Neural Processing Letters (2020) 51:657–674
https://doi.org/10.1007/s11063-019-10110-z

Unsupervised Video Object Segmentation Based on Mixture Models and Saliency Detection

Guofeng Lin[1] · Wentao Fan[1]

Published online: 28 August 2019

Abstract
In this paper, we propose an unsupervised video object segmentation approach which is mainly based on a saliency detection method and the Gaussian mixture model with Markov random field. In our approach, the saliency detection method is developed as a preprocessing technique to calculate the probability of each pixel as the target object. In contrast to traditional saliency detection methods which are normally difficult to obtain the object's precise boundary and are therefore hard to segment consistent objects, the developed saliency detection method can calculate the saliency of each frame in the video sequence and extract the position and region of the target object with more accurate object boundary. The refined extracted object region is then taken as the prior information and incorporated into the Gaussian mixture model with Markov random field to obtain the precise pixel-wise segmentation result of each frame. The effectiveness of the proposed unsupervised video object segmentation approach is validated through experimental results using both the SegTrack and the SegTrack v2 data sets.

Keywords Video object segmentation · Gaussian mixture model · Markov random field · saliency detection

1 Introduction

Video object segmentation is the process of automatically segmenting the object of interest from the entire video sequence, which is a critical step in various computer vision applications, such as video surveillance, behavioral understanding, activity recognition, video summarization and video retrieval. Broadly speaking, video object segmentation approaches can be divided into different categories based on the styles of segmentation: manual segmentation, semi-automatic segmentation, and fully automatic segmentation. The manual segmentation approaches mainly depend on the operator's experience, and are normally

✉ Wentao Fan
fwt@hqu.edu.cn

Guofeng Lin
17014083009@hqu.edu.cn

[1] Department of Computer Science and Technology, Huaqiao University, Xiamen, China

time-consuming and laborious. On the other hand, semi-automatic video object segmentation approaches [2,32,45] need to annotate the target object in the key frame for initialization, and then use the motion and appearance constrained optimization techniques to propagate annotations throughout the video. Although these semi-automatic approaches can normally provide promising segmentation results, most computer vision applications have to process a large amount of video data, and the cost of manually annotating video frames is particularly expensive. To tackle this problem, various fully automated methods for video object segmentation have emerged. For instance, several fully automated segmentation methods process each frame of given videos by adopting the appearance and motion constraints to make the bottom-up segmentation [3,20,39].

A variety of other automatic segmentation methods have also been proposed, such as graph-based methods [12,35], segmentation through clustering [5], binary partition tree [22] and so on. In recent years, deep learning techniques have shown promising performance in many applications, such as place recognition [44], dimension reduction [48], image ranking [43], human pose recovery [18], etc. There are many video object segmentation approaches based on deep learning have also been proposed. For instance, in [31], a method has been developed for object segmentation in videos by using a convolutional neural network trained with static images only. In [4], Caelles et al. have proposed the one-Shot video object segmentation approach, based on a fully-convolutional neural network architecture. This approach is able to successively transfer generic semantic information to the task of foreground segmentation, and finally to learn the appearance of a single annotated object of the test sequence. In [29], an efficient video object segmentation approach has been presented based on a deep Siamese encoder-decoder network that is designed to take the advantage of mask propagation and object detection while avoiding the weaknesses of both approaches. Although segmentation based on deep learning methods have demonstrated good results, the involved models often contain lots of parameters that require designated training steps and a large amount of time to learn.

During the last decade, various video object segmentation approaches based on saliency detection models have been developed [13,23,25,33,34,40,41,47] in which an explicit concept of how a foreground object looks like is formulated in the given video sequence. The basic idea of this kind of approach is that normally we are interested only in some particular regions of a given video. These regions correspond to noticeable objects that most attract users' interest and best represent the content of the video. Thus, in these video object segmentation approaches, saliency detection models are used to locate object-like regions from each frame. The problem of video object segmentation has then been recast as the problem of object region selection. The *objectiveness* (i.e. the possibility of each pixel in the frame to be an object of interest) is measured based on both motion and appearance based cues. The region of interest in a video frame is considered to be an object of interest if it has a high similarity across the frames with high objectiveness. Consequently, the obtained object of interest would provide a reliable priori information for the video object segmentation task. However, as indicated in [40], one major limitation of this kind of segmentation approach is that the dependency of the extracted potential regions of interest in adjacent frames is not taken into account which would downgrade the video segmentation performance. Moreover, it is very difficult to define a precise boundary between the object and the background in the saliency map.

Model-based segmentation algorithms have also been proposed in the past few decades and drawn considerable attention continuously. One of the most representative model-based segmentation methods is based on mixture models [26]. Various mixture models have been applied to different problems, ranging from visual scenes categorization [8], video background subtraction [9], gene expression clustering [11], to image segmentation [10,19]. The main advantage of mixture model-based approaches is that they can incorporate the prior knowledge to model unknown uncertainties in a probabilistic manner. Although conventional mixture model (such as Gaussian mixture model) is efficient for segmentation, it does not take the spatial information between neighbouring pixels into account, which results in that the segmentation performance is quite sensitive to noise. In recent years, mixture models based on Markov random field (MRF) have received great attention in image segmentation [6,15,28]. In these approaches, in order to reduce the segmentation sensitivity to noise, the prior distribution of the calculated pixels is related to the corresponding parameters of its neighboring pixels.

Inspired from aforementioned works, we propose an unsupervised video object segmentation approach that is based on saliency detection and mixture models. In our approach, we first obtain saliency maps for input frames by extracting the spatial static edges in the same frame and the estimated motion boundary edges between adjacent frames. Next, potential regions of interest are generated according to the self-adaptive method and the object of interest is located. Then, the information of the obtained object of interest is used as the prior and is incorporated into the Gaussian mixture model with MRF to acquire accurate pixel-wise segmentation results. The contributions of our work can be summarized as follows. First, an unsupervised video object segmentation approach is developed based on saliency detection, Gaussian mixture models and MRF. Second, in order to identify and extract the region of the target object (i.e. the object of interest) in a given video among several candidate regions, a method of identifying the region of the object of interest is proposed. Third, the effectiveness of the proposed unsupervised video object segmentation approach is validated through experimental results using both the SegTrack [36] and SegTrack v2 [21] data sets .

The rest of this work is organized as follows. In Sect. 2, we provide details of our unsupervised video object detection approach. In Sect. 3, experiments conducted on the SegTrack and SegTrack v2 data sets are used to evaluate the proposed segmentation approach. Finally, conclusion is presented in Sect. 4.

2 The Proposed Approach

2.1 The Framework

The framework of our method can be mainly divided into three steps. First, a spatiotemporal saliency map of the input frame is obtained by extracting the spatial static edges in the same frame and the estimated motion boundary edges between adjacent frames. Second, potential regions of interest are generated according to the self-adaptive method and the object of interest is located. Third, the information of the obtained object of interest is used as the prior and is incorporated into the Gaussian mixture model with MRF to acquire accurate pixel-wise segmentation results. The overview of the proposed unsupervised video object segmentation method is shown in Fig. 1.

2.2 Saliency-Aware Segmentation

The method that we used to calculate the saliency map of the input frame is mainly based on [40]. The framework for obtaining the saliency map is shown in Fig. 2. First, input frames are partitioned into superpixels through the SLIC superpixel method [1]. Then, based on the discontinuity of color and motion, we compute the edge probability and the optical flow to

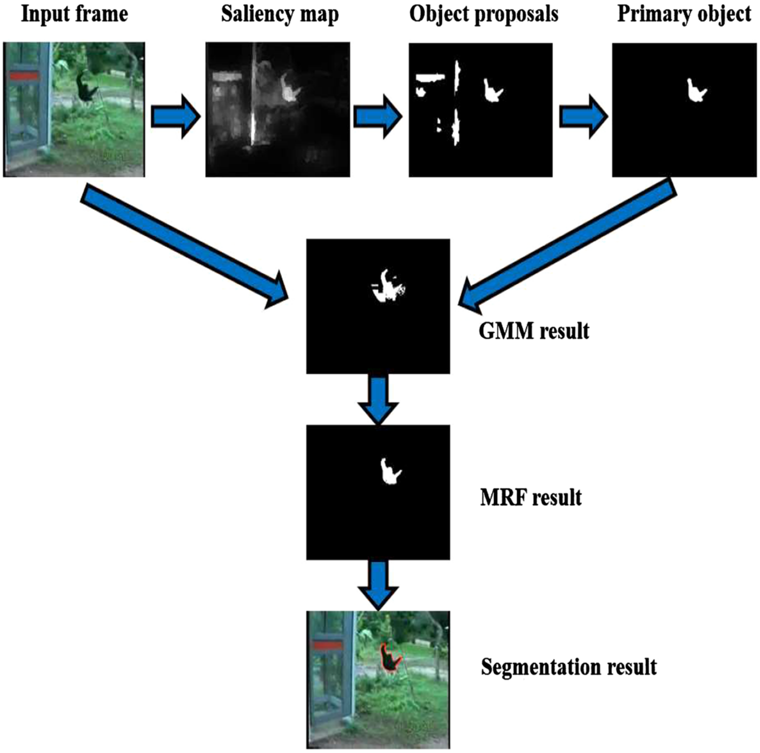

Fig. 1 Overview of the proposed unsupervised video object segmentation method

extract two types of edges: the spatial static edge in the same frame and the estimated motion boundary edges between adjacent frames. After that, we combine the two edge maps into a spatiotemporal edge probability map. Based on the probability map, the intra-frame graph and inter-frame graph are constructed to calculate the object probability of each super pixel, thereby obtaining the saliency of the current frame.

2.2.1 SLIC Superpixel Method

In our work, we adopt the SLIC method [1] to compute superpixels from each frame. Superpixel methods exploit the similarity between the features of the pixels to group these pixels, and use a small number of superpixels instead of a large number of pixels to express the image features. The motivation of using the SLIC superpixel method in our segmentation method is that, the superpixels formed by SLIC are more compact and can better maintain the original outline of the target object. Moreover, it has fewer parameters and faster computational time than many existing superpixel methods. The main steps of the SLIC method are described as follows: (1) initialize the seed point (cluster center); (2) reselect the seed point in the neighborhood of the seed point: calculate the gradient value of all the pixels in the neighborhood, and move the seed point to the place with the smallest gradient in the neighborhood; (3) assign a class label to each pixel in the neighborhood around each seed point; (4) calculate the distance D between the pixel and the seed point, including the color distance and the spatial distance. The seed point corresponding to the minimum value of the

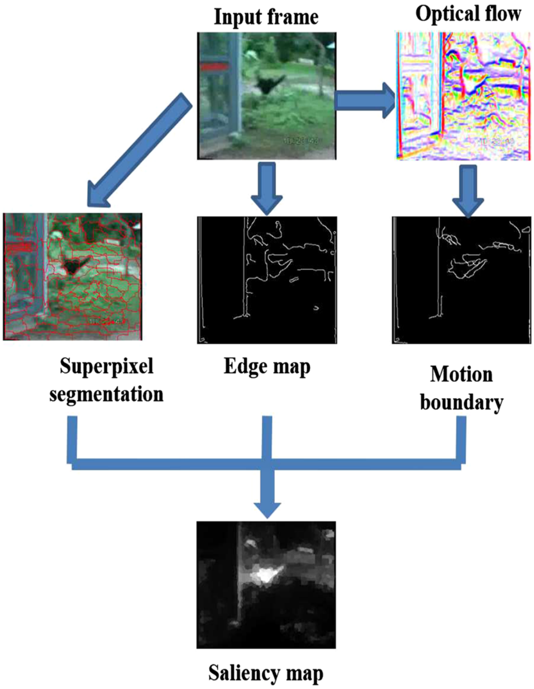

Fig. 2 The framework for obtaining the saliency map of a given input frame

distance is taken as the cluster center of the pixel; (5) the iterative optimization is performed until the cluster center of each pixel no longer changes. The distance in step 4 is calculated by

$$d_c = \sqrt{(r_j - r_i)^2 + (g_j - g_i)^2 + (b_j - b_i)^2} \tag{1}$$

$$d_s = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \tag{2}$$

$$D = \sqrt{\left(\frac{d_c}{Nc}\right)^2 + \left(\frac{d_s}{N_s}\right)^2} \tag{3}$$

where d_c represents the color distance between the pixel i and the seed point j in terms of RGB color information, d_s represents the spatial distance, N_s denotes the maximum spatial distance within the class, and N_c is the maximum color distance.

2.2.2 Intra-Frame Graph Construction

For the kth frame, an undirected weighted intra-frame graph G^k is constructed by considering superpixels within the kth frame as the nodes. The weight between two nodes in the kth frame is denoted as W^k_{mn}. In this framework, an intra-frame graph is constructed to represent the foreground probability map for locating foreground object and the geodesic distance [40] of the shortest path between two superpixels on the image is used to calculate the objectiveness of each superpixel. This is mainly based on the assumption that object region normally has a high spatiotemporal edge value or is surrounded by an area with a high spatiotemporal edge value. For each superpixel y_n^k in the kth frame, the probability that y_n^k belongs to the foreground object is calculated by

$$p^k(y_n^k) = \min_{t \in T^k} d_{geo}(y_n^k, t, G_k) \tag{4}$$

where T^k indicates the superpixels along the four boundaries of the kth frame. The geodesic distance between any two nodes (i.e. superpixels) v_1 and v_2 in graph G_k is defined by

$$d_{geo}(y_n^k, t, G_k) = \min_{C_{v_1,v_2}} \sum_{mn} W_{mn}^k, \quad mn \in C_{v_1,v_2} \tag{5}$$

where C_{v_1,v_2} denotes a path connecting the nodes v_1 and v_2. The weight W_{mn}^k is defined by

$$W_{mn}^k = \|E^k(y_m^k) - E^k(y_n^k)\| \tag{6}$$

where $E^k(y_m^k)$ and $E^k(y_n^k)$ denote the spatiotemporal boundary probability of superpixels y_m^k and y_n^k, respectively.

2.2.3 Inter-Frame Graph Construction

We construct an undirected weighted inter-frame graph G'^k for each pair of the kth frame and the $k+1$th frame by treating all the superpixels in these two frames as the nodes. Two kinds of edges are defined: all the spatially adjacent superpixels are connected by intra-frame edges whereas all the temporally adjacent superpixels are linked by inter-frame edges. The edge weights are specified as the Euclidean distance between the average colors in the CIE-Lab color space.

For the kth frame, a self-adaptive threshold σ^k for decomposing the kth frame into object-like regions and background regions is calculated through the average of the probability map p^k. Therefore, the object-like regions F^k and background regions B^k in the kth frame are defined as:

$$F^k = \{y_n^k | p^k(y_n^k) > \sigma^k\} \cup \{y_n^k | y_n^k \text{ is temporally connected to } F^{k-1}\} \tag{7}$$

and

$$B^k = Y^k - F^k \tag{8}$$

Then, in the inter-frame graph, the saliency of the kth frame is calculated by

$$S^k(y_n^k) = \min_{b \in B^k \cup B^{K+1}} d_{geo}(y_n^k, b, G'_k) \tag{9}$$

Finally, a saliency map is obtained by calculating the saliency of each superpixel.

2.3 Object Region Extraction

After applying the saliency detection method to calculate the saliency map for the input frame, the saliency map acquires the background and foreground labels of the frame according to the self-adaptive threshold. However, multiple object regions may be located in the extraction result because the camera moves following the object's movement, and another active object or dynamic background environment may also appear in the video. In order to identify and extract the region of the object of interest in the video among several candidate regions, a method of identifying the region of the object of interest is proposed in this section.

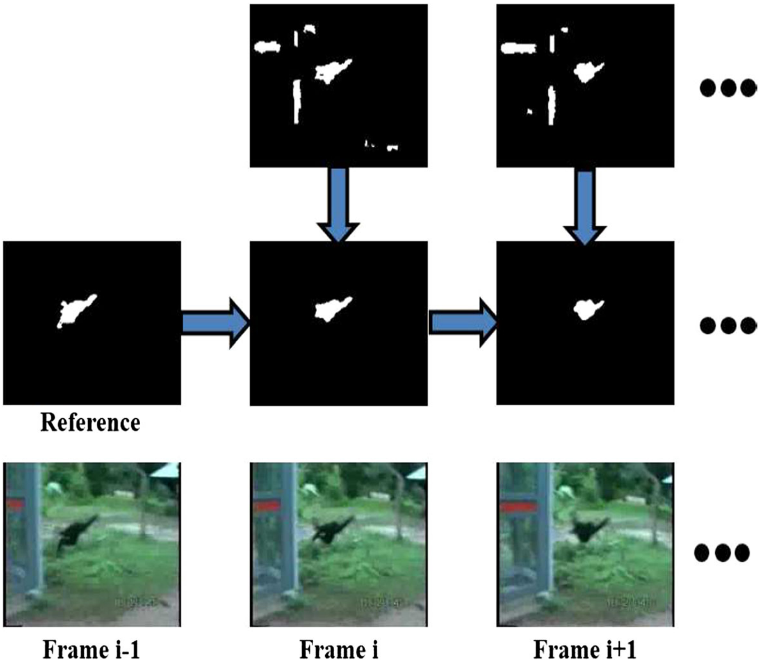

Fig. 3 The example of detecting object of interest in the ith frame based on the object of interest region that was found in the $i-1$th frame as the reference

The main assumption of our method for identifying the region of the object of interest in the saliency map is that, there is no significant movement of the camera for the first several frames (i.e., normally 5–10 frames) of the video, and thus only the primary active object will be detected. Therefore, we will treat the first frame that only contains the object of interest as the reference for the following frames in order to refine the saliency map to have only the target object. The object detected in the current frame may change in shape and position compared to the one detected from the previous frame, but the displacement of the centroid (i.e., the center position) of the object region in two consecutive frames is not significant due to the limited time period. Based on this idea, when multiple object regions are detected in one frame, we calculate the Euclidean distance from the centroid of each object detected in this frame to the centroid of the object of interest found in the previous frame (i.e., the reference frame), and then consider the object with the minimum value of the distance as the true object of interest. For example, as shown in Fig. 3, if J object regions $(r_1, \ldots, r_J)$ were found after applying the self-adaptive threshold for the ith frame, then we can use the object of interest region that was found in the $i-1$th frame as the reference to locate the object of interest region in the ith frame. Specifically, in order to locate the correct object of interest among all obtained objects that were found in the ith frame, we first compute the centroid c_j for each object region as

$$c_j^{(i)}(x_j, y_j) = \left(\frac{\sum_{n=1}^{N_j} x_{nj}}{N_j}, \frac{\sum_{n=1}^{N_j} y_{nj}}{N_j} \right) \tag{10}$$

where (x_j, y_j) represents the coordinates of the centroid of the jth object region, N_j denotes the number of pixels in the object region j, x_{nj} and y_{nj} indicate the position of the nth pixel in the jth object region. Next, we calculate the Euclidean distance between the centroid of object of interest c_p that we found in the previous frame with each centroid of the object region obtained in the ith frame as

$$c_j^{(i)}(x_j, y_j) - c_p^{(i-1)}(x_p, y_p) = \sqrt{(x_j - x_p)^2 + (y_j - y_p)^2} \tag{11}$$

where (x_p, y_p) denotes the position of the object of interest region of the reference frame. Then, the object of interest region p^i in the ith frame corresponds to the one with the smallest distance among all J regions

$$p^{(i)} = \arg\min_j \left[c_j^{(i)}(x_j, y_j) - c_p^{(i-1)}(x_p, y_p)\right] \tag{12}$$

The object of interest in the $i + 1$th frame is obtained using the same fashion.

2.4 Segmentation Via Gaussian Mixture Model

In this step, Gaussian mixture model is adopted to perform object segmentation for input frames by taking the detected object of interest in the previous step as the prior information. We use the Gaussian mixture model to model the feature vectors of the object region and the background region in each frame and divide them into two classes of labels (foreground and background). These feature vectors represent the information of the video (such as pixel value, location coordinate, etc.) where labels and pixels are independent of each other. In our method, feature vectors in the foreground and background are subjected to multiple Gaussian distributions, which are weighted and linearly combined together as a mixture model in order to model the foreground and background. Assuming that there are K different regions in background B and the vector in the jth region obeys the Gaussian distribution $\mathcal{N}(B|\mu_j, \sigma_j)$ with parameters μ_j (mean) and Σ_j (covariance matrix). Then, both the background region of the video and the foreground object region can be expressed by mixtures of K Gaussian distributions. Therefore, in order to have the probability of object region and background, it is necessary to infer the parameters of Gaussian mixture model. Specifically, the Gaussian mixture model representing the background B is given by

$$p(B) = \sum_{j=1}^{K} \pi_j \mathcal{N}(B|\mu_j, \Sigma_j) \tag{13}$$

where $\mathcal{N}(B|\mu_j, \Sigma_j)$ is the Gaussian distribution associated with the jth component of the mixture model. The parameters π_j in Eq. (13) are called mixing coefficients which must satisfy the following constraints

$$0 \leq \pi_j \leq 1 \quad \text{and} \quad \sum_{j=1}^{K} \pi_j = 1 \tag{14}$$

Then, the likelihood function of the Gaussian mixture model is given by

$$L(B) = \sum_{n=1}^{N} \log \sum_{j=1}^{K} \pi_j \mathcal{N}(B_n|\mu_j, \Sigma_j) \tag{15}$$

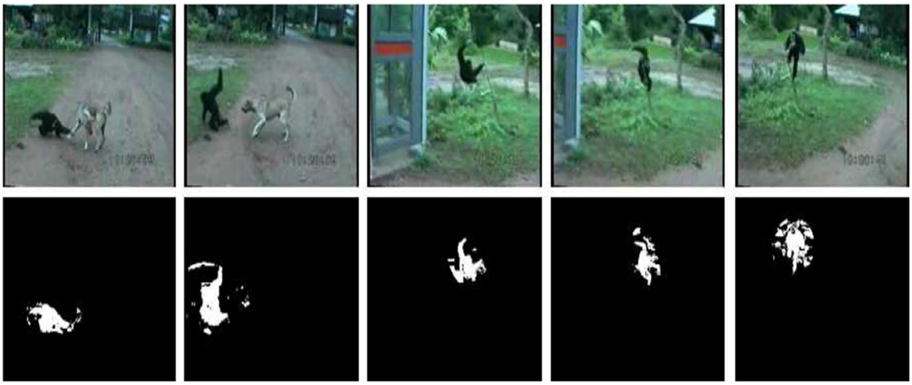

Fig. 4 Sample segmentation results obtained by the Gaussian mixture model for the “monkeydog” video of the SegTrack data set

where N denotes the total number of pixels in the background B. The parameters of the Gaussian mixture model are obtained by maximizing the likelihood function as

$$(\mu^*, \Sigma^*, \pi^*) = \arg\max_{\mu,\Sigma,\pi} L(B) \tag{16}$$

where the optimal parameters μ^*, Σ^*, π^* can be obtained by using the expectation maximization (EM) algorithm [27]. Then, based on the prior information obtained from the previous step and the Gaussian mixture models defined for the background and the foreground, the foreground and background probabilities based on color and location information are calculated respectively for the pixels of the original input frames. When the difference between the probabilities of foreground and background is greater than 0, the label of the pixel is set to foreground, otherwise it denotes background.

2.5 Markov Random Field

Although the Gaussian mixture model is an effective approach for segmentation, its segmentation results may contain noise, for instance as shown in Fig. 4. This is due to the fact that the image segmentation based on the Gauss mixture model considers pixels separately, and does not take the spatial relationship between nearby pixels into account. In order to tackle this problem, we adopt Markov random field (MRF) [14] to redefine the segmentation results. MRF considers that spatial information between pixels can distinguish different texture distributions, and effectively solves the problem of noise.

In our case, we apply the pairwise potential MRF to redefine the segmentation results to improve the segmentation accuracy [16]. The frame is represented by an array $X = (x_1, x_2, \ldots, x_N)$, x_n represents the pixel value at pixel n, and the image segmentation result is represented by an array $Y = (y_1, y_2, \ldots, y_N)$, where $y_n \in (0, 1)$ such that 0 represents the background and 1 denotes the foreground. In MRF, the image segmentation problem is summarized as the optimization problem of the MRF-Gibbs energy function.

The unary potential is calculated by the Gaussian mixture model, which indicates whether the pixel belongs to the category (background or foreground) described by the model. The calculation of potential energy is performed using an isotropic second-order neighborhood system (eight neighborhoods), indicating the consistency of the type of two pixel points. The unary potential and pairwise potential are defined by

$$E_{unary}(X, Y) = \sum_{n=1}^{N} U_n(y_n|x_n) \tag{17}$$

and

$$E_{pairwise}(X, Y) = \sum_{n=1}^{N} \sum_{j \in \varepsilon_n} V_{nj}(y_n, y_j|x_n, x_j) \tag{18}$$

where $U_n(y_n|x_n)$ is the negative logarithm of the probability of y_n. ε indicates the neighborhoods of the system, V_{nj} is obtained by calculating the difference between the pixel n and the jth neighboring pixel. The closer the two pixels are, the larger the potential energy is.

3 Experimental Results

In this work, we propose an unsupervised method that can automatically detect and extract the moving objects in video sequences. The goal of this section is to validate the proposed method by conducting experiments on the SegTrack data set [36] and the SegTrack v2 data set [21]. We also compare our segmentation results with several other video segmentation methods to demonstrate its advantages. All experiments were conducted using Matlab and tested using a PC with Windows platform (Core i7, running at 2.78 GHz with 32 GB of RAM).

3.1 Experiments on the SegTrack Data Set

We first conducted our experiments on the SegTrack data set, which is a popular video segmentation data set with full pixel-level annotations on multiple objects at each frame within each video. Originally, the SegTrack data set contains six video sequences. Since one of the videos ("penguin") that does not have the ground-truth, it is not considered in our experiments. Therefore, five video sequences in total with different characteristics from the SegTrack data set were used in our experiments including "birdfall" (small object), "cheetah" (object with fast motion patterns), "girl" (object with large shape deformation), "monkeydog" (object with large camera motion) and "parachute" (object with color overlap). We applied a similar setting as in [46], such that videos were divided into two categories: videos with static camera and with dynamic camera, respectively. For the video with static camera (the "birdfall" video), we used background subtraction to extract the moving target region in the video. For videos with the dynamic camera, we used the saliency segmentation and object of interest extraction method as we introduced in Sects. 2.2 and 2.3 to extract the region of the object of interest. After the region of the object of interest was detected, the extracted object region was optimized and redefined by using the Gaussian mixture model and MRF. In our case, the Gaussian mixture model was initialized with the k-means algorithm with the number of centers set to 10.

The qualitative results for the proposed unsupervised video segmentation method are shown in Figs. 5, 6, 7, 8 and 9 for the SegTrack data set. Fig. 10 shows the segmented results with the red boundaries in original frames. Based on these results, we can observe that the proposed video segmentation method is able to successfully extract primary moving object

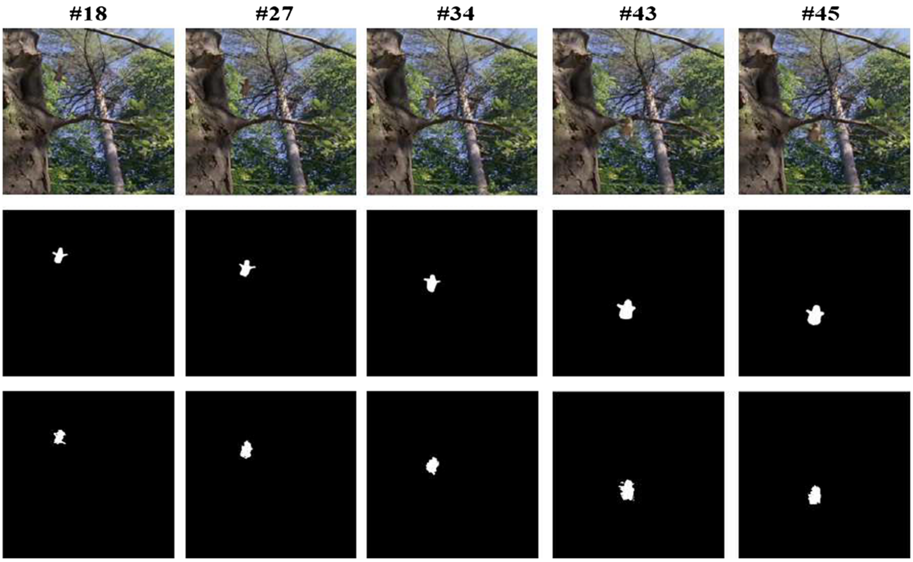

Fig. 5 The result of the segmentation for the “birdfall” video. The first row illustrates the original input frames, the second row shows the ground-truth, and the third row demonstrates the segmentation results using the proposed method

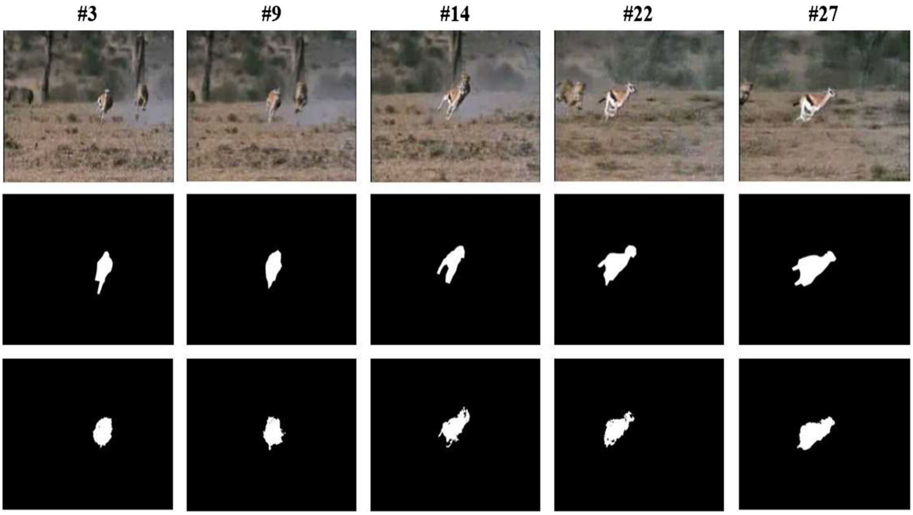

Fig. 6 The result of the segmentation for the “cheetah” video. The first row illustrates the original input frames, the second row shows the ground-truth, and the third row demonstrates the segmentation results using the proposed method

in given video sequences. Next, a visual comparison of the proposed method with several other video segmentation methods which include [13,40,46] and [42], is provided in Fig. 11, where higher saliency probabilities are denoted by brighter pixels. As shown in this figure, the proposed method has better performance than other tested methods in terms of more accurately estimated saliency maps at pixel level within and on the contour of the objects

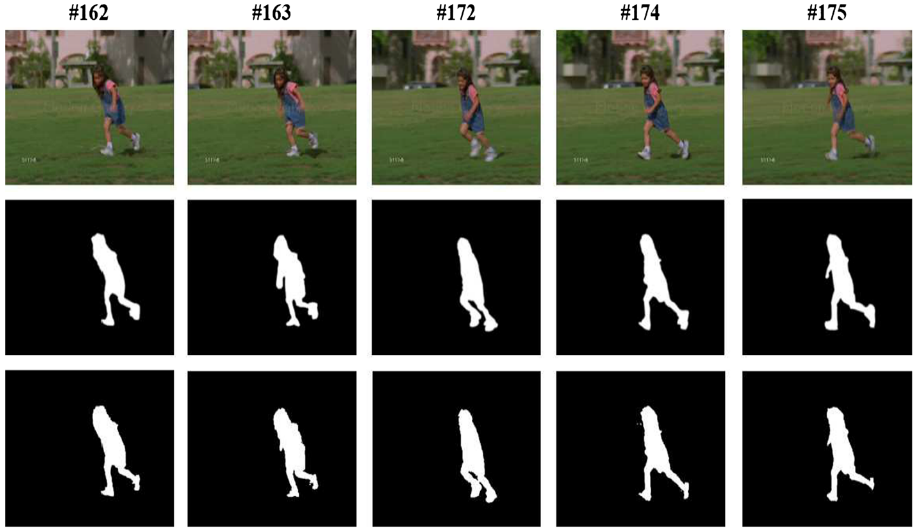

Fig. 7 The result of the segmentation for the "girl" video. The first row illustrates the original input frames, the second row shows the ground-truth, and the third row demonstrates the segmentation results using the proposed method

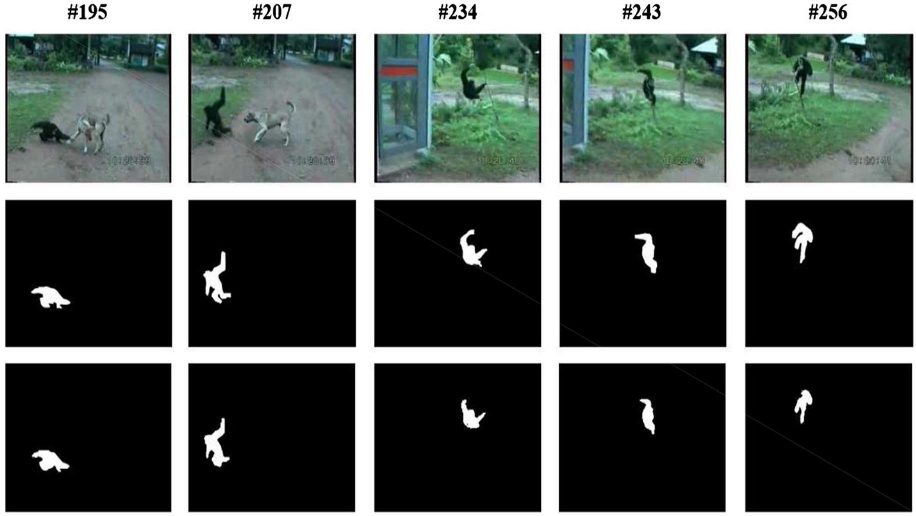

Fig. 8 The result of the segmentation for the "monkeydog" video. The first row illustrates the original input frames, the second row shows the ground-truth, and the third row demonstrates the segmentation results using the proposed method

in cluttered backgrounds. Another observation is that, the image saliency method [42] has obtained the worst performance among all tested methods, where the foreground objects cannot be precisely detected by saliency maps. This is due to the fact that the method of [42] does not take motion information into account and therefore results in degraded performance in locating object, especially when background and foreground have similar colors.

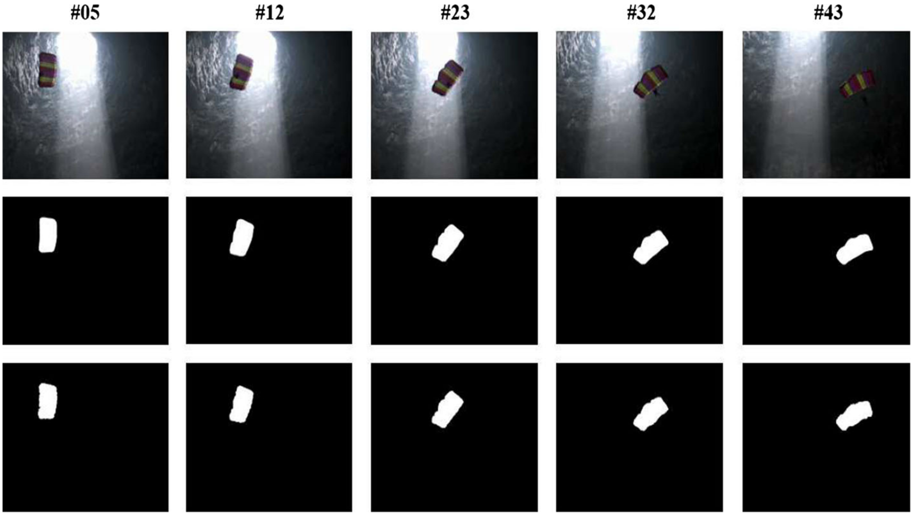

Fig. 9 The result of the segmentation for the "parachute" video. The first row illustrates the original input frames, the second row shows the ground-truth, and the third row demonstrates the segmentation results using the proposed method

Table 1 The average per-frame pixel error rate using the SegTrack data set by different segmentation methods

Video	Ours	[46]	[40]	[30]	[37]	[7]
Birdfall	**148**	155	209	189	252	454
Cheetah	**613**	633	796	806	1142	1217
Girl	1237	1488	**1040**	1698	1304	1755
Monkeydog	**345**	365	562	472	563	683
Parachute	**201**	220	207	221	235	502

Bold value indicates the best performance in terms of the lowest per-frame pixel error rate

Furthermore, in order to quantitatively compare with other experimental results, we utilized the average per-frame pixel error rate for evaluation, which is the number of pixels misclassified comparing to the ground-truth segmentation [46], and is defined by

$$Error = \frac{XOR(m, GT)}{M} \tag{19}$$

where m is the final result of each frame segmentation, and GT is the ground-truth segmentation result of the video, and M is the total number of frames in the video. The average per-frame pixel error rate can effectively estimate the approximation between segmentation results and the corresponding ground-truth. The smaller the error, the closer the segmentation result is to the ground truth.

We compare the proposed method quantitatively with other recent video segmentation methods including [7] and [30,37,40,46]. Among those tested methods, [30] and [40,46] are unsupervised, while [7] and [37] are supervised (i.e. an initial annotation is required for the first frame). In our experiments, for the tested segmentation methods, we adopted the same settings as in their original works. The comparison results are shown in Table 1. According to this table, for most cases, the proposed method provided better segmentation performance than the tested methods in terms of lower average per-frame pixel error rates. We

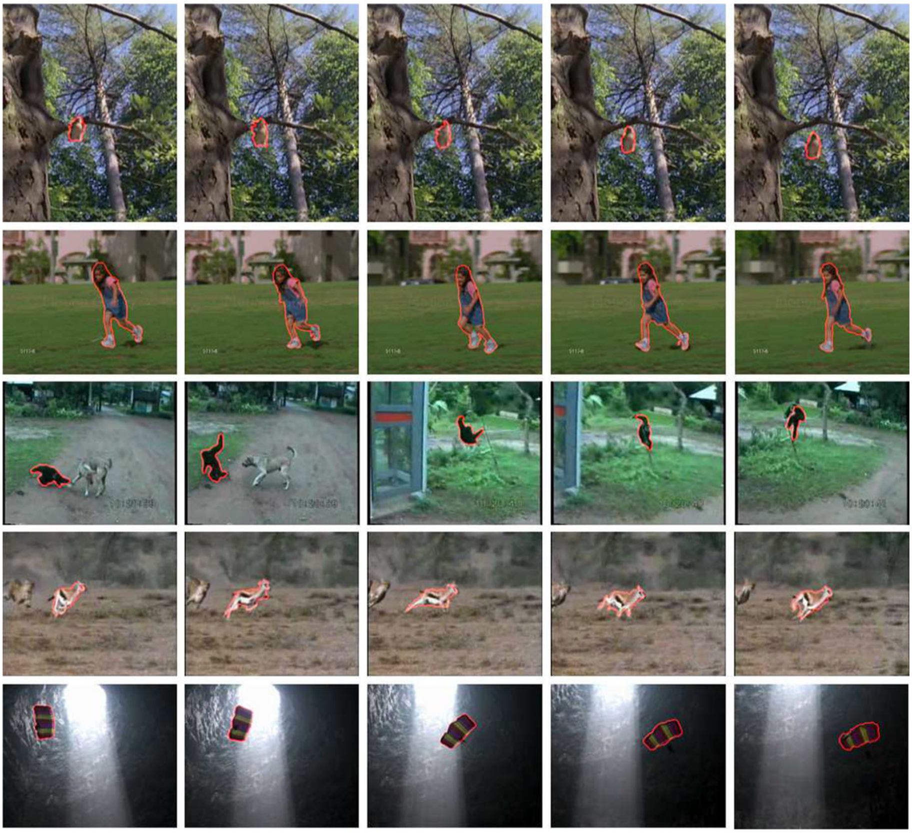

Fig. 10 The segmentation results obtained by the proposed method for the SegTrack data set. The region within the red boundary corresponds to the object of interest in the video. (Color figure online)

may notice that our method performed slightly worse than [40] for the "girl" sequence, this might be caused by the large shape deformation of the primary object (i.e., the running girl) and therefore degraded the performance of the object region extraction step in our method. However, as we can observe from Table 1, our method was able to obtain better performance than [40] for the other tested video sequences.

3.2 Experiments on the SegTrack v2 Data Set

To further demonstrate the effectiveness of the proposed unsupervised video object segmentation method, more experiments were conducted on the SegTrack v2 data set, which is an updated version of the SegTrack data set. In addition to the five video sequences in the SegTrack data set, eight new sequences are introduced in the SegTrack v2 data set including "frog", "worm", "soldier", "monkey", "bird of paradise", "drifting car", "hummingbird", and "BMX". Our method was compared with several state-of-the-art video object segmentation methods including [17] and [24,29,31,38]. We report the experimental results by different methods on the SegTrack v2 data set in terms of the computational time and another segmentation evaluation metric namely Intersection over Union (IoU) which is defined by

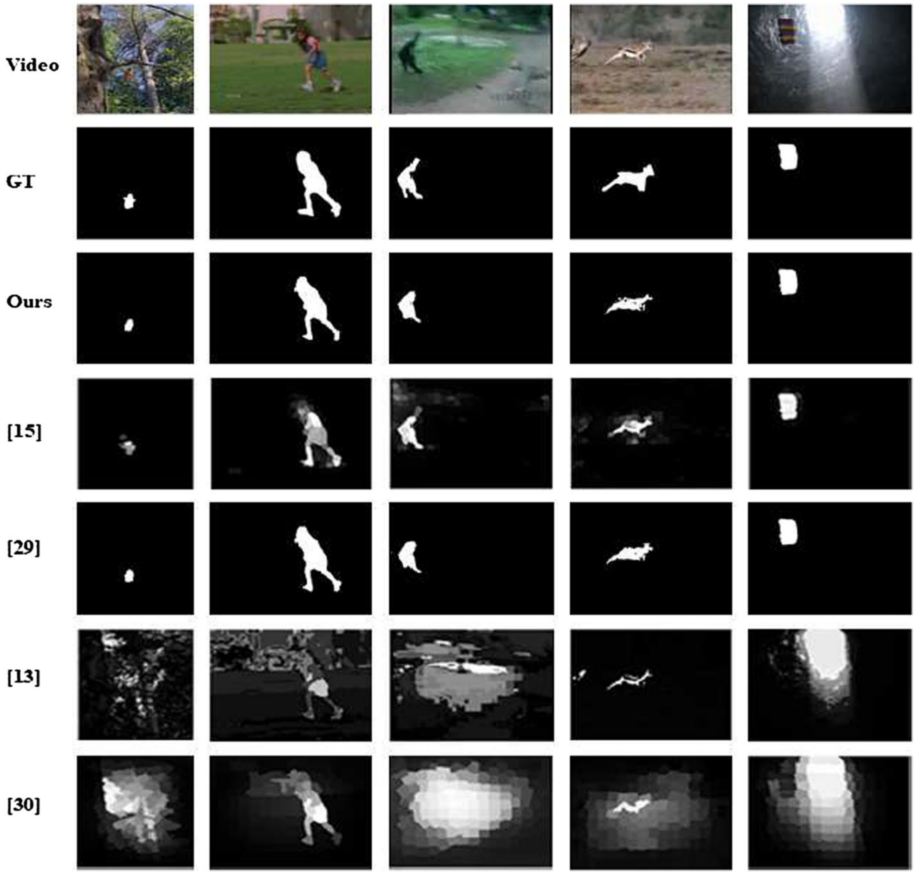

Fig. 11 The segmentation results obtained by different methods using the SegTrack data set with ground truth

Table 2 The segmentation performance in terms of the IoU metric and the computational time on the SegTrack v2 data set by different methods

Methods	IoU	Time (s)
[24]	58.4	0.37
[38]	67.5	42.20
[31]	70.3	12.00
[4]	65.4	10.00
[29]	71.1	**0.13**
[17]	57.1	0.21
Ours	**73.4**	1.60

Bold value indicates the best performance in terms of the highest IoU value and the fastest runtime

$$IoU = \frac{\text{Segmentation result} \cap \text{Ground-truth}}{\text{Segmentation result} \cup \text{Ground-truth}} \tag{20}$$

The experimental results of our method and other tested ones are shown in Table 2 in terms of the IoU metric and the computational time for segmenting one frame. As we can see from this table, the proposed method is able to obtain the highest IoU value among all tested methods. We may also notice that although our method is relatively slower than [17,29] and [24], it is significantly more computationally efficient than [4] and [31,38].

4 Conclusion

In this paper, an unsupervised video object segmentation approach was proposed based on the Gaussian mixture model with MRF. In our approach, a saliency detection method was developed to locate the object of interest. The developed saliency detection method can calculate the saliency of each frame in the video sequence and extract the position and region of the object of interest with more accurate object boundary. The refined extracted object region was then taken as the prior information and incorporated into our Gaussian mixture model and MRF to obtain the precise pixel-wise segmentation result of each frame. The effectiveness of the proposed unsupervised video object segmentation approach was validated by conducting experiments on both the SegTrack and the SegTrack v2 data sets.

Acknowledgements The completion of this work was supported by the National Natural Science Foundation of China (61876068), the Natural Science Foundation of Fujian Province (2018J01094) and the Promotion Program for Young and Middle-aged Teacher in Science and Technology Research of Huaqiao University (ZQN-PY510).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Achanta R, Shaji A, Smith K, Lucchi A, Fua P, Süsstrunk S (2012) Slic superpixels compared to state-of-the-art superpixel methods. IEEE Trans Pattern Anal Mach Intell 34(11):2274–2282
2. Bai X, Wang J, Simons DP, Sapiro G (2009) Video snapcut: robust video object cutout using localized classifiers. ACM Trans Graph 28(3):70
3. Brendel W, Todorovic S (2009) Video object segmentation by tracking regions. In: 2009 IEEE 12th international conference on computer vision, pp 833–840
4. Caelles S, Maninis K, Pont-Tuset J, Leal-Taixé L, Cremers D, Gool LV (2017) One-shot video object segmentation. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), pp 5320–5329
5. Cai W, Chen S, Zhang D (2007) Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation. Pattern Recognit 40(3):825–838
6. Celeux Gilles, Forbes Florence, Peyrard Nathalie (2001) Em procedures using mean field-like approximations for Markov model-based image segmentation. Pattern Recognit 36(1):131–144
7. Chockalingam P, Pradeep N, Birchfield S (2009) Adaptive fragments-based tracking of non-rigid objects using level sets. In: 2009 IEEE 12th international conference on computer vision, pp 1530–1537
8. Fan W, Bouguila N (2016) Model-based clustering based on variational learning of hierarchical infinite beta-liouville mixture models. Neural Process Lett 44(2):431–449
9. Fan W, Bouguila N (2019) Nonparametric hierarchical Bayesian models for positive data clustering based on inverted Dirichlet-based distributions. IEEE Access 7:83600–83614
10. Fan W, Hu C, Du J, Bouguila N (2018) A novel model-based approach for medical image segmentation using spatially constrained inverted Dirichlet mixture models. Neural Process Lett 47(2):619–639
11. Fan W, Bouguila N, Du J, Liu X (2019) Axially symmetric data clustering through Dirichlet process mixture models of Watson distributions. IEEE Trans Neural Netw Learn Syst 30(6):1683–1694
12. Felzenszwalb PF, Huttenlocher DP (2004) Efficient graph-based image segmentation. Int J Comput Vis 59(2):167–181

13. Fu H, Cao X, Tu Z (2013) Cluster-based co-saliency detection. IEEE Trans Image Process 22(10):3766–3778
14. Geman S, Geman D (1984) Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Trans Pattern Anal Mach Intell 6(6):721–741
15. Greggio N, Bernardino A, Laschi C, Dario P, Santos-Victor J (2012) Fast estimation of Gaussian mixture models for image segmentation. Mach Vis Appl 23(4):773–789
16. He H, Lu K, Lv B (2006) Gaussian mixture model with Markov random field for mr image segmentation. In: 2006 IEEE international conference on industrial technology, pp 1166–1170
17. He K, Gkioxari G, Dollar P, Girshick R (2018) Mask R-CNN. IEEE Trans Pattern Anal Mach Intell. https://doi.org/10.1109/TPAMI.2018.2844175
18. Hong C, Yu J, Wan J, Tao D, Wang M (2015) Multimodal deep autoencoder for human pose recovery. IEEE Trans Image Process 24(12):5659–5670
19. Hu C, Fan W, Du J, Zeng Y (2018) Model-based segmentation of image data using spatially constrained mixture models. Neurocomputing 283:214–227
20. Huang Y, Liu Q, Metaxas D (2009) Video object segmentation by hypergraph cut. In: 2009 IEEE conference on computer vision and pattern recognition, pp 1738–1745
21. Li F, Kim T, Humayun A, Tsai D, Rehg JM (2013) Video segmentation by tracking many figure-ground segments. In: 2013 IEEE international conference on computer vision, pp 2192–2199
22. Lu H, Woods JC, Ghanbari M (2007) Binary partition tree for semantic object extraction and image segmentation. IEEE Trans Circuits Syst Video Technol 17(3):378–383
23. Mahadevan V, Vasconcelos N (2010) Spatiotemporal saliency in dynamic scenes. IEEE Trans Pattern Anal Mach Intell 32(1):171–177
24. Marki N, Perazzi F, Wang O, Sorkine-Hornung A (2016) Bilateral space video segmentation. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR), pp 743–751
25. Mathe S, Sminchisescu C (2012) Dynamic eye movement datasets and learnt saliency models for visual action recognition. Comput Vis ECCV 2012:842–856
26. McLachlan G, Peel D (2000) Finite Mixture Models. Wiley, New York
27. Moon TK (1996) The expectation-maximization algorithm. IEEE Signal Process Mag 13(6):47–60
28. Nikou C, Galatsanos NP, Likas AC (2007) A class-adaptive spatially variant mixture model for image segmentation. IEEE Trans Image Process 16(4):1121–1130
29. Oh SW, Lee J, Sunkavalli K, Kim SJ (2018) Fast video object segmentation by reference-guided mask propagation. In: 2018 IEEE/CVF conference on computer vision and pattern recognition, pp 7376–7385
30. Papazoglou A, Ferrari V (2013) Fast object segmentation in unconstrained video. In: 2013 IEEE international conference on computer vision, pp 1777–1784
31. Perazzi F, Khoreva A, Benenson R, Schiele B, Sorkine-Hornung A (2017) Learning video object segmentation from static images. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), pp 3491–3500
32. Price BL, Morse BS, Cohen S (2009) Livecut: learning-based interactive video segmentation by evaluation of multiple propagated cues. In: IEEE international conference on computer vision, pp 779–786
33. Rahtu E, Kannala J, Salo M, Heikkilä J (2010) Segmenting salient objects from images and videos. Comput Vis- ECCV 2010:366–379
34. Ramadan H, Tairi H (2016) Moving object segmentation in video using spatiotemporal saliency and Laplacian coordinates. In: 2016 IEEE/ACS 13th international conference of computer systems and applications (AICCSA), pp 1–7
35. Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22(8):888–905
36. Tsai D, Flagg M, Mrehg J (2010a) Motion coherent tracking with multi-label mrf optimization. BMVC
37. Tsai D, Flagg M, Rehg J (2010b) Motion coherent tracking with multi-label mrf optimization. In: Proc. BMVC, pp 56.1–11
38. Tsai Y, Yang M, Black MJ (2016) Video segmentation via object flow. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR), pp 3899–3908
39. Vazquez-Reina A, Avidan S, Pfister H, Miller E (2010) Multiple hypothesis video segmentation from superpixel flows. Comput Vis ECCV 2010:268–281
40. Wang W, Shen J, Porikli F (2015) Saliency-aware geodesic video object segmentation. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR), pp 3395–3402
41. Wang W, Shen J, Yang R, Porikli F (2018) Saliency-aware video object segmentation. IEEE Trans Pattern Anal Mach Intell 40(1):20–33
42. Yang C, Zhang L, Lu H, Ruan X, Yang M (2013) Saliency detection via graph-based manifold ranking. In: 2013 IEEE conference on computer vision and pattern recognition, pp 3166–3173

43. Yu J, Yang X, Gao F, Tao D (2017) Deep multimodal distance metric learning using click constraints for image ranking. IEEE Trans Cybern 47(12):4014–4024
44. Yu J, Zhu C, Zhang J, Huang Q, Tao D (2019) Spatial pyramid-enhanced NetVLAD with weighted triplet loss for place recognition. IEEE Trans Neural Netw Learn Syst. https://doi.org/10.1109/TNNLS.2019.2908982
45. Yuen J, Russell B, Liu C, Torralba A (2009) Labelme video: Building a video database with human annotations. In: IEEE international conference on computer vision, pp 1451–1458
46. Zhang D, Javed O, Shah M (2013) Video object segmentation through spatially accurate and temporally dense extraction of primary object regions. In: 2013 IEEE conference on computer vision and pattern recognition, pp 628–635
47. Zhang L, Liu Y, Han S (2017) Video segmentation based on strong target constrained video saliency. In: 2017 2nd International conference on image, vision and computing (ICIVC), pp 356–360
48. Zhang J, Yu J, Tao D (2018) Local deep-feature alignment for unsupervised dimension reduction. IEEE Trans Image Process 27(5):2420–2432

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.