that extracting salient objects or regions is more beneficial to a wide range of computer vision applications. Saliency detection methods in general can be categorized as either bottom-up or top-down approaches. Top-down approaches [32], [33], [34], [35] are goal-directed and require an explicit understanding of the context of the image. Supervised learning with a specific class is therefore a frequently adopted principle. Most of the saliency detection methods [36], [37] are based on bottom-up visual attention mechanisms, which are independent of the knowledge of the content in the image.

Inspired by visual perception studies that indicate *contrast* is a major factor in visual attention mechanisms, numerous bottom-up models have been proposed based on different mathematical formulations of contrast. Many methods [32], [39], [40] assumed that globally infrequent features are more salient, and adopted various low level features, such as intensity, color and orientation. More specially, in [41], a content-aware saliency detection with the consideration of the contrast from both local and global perspectives was built. [42] presented a saliency method based on the fusion of different feature channels and local center-surround hypothesis. In [43], two saliency indicators, global appearance contrast and spatially compact distribution, were considered. Recently, several approaches [44], [45], [46] exploited background information, called *boundary prior*. These methods use image boundaries as background, further enhancing saliency computation.

While image saliency detection has been extensively studied, computing spatiotemporal saliency for videos is a relatively new problem. Different from image saliency detection, moving objects catch more attention of human beings than static ones. In other words, motion is the most important cue for video saliency detection, which makes deeper exploration of the inter-frame information crucial. Gao *et al.* [13] extended their image saliency model [47] by adding the motion channel for prediction of human eye fixations in dynamic scenes based on the center-surround hypothesis. Similarly, Mahadevan *et al.* [14] combined center-surround saliency with dynamic textures for spatiotemporal saliency using the saliency model in [47]. The phase spectrum of the Fourier transform is considered to be the key element in obtaining the location of salient regions in [38]. Seo *et al.* [15] computed the local regression kernels from the given video, measuring the likeness of a pixel (or voxel) to its surrounding. They extended their model for video saliency detection by extracting a feature vector from each 3-D cube. Recently, [5] used a statistical framework and local contrast in illumination, color, and motion for formulating final saliency maps. [48] proposed a cluster-based saliency method, where three visual attention cues, contrast, spatial, and global correspondence, are devised to measure the cluster saliency. [49] adopted space-time saliency to generate a low-frame-rate video from a high-frame-rate input using various low-level features and region-based contrast analysis. In [50], gradient flow field is proposed for detecting salient object regions in video sequences with global optimization.

3 Spatiotemporal Saliency Prior

Our video object segmentation method consists of two stages: superpixel based spatiotemporal saliency prior detection and pixel based binary labeling. Here, we explain the saliency stage first.

To achieve reliable saliency estimation, our method combines psychophysically motivated low-level features, such as color, edge, and motion boundary in a unified geodesic distance based framework. Fig. 2 shows intermediate stages of our video saliency. First, input frames are partitioned into superpixels for computational efficiency (Fig. 2-b). We then extract two types of edges: spatial edges (Fig. 2-c) within the same frame, and motion boundary edges (Fig. 2-d) across neighboring frames. These two features are explicitly integrated into a single spatiotemporal edge map (Fig. 2-e) as described in Section 3.1. In Section 3.2, geodesic distance is adopted in an intra-frame graph for computing rough object probability of each superpixel as given Fig. 2-f. To improve the saliency estimation, in Section 3.3, an inter-frame graph is incorporated with geodesic measure for producing an initial spatiotemporal saliency map as shown in Fig. 2-g. Finally, we apply a skeleton abstraction method that amplifies the saliency values of central skeleton regions based on geodesic distances to incorporate weak object structure, which can be seen in Fig. 2-h and is detailed in Section 3.4.

3.1 Spatiotemporal Edge Generation

Human visual perception [51], [52] suggest that basic visual features such as motion and edges are processed at the human pre-attentive stage for visual attention, which motivates us to combine spatial edge and motion boundary cues into a coalescent spatiotemporal edge map. Both color and motion discontinuities provide valuable evidence in predicting object boundaries. As shown in Fig. 2, spatial color discontinuities in a single frame and optical flow field estimated from two consecutive frames reveal the important regions of the video frames. We build our approach on these two indicators.

Given an input video sequence $\{F^1, F^2, \cdots\}$, we compute a spatial edge probability map E_c^k of k-th frame F^k using [53]. The value of $E_c^k(x)$, normalized to [0, 1], represents the probability of edge at the corresponding pixel x. The optical flow between the pairs of subsequent frames are obtained by the large displacement motion estimation algorithm [54]. Let V^k be the optical flow field of F^k, and we compute the motion gradient magnitude E_o^k of V^k as $E_o^k(x) = \|\nabla V^k(x)\|$. We oversegment each frame into superpixels using SLIC [55]. Let $\mathbf{Y}^k = \{y_1^k, y_2^k, \cdots\}$ be the superpixel set of F^k. Given the pixel edge map E_c^k, the edge probability of superpixel y_n^k is computed as the average value of the pixels within y_n^k. This generates a superpixel-wise edge map E_c^k. Similarly, the optical flow magnitude map E_o^k is re-computed on superpixel level. Then, we generate a spatiotemporal edge map E^k as:

$$\mathrm{E}^k = E_c^k \cdot E_o^k. \tag{1}$$

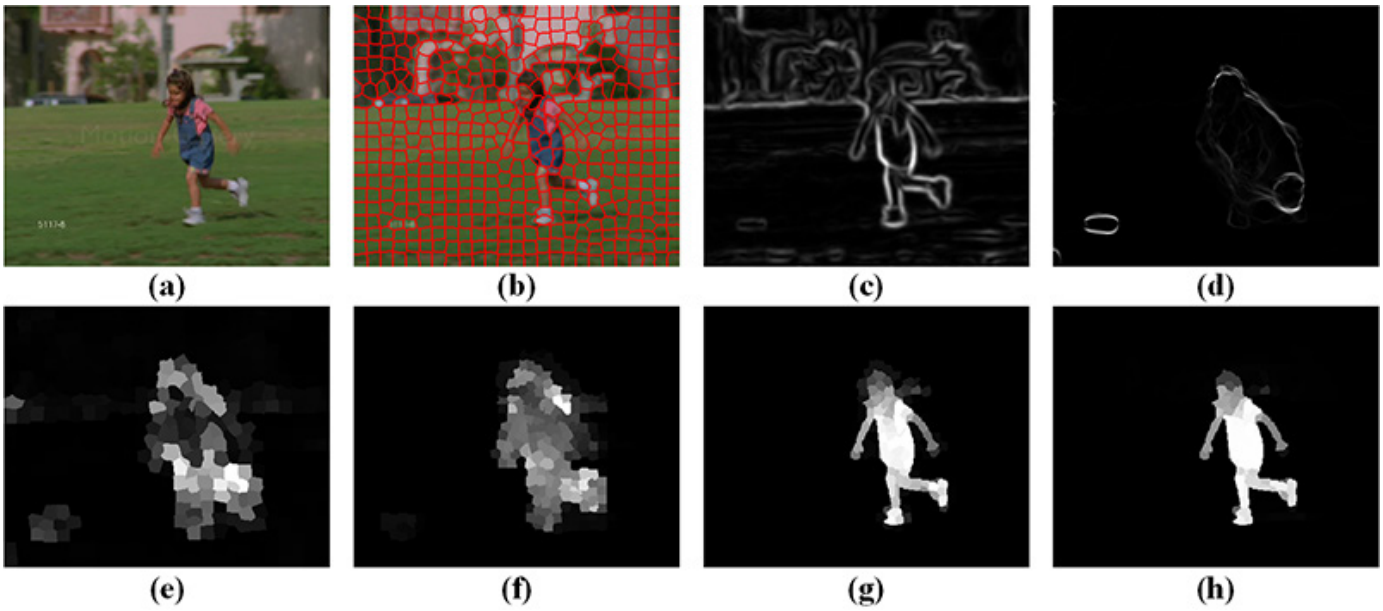

Fig. 2. Overview of geodesic distance based spatiotemporal saliency prior. (a) Input frame F^k. (b) Oversegmentation of F^k into superpixels $\mathbf{Y}^k$. (c) Spatial edge probability map E_c^k of F^k. (d) Gradient magnitude E_o^k of optical flow of F^k. (e) Spatiotemporal edge map E^k via (1). (f) Object result P^k via intra-frame graph. (g) Saliency result S^k via inter-frame graph. (h) Final video saliency via the proposed skeleton abstraction method.

The intuition behind the design of (1) is that, distinct motion patterns and spatial gradients are indicators of the location of salient foreground object. This can be easily observed in Fig. 2-e, where object superpixels either have high spatiotemporal edge map values or are surrounded by those high valued superpixels.

3.2 Intra-frame Graph Construction

To highlight the foreground regions that have high spatiotemporal edge values or are surrounded by regions with high spatiotemporal edge values, we employ geodesic distance to compute a rough object probability map. The geodesic distance $d_{geo}(v_1, v_2, \mathcal{G})$ between any two nodes v_1, v_2 in graph $\mathcal{G}$ is the smallest integral of a weight function W over all possible paths between v_1 and v_2:

$$d_{geo}(v_1, v_2, \mathcal{G}) = \min_{C_{v_1,v_2}} \int_{v_2}^{v_1} |W(m) \cdot \dot{\mathcal{C}}_{v_1,v_2}(m)| dm, \quad (2)$$

where $\mathcal{C}_{v_1,v_2}(m)$ is a path connecting the nodes v_1, v_2.

For frame F^k, we construct an undirected weighted graph $\mathcal{G}^k = \{\mathcal{V}^k, \mathcal{E}^k\}$ with superpixels $\mathbf{Y}^k$ as nodes $\mathcal{V}^k$ and the links between adjacent nodes as edges $\mathcal{E}^k$. Based on the graph structure, we derive a $|\mathcal{V}^k| \times |\mathcal{V}^k|$ weight matrix W^k, where $|\mathcal{V}^k|$ is the number of nodes in $\mathcal{V}^k$. The (m, n)-th element of W^k indicates the weight of edge $e_{mn}^k \in \mathcal{E}^k$ between adjacent superpixels Y_m^k and Y_n^k:

$$W_{mn}^k = \|\mathrm{E}^k(y_m^k) - \mathrm{E}^k(y_n^k)\|, \quad (3)$$

where $\mathrm{E}^k(Y_m^k)$ and $\mathrm{E}^k(Y_n^k)$ correspond to the spatiotemporal boundary probability of superpixels Y_m^k and Y_n^k, separately.

For superpixel y_n^k, the probability $P^k(y_n^k)$ of being foreground is computed by the shortest geodesic distance to the image boundaries using

$$P^k(y_n^k) = \min_{q \in \mathbf{Q}^k} d_{geo}(y_n^k, q, \mathcal{G}^k), \quad (4)$$

where $\mathbf{Q}^k$ indicate the superpixels along the four boundaries of F^k. The geodesic distance $d_{geo}(v_1, v_2, \mathcal{G}^k)$ between any two superpixels $v_1, v_2 \in \mathcal{V}^k$ in graph $\mathcal{G}^k$ can be computed as:

$$d_{geo}(v_1, v_2, \mathcal{G}^k) = \min_{C_{v_1,v_2}} \sum_{m,n} W_{mn}^k, \quad m, n \in \mathcal{C}_{v_1,v_2}. \quad (5)$$

which can be seen as the accumulated edge weights along their shortest path on graph $\mathcal{G}^k$.

If a superpixel is outside the desired object, its probability value is small because there exists a pathway to image boundaries that does not pass the regions with high spatiotemporal edge value. Whereas, if a superpixel is inside the object, this superpixel is surrounded by the regions with large probabilities of edges, which increases the geodesic distance to image boundaries. Since our graph is very sparse, the shortest paths of all superpixels are efficiently computed by the Johnson algorithm [56].

3.3 Inter-frame Graph Construction

The foreground probability map P^k reveals the foreground object region but it is not complete and precise. In particular, probability values of the true background regions near the object boundary may have high values due to the oversegmentation process. Besides, inaccurate optical flow estimation may result in erroneous values. By the definition of saliency, foreground and background regions should be visually different, and object regions should be temporally continuous between consecutive frames. These motivate us to estimate saliency between pairs of adjacent frames.

For each pair of adjacent frames F^k and F^{k+1}, we construct an undirected weighted graph $\mathcal{G}'^k = \{\mathcal{V}'^k, \mathcal{E}'^k\}$. The nodes $\mathcal{V}'^k$ consist of the superpixels $\mathbf{Y}^k$ of F^k and the superpixels $\mathbf{Y}^{k+1}$ of F^{k+1}. There are two types of edges: intra-frame edges that link spatially adjacent superpixels and inter-frame edges that connect temporally adjacent superpixels. The superpixels are spatially connected if they are adjacent in the same frame. Temporally adjacent superpixels refer to the superpixels which belong to

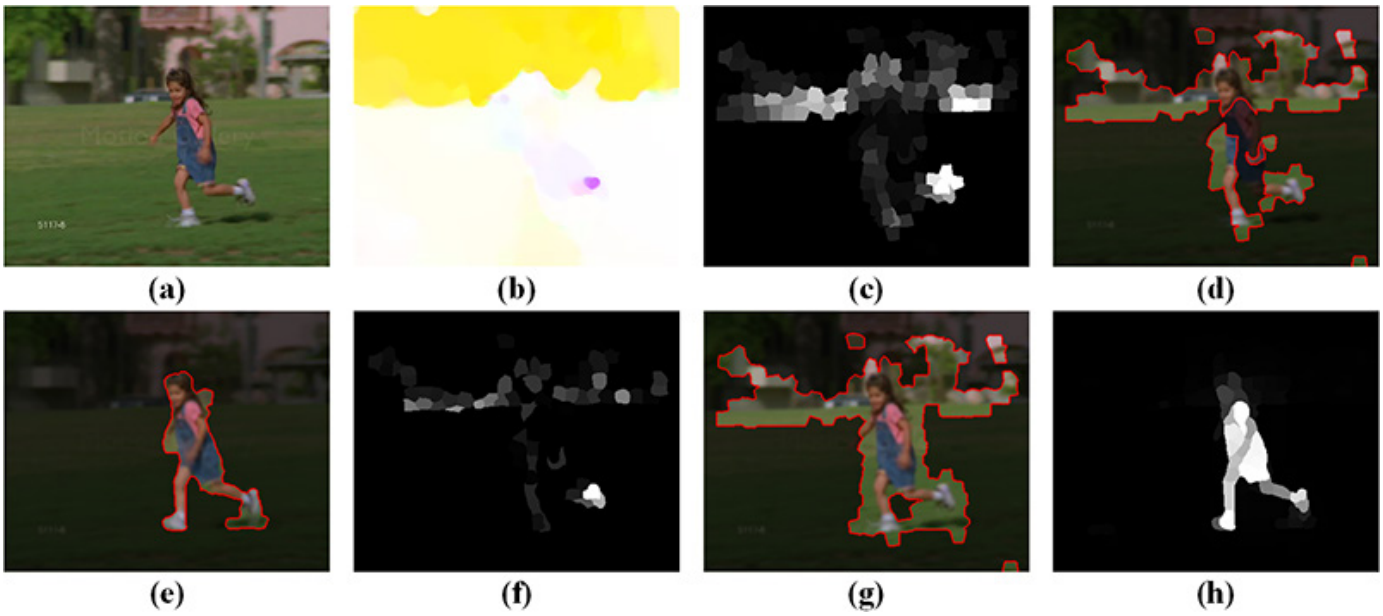

Fig. 3. Illustration of inter-frame graph construction. (a) Frame F^k. (b) Optical flow flied V^k. (c) When the optical flow estimation is not accurate, object probabilities P^k are degraded. (d) F^k is decomposed into background regions $\mathbf{B}^k$ and object-like regions $\mathbf{U}^k$ by self-adaptive threshold σ^k in (6). The black regions indicate $\mathbf{B}^k$, while the bright regions indicate $\mathbf{U}^k$. (e) The decomposition of previous frame F^{k-1}. (f) The object-like regions $\mathbf{U}^{k-1}$ of F^{k-1} are projected onto F^k. (g) Spatiotemporal saliency result S^k for F^k with consideration of (d) and (e). (h) Spatiotemporal saliency result S^k for F^k with consideration of (e) and (f).

different frames but have overlap. We assign the edge weight as the Euclidean distance between their mean colors in the CIE-Lab color space. For each frame, we use a self-adaptive threshold to decompose frame F^k into background regions $\mathbf{B}^k$ and object-like regions $\mathbf{U}^k$ through the probability map P^k. The threshold σ^k for F^k is computed as

$$\sigma^k = \mu(P^k), \tag{6}$$

where $\mu(\cdot)$ is the mean probability of all pixels within the frame F^k. We assign $\mathbf{U}^k$ and $\mathbf{B}^k$ of k-th frame as

$$\begin{aligned} \mathbf{U}^k = \; & \{y_n^k | P^k(y_n^k) > \sigma^k\} \\ & \cup \{y_n^k | y_n^k \text{ is temporally connected to } \mathbf{U}^{k-1}\}, \\ \mathbf{B}^k = \; & \mathbf{Y}^k - \mathbf{U}^k. \end{aligned} \tag{7}$$

In a causal system, previously determined object regions offer valuable information to eliminate artifacts due to inaccurate optical flow estimation. Therefore, we project object-like regions of prior frame F^{k-1} onto frame F^k. Our motivation can be observed in Fig. 3. The object estimation result of frame F^k (Fig. 3-c) is not ideal, due to the incorrect optical flow estimation (Fig. 3-b). If F^k is segmented using only the self-adaptive threshold T^k defined in (7), an inferior decomposition is generated (Fig. 3-d), further leading into incorrect saliency result (Fig. 3-g). When the previous estimation is projected, a more correct decomposition is obtained (Fig. 3-f), and more consistent saliency is attained (Fig. 3-h).

Based on the graph $\mathcal{G}'^k$, we compute saliency map S^k (S^{k+1}) for frame F^k (F^{k+1}) as follows:

$$\begin{aligned} S^k(y_n^k) &= \min_{b \in \mathbf{B}^k \cup \mathbf{B}^{k+1}} d_{geo}(y_n^k, b, \mathcal{G}'^k), \\ S^{k+1}(y_n^{k+1}) &= \min_{b \in \mathbf{B}^k \cup \mathbf{B}^{k+1}} d_{geo}(y_n^{k+1}, b, \mathcal{G}'^k). \end{aligned} \tag{8}$$

The rationale behind (8) is that the saliency value of a superpixel is measured by its shortest path to background regions in color space considering both spatial and temporal information. We update P^k and P^{k+1} for frame F^k and F^{k+1} with S^k and S^{k+1}, and keep iterating this process for the following two adjacent frames F^{k+1} and F^{k+2} until the final frame.

3.4 Skeleton Abstraction

To further refine the saliency estimates above, we use a geodesic distance based abstraction scheme that augments core regions with higher saliency values. We decompose (Fig. 4-c) frame F^k into two parts: background regions $\mathbf{B}'^k$ and object-like regions $\mathbf{U}'^k$ using a threshold similar to the one in (6) yet computed by the saliency result S^k as

$$\begin{aligned} \sigma'^k &= \mu(S^k), \\ \mathbf{U}'^k &= \{y_n^k | S^k(y_n^k) > \sigma'^k\}, \\ \mathbf{B}'^k &= \mathbf{Y}^k - \mathbf{U}'^k. \end{aligned} \tag{9}$$

As the saliency result S^k is more accurate than P^k, we decompose frame F^k through an efficient thresholding strategy. The skeleton region abstraction is an iterative process based on the undirected weighted graph $\mathcal{G}^k$ defined in Section 3.2. The base skeleton region should have two properties. First, this region should be as far away from $\mathbf{B}'^k$ as possible; second, it should be close to foreground regions $\mathbf{U}'^k$. Based on this condition, the base skeleton region is selected by

$$\mathbf{O}^k \leftarrow \left\{ \operatorname*{argmin}_{o \in \mathbf{U}'^k} \frac{\max_{u' \in \mathbf{U}'^k} d_{geo}(o, u', \mathcal{G}^k)}{\min_{b' \in \mathbf{B}'^k} d_{geo}(o, b', \mathcal{G}^k)} \right\}. \tag{10}$$

After obtaining the base skeleton region (Fig. 4-d), we select the other skeleton regions. These regions are as far away from $\mathbf{B}'^k$ and previous skeleton regions as possible. This induces the skeleton regions to cover object regions that may have different

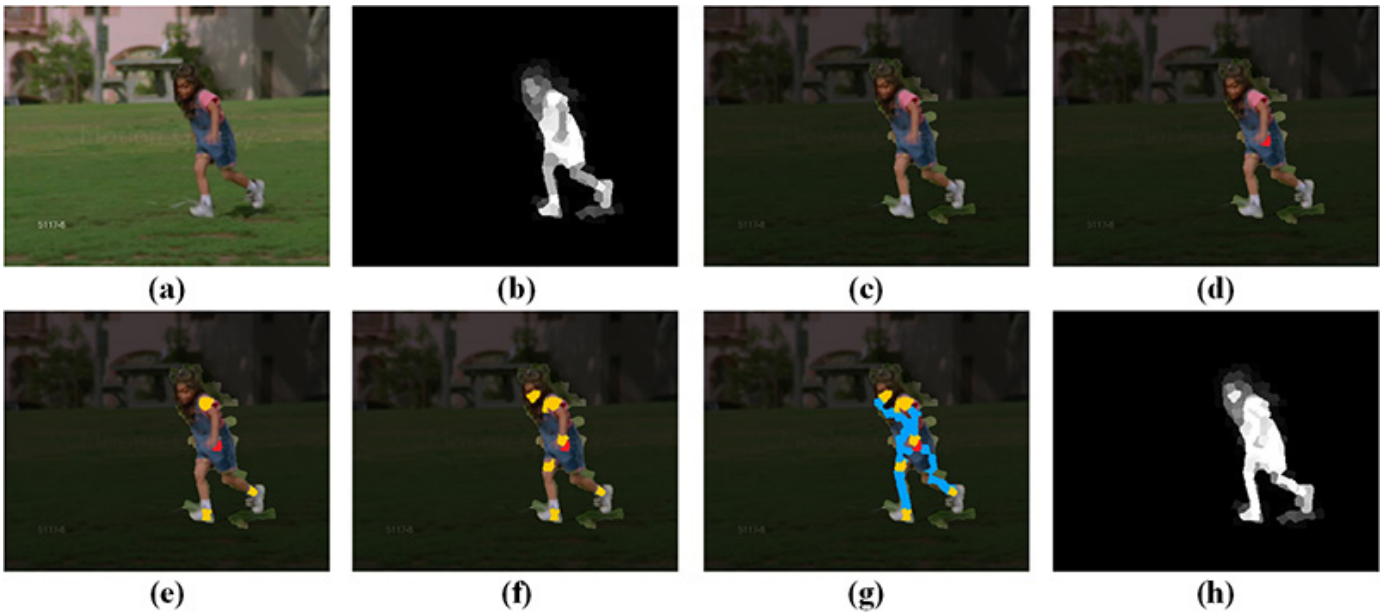

Fig. 4. Illustration of skeleton abstraction process. (a) Frame F^k. (b) Saliency results S^k of (a) obtained via (8). (c) F^k is decomposed into background regions $\mathbf{B}'^k$ (black area) and object-like regions $\mathbf{U}'^k$ (bright area) via (9). (d) The red region corresponds to the first selected skeleton region by (10). (e) The yellow regions correspond to the subsequently selected skeleton regions by (11). (f) We iteratively add skeleton regions until the number of selected skeleton regions reaches 10% of object-like regions $\mathbf{U}'^k$. (g) The blue regions are the other skeleton regions that lie on the shortest geodesic path between the base and the selected skeleton regions. (h) The enhanced saliency values of the skeleton regions.

appearances. Therefore, the skeleton regions are selected in a greedy fashion:

$$\mathbf{O}^k \leftarrow \mathbf{O}^k \cup \Big\{ \underset{o \in \mathbf{U}'^k}{\operatorname{argmax}} \big(\min_{o' \in \mathbf{O}^k} d_{geo}(o, o', \mathcal{G}^k) \cdot \min_{b' \in \mathbf{B}'^k} d_{geo}(o, b', \mathcal{G}^k) \big) \Big\}. \tag{11}$$

As shown in Fig. 4-e, each of the subsequent skeleton regions is selected to maximize its geodesic distance to background and previously selected skeleton regions. This process continues until a small percentage (10%) of the object-like regions $\mathbf{U}'^k$ are selected as skeleton. All object-like regions that lie on the shortest geodesic path between the base and subsequently chosen skeleton regions are also selected as skeleton regions. Finally, we increase the saliency values of the skeleton regions ($2\times$) in Fig. 4-h. A quantitative evaluation of the improvement of each step of our saliency scheme is presented in Section 5.4.

4 Pixel Labeling Energy Function

In the second stage of our segmentation method, we perform binary video segmentation based on the saliency results from Section 3. Global appearance models for foreground and background are established using our saliency prior. Dynamic location model for each frame is estimated from motion information extracted from subsequent frames. Finally, the spatiotemporal saliency maps, global appearance models and dynamic location model are combined into an energy function for binary segmentation.

We formulate the segmentation task as a pixel labeling problem. Each pixel x_i^k in F^k takes a label $l_i^k \in \{0, 1\}$, where 1 corresponds to foreground. A labeling $\mathbf{L} = \{l_i^k\}_{k,i}$ of pixels from all frames represents a partitioning of the entire video. Similar to other segmentation works [7], [57], we define an energy function for labeling $\mathbf{L}$ of all the pixels as

$$\begin{aligned} \mathcal{F}(\mathbf{L}) = &\sum_{k,i} \mathcal{U}^k(l_i^k) + \lambda_1 \sum_{k,i} \mathcal{A}^k(l_i^k) + \lambda_2 \sum_{k,i} \mathcal{L}^k(l_i^k) \\ &+ \lambda_3 \sum_{(i,j) \in \mathbf{N}_s} \mathcal{V}^k(l_i^k, l_j^k) + \lambda_4 \sum_{(i,j) \in \mathbf{N}_t} \mathcal{W}^k(l_i^k, l_j^{k+1}), \end{aligned} \tag{12}$$

where the spatial pixel neighborhood $\mathbf{N}_s$ consists of 8 neighboring pixels within the same frame, the temporal pixel neighborhood $\mathbf{N}_t$ consists of the forward-backward 9 neighbors in adjacent frames, and i, j are indices of pixels.

This energy function consists of three unary terms, $\mathcal{U}^k$, $\mathcal{A}^k$ and $\mathcal{L}^k$, and two pairwise terms $\mathcal{V}^k$ and $\mathcal{W}^k$, which depend on the labels of spatially and temporally neighboring pixels. The purpose of $\mathcal{U}^k$ is to evaluate how likely a pixel is foreground according to the spatiotemporal saliency maps computed in the previous step. The unary appearance term $\mathcal{A}^k$ encourages labeling pixels that have similar colors according to their global appearance models. The third unary term $\mathcal{L}^k$ is for labeling pixels according to the location priors estimated from the dynamic location models. The pairwise terms $\mathcal{V}^k$ and $\mathcal{W}^k$ encourage spatial and temporal smoothness, respectively. The scalar parameters λ weight the various terms, which can be set according to the characteristic of the video content. Having described the separate terms of the complete $\mathcal{F}$ below, we use graph-cuts [58] to compute the optimal binary labeling and obtain the final segmentation (Fig. 5-h).

Saliency term $\mathcal{U}^k$: The unary saliency term $\mathcal{U}^k$ is based on the saliency estimation results and penalizes assignments of pixels with low saliency to the foreground. The term $\mathcal{U}^k$ has the following form

$$\mathcal{U}^k(l_i^k) = \begin{cases} -\log(1 - S^k(x_i^k)) & \text{if } l_i^k = 0; \\ -\log(S^k(x_i^k)) & \text{if } l_i^k = 1. \end{cases} \tag{13}$$

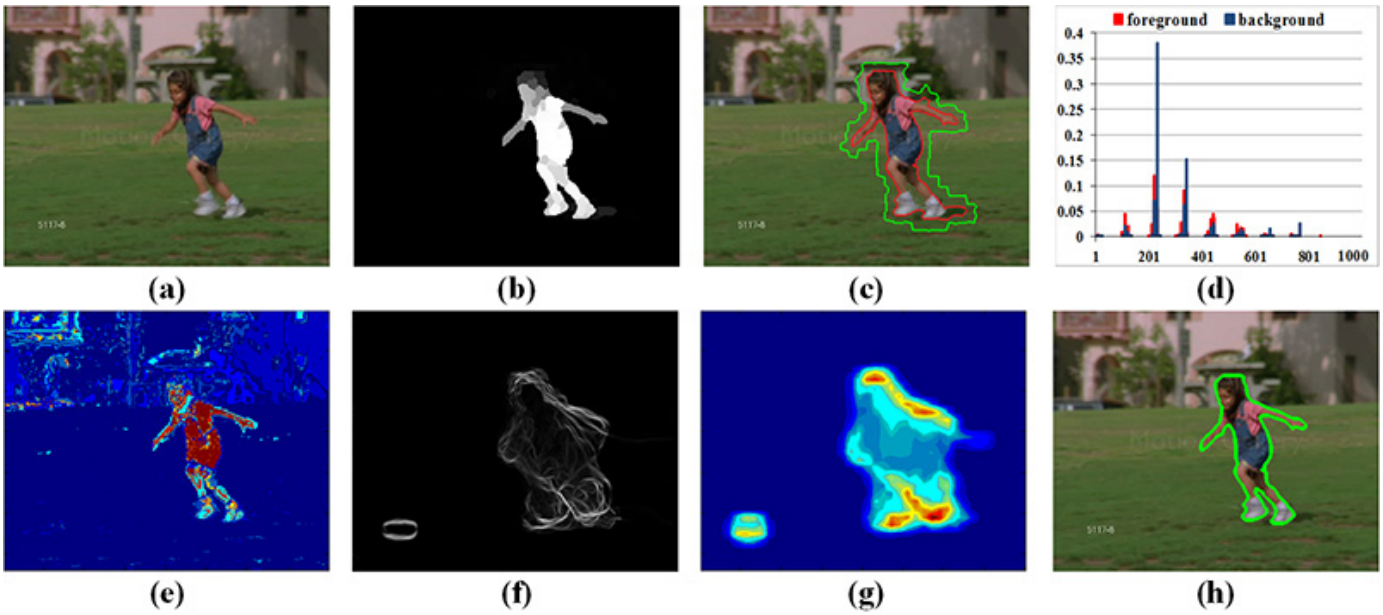

Fig. 5. Illustration of video segmentation. (a) Input frame F^k. (b) Video saliency map S^k. (c) The regions within the red boundaries have higher saliency values than adaptive threshold, which are used for establishing foreground histogram model. The regions between the green boundaries and red boundaries are for building background histogram model. (d) Global appearance models $\{H_f, H_b\}$ estimated from (c). (e) Foreground probability computed via appearance model in (d). (f) Accumulated optical flow gradient magnitude $\widehat{E}^k$ yields trajectory of the object within few subsequent frames. (g) Dynamic location prior L^k obtained via intra-frame graph described in Section 3.2. (h) Final segmentation results by (13), which consists of the saliency term (b), the appearance term (e), and the location term (g), and two pairwise terms.

Appearance term $\mathcal{A}^k$: To model the foreground and background appearance, two weighted color histograms H_f and H_b are computed in RGB color space. Each color channel is uniformly quantized into 10 bins, and there is a total of 10^3 bins in the joint histogram. Each pixel contributes into these histograms H_f and H_b according to its color values with weights $S^k(x)$ and $1 - S^k(x)$, respectively.

To construct the foreground (background) histogram, we only use pixels from the superpixels spatially connected to the former foreground (background) superpixels and have saliency values larger (smaller) than the adaptive threshold, which is defined as the mean value of spatiotemporal saliency map. This strategy makes better use of the information of spatiotemporal saliency results and minimizes adverse effects of background regions with similar color to the foreground contaminating the foreground histogram (Fig. 5-c,e). Finally, the histograms are normalized. Denoting $c(x_i^k)$ as the histogram bin index of RGB color value at pixel x_i^k, the unary appearance term $\mathcal{A}^k$ is defined as:

$$\mathcal{A}^k(l_i^k) = \begin{cases} -\log\left(\dfrac{H_b(c(x_i^k))}{H_f(c(x_i^k)) + H_b(c(x_i^k))}\right) & \text{if } l_i^k = 0; \\ -\log\left(\dfrac{H_f(c(x_i^k))}{H_f(c(x_i^k)) + H_b(c(x_i^k))}\right) & \text{if } l_i^k = 1. \end{cases} \tag{14}$$

Location term $\mathcal{L}^k$: For the cases of cluttered scenes and background regions having similar appearance models with the foreground, the object motion consistency provides a valuable prior to locate the areas likely to contain the object. Thus, we estimate the location of foreground object with respect to motion information from a small number of neighboring frames.

For k-th frame, we accumulate the optical flow gradient magnitudes within a temporal window of $\pm t$ frames to obtain relatively longer term motion information of the foreground regions as

$$\widehat{E}^k = \sum_{i=k-t}^{k+t} E_o^i = \sum_{i=k-t}^{k+t} \|\nabla V^i\|. \tag{15}$$

Having a larger temporal window provides some robustness to individual pixel-wise unreliable optical flow estimates. However, this may also cause $\widehat{E}^k$ loses its discriminative power since motion cue spans out on too many frames. In our experiments, we set $t = 5$. We use the intra-frame graph construction described in Section 3.1 to compute a dynamic location model for each frame (Fig. 5-f,g). Finally, we determine the location prior L_i^k for pixel x_i^k and the unary location term $\mathcal{L}^k$ as

$$\mathcal{L}^k(l_i^k) = \begin{cases} -\log(1 - L^k(x_i^k)) & \text{if } l_i^k = 0; \\ -\log(L^k(x_i^k)) & \text{if } l_i^k = 1. \end{cases} \tag{16}$$

Pairwise terms $\mathcal{V}^k$, $\mathcal{W}^k$: These terms impose label smoothness by constraining the segmentation labels to be both spatially and temporally consistent. They are contrast-modulated Potts potentials [7], [22], [57], which favor assigning the same label to neighboring pixels that have similar color. The spatial consistency term $\mathcal{V}^k$ between spatially adjacent pixels x_i and x_j is defined as

$$\mathcal{V}^k(l_i^k, l_j^k) = \delta(l_i^k, l_j^k) \exp^{-\theta||C(x_i^k) - C(x_j^k)||^2}, \tag{17}$$

where $C(x_i^k)$ is the color vector of the pixel x_i^k and $\delta(\cdot)$ denotes the Dirac delta function, which is 0 when $l_i^k \neq l_j^k$. The constant ϕ is chosen [57] to be

$$\theta = (2 \sum_{(i,j) \in \mathbf{N}_s} ||C(x_i^k) - C(x_j^k)||^2)^{-1}, \tag{18}$$

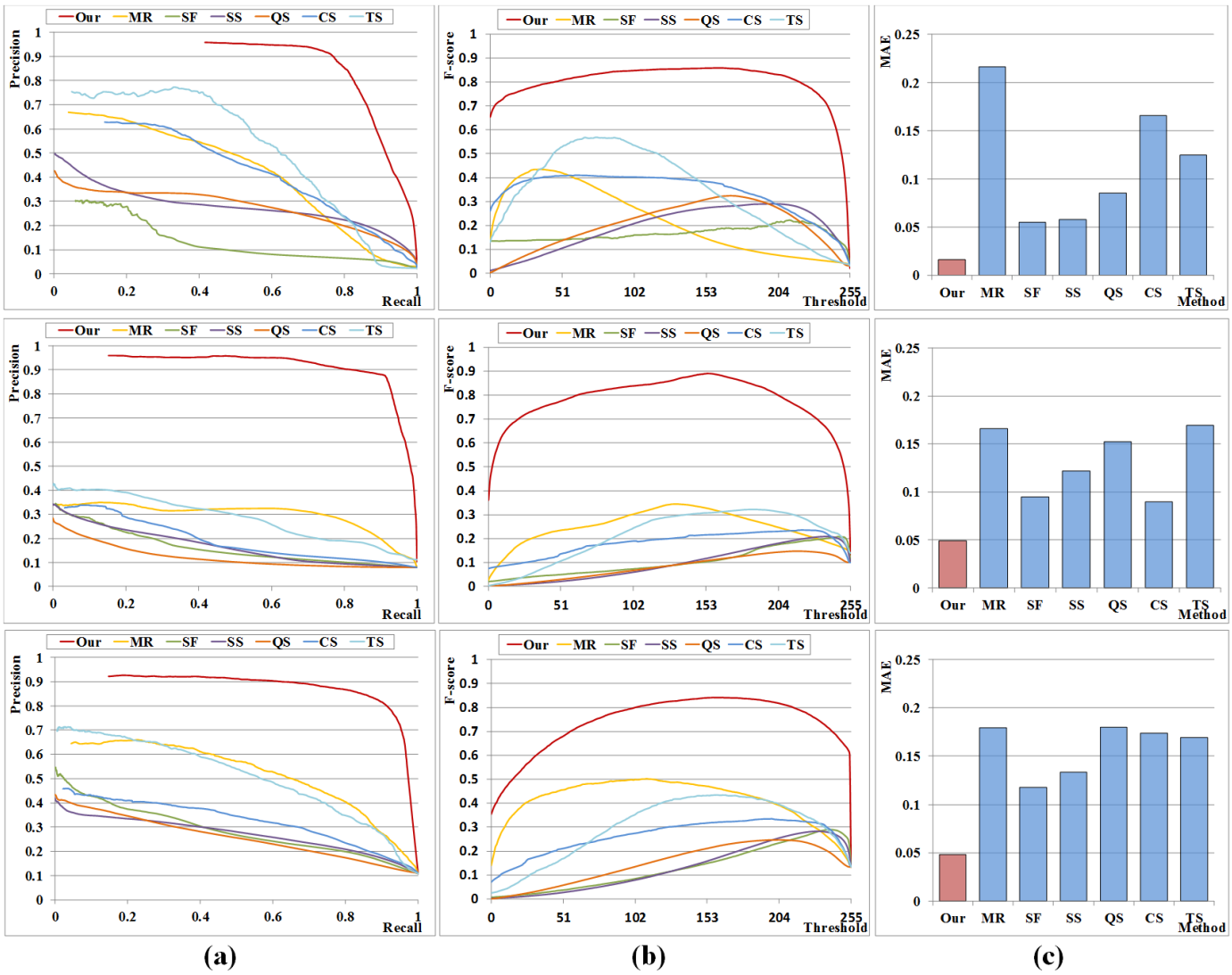

Fig. 6. Comparison of saliency detection methods using SegTrack [59] (**top**), extended SegTrack [60] (**middle**) and FBMS [1] (**bottom**) with pixel-level ground-truth: (a) average precision recall curve by segmenting saliency maps using fixed thresholds, (b) F-score, (c) average MAE.

To ensure the exponential term in (17) switches appropriately between high and low contrast. Similarly, the temporal consistency term $\mathcal{W}^k$ is defined as

$$\mathcal{W}^k(l_i^k, l_j^{k+1}) = \delta(l_i^k, l_j^{k+1}) \exp^{-\theta||C(x_i^k) - C(x_j^{k+1})||^2}. \quad (19)$$

5 Experimental Evaluations

We first evaluate the effectiveness of our spatiotemporal saliency estimation method by comparing against some state-of-the-art saliency methods in Section 5.1. After that, we compare both quantitatively and qualitatively our overall segmentation method with serveral well-known video segmentation approaches (in Section 5.2). Then we offer more detailed exploration and dissect various parts of our approach. In Section 5.3, we assess its computational load. In Section 5.4, we evaluate the effectiveness of each step of the proposed framework.

We performed experiments on four benchmark datasets: the SegTrack [59], the extended SegTrack [60], and Freiburg-Berkeley Motion Segmentation Dataset (FBMS) [1]. The SegTrack dataset contains 6 videos where full pixel-level segmentation ground-truth for each frame is available. We follow the common protocol [7], [8], [22] and use 5 video sequences (*Birdfall*, *Cheetah*, *Girl*, *Monkeydog* and *Parachute*) for evaluations. The extended SegTrack dataset consists of 8 additional sequences, which have complex backgrounds and varying object motion patterns. We select five sequences (*Bird of Paradise*, *Frog*, *Monkey*, *Soldier* and *Worm*), each of which contains a single dominant object. The widely used FBMS dataset, containing 59 video clips, covers various challenges such as large foreground and background appearance variation, significant shape deformation, and large camera motion.

5.1 Evaluation of Spatiotemporal Saliency

Since spatiotemporal saliency detection is an important step of our video segmentation approach, we assess its performance against the existing saliency methods. Using the original implementations obtained from the corresponding authors, we make comparisons between 6 alternative approaches including manifold ranking saliency model (MR) [45], saliency filter (SF) [40], self-resemblance based saliency (SS) [15], saliency via quaternion Fourier transform (QS) [38], cluster-based co-saliency (CS) [48], and space-time saliency for time-mapping (TS) [49]. The former

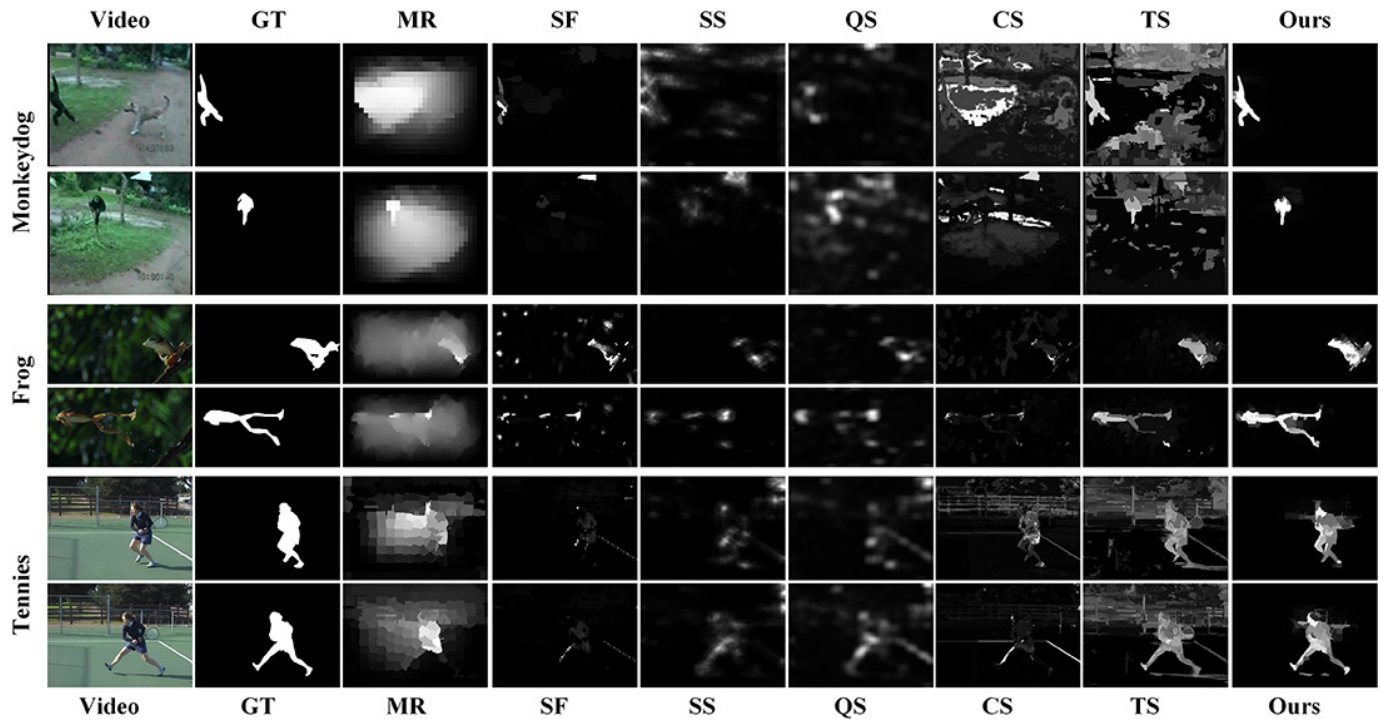

Fig. 7. Qualitative comparison against the state-of-the-art methods on the SegTrack benchmark [59], the extended SegTrack [60] and the famous FBMS dataset [1] with pixel-level ground-truth labels. Our saliency method yields continuous saliency maps that are most similar to the ground-truth.

TABLE 1
APFPER results on SegTrack dataset [59] compared to the ground-truth. Lower values are better. The best and the second best results are boldfaced and underlined, respectively.

	video	frames	unsupervised										supervised	
			Ours	[1]	[7]	[8]	[9]	[23]	[22]	[27]	[60]	[61]	[59]	[62]
SegTrack	Birdfall	30	**140**	217	288	468	155	606	189	144	199	468	252	454
	Cheetah	29	622	890	905	1175	633	11210	806	617	**599**	1968	1142	1217
	Girl	21	**991**	3859	1785	5683	1488	26409	1698	1195	1164	7595	1304	1755
	Monkeydog	71	350	**284**	521	1434	365	12662	472	354	322	1434	563	683
	Parachute	51	**195**	855	201	1595	220	40251	221	200	242	1113	235	502
	Avg.	-	**459**	1221	740	2071	572	18228	677	502	505	2516	699	922

two of these methods aim at image saliency while the latter four are designed for video saliency.

We report results on three widely used performance measures including precision-recall (PR) curve, F-score [39], and MAE (mean absolute errors). *Precision* is the fraction of the correctly labeled foreground pixels among the all pixels labeled as foreground by the algorithm, while *recall* is the fraction of correctly labeled foreground pixels among the ground-truth foreground pixels. We generate binary saliency maps from each method and plot the corresponding PR curves by varying the operating point threshold.

In general, a high recall response may come at the expense of reduced precision, and vice versa. Therefore, we also estimate F-score for evaluating precision and recall simultaneously. F-score evaluates precision and recall is defined as

$$\text{F-score} = \frac{(1+\beta^2) \times \text{precision} \times \text{recall}}{\beta^2 \times \text{precision} + \text{recall}}, \qquad (20)$$

where we set β^2 to 0.3 to assign a higher importance to precision as suggested in [39].

For a complete analysis, we follow [40] to evaluate the *mean absolute error* (MAE) between a real-valued saliency map $\mathbb{S}$ and a binary ground-truth $\mathbb{G}$ for all pixels as $\text{MAE} = |\mathbb{S} - \mathbb{G}|/N$, where N is the number of pixels. The MAE estimates the approximation degree between the saliency map and the ground-truth map, and it is normalized to [0, 1]. The MAE provides a better estimate of conformity between estimated and ground-truth maps.

The precision-recall curves of all methods are reported in Fig. 6-a. As shown, our method significantly outperforms the state-of-the-art. The minimum recall value in these curves can also be regarded as an indicator of robustness. A high precision score at the minimum recall value means a good separation between the foreground and background confidence values, as most of the high confidence saliency values (close to 1) are correctly estimated the foreground object. As can be seen, when the threshold is close to 255, the recall scores of other saliency models become very small, and the recall scores of SS [15] and QS [38] shrinks to 0. To our advantage, the minimum recall of the our method does not drop to 0. This demonstrates our saliency maps align better with the correct objects. In addition, our saliency method achieves the best precision rates above 0.9, which shows it is more precise to the actual salient information. Similar conclusions can be drawn from the F-score, as shown in Fig. 6-b. Our F-score is well above the performance of other methods. The MAE results are presented in

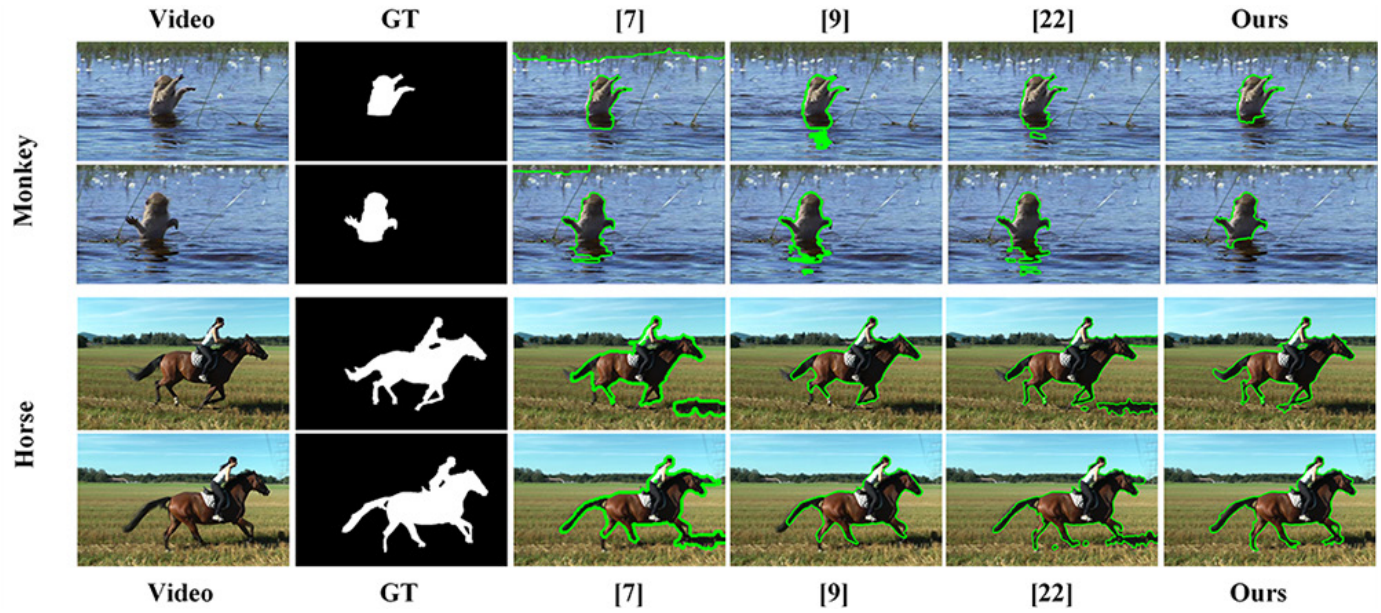

Fig. 8. Our segmentation results on extended SegTrack dataset [60] (*Monkey*), and FBMS [1] (*Horse*) with pixel-level ground-truth masks. The pixels within the green boundaries are segmented as foreground.

TABLE 2
IoU scores on SegTrack dataset [59] and extended SegTrack dataset [60] compared to the ground-truth. Higher values are better. The best and the second best results are boldfaced and underlined, respectively.

	video	frames	unsupervised							supervised			
			Ours	[7]	[9]	[22]	[29]	[27]	[25]	[63]	[64]	[65]	[66]
SegTrack	Birdfall	30	**74.5**	48.7	71.4	37.4	72.5	73.2	57.4	78.7	57.4	56.0	32.5
	Cheetah	29	**64.3**	43.4	58.8	40.9	61.2	64.2	24.4	66.1	33.8	46.1	33.1
	Girl	21	**88.7**	77.5	81.9	71.2	86.4	86.7	31.9	84.6	87.9	53.6	52.4
	Monkeydog	71	**78.0**	64.3	74.2	73.6	74.0	76.1	68.3	82.2	54.4	61.0	22.1
	Parachute	51	94.8	94.3	93.9	88.1	**95.9**	94.6	69.1	94.4	94.5	85.6	69.9
Extended SegTrack	Bird of Paradise	98	**94.5**	22.3	35.2	85.4	90.0	93.9	86.8	93.0	95.2	5.1	44.3
	Frog	279	**83.3**	71.0	76.3	69.4	80.2	81.5	67.1	56.3	81.4	14.5	45.2
	Monkey	31	**84.1**	38.6	61.4	69.6	83.1	63.9	61.9	86.0	88.6	73.1	61.7
	Soldier	32	**79.2**	10.0	51.4	47.4	76.3	36.8	66.5	81.1	86.4	70.7	43.0
	Worm	243	74.8	40.5	53.9	73.0	**82.4**	61.7	34.7	79.3	89.6	36.8	27.4
	Avg.	-	**81.6**	51.1	65.8	65.6	80.2	73.3	56.8	80.1	76.9	50.2	43.1

Fig. 6-c. Our saliency maps successfully reduce the MAE almost by 75% compared to the second best method (SF [40]).

Fig. 7 shows a qualitative comparison of different methods, where brighter pixels indicate higher saliency probabilities. It is observed that image saliency methods (MR [45], SF [40]) applied independently to each frame produce unstable outputs, some saliency maps even completely miss the foreground object, mainly because temporal coherence in video can convey important information for identifying salient objects. In contrast, video saliency methods (SS [15], QS [38], CS [48], and TS [49]) perform relatively better as they utilize motion information. However, video saliency maps from previous models are often generated in lower precision and tend to assign lower foreground probabilities to pixels inside the salient objects. This is due to the fact that optical flow estimations are unreliable.

Based on above, we draw two important conclusions: (1) motion information gives effective guidance for detecting foreground object; (2) making methods rely heavily on motion information is not the optimal choice. Comprehensive utilization of various features in spatial and temporal space (color, edges, motion, etc.) produces more satisfactory segmentation results. Our model can estimate more accurate saliency maps within and on the boundaries of the target objects in cluttered backgrounds. In addition, the assigned saliency values have higher confidence values, which also reflects in the quantitative analysis.

5.2 Evaluation of Pixel Labeling

Our framework produces both spatially and temporally coherent video object segmentation results in a fully unsupervised way. We use the average per-frame pixel error rate (APFPER) by [59] to evaluate the performance on the SegTrack dataset. This error rate measures the number of misclassified pixels and used in [8], [9], [22]. As discussed in [60], the intersection-over-union overlap (IoU) metric, which is the intersection over union of the estimated and ground-truth segmentation maps, is an informative indicator of the performance. This metric is also widely used for evaluating the segmentation performance. Therefore, we report our performance on the IoU metric for the SegTrack [59], extended SegTrack [60], and FBMS [1] by computing the score for each frame and then averaging it over all frames.

The APFPER results of ours and [1], [7], [8], [9], [23], [22], [27], [60], [61], [59], [62] on the SegTrack are shown in Table 1.

TABLE 3
IoU scores on a representative subset of the FBMS dataset [1], and the average computed over the 59 video sequences.

	video	Ours	[7]	[9]	[22]
FBMS	Bear2	70.1	**87.5**	21.0	86.8
	Cars5	38.5	10.7	**38.7**	17.4
	Cars9	**60.0**	19.5	28.9	52.4
	Cars10	55.9	65.7	74.9	**79.0**
	Cats1	**85.7**	19.8	81.5	83.1
	Dogs2	**91.7**	90.8	83.7	86.3
	Horses1	**89.4**	77.6	83.5	77.5
	Horses2	**92.7**	13.5	86.7	91.5
	People1	**68.1**	56.0	64.8	53.3
	People2	**68.3**	47.1	56.5	48.0
	People4	**86.4**	82.1	83.8	79.4
	People5	56.4	10.7	**84.4**	51.8
	Rabbits1	90.8	92.4	91.6	**92.9**
	Rabbits2	**71.0**	20.4	47.8	28.3
	Rabbits5	88.1	55.1	84.7	**90.1**
	Avg.	**63.3**	52.3	54.3	47.7

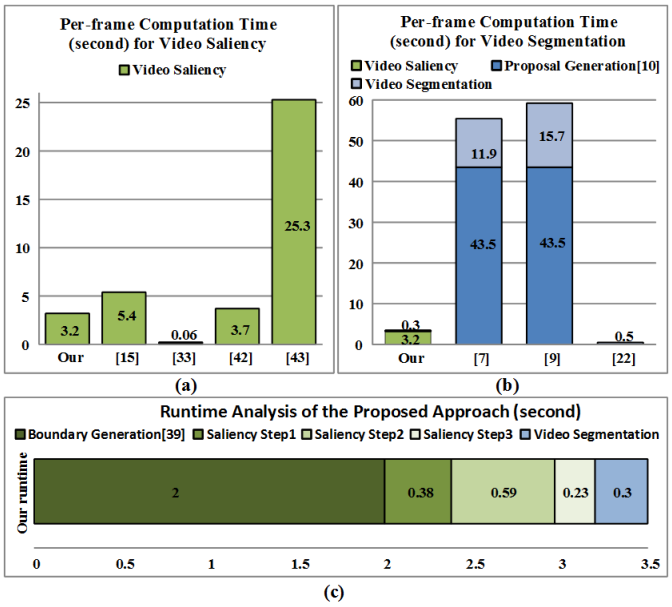

Fig. 9. Computational load of our method and the state-of-the-art for 320×240 video. (a) Execution time of video saliency estimation stage compared against other video saliency methods [15], [38], [48], [49]. (b) Execution time of overall method compared against other video segmentation methods [7], [9], [22]. (c) Execution time of each intermediate steps. *Step1* and *Step2* are saliency estimations via intra-frame graph and inter-frame graph, respectively. *Step3* is the final saliency step.

The segmentation methods in [1], [7], [8], [9], [23], [22], [27], [60], [61] and our method are unsupervised, while other methods in [59], [62] are supervised. As seen, our method outputs promising results on most video sequences, compared with existing top-performing unsupervised algorithms. Furthermore, our algorithm is better or on a par with the supervised approaches [59], [62], which indicates the robustness of the proposed approach.

Table 2 presents the IoU scores of our method and [7], [9], [22], [29], [27], [25], [63], [64], [65], [66] on the SegTrack and the extended SegTrack. Our approach outperforms the state-of-the-art methods and achieves the highest overall IoU score (81.6). The IoU scores for representative clips of the FBMS and the average performance over the entire dataset are demonstrated in Table 3. The proposed method achieves the best score on most videos and

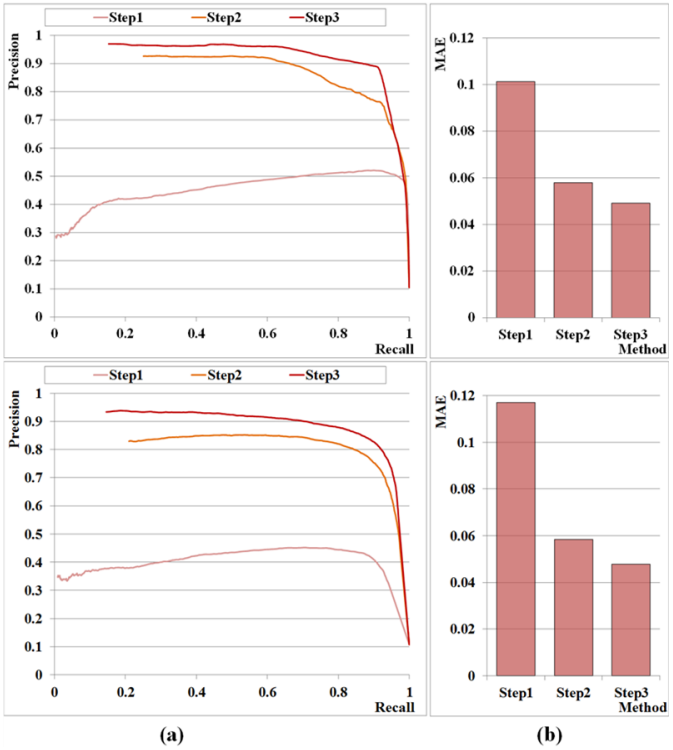

Fig. 10. Assessment of individual steps of our saliency estimation by (a) precision-recall curves, and (b) MAE scores. *Step1* and *Step2* refer to saliency via intra-frame and inter-frame graphs, respectively. *Step3* is the skeleton abstraction. **Top:** evaluation results on the extended SegTrack [60]. **Bottom:** evaluation results on the FBMS [1].

performs comparably or better than other methods. Representative pixel labeling results are shown in Fig. 8, and target foregrounds in various scenarios are segmented accurately by our algorithm. In contrast, existing methods [7], [9], [22] either mislabel background pixels as foreground or miss foreground pixels.

5.3 Computational Load

Our method is tested on a Dell T5610 workstation with an Intel Xeon E5 CPU of 2.50 GHz. We analyze the computational load of the steps in the proposed pipeline. We also include 4 video saliency methods [15], [38], [48], [49] and 3 video segmentation methods [7], [9], [22] to provide a comprehensive view of execution times of existing approaches.

The execution times are presented in Fig. 9 (excluding optical flow computations for all algorithms). Fig. 9-a shows the execution time comparisons of our and other saliency methods. Our saliency method is one of the fastest solutions and only slower than the frequency domain based method [38]. Fig. 9-b reports the per-frame processing times of the overall segmentation procedures. All solutions use the optical flow estimation method of [54]. Our method (3.5 seconds per frame) is much faster than [7], [9] but only slower than [22]. The object proposal based segmentation methods of [7], [9] require computationally expensive object proposal generation and inference stage [10] costing 43.5 seconds additional time per frame. Clearly, running time efficiency is the major bottleneck for the usability of those video segmentation algorithms, as a substantial amount of time is spent preprocessing frames to generate object proposals.

The execution time of each part of our whole scheme is shown in Fig. 9-c. The whole segmentation pipeline takes about 3.5

seconds for each frame, where over 60% of the runtime is spent on the edge generation [53]. Saliency detection takes a total of 1.2 seconds: 0.38 seconds for computing the saliency via intra-frame graph (*Step1*), 0.59 seconds for improving saliency results via inter-frame graph (*Step2*), and 0.23 seconds for generating final saliency via abstracting skeleton regions (*Step3*).

5.4 Validation of the Proposed Algorithm

To exhibit more details of our algorithm and objectively evaluate the contribution of different parts in the proposed saliency model to the saliency detection performance, we report the evaluation of each stage of our algorithm on the extended SegTrack [60] and the FBMS [1] datasets. We report the performance improvement of each step in Fig. 10. $Step1$ and $Step2$ refer to the initial saliency via the intra-frame graph (Section 3.2) and the refined saliency via the inter-frame graph (Section 3.3). $Step3$ corresponds to our final saliency results (Section 3.4). Compared to the PR curve for $Step1$, the performance of $Step2$ is elevated and $Step3$ achieves the best performance. This demonstrates the performance improvement of our saliency refinement via inter-frame graph and object skeleton abstraction scheme based saliency optimization. The MAE measure results show similar conclusions. Overall, the performance of each step improves progressively, which demonstrates that the combination of all steps effectively improves the overall performance.

6 Conclusion

We have presented an unsupervised approach that incorporates geodesic distance into saliency-aware video object segmentation. As opposed to the traditional video segmentation methods that heavily rely on cumbersome object inference and motion analysis, our method emphasizes the importance of video saliency to offer reliable cues for pixel labeling of foreground video objects.

The proposed method incorporates intra-graph edge and inter-graph motion boundary information into a spatiotemporal edge map. In intra-frame graph, the geodesic distance between the superpixel and frame boundary is exploited to estimate the foreground probability. In inter-frame graph, geodesic distance to the estimated background is utilized to update the spatiotemporal saliency map for each pair of adjacent frames. The geodesic distance is also employed to extract the foreground superpixels in the skeleton abstraction step to further enhance the saliency scores. In the pixel labeling stage, an energy function that combines global appearance models, dynamic location models and spatiotemporal saliency maps is defined and minimized via graph-cuts to obtain the final segmentation results. We have evaluated our methods on four benchmarks, namely SegTrack [59], extended SegTrack [60], and FBMS [1]. The extensive experimental evaluations show that our approach achieves higher performance scores than many other existing methods.

References

[1] T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories," in *European Conference on Computer Vision (ECCV)*, 2010.

[2] J. Lezama, K. Alahari, J. Sivic, and I. Laptev, "Track to the future: Spatio-temporal video segmentation with long-range motion cues," in *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[3] K. Fragkiadaki, G. Zhang, and J. Shi, "Video segmentation by tracing discontinuities in a trajectory embedding," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[4] W. Brendel and S. Todorovic, "Video object segmentation by tracking regions," in *Computer Vision (ICCV)*, 2009.

[5] E. Rahtu, J. Kannala, M. Salo, and J. Heikkilä, "Segmenting salient objects from images and videos," in *European Conference on Computer Vision (ECCV)*, 2010.

[6] C. Xu, C. Xiong, and J. J. Corso, "Streaming hierarchical video segmentation," in *European Conference on Computer Vision (ECCV)*, 2012.

[7] Y. J. Lee, J. Kim, and K. Grauman, "Key-segments for video object segmentation," in *Computer Vision (ICCV)*, 2011.

[8] T. Ma and L. J. Latecki, "Maximum weight cliques with mutex constraints for video object segmentation," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[9] D. Zhang, O. Javed, and M. Shah, "Video object segmentation through spatially accurate and temporally dense extraction of primary object regions," in *Computer Vision and Pattern Recognition (CVPR)*, 2013.

[10] I. Endres and D. Hoiem, "Category independent object proposals," in *European Conference on Computer Vision (ECCV)*, 2010.

[11] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?" in *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[12] J. Carreira and C. Sminchisescu, "Constrained parametric min-cuts for automatic object segmentation," in *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[13] D. Gao, M. Vijay, and V. Nuno, "The discriminant center-surround hypothesis for bottom-up saliency," *Advances in neural information processing systems*, pp. 497–504, 2008.

[14] V. Mahadevan and N. Vasconcelos, "Spatiotemporal saliency in dynamic scenes," *IEEE Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 171–177, Jan 2010.

[15] H. J. Seo and P. Milanfar, "Static and space-time visual saliency detection by self-resemblance," *Journal of vision*, vol. 9, no. 12, p. 15, 2009.

[16] X. Bai and G. Sapiro, "A geodesic framework for fast interactive image and video segmentation and matting," in *Computer Vision (ICCV)*, 2007.

[17] B. Price, B. Morse, and S. Cohen, "Geodesic graph cut for interactive image segmentation," in *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[18] C. Antonio, S. Toby, and B. Andrew, "Geos: geodesic image segmentation," in *European Conference on Computer Vision (ECCV)*, 2008.

[19] A. Criminisi, T. Sharp, C. Rother, and P. Perez, "Geodesic image and video editing," *ACM Transactions on Graphics*, vol. 29, no. 5, 2010.

[20] W. Wang, J. Shen, and F. Porikli, "Saliency-aware geodesic video object segmentation," in *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[21] P. Ochs and T. Brox, "Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions," in *Computer Vision (ICCV)*, 2011.

[22] A. Papazoglou and V. Ferrari, "Fast object segmentation in unconstrained video," in *Computer Vision (ICCV)*, 2013.

[23] P. Ochs and T. Brox, "Higher order motion models and spectral clustering," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[24] W. Wang, J. Shen, X. Li, and F. Porikli, "Robust video object co-segmentation," *IEEE Trans. on Image Processing*, vol. 24, no. 10, pp. 3137-3148, 2015.

[25] M. Grundmann, V. Kwatra, M. Han, and I. Essa, "Efficient hierarchical graph-based video segmentation," in *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[26] A. Faktor and M. Irani, "Video segmentation by non-local consensus voting." in *British Machine Vision Conference (BMVC)*, 2014.

[27] W.-D. Jang, C. Lee, and C.-S. Kim, "Primary object segmentation in videos via alternate convex optimization of foreground and background distributions," in *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[28] K. Fragkiadaki, P. Arbeláez, P. Felsen, and J. Malik, "Learning to segment moving objects in videos," in *Computer Vision (ICCV)*, 2015.

[29] F. Xiao and Y. J. Lee, "Track and segment: An iterative unsupervised approach for video object proposals," in *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[30] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. V. Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology

for video object segmentation," in *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[31] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[32] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," *Advances in neural information processing systems*, pp. 545–552, 2006.

[33] A. Borji, D. Sihite, and L. Itti, "Probabilistic learning of task-specific visual attention," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[34] D. Zhang, J. Han, J. Han, L. Shao, "Co-saliency Detection Based on Intrasaliency Prior Transfer and Deep Intersaliency Mining," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 27, no. 6, pp. 1163-1176, 2016.

[35] D. Zhang, J. Han, C. Li, J. Wang, and X. Li, "Detection of Co-salient Objects by Looking Deep and Wide," *International Journal of Computer Vision*, vol. 120, pp. 215-232, 2016.

[36] W. Wang, J. Shen, L. Shao, and F. Porikli, "Correspondence driven saliency transfer," *IEEE Trans. on Image Processing*, vol. 25, no. 11, pp. 5025-5034, 2016.

[37] W. Wang and J. Shen, Y. Yu, and K-L. Ma, "Stereoscopic thumbnail creation via efficient stereo saliency detection," *IEEE Trans. on Visualization and Computer Graphics*, in press, doi://10.1109/TVCG.2016.2600594, 2016.

[38] C. Guo, Q. Ma, and L. Zhang, "Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform," in *Computer Vision and Pattern Recognition (CVPR)*, 2008.

[39] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Computer Vision and Pattern Recognition (CVPR)*, 2009.

[40] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[41] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," in *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[42] D. Klein and S. Frintrop, "Center-surround divergence of feature statistics for salient object detection," in *Computer Vision (ICCV)*, 2011.

[43] M. Cheng, J. Warrell, and W. Lin, "Efficient salient region detection with soft image abstraction," in *Computer Vision (ICCV)*, 2013.

[44] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic saliency using background priors," in *European Conference on Computer Vision (ECCV)*, 2012.

[45] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Computer Vision and Pattern Recognition (CVPR)*, 2013.

[46] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Computer Vision and Pattern Recognition (CVPR)*, 2014.

[47] D. Gao and N. Vasconcelos, "Bottom-up saliency is a discriminant process," in *Computer Vision and Pattern Recognition (CVPR)*, 2007.

[48] H. Fu, X. Cao, and Z. Tu, "Cluster-based co-saliency detection," *IEEE Transactions on Image Processing*, vol. 22, no. 10, pp. 3766–3778, Oct 2013.

[49] F. Zhou, S. B. Kang, and M. F. Cohen, "Time-mapping using space-time saliency," in *Computer Vision and Pattern Recognition (CVPR)*, 2014.

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

[51] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognitive psychology*, vol. 12, no. 1, pp. 97–136, 1980.

[52] P. Mital, T. J. Smith, S. Luke, and J. Henderson, "Do low-level visual features have a causal influence on gaze during dynamic scene viewing?" *Journal of Vision*, vol. 13, no. 9, pp. 144–144, 2013.

[53] M. Leordeanu, R. Sukthankar, and C. Sminchisescu, "Efficient closed-form solution to generalized boundary detection," in *European Conference on Computer Vision (ECCV)*, 2012.

[54] T. Brox and J. Malik, "Large displacement optical flow: Descriptor matching in variational motion estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 500–513, March 2011.

[55] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[56] D. B. Johnson, "Efficient algorithms for shortest paths in sparse networks," *J. ACM*, vol. 24, no. 1, pp. 1–13, Jan. 1977.

[57] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, 2004.

[58] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1222–1239, 2001.

[59] D. Tsai, M. Flagg, A. Nakazawa, and J. M. Rehg, "Motion coherent tracking using multi-label MRF optimization," *International Journal of Computer Vision*, vol. 100, no. 2, pp. 190–202, 2012.

[60] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg, "Video segmentation by tracking many figure-ground segments," in *Computer Vision (ICCV)*, 2013.

[61] O. Barnich and M. Van Droogenbroeck, "Vibe: A universal background subtraction algorithm for video sequences," *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1709–1724, June 2011.

[62] P. Chockalingam, N. Pradeep, and S. Birchfield, "Adaptive fragments-based tracking of non-rigid objects using level sets," in *Computer Vision (ICCV)*, 2009.

[63] L. Wen, D. Du, Z. Lei, S. Z. Li, and M.-H. Yang, "Jots: Joint online tracking and segmentation," in *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[64] Y.-H. Tsai, M.-H. Yang, and M. J. Black, "Video segmentation via object flow," in *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[65] M. Godec, P. M. Roth, and H. Bischof, "Hough-based tracking of non-rigid objects," in *Computer Vision (ICCV)*, 2011.

[66] S. Wang, H. Lu, F. Yang, and M.-H. Yang, "Superpixel tracking," in *Computer Vision (ICCV)*, 2011.

Wenguan Wang received the B.S. degree in computer science and technology from the Beijing Institute of Technology in 2013. He is currently working toward the Ph.D. degree in the School of Computer Science, Beijing Institute of Technology, Beijing, China. His current research interests include salient object detection and object segmentation for image and video.

Jianbing Shen (M'11-SM'12) is a Professor with the School of Computer Science, Beijing Institute of Technology, Beijing, China. He has published about 70 journal and conference papers such as *IEEE CVPR*, *IEEE ICCV*, and *IEEE Transactions*. He has also obtained many flagship honors including the Fok Ying Tung Education Foundation from Ministry of Education, the Program for Beijing Excellent Youth Talents from Beijing Municipal Education Commission, and the Program for New Century Excellent Talents from Ministry of Education. His research interests include computer vision and multimedia processing. He serves as an associate editor of *Neurocomputing*.

Ruigang Yang is currently a full professor of Computer Science at the University of Kentucky. His research interests span over computer vision and computer graphics. He has received a number of awards, including the US National Science Foundation Faculty Early Career Development (CAREER) Program Award in 2004 and the best Demonstration Award at CVPR 2007. He is currently an associate editor of the *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Fatih Porikli is an IEEE Fellow and a Professor with the Research School of Engineering, Australian National University, Canberra, ACT, Australia. He is also acting as the Computer Vision Group Leader at NICTA, Australia. He has contributed broadly to object and motion detection, tracking, and video analytics. Prof Porikli was the recipient of the R&D 100 Scientist of the Year Award in 2006. He has won 4 best paper awards at premier IEEE conferences including the Best Paper Runner-Up at IEEE CVPR in 2007. He serves as the Associate Editor of 5 premier journals including IEEE Signal Processing Magazine and SIAM Imaging Sciences. He served at the organizing committees of several flagship conferences including ICCV, ECCV, and CVPR.