Vis Comput (2017) 33:1103–1119
DOI 10.1007/s00371-016-1268-2

ORIGINAL ARTICLE

Multiple cues-based active contours for target contour tracking under sophisticated background

Peng Lv[1] · **Qingjie Zhao**[1] · **Yanming Chen**[1] · **Liujun Zhao**[1]

Published online: 6 June 2016

Abstract In this paper, we propose a novel target contour tracking method under sophisticated background using the multiple cues-based active contour model. To locate the target position, a contour-based mean-shift tracker is designed which combines both color and texture information. To reduce the adverse impact of sophisticated background and also accelerate the curve motion, we propose a two-layer-based target appearance model that combines both discriminative pre-learned-based global layer and voting-based local layer. The proposed appearance model is able to extract rough target region from the complex background, which provides important target region information for our active contour model. We subsequently introduce a dynamical shape model to provide prior target shape information for more stable segmentation. To obtain accurate target boundaries, we design a new multiple cues-based active contour model which integrates with target edge, discriminative region, and shape information. The experimental results on 30 video sequences demonstrate that the proposed method outperforms other competitive contour tracking methods under various tracking environment.

Keywords Object contour tracking · Active contours · Level sets · Segmentation

Electronic supplementary material The online version of this article (doi:10.1007/s00371-016-1268-2) contains supplementary material, which is available to authorized users.

✉ Peng Lv
p1v@bit.edu.cn

[1] Beijing Key Laboratory of Intelligence Information Technology, School of Computer Science, Beijing Institute of Technology, Beijing 100081, China

1 Introduction

Target tracking is an important and challenging task in computer vision, such as robotics [36], video surveillance [19], and human–computer interaction [30]. Tracking target in real-world is typically difficult due to many factors, including illumination variation, appearance variation, pose changes, occlusion, and camera noise, etc. To overcome above-mentioned challenges and achieve robust target tracking, a large body of methods have been published in the literature during last two decades. For a survey of early tracking methods, we refer the readers to [39].

Typically, in conventional generative-based tracking method, a group of target templates are built and updated online based on previous target observations during tracking. Because the target appearance often changes and the background is sometimes sophisticated in the real-world tracking condition, generative-based tracking methods are hard to measure the target state correctly. Recently, discriminative-based tracking framework [2] draws more and more attention owing to its robust performance in complex scenarios. In the framework, discriminative technique plays an important role in building target appearance model.

However, these traditional tracking methods (both generative based and discriminative based) use rectangle or other rigid shape to represent the target, which lose detailed shape and boundary information of the target. Furthermore, these rigid shapes contain a large number of background pixels outside the target regions, which may reduce the tracking accuracy. A better manner to cope with this problem is to use contours or silhouettes to represent the deformable targets. Recently, many contour-based tracking methods [1,4–6,13,16,26,31,34] have been proposed to catch the detailed target shape information dynamically during tracking. Early work tends to use the parametric-based contour model [17]

to track a set of marked points around the target. Parametric-based contour model is not able to perform well when the target undergoes sophisticated background. Recently, level set-based active contour model [5,6,10,25,26] is widely applied because it could flexibly represent the target's topological changes, such as splitting and merging. Nevertheless, conventional level set methods mainly have two drawbacks: (a) they need to be reinitialized after several iterations to ensure the accuracy of segmentation results, which costs lots of extra computing time. (b) They only use edge, region, or shape information to guide curve motion during evolution, which may lead to false segmentation. Therefore, it is difficult for these methods to obtain stable and robust tracking performance in real-world tracking condition.

In this paper, we aim to track and segment the target under sophisticated background. we propose a novel and stable level sets-based framework for tracking non-rigid target boundaries using edge, region, and shape information. The key contributions that different from the other contour tracking methods are listed as follows:

1. To fast predict the target position accurately and pre-learn the target appearance changes, we propose a contour-based mean-shift target locating algorithm which integrates joint color and texture cues.
2. To extract discriminative rough region information for our active contour model, we propose a novel superpixel-based dynamic appearance model using both global and local layers to extract the discriminative rough target region. In the appearance model, an AdaBoost-based pre-learned model and a voting algorithm are embedded into the global and local layers, respectively.
3. To obtain more stable and accurate segmentation result under sophisticated background, we design a new multiple cues-based active contour model which combines edge, discriminative region and shape information to segment the target.

2 Related work

Mean-shift-based tracking methods As a fundamental task in the field of computer vision, object tracking has attracted much attention. In recent years, mean-shift methods have gained wide popularity in object tracking and video segmentation. Comaniciu et al. [11] propose mean-shift-based optimization framework to find the target location. To solve the limitation that the target scale and orientation could not be estimated efficiently, Yilmaz [38] uses an asymmetric kernel-based tracker to improve the tracking performance. However, these methods, which use fixed shape to represent the target, may result in inaccurate tracking performance when target shape changes during tracking. That is because the fixed shape may contain some background regions which might confuse the tracker. To obtain accurate target position, our method uses contour-based mean-shift tracker to locate the target that integrated with color and texture features.

Video segmentation methods Recently, video segmentation has received significant interest due to its critical importance in multimedia applications. Video segmentation aims to extract accurate target region from video sequence mainly using offline approaches. Generally, graph-based approaches [15] are among the top-performing methods for the task of segmentation. In [23], authors transform the problem of video target segmentation into the task to find a maximum weight clique in a weighted region graph. Lee et al. [20] introduce a method to estimate a pixel-level target segmentation based on a series of binary partitions among some key segments. To improve the segmentation performance, Ramakanth et al. [29] present an energy function based on patch seams across frames to solve the video segmentation task. However, these methods need to process the whole video together, which limits their effectiveness for applications that entail online processing, such as surveillance and action recognition. Another kind of approach for video target segmentation is video matting [3,37,41], which needs some human–computer interactions to obtain good segmentation performance. Nevertheless, in our work, we aim to automatically achieve online segmenting the moving target.

Contour-based online tracking methods Paragios et al. [27] firstly use geodesic active contour model [8] to drive the curve to target boundaries during evolution. However, edge-based contour model might not be able to drive the curve to target boundaries under sophisticated background in real-world tracking. Zhang et al. [40] introduce a background mismatching-based method to segment the moving target. Bibbly et al. [5] propose a pixel-based contour tracking method that uses a generative model; however, without the edge information, the tracker may lose precise target boundary information. These region-based contour tracking methods also have the limitation that the contour is sensitive to similar regions in foreground and background. In addition to the edge and region clues, Cremers [12] introduces a statistical shape knowledge into level set-based tracking method. Mahmoodi [24] also proposes a shape-based active contour model for video segmentation. Because that only shape information is used, the methods might be hard to track the target in complex tracking environment. Afterwards, Cai et al. [6] propose a contour tracking framework by combing both region and edge information. However, when a target undergoes variations caused by camera noise, shape variation, or self-shadowing, the tracker may generate inaccurate segmentation result. To obtain precise target boundary information, segmentation technique is also applied to contour tracking process. Fan et al. [13] introduce an image matting

model for tracking the target region on a scribble trimap. Godec et al. [14] present a hough-transform-based contour tracking method that integrates voting-based detection and back-projection into object segmentation process. However, these methods might generate over- or under-segmentation results under sophisticated background, which caused by the lack of discriminative target appearance information. Unlike traditional contour tracking methods, we build an target discriminative appearance model combining global and local layers to generate important discriminative region information, and then integrate the edge, region, and shape information into multi-cues-based active contour model to segment the target from sophisticated background.

The rest of the paper is organized as follows: Sect. 3 describes our target contour tracking framework. Section 4 introduces the evaluation metrics and also analyses the parameters in our model. We show the qualitative and quantitative results in Sect. 5. Section 6 summarizes the paper.

3 Proposed method

3.1 System overview

The framework of our method is shown in Fig. 1. After manual initializing the target contour, the new target position is located by the mean-shift tracker which combines color and texture clues. To capture the appearance changes and extract the rough target region, we propose a discriminative appearance model by combing both global and local target information. In the global layer, an AdaBoost-based pre-learned model is trained to extract the rough target region, while, in the local layer, voting-based algorithm is applied to retain the target local information. In addition, we also trained a shape model based on the prior segmentation results, which is helpful for guiding the curve motion in the curve evolution. Integrating with the discriminative region, edge and shape information, a new active contour model is proposed to accurately segment the target. During tracking, we update the appearance model using the segmenting result in each frame.

3.2 Contour-based mean-shift target locating

To reduce the impact of complex background and the time cost of the target segmenting, we firstly locate the target region before extracting its boundaries. Moreover, we also pre-learn the target appearance changes after locating the target, which would be benefit to extracting the rough target region in our appearance model. A natural approach to track and locate the target position would simply applying the mean-shift tracker which uses the rigid or elliptical region to represent the target. However, this approach has two drawbacks: (a) important target contour information may be lost during tracking; and (b) the tracker may be interfered by the background in target bounding-box. To cope with these two problems, we use non-rigid region to represent the target in our mean-shift tracker.

As shown in Fig. 2, at frame $t+1$, we use the non-rigid target region $I_{\mathcal{C}}(t)$ in frame $I(t)$ as the target template, which provides precise target information. To enable our tracker to achieve more robust tracking under various environment, we extract both color and texture information from the target region. We use a color histogram and LBP feature [32] to represent the colore and texture information, respectively:

$$\begin{cases} \mathbf{f}_R = \{\mathbf{f}_{\text{color}}, \mathbf{f}_{\text{texture}}\} \\ \mathbf{f}_{\text{color}} = f_{\text{RGB\&HSV}} \\ \mathbf{f}_{\text{texture}} = \text{LBP}(I_{\mathcal{C}}(t)). \end{cases} \tag{1}$$

To measure the similarity between template region and candidates, we use the following distance:

$$d(\mathbf{f}_R, \mathbf{f}) = \sqrt{1 - \rho[\mathbf{f}_R, \mathbf{f}]}, \tag{2}$$

where $\mathbf{f}$ is the feature of template region, $\rho[\cdot]$ is the Bhattacharyya distance between two discrete distributions, which

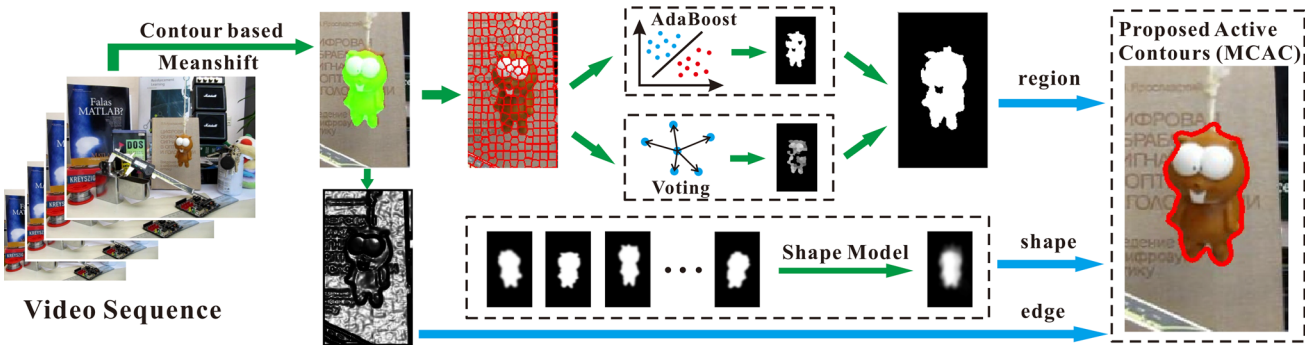


Fig. 1 Framework of the proposed contour tracking method

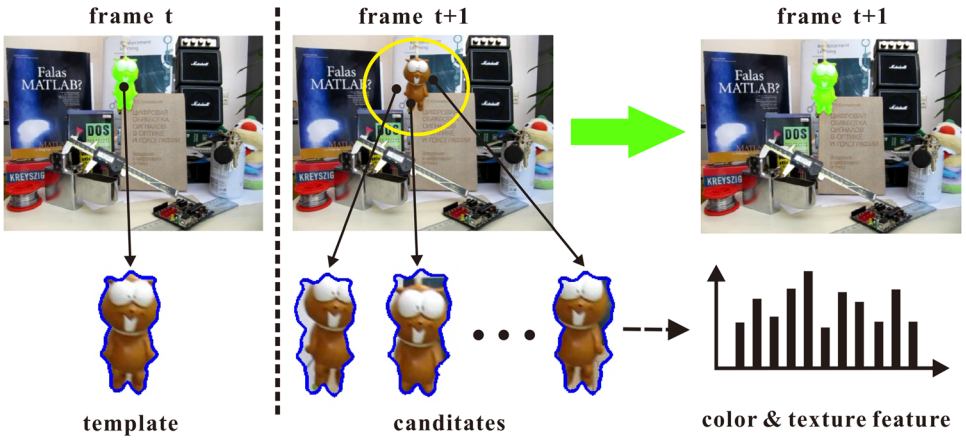


Fig. 2 Illustration of our contour-based mean-shift tracker

defined as:

$$\rho[\mathbf{f}_R, \mathbf{f}] = \sum_{i=1}^{N} \sqrt{\mathbf{f}_{i,R} \cdot \mathbf{f}_i}. \tag{3}$$

Then, we use mean-shift algorithm to find the target position $\mathbf{y}'_R$ in frame $t+1$ as follows:

$$\mathbf{y}'_R = \frac{\Sigma_{i=1}^{n} \mathbf{y}_{i,R} w_i g(\cdot)}{\Sigma_{i=1}^{n} w_i g(\cdot)}, \tag{4}$$

where w_i is the candidates weights, n is the number of samples, and $g(\cdot)$ is the kernel, respectively. After several iterations, a new non-rigid target position can be obtained in frame $I(t+1)$, as shown in Fig. 2. In our method, this non-rigid target region provides important information for our appearance model, which will be described more detailedly in the next section.

3.3 Appearance model combing global and local layers

Since the sophisticated background may affect the curve motion in segmentation procedure, we propose an appearance model to extract rough target region from the upcoming frame $I(t+1)$. Some prior works tend to use pixel-based or sparse-based models to represent the target; however, these models are hard to effectively represent the detailed target boundary information. In our method, we build a target appearance model based on superpixels to retain both target region and boundary information simultaneously. Besides, traditional contour tracking methods usually use single-layer-based appearance model, and thereby lose global or local region information. That may lead to unstable segmentation result. Rather than the single-layer appearance model, we combine both global and local layers to extract the rough target region. In our model, the global layer is able to provide primary region information when the target undergoes complex shape deformation. The local layer, meanwhile, provides important local region information when the global layer generates false classification results under sophisticated background. Such two layers provide important region information for our active contour model.

Discriminative pre-learning-based global layer In the global layer, we use a discriminative method to extract the global rough target region. Let $\mathrm{sp}_{i,t}$ stands for the i-th superpixel in frame $I(t)$. For every superpixel $\mathrm{sp}_{i,t}$, a histogram-based feature descriptor $\mathbf{s}_{i,t}$ is extracted in RGB and HSV color space. The feature descriptor $\mathbf{s}_{i,t}$ is labeled by $l_i = \{+1, -1\}$ according to the following criteria:

$$l_i = \begin{cases} +1, & \text{if } \frac{\mathrm{sp}_{i,t} \cap Target}{\mathrm{sp}_{i,t}} \geqslant \eta; \\ -1, & \text{if } \frac{\mathrm{sp}_{i,t} \cap Target}{\mathrm{sp}_{i,t}} < \eta, \end{cases} \tag{5}$$

where $\mathrm{sp}_{i,t} \cap Target$ means the intersection of superpixel $\mathrm{sp}_{i,t}$ and target region.

However, the target appearance may change during tracking, which may lead to false classification. To avoid this problem, we pre-learned the target appearance from the upcoming frame $I(t+1)$ before classifying superpixels. As discussed in Sect. 3.2, the mean-shift tracker locates the non-rigid target region in frame $I(t+1)$, which enables us to use this information to update the AdaBoost classifier. In the pre-learned procedure, we randomly select some unlabeled superpixels from the region $I_C(t+1)$ in the next frame as the positive examples to update the classifier. What is more, the internal superpixels have higher probability to be selected than ones closed to the periphery. After pre-learning the target appearance, our model could capture changes of the target and extract the rough target region $R_{t+1}^{\mathrm{global}}$, as shown in Fig. 3d.

To allow our model to adapt to various tracking conditions, the classifier is dynamically updated. Assuming that we have got the tracking result at frame $I(t)$, we randomly select some superpixels from the foreground and background as the

Fig. 3 Illustration of our global- and local-based appearance model: **a** the located position by our mean-shift tracker; **b** superpixel segmentation; **c** the target edge information; **d** discriminative region of the global layer; **e** result of voting in local layer; **f** the final rough target region; **g** final segmentation result on target region

positive and negative samples, respectively. These samples are used to update the AdaBoost classifier.

Voting-based local layer Under various tracking conditions, we find that the global layer may miss some local regions, and thereby leads to false segmentation. To reduce the adverse impact of noises caused by global layer, we propose a local voting algorithm to extract the target region.

In our model, to retain the local features of the target, each unlabeled superpixel in the upcoming frame $I(t+1)$ is voted by the surrounded labeled superpixels in the prior frame. We use the following distance to measure the similarity between two superpixels:

$$d_{\mathrm{sp}}(\mathbf{s}_i, \mathbf{s}_j) = \exp\left\{-\frac{\chi_2(\mathbf{s}_i, \mathbf{s}_j)^2}{\sigma}\right\}, \tag{6}$$

where χ_2 is the Chi-square distance [18]. For every superpixel in $I(t+1)$, the score which voted by the surrounded superpixels in $I(t)$ is computed by the following formula:

$$\mathrm{Score}(\mathrm{sp}_{i,t+1}) = \frac{\sum_j^{\mathrm{sp}_{j,t}\in\Omega_r} l_i \cdot \Gamma(d_{\mathrm{sp}}(\mathbf{s}_i, \mathbf{s}_j))}{\|\Gamma(d_{\mathrm{sp}}(\mathbf{s}_i, \mathbf{s}_j))\|_0}, \tag{7}$$

where Ω_r is the region of radius r surrounding the superpixel $\mathrm{sp}_{i,t+1}$ in frame $I(t)$. Besides, the kernel function $\Gamma(\cdot)$ is given by

$$\Gamma(d_{\mathrm{sp}}(\mathbf{s}_i, \mathbf{s}_j)) = \begin{cases} d_{\mathrm{sp}}(\mathbf{s}_i, \mathbf{s}_j), & \text{if } d_{\mathrm{sp}}(\mathbf{s}_i, \mathbf{s}_j) \geqslant \zeta; \\ 0, & \text{if } d_{\mathrm{sp}}(\mathbf{s}_i, \mathbf{s}_j) < \zeta. \end{cases} \tag{8}$$

This kernel function indicates that we only use credible pairwise superpixel for voting, which ensures our appearance model more stable. After the voting procedure, the local target region R_{t+1}^{local} is obtained, as shown in Fig. 3f.

To obtain more stable target region, we combine the global and local layers as follows: $R_{t+1} = R_{t+1}^{\mathrm{global}} \cup R_{t+1}^{\mathrm{local}}$. This rough target region provides important region information for our active contour model, which will be discussed more detail in Sect. 3.5. Moreover, to reduce the noises caused by these two layers, opening operator is applied to the expanded rough target region:

$$R_t' = (R_t \ominus B_1) \oplus B_2, \tag{9}$$

where B_1 and B_2 denote the erosion and dilation structuring element, respectively. Figure 3f, g shows that after integrating both global and local information into the appearance model, target can be extracted accurately.

3.4 Dynamic shape model

During the curve evolution in segmentation process, various noises such as illumination and target appearance changes may affect the curve evolution, which would result in the false segmentation. What is more, some false-negative regions generated by our appearance model may also cause under- or over-segmentation. In this section, we build a dynamic shape model to guide curve motion during evolution.

In our shape model, we use a gray image which fuses a series of target segmentation results to represent the target shape template. For a target shape $\mathcal{S}_t$ at time t, a gaussian kernel is applied to the target region: $\mathcal{S}_t = G(\mathcal{C}_t)$, where $\mathcal{C}_t$ is the target region mask which is labeled by 1s and 0s. To enable the shape model to adapt to target deformation, we update the model as follows:

$$\begin{aligned} \mathcal{S}_t &= p\mathcal{C}_t + (1-p)\mathcal{S}_{t-1} \\ &= p\mathcal{C}_t + p(1-p)\mathcal{C}_{t-1} + (1-p)^2\mathcal{S}_{t-2} \\ &= p\mathcal{C}_t + p(1-p)\mathcal{C}_{t-1} + p(1-p)^2\mathcal{C}_{t-2} + (1-p)^3\mathcal{S}_{t-3} \\ &= p\mathcal{C}_t + p(1-p)\mathcal{C}_{t-1} + \cdots \\ &\quad + p(1-p)^{t-1}\mathcal{C}_1 + (1-p)^t\mathcal{S}_0 \\ &= \sum_{k=1}^{t} p(1-p)^{t-k}\mathcal{C}_k + (1-p)^t\mathcal{S}_0, \end{aligned} \tag{10}$$

where p is the update ratio and $\mathcal{S}_0$ is the initialized target region. Our shape model $\mathcal{S}_t$ is able to give assistance to guide the curve evolution in our active contour model, which would be discussed more detail in the next section.

3.5 Multi-cues active contours and curve evolution

Although we have obtained the rough target region by our appearance model, the false positive and negative regions may affect the segmentation performance. To get accurate target boundaries, this section introduces our multi-cues active contour model which combines edge, discriminative region, and shape information. Because the conventional active contour models [8,9,21,40] only consider edge or region information, the curve is vulnerable to be interfered by the complicated background or obvious boundaries, and thereby stops at the false position after evolution. On the other hand, the active contours combined with shape model, to some extent, perform well under sophisticated background; nevertheless, it may not be adapted to complex deformation. To accurately segment the target under various conditions, such as sophisticated background and large deformation, we embed our dynamic appearance model and shape model into the proposed active contour model.

Edge term As many works [8,9] refer, an edge-detector is defined for extracting the image boundaries: $g(|\nabla I|) = 1/(1+|\nabla \hat{I}|^2)$. Note that the rough expanded target region R'_t, which is obtained in our appearance model as described in Sect. 3.3, could reduce the negative effect of the background. Therefore, to accelerate the curve evolution, we just let the curve move on the extended rough target region $I'_R(t)$, where $I'_R(t) = R'_t \cdot I(t)$. Then, the edge information of the rough target region can be represented as follows:

$$g_{\text{edge}} = \frac{1}{1+|\nabla \widehat{R'_t \cdot I(t)}|^2} = R'_t \cdot g(|\nabla I(t)|) - R'_t + 1. \tag{11}$$

Motivated by [21], we define an edge term in our active contour model based on the edge information g_{edge}:

$$\begin{aligned}\mathcal{F}_1 &\triangleq \int_\Omega g_{edge} \cdot \delta(\varphi)|\nabla\varphi|\mathrm{d}x \\ &\triangleq \int_\Omega (R'_t \cdot g(|\nabla I(t)|) - R'_t + 1) \cdot \delta(\varphi)|\nabla\varphi|\mathrm{d}x.\end{aligned} \tag{12}$$

Region term In many situations, it is hard to extract target boundaries due to the blurred edge or sophisticated background, which would affect the curve motion during evolution. To enable the curve to correctly stop at the target boundaries, target region information is embedded into our active contour model.

Recall that in Sect. 3.3, the rough target region R_t provides important information of target region for the active contour model. However, this region information cannot be straightly embedded into the edge-based geodesic active contour model. To address this constraint and embed the region information into our model, we transform the region R_t into homologous edge information beforehand:

$$\begin{aligned}g_{\text{region}} &= g(|\nabla R_t \cdot I(t)|) + g(|\nabla R_t|) - 1 \\ &= R_t \cdot g(|\nabla I(t)|) + g(|\nabla R_t|) - R_t.\end{aligned} \tag{13}$$

Then, we define the following region term in our active contour model:

$$\begin{aligned}\mathcal{F}_2 &\triangleq \int_\Omega g_{\text{region}} \cdot \delta(\varphi)|\nabla\varphi|\mathrm{d}x \\ &\triangleq \int_\Omega (R_t \cdot g(|\nabla I(t)|) + g(|\nabla R_t|) - R_t) \cdot \delta(\varphi)|\nabla\varphi|\mathrm{d}x.\end{aligned} \tag{14}$$

Shape term During the tracking, our appearance model may generate some false-negative regions. Due to the false-negative regions information, the curve may move across the target boundaries. To cope with this problem, we add the target shape information to the active contour model:

$$\begin{cases} g'_{\text{edge}} = \mathcal{S}_t \cdot g_{\text{edge}}; \\ g'_{\text{region}} = \mathcal{S}_t \cdot g_{\text{region}}, \end{cases} \tag{15}$$

where $\mathcal{S}_t$ is the target shape model. Then, we use Eq. (15) to update Eqs. (12) and (14), respectively. There are two advantages to embed the shape term into our active contour model: (a) the shape term is able to allow the curve to move toward the target boundary outside the target region; (b) and also ensures that the curve would not continue to converge inside the target region, which effectively improve the segmentation performance. After integrating with shape information, the proposed active contour model could produce more stable results.

Energy functional and curve evolution By combining the edge, region, and shape information, we propose a multi-cues active contour model (MCAC):

$$\mathcal{E}(\varphi) = \alpha\mathcal{F}_1(\varphi) + \beta\mathcal{F}_2(\varphi) + \mu\mathcal{R}(\varphi) + \tau\mathcal{A}(\varphi), \tag{16}$$

where $\mathcal{A}(\varphi)$ and $\mathcal{R}(\varphi)$ are area accelerate term and non-reinitialization term to speed up the curve evolution procedure, respectively. These two terms are given by

$$\mathcal{A}(\varphi) \triangleq \int_\Omega g(|\nabla I(t)|)H(-\varphi)\mathrm{d}x, \tag{17}$$

$$\mathcal{R}(\varphi) \triangleq \int_\Omega p(|\nabla\varphi|)\mathrm{d}x, \tag{18}$$

where $H(\cdot)$ is the Heaviside function and $p(\cdot)$ is a potential function defined in [21].

Then, the Eq. (16) could be minimized by solving the following gradient flow:

$$\frac{\partial \varphi}{\partial t} = \delta_\epsilon(\varphi)\left[\alpha \mathrm{div}\left(\mathcal{S}_t \cdot g_{\mathrm{edge}} \cdot \mathbf{F}\right) + \beta \mathrm{div}\left(\mathcal{S}_t \cdot g_{\mathrm{region}} \cdot \mathbf{F}\right)\right] + \mu \mathrm{div}(d_p(|\nabla\varphi|)\nabla\varphi) + \tau g(|\nabla I|)\delta_\epsilon(\varphi), \quad (19)$$

where $\mathbf{F} = \nabla\varphi/|\nabla\varphi|$. By applying the finite difference calculation framework, the energy $\mathcal{E}(\varphi)$ will slow down the shrinking or expanding the zero level contour when the curve arrives at target boundaries.

4 Evaluation criteria and parameter analysis

4.1 Evaluation metrics

To quantitatively and effectively evaluate the performance of the implemented tracking methods compared to the manual segmentation groundtruth, we report the following contour-based criteria in our experiments: Intersection-over-Union (IoU), Dice coefficient (Dice), Mean Absolute Distance (MAD), and the Hausdorff Distance (HD). Let $\mathcal{C}_1'$ and $\mathcal{C}_2'$ denote the contours of regions $\mathcal{C}_1$ and $\mathcal{C}_2$, and the contour-based criteria can be defined as follows:

$$\mathrm{IoU}(\mathcal{C}_1', \mathcal{C}_2') = \frac{|\mathcal{C}_1 \cap \Omega_2|}{|\mathcal{C}_1| \cup |\mathcal{C}_2|}, \quad (20)$$

$$\mathrm{Dice}(\mathcal{C}_1', \mathcal{C}_2') = \frac{2|\mathcal{C}_1 \cap \mathcal{C}_2|}{|\mathcal{C}_1| + |\mathcal{C}_2|}, \quad (21)$$

$$\mathrm{MAD}(\mathcal{C}_1', \mathcal{C}_2') = \frac{\int_0^1 d(\mathcal{C}_1(s), \mathcal{C}_2)|\mathcal{C}_1'(s)|ds}{2|\mathcal{C}_1|} + \frac{\int_0^1 d(\mathcal{C}_2(s), \mathcal{C}_1)|\mathcal{C}_2'(s)|ds}{2|\mathcal{C}_2|}, \quad (22)$$

$$\mathrm{HD}(\mathcal{C}_1', \mathcal{C}_2') = \max\left\{\sup_{s_1}\inf_{s_2} d(\mathcal{C}_1(s_1), \mathcal{C}_2(s_2)), \sup_{s_2}\inf_{s_1} d(\mathcal{C}_1(s_1), \mathcal{C}_2(s_2))\right\}, \quad (23)$$

where $d(\mathcal{C}_1(s), \mathcal{C}_2)$ denotes the minimum distance between point $\mathcal{C}_1(s)$ and contour $\mathcal{C}_2$, $|\mathcal{C}_1|$ and $|\mathcal{C}_1|$ represent the contour length and the area of region $\mathcal{C}_1$, respectively. The IoU and Dice metrics are used to intuitively evaluate the tracking performance by computing overlap rate between two regions, which is commonly used in target tracking and image segmentation. To reasonably evaluate the segmentation performance, we adopt MAD and HD metrics to, respectively, indicate the mean and peak errors of the experiment results compared with the groundtruth.

4.2 Parameter analysis

Size of superpixel Choosing an appropriate size for superpixels is very important. When the superpixel size is too small, the feature descriptor would have low discriminative ability, which may result in bad classification in our discriminative appearance model. In contrast, if the superpixel size is too large, the false-positive and false-negative superpixels would significantly interfere the curve motion in our active contour model. What is more, large superpixel may lose detailed target region information. To find an appropriate value for the superpixel size, we test different superpixel sizes on 30 video sequences and subsequently report the tracking results under four metrics (IoU, Dice, MAD, and HD) in Fig. 4. It is shown that good results (high overlap rate and low pixel error) can be obtain when we set the size of superpixel between 10 and 15.

The Ratio of α to β In the experiment, we find that the ratio ν between α and β in Eq. (16) has great relevance to the segmentation performance. When ν is too large, the curve tends to stop at the boundaries of the sophisticated background. Conversely, when ν is too small, the curve motion might be seriously influenced by the false negative and positive regions and would, therefore, result in inaccurate segmentation. To obtain stable performance, we test different values for parameter ν. Figure 5 reports the tracking results under different metrics using different values for ν, where we can see that the tracking performance is more stable when we set $\nu \in [2, 5]$.

Parameters μ and τ As discussed in [21], the active contour model is not sensitive to the choice of μ, thus in our experiment, we set $\mu = 1$. Traditionally, the parameter τ needs to be tuned according to the boundaries of the target in different tracking conditions. For target with weak boundaries, the value of τ should be chosen relatively small to avoid boundary leakage. However, in our method, the appear-

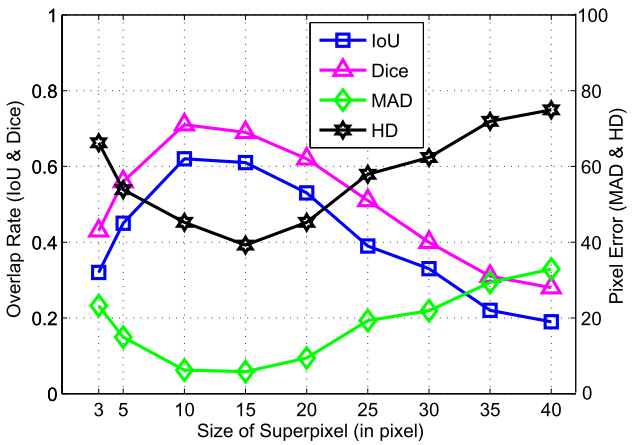


Fig. 4 Comparison of the tracking results under four metrics with different superpixel sizes. IoU and Dice are based on *left Y* axis, while MAD and HD are based on *right Y* axis

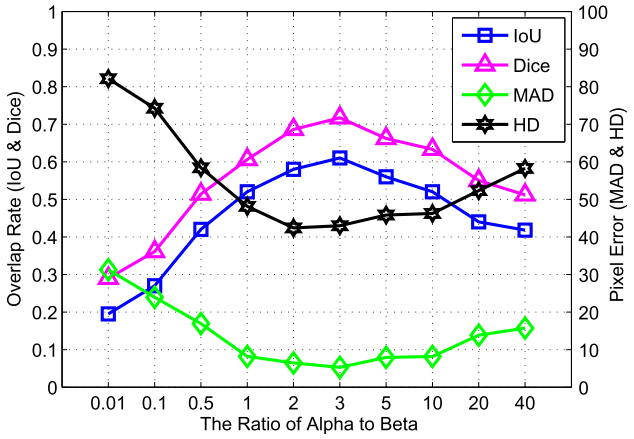


Fig. 5 Comparison of the tracking results under four metrics with different ratios of α to β. IoU and Dice are based on *left Y* axis, while MAD and HD are based on *right Y* axis

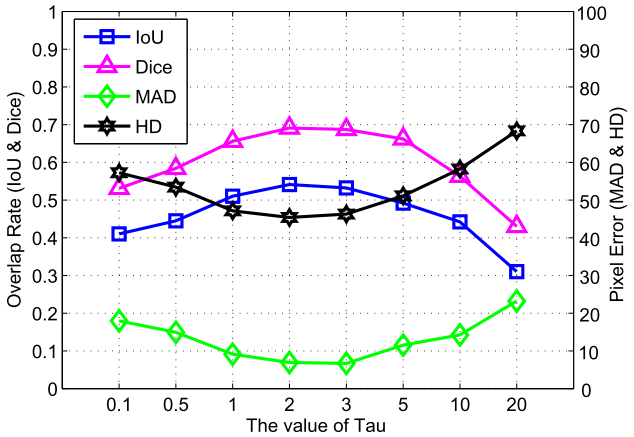


Fig. 6 Comparison of the tracking results under four metrics with different values of τ. IoU and Dice are based on *left Y* axis, while MAD and HD are based on *right Y* axis

ance model and shape model could provide additional target boundary information. So the impact of the value τ variation is not significant. Figure 6 reports the tracking performance using different values for τ, where we can see that the proposed method generates stable results when $\tau \in [1, 5]$.

5 Experimental results

5.1 Experimental setup

The proposed method is implemented in MATLAB R2010b under Red Hat Enterprise Linux platform on a Intel (R) Core (TM) i7 3.4GHz processor with 3GB memory. In addition, the tracking results of the proposed method are available on the website: https://github.com/plvmail/MultiCuesActiveContour.

Parameters setting In Sect. 3.3, the radius r of voting region Ω_r is set to 20. We set $\sigma = 0.4$ and $\zeta = 0.3$. In Eq. (9), the erosion and dilation structuring element are 5×5 and 12×12, respectively. The updating parameter p in our dynamic shape model is set to 0.7. Besides, we set $\alpha = 1$, $\beta = 3$, $\mu = 1$, and $\tau = 2$ in the energy functional Eq. (16) of the proposed active contour. During the evolution, we set number of the inner and outer iteration steps as 8 and 40, respectively.

Compared Algorithms To objectively evaluate the improvement of the proposed method, five contour tracking algorithms and three baseline methods are compared: (a) background mismatch-based method (Mismatch) [40]; (b) superpixel-based method (SPT) [35]; (c) dynamic graph-based method (DGT) [7]; (d) hough-based method (HT) [14]; (e) Scribble tracker based on matting-based method (Scribble) [13]. Note that when we implement other algorithms, the parameters are set to the default values suggested in the original papers. Moreover, to better analyse the improvement of the proposed active contour model, we also build three baseline methods: (f) our method with edge-based distance regularized level set evolution (DRLSE) [21]; (g) our method with region-based active contours (GACV) [9]; (h) our method without shape information (w/o shape).

Dataset For a more comprehensive evaluation of the tracking performance, we implement the proposed method on SegTrack v2 dataset [22] and seven extra traditional video sequences in our experiments. SegTrack v2 dataset is an extension version of the SegTrack dataset [33] with more annotated objects and video sequences, which is widely used in video segmentation algorithms. SegTrack v2 dataset consists of 14 sequences with 24 objects over 947 annotated frames including different challenges, including appearance variation (*Bird of Paradise* and *Birdfall*), similar objects (*Penguin*), complex deformation (*Worm*, *Hummingbird*, *Soldier*, *Monkey*, *Frog*, and *BMX*), show-motion (*Frog*), and occlusion (*Cheetah* and *Penguin*).

5.2 Quantitative comparison with segmentation-based methods

We report the quantitative results of the proposed method and state-of-the-art methods under IoU and Dice metrics in Table 1. Table 2 summarises the mean and peak errors of the results under MAD and HD metrics. It is shown that in both SegTrack v2 dataset and traditional video sequences, the proposed method outperforms other online target tracking and segmentation methods. To clearly and immediately analyse the improvement of our method against different challenges, more detailed discussions are presented next.

Complex deformation It is really an important and challenging task to track and segment deformable targets in

Table 1 Comparison of the proposed method with five state-of-the-art methods on 30 video sequences under Intersection-over-Union (IoU) and Dice coefficient (Dice) metrics

Methods	Mismatch [40]		Scribble [13]		HT [14]		SPT [35]		DGT [7]		Proposed	
Metrics (%) ↑	IoU	Dice	IoU	Dice	IoU	Dice	IoU	Dice	IoU	Dice	IoU	Dice
Girl	19.42	31.01	47.90	64.60	39.19	51.78	59.10	73.83	**72.02**	**83.59**	*69.52*	*81.84*
Frog	45.08	60.06	*45.72*	*62.30*	40.93	57.30	40.58	55.54	40.59	55.30	**66.43**	**78.19**
Birdfall	12.71	18.12	5.89	10.40	5.46	8.45	12.96	20.94	**61.03**	**75.23**	*54.47*	*70.23*
Monkey	38.68	55.04	33.16	49.44	42.92	56.93	59.00	73.84	*67.11*	*80.16*	**75.53**	**85.91**
Bird of Paradise	29.11	41.20	52.61	68.52	43.99	55.28	38.88	53.20	*82.75*	*90.00*	**83.68**	**91.36**
BMX-Person	28.51	43.57	37.29	54.22	1.86	2.31	42.15	59.04	*60.39*	*75.18*	**73.59**	**84.51**
BMX-Bike	*6.39*	*9.89*	4.31	7.29	0.89	1.53	4.17	7.65	5.09	9.15	**15.32**	**21.27**
Cheetah-Deer	21.55	34.21	*46.25*	*62.79*	38.65	53.24	28.91	43.93	33.62	47.54	**58.22**	**73.06**
Cheetah-Cheetah	6.47	11.46	*19.28*	*27.88*	15.06	19.77	10.72	17.43	–	–	**44.92**	**58.86**
Drift-Green	25.97	37.92	**58.11**	*72.98*	48.36	63.96	22.54	34.03	*58.03*	**73.16**	56.53	71.49
Drift-Red	6.78	12.28	11.09	16.99	**39.82**	**56.52**	25.91	39.50	6.01	10.03	**38.84**	*54.21*
Hummingbird-Left	24.09	36.81	37.08	52.19	12.09	21.14	36.23	49.83	*56.59*	*71.49*	**64.28**	**77.15**
Hummingbird-Right	24.54	37.04	20.91	31.93	18.16	28.98	*37.09*	*53.14*	15.98	26.56	**57.74**	**72.64**
Monkeydog-Monkey	**69.91**	**81.38**	21.53	29.12	29.28	42.21	22.32	28.82	56.32	70.47	*67.41*	*79.24*
Monkeydog-Dog	10.76	14.86	**72.32**	**79.22**	5.91	9.87	12.44	14.35	1.95	3.56	*25.11*	*32.37*
Parachute	37.14	49.98	64.32	76.82	64.55	78.35	66.41	77.90	**89.43**	**94.29**	*82.69*	*90.31*
Penguin-#1	17.74	28.48	*25.93*	*39.29*	24.16	35.99	9.21	15.61	–	–	**74.63**	**85.42**
Penguin-#2	56.82	70.78	17.99	28.16	**78.66**	**88.03**	7.71	13.49	3.27	6.29	*75.06*	*85.65*
Penguin-#3	35.35	47.46	*38.39*	*54.42*	6.22	10.70	3.03	5.54	1.49	2.93	**55.49**	**70.64**
Penguin-#4	46.49	62.06	45.48	62.20	*49.37*	*66.00*	2.50	4.67	14.88	25.72	**61.50**	**75.88**
Penguin-#5	13.74	22.75	23.54	36.79	*26.55*	*41.82*	5.93	10.60	1.26	2.43	**43.64**	**60.20**
Penguin-#6	22.88	35.32	*39.56*	*55.72*	13.15	22.38	29.05	44.73	–	–	**60.36**	**74.61**
Soldier	29.34	42.98	40.03	55.80	57.83	71.96	59.91	73.81	*70.77*	*82.63*	**81.15**	**89.55**
Worm	16.58	28.11	37.65	54.19	42.63	55.20	37.34	53.09	*67.69*	*80.41*	**74.25**	**84.83**
Bird2	34.62	49.85	47.52	63.47	25.75	39.97	42.66	59.42	*48.89*	*65.00*	**80.20**	**88.91**
Panda	18.13	28.88	58.76	72.98	59.84	69.70	57.53	72.27	*71.43*	*82.97*	**85.77**	**92.20**
Pedxing1	50.15	66.53	52.68	68.87	59.70	74.39	54.71	70.47	*61.70*	*76.11*	**83.90**	**91.21**
Surfer	4.48	7.79	*57.44*	*72.40*	41.90	57.00	42.98	59.20	53.31	68.61	**65.13**	**78.42**
Seq_sb	12.79	22.20	19.88	31.27	61.21	74.80	20.69	27.12	*78.10*	*87.17*	**79.70**	**88.03**
Lemming	23.21	32.39	62.38	71.08	71.95	80.32	75.54	82.35	*80.16*	*87.29*	**83.58**	**89.22**
Mean per object	26.31	37.35	38.17	51.11	35.53	46.53	32.27	43.18	*46.66*	*56.79*	**64.62**	**75.91**
Mean per sequence	25.80	36.91	41.09	54.28	39.79	51.27	39.96	52.13	*55.70*	*66.77*	**68.94**	**79.40**

"–" Indicates that the result is not reported in the sequence. The best two results are labeled with bold and italic values, respectively

video sequence. In our experiments, we firstly implement the methods on *Girl*, *Frog*, *Monkey*, *BMX-Person*, *Hummingbird*, *Worm*, *Soldier*, and *Surfer* sequences, wherein the targets undergo large deformations. The superpixel-based trackers (DGT and SPT) perform better than the pixel-based trackers (Mismatch, Scribble, and HT). It is because the pixel-based trackers do not build effective appearance model, which might lead to inaccurate segmentation when target shape changes. However, as shown in Table 2, both SPT and DGT have large peak errors (under HD metric) which indicate that they could not generate stable segmentation results. Mismatch performs well on sequence *Frog*, wherein the background is clear and the target undergoes slow motion. Both of these two conditions are beneficial for Mismatch to drive the curve toward the target boundary; however, in other sequences, the tracker fails to segment the target. Notwithstanding the cluttered background or slow motion in these sequences, the proposed method performs better. That is because our appearance model is able to extract the rough target region by combing global and local region information, which provides important target region information to guide the curve motion.

Appearance variation To demonstrate the improvement of the proposed method when target appearance changes, we

Table 2 Comparison of the proposed method with five state-of-the-art methods on 30 video sequences under Mean Absolute Distance (MAD) and Hausdorff Distance (HD) metrics

Methods	Mismatch [40]		Scribble [13]		HT [14]		SPT [35]		DGT [7]		Proposed	
Metrics (in pixel) ↓	MAD	HD	MAD	HD	MAD	HD	MAD	HD	MAD	HD	MAD	HD
Girl	21.40	163.2	13.08	70.11	14.45	72.10	16.66	140.6	*4.29*	*61.12*	**3.64**	**33.73**
Frog	36.56	121.2	23.64	*96.40*	38.64	123.6	38.49	100.7	*23.13*	149.9	**9.29**	**70.70**
Birdfall	27.75	126.3	38.55	104.4	45.99	64.36	27.00	60.37	*2.13*	*13.13*	**1.70**	**7.87**
Monkey	17.73	103.7	19.06	86.94	9.56	*46.47*	*5.13*	49.64	6.68	59.87	**2.42**	**23.50**
Bird of Paradise	11.90	117.3	14.79	97.78	21.30	104.7	13.36	105.4	*8.06*	*95.40*	**6.98**	**69.57**
BMX-Person	48.86	275.5	*19.66*	*126.7*	156.8	242.2	22.23	168.9	29.46	246.4	**2.89**	**26.01**
BMX-Bike	49.17	210.7	69.31	143.7	87.45	172.4	53.22	132.5	*48.74*	*130.4*	**15.92**	**87.23**
Cheetah-Deer	14.97	98.81	*4.11*	*28.57*	6.13	29.15	6.41	40.21	6.81	35.66	**2.39**	**16.92**
Cheetah-Cheetah	48.31	197.1	*32.52*	*65.38*	81.37	104.18	59.04	111.7	–	–	**6.32**	**26.96**
Drift-Green	21.35	178.9	**6.65**	*49.96*	16.16	81.98	24.85	143.1	12.70	131.9	*7.51*	**46.78**
Drift-Red	57.95	380.8	81.50	218.7	*14.81*	*92.64*	19.50	92.68	60.56	189.2	**12.51**	**76.78**
Hummingbird-Left	36.56	190.9	23.64	*118.6*	38.64	174.6	38.49	151.6	*23.13*	170.6	**9.29**	**82.23**
Hummingbird-Right	18.35	118.7	25.56	*95.81*	21.92	109.7	*12.54*	102.7	38.51	191.7	**6.68**	**57.66**
Monkeydog-Monkey	*3.94*	62.11	31.95	68.78	9.17	*36.33*	55.63	94.60	10.35	66.09	**1.77**	**14.75**
Monkeydog-Dog	17.35	*70.32*	**7.09**	**21.32**	84.92	180.6	62.46	125.8	62.15	122.7	*14.82*	75.32
Parachute	32.83	162.2	4.62	28.23	3.73	*21.59*	2.46	24.60	*2.42*	29.25	**1.41**	**12.98**
Penguin-#1	55.19	236.2	18.99	*66.12*	*17.93*	75.97	21.49	86.91	–	–	**3.26**	**21.82**
Penguin-#2	7.62	58.25	18.00	62.63	**2.52**	**17.09**	23.69	81.39	23.94	83.62	*3.13*	*20.71*
Penguin-#3	24.25	122.6	*15.28*	*76.64*	41.40	107.6	42.49	109.8	29.54	86.83	**5.44**	**33.55**
Penguin-#4	*6.42*	51.41	8.83	51.39	6.95	*28.62*	45.91	111.3	14.75	54.12	**4.18**	**20.74**
Penguin-#5	29.60	146.3	15.52	72.27	*12.63*	*55.03*	30.11	81.82	78.26	167.9	**5.20**	**28.92**
Penguin-#6	37.43	194.8	13.74	*66.87*	18.55	75.95	*10.27*	73.25	–	–	**4.24**	**24.35**
Soldier	12.52	103.8	11.84	66.69	10.81	73.53	8.67	89.46	*3.49*	*47.01*	**1.30**	**22.53**
Worm	32.06	254.4	*11.46*	60.89	14.90	*50.49*	16.72	109.7	18.26	156.5	**1.96**	**14.64**
Bird2	8.41	59.73	*4.45*	*32.08*	7.59	35.06	6.70	52.58	7.24	52.81	**0.81**	**15.75**
Panda	32.85	175.8	*3.75*	22.71	4.26	*18.30*	6.17	48.53	5.52	45.32	**1.01**	**8.49**
Pedxing1	18.22	140.1	5.45	39.67	5.12	35.66	*2.97*	**31.23**	4.92	50.04	**0.98**	**8.36**
Surfer	72.22	305.9	*3.09*	*19.74*	6.75	35.18	4.36	31.07	5.16	36.31	**2.01**	**15.42**
Seq_sb	17.29	58.73	13.36	47.95	*2.88*	*14.10*	21.95	55.75	4.78	42.98	**1.05**	**8.17**
Lemming	26.12	69.94	15.98	31.84	8.28	29.83	18.87	49.82	*3.82*	*12.83*	**2.99**	**9.81**
Mean per object	28.17	151.86	*19.18*	*71.29*	27.05	76.97	23.93	88.59	19.96	93.69	**4.77**	**32.74**
Mean per sequence	27.65	149.5	17.46	*67.01*	23.48	69.85	19.78	81.10	*14.31*	80.54	**4.09**	**30.09**

"–" Indicates that the result is not reported in the sequence. The best two results are labeled with bold and italic values, respectively

run the compared methods on sequences *Parachute*, *Bird of Paradise*, *Drift*, and *Seq_sb*. Note that *Drift* sequence also has sophisticated background. As shown in Table 1, appearance variations on the targets bring a lot of difficulties to the methods that do not have effective appearance models, such as Mismatch, SPT, and Scribble. For this reason, these three methods generate accumulated errors and have low overlap rates. Although HT could capture the appearance variation using hough voting mechanism, nevertheless, due to lack of target shape information the method could not accurately segment the target. DGT, which benefits from the graph matching-based appearance model, performs well on the tested sequences. However, DGT has larger mean and peak errors compared to the proposed method, as shown in Table 2. Overall, profiting from the pre-learned procedure, our dynamic appearance model could capture the appearance changes promptly, which enables the proposed active contour model to accurately segment the target.

Similar objects In visual tracking, similar objects confuse many methods to correctly track the target, and the same problem also occurs in segmentation-based tracking methods. In the experiments, similar objects occur in sequences *Penguin*. From Table 1, we can see that both superpixel-

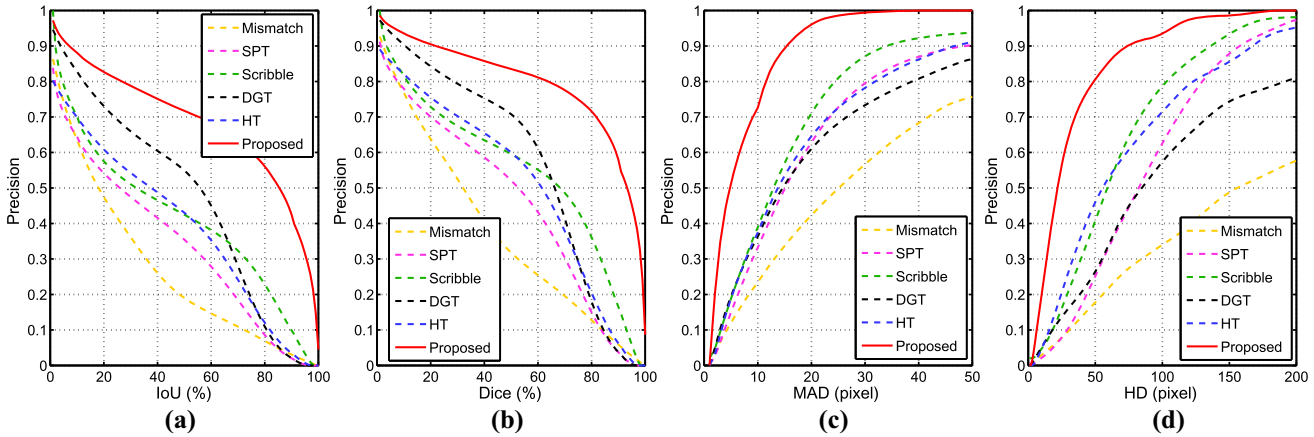


Fig. 7 Precision comparison of the proposed method with five state-of-the-art methods under four metrics: **a** Intersection-over-Union (IoU); **b** dice coefficient (Dice); **c** Mean Absolute Distance (MAD); and **d** Hausdorff Distance (HD)

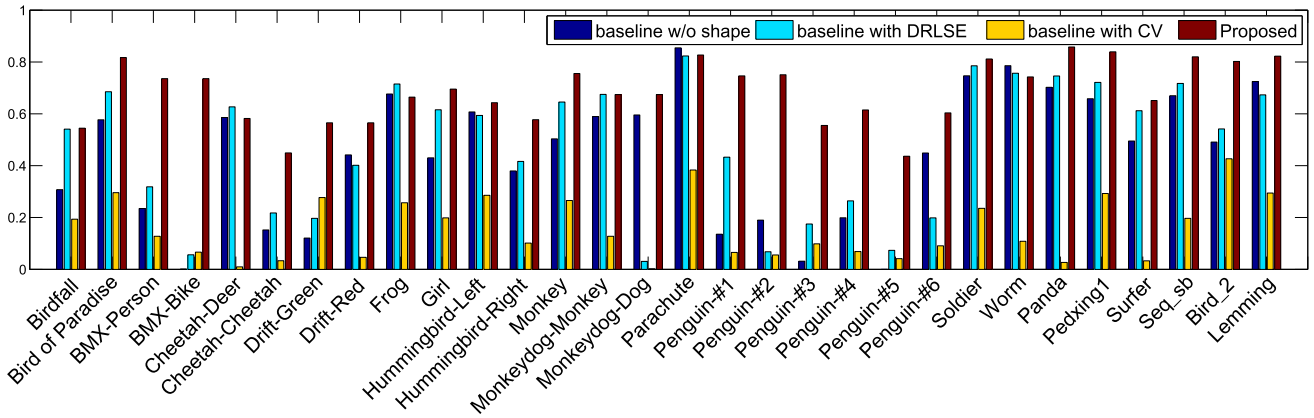


Fig. 8 Comparison of the proposed method with three baseline methods on 30 tested video sequences under IoU metric

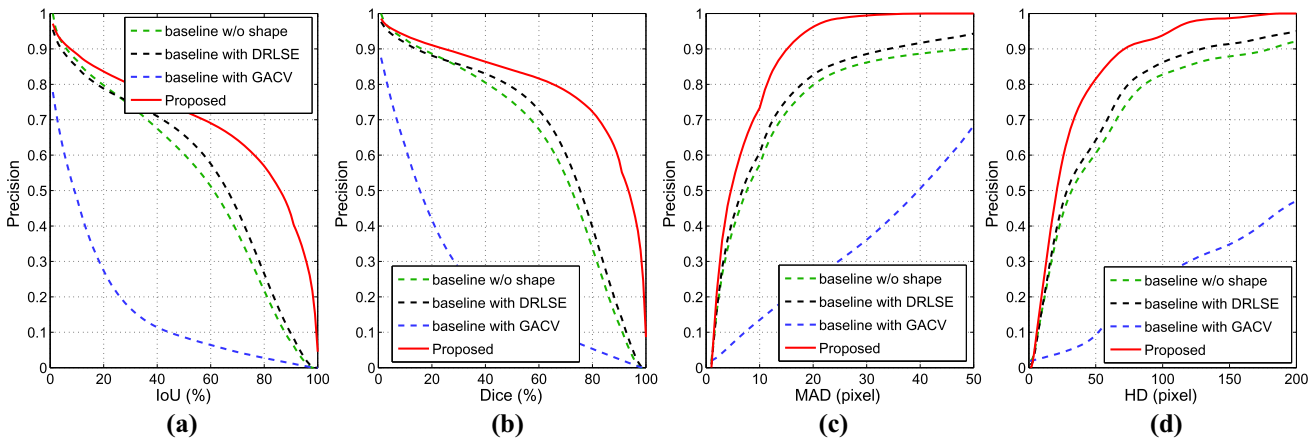


Fig. 9 Precision comparison of the proposed method with three baseline methods under four metrics: **a** Intersection-over-Union (IoU); **b** Dice coefficient (Dice); **c** Mean Absolute Distance (MAD); and **d** Hausdorff Distance (HD)

based methods (SPT and DGT) fail to segment the target. That is because SPT and DGT consider neither the global shape information nor local superpixel position information, which makes the methods hard to distinguish similar objects. Because Mismatch, Scribble, and HT are based on pixel-level and local search, therefore, they are more robust against similar objects. However, these methods, especially HT, might fail to generate accurate results due to the accumulated error.

In the proposed method, both priori shape information and local region information are considered, which helps the active contour model to correctly segment the target. Tables 1 and 2 show that the proposed method outperforms other methods.

Occlusion The targets are occluded in sequences *Bird2*, *Cheetah*, and *Penguin*. By combining discriminative global and local appearance information, the proposed method could handle the occluded cases and generate good performance, as presented in Table 1. Due to the lack of precise region information, Mismatch, HT, and SPT could not extract the target region correctly. Scribble and DGT perform slightly better on sequences *Bird2* and *Cheetah-Deer*, which benefit from their effective appearance model. However, DGT has low overlap rate on the sequence *Penguin*, which includes similar objects. In contrast, relying on the prior shape information and the multi-cues active contour model, our method could finally obtain the accurate segmentation results.

To better compare the tracking performance of the implement methods, we also show the precision curves under four metrics (IoU, Dice, MAD, and HD) in Fig. 7. The precision curves under IoU and Dice metrics are shown in Fig. 7a, b, where we can see that Mismatch method could not correctly segment the target in most cases (about 70 % of the results have low overlap rate where IoU $< 40\,\%$ and Dice $< 50\,\%$). It is because the Mismatch is easily to be confused by the complex background, occlusion, and appearance variation. Compared with Mismatch, methods HT, Scribble, and SPT perform better. It should be noted that Scribble presents slightly more stable results than the other two methods (HT and SPT), and it also can be reflected in Fig. 7c, d, where Scribble results in low mean and peak errors. Method DGT has higher overlap rate than Mismatch and HT under IoU and Dice metrics, whereas its precision rapidly decreased when IoU $> 60\,\%$ and Dice $> 60\,\%$, which means that DGT might not be able to generate accurate and stable segmentation results. Benefitting from the discriminative global and local region information, the proposed method performs significant better than other methods. Besides that, the prior target shape information allows our active contour model to handle the noises originated from sophisticated background; therefore, our method lowers mean and peak errors on test video sequences, as shown in Fig. 7c, d.

5.3 Quantitative comparison with baseline methods

To demonstrate the improvement of the proposed active contour model in our tracking framework, we compare the proposed method with three baseline methods: baseline framework with DRLSE [21], baseline framework with GACV [9], and baseline framework without shape model.

Table 3 Comparison of the proposed method with four offline video segmentation methods on SegTrack v2 dataset under Intersection-over-Union (IoU) metric

Methods	[20]	[15]	[22]	[28]	Proposed
Girl	*87.7*	31.9	**89.1**	83.4	69.5
Frog	–	*67.1*	65.8	**69.0**	66.4
Birdfall	49.0	*57.4*	**62.0**	47.8	54.5
Monkey	*79.0*	61.9	**84.1**	70.9	75.5
Bird of Paradise	**92.2**	86.8	*88.2*	81.1	83.7
BMX-Person	**87.4**	39.2	*75.1*	74.5	73.6
BMX-Bike	**38.6**	*32.5*	24.6	30.9	15.3
Cheetah-Deer	*44.5*	18.8	17.4	18.3	**58.2**
Cheetah-Cheetah	11.7	24.4	*41.3*	22.2	**44.9**
Drift-Green	63.7	55.2	**73.8**	*65.4*	56.5
Drift-Red	30.1	27.2	**58.4**	**59.8**	38.8
Hummingbird-left	**74.0**	25.2	65.2	*65.8*	64.3
Hummingbird-right	*46.3*	13.7	45.4	35.0	**57.7**
Monkeydog-Monkey	**74.3**	*68.3*	58.8	24.1	67.4
Monkeydog-Dog	4.9	18.8	*17.4*	16.5	**25.1**
Parachute	**96.3**	69.1	*93.2*	91.3	82.7
Penguin-#1	12.6	*72.0*	51.4	59.3	**74.6**
Penguin-#2	11.3	**80.7**	73.2	*79.1*	75.1
Penguin-#3	11.3	*75.2*	69.6	**75.6**	55.5
Penguin-#4	7.7	**80.6**	57.6	47.1	*61.5*
Penguin-#5	4.2	*62.7*	**63.4**	45.8	43.6
Penguin-#6	8.5	**75.5**	48.6	56.7	*60.4*
Soldier	66.6	66.5	**83.0**	50.7	*81.2*
Worm	**84.4**	34.7	*75.6*	59.5	74.3
Mean per object	47.2	51.9	**61.8**	55.4	*60.8*
Mean per sequence	57.3	50.8	**68.0**	58.6	*64.3*

"–" Indicates that the result is not reported in the sequence. The best two results are labeled with bold and italic values, respectively

The overlap rate (under IoU metric) of the baseline methods on tested video sequences is shown in Fig. 8. Because the region-based traditional active contour method GACV does not consider the target edge and shape information, it is hard for this method to obtain good segmentation results on the sequence wherein the target appearance is not obvious. Figure 8 shows that baseline method with GACV fails to segment the target in most sequences. On the other hand, edge-based baseline method with DRLSE also meets the similar problem. Lacking of the constraints of target region and shape information, the method would be interfered when target boundary is blurred or the background is complex, such as *Birdfall*, *Cheetah-Deer*, and *Penguin*. In our method, the shape model is of great significance to reduce noise, and also ensures the stability of the proposed active contour model. Without the shape model, the baseline method fails to segment the target under complex scenes (*Penguin*, *BMX*, and *Drift*). Compared with the other three baseline methods, the

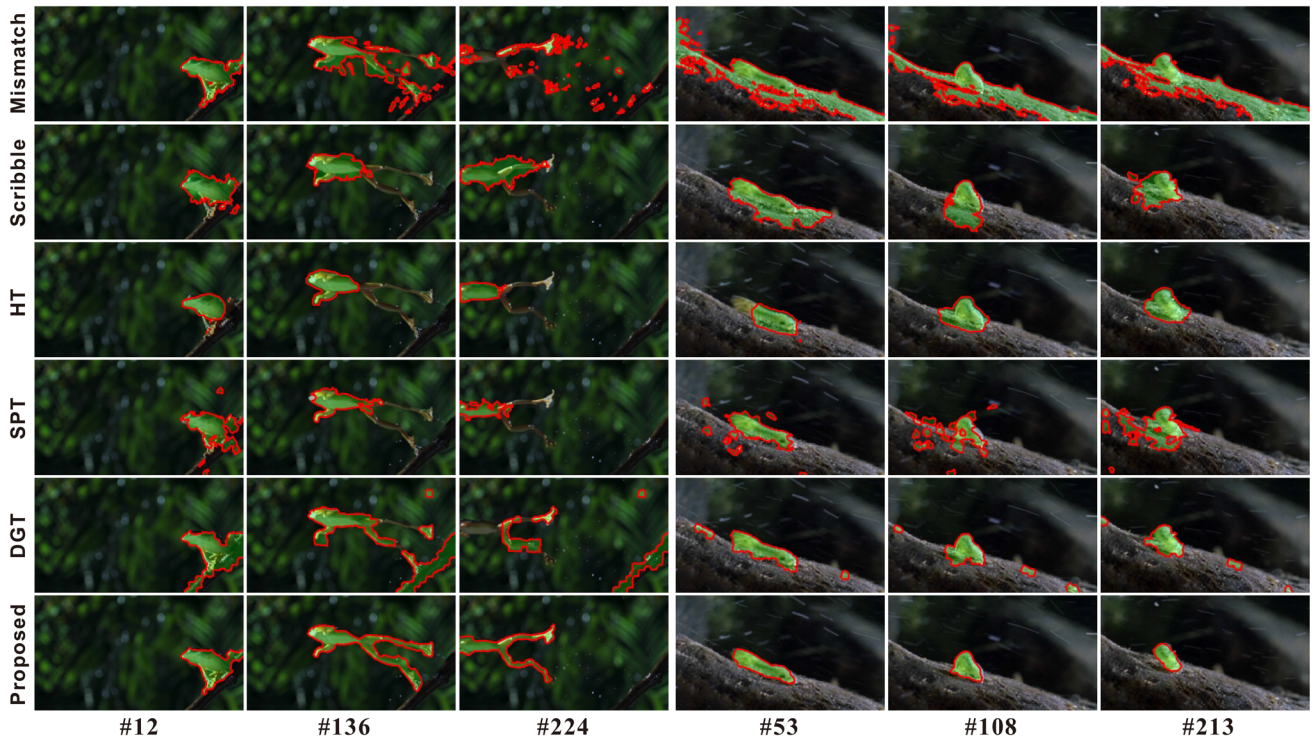


Fig. 10 Tracking and segmentation performance on sequences *Frog* (*left three columns*) and *Worm* (*right three columns*) of sex methods (from *top* to *bottom*): Mismatch [40], Scribble [13], HT [14], SPT [35], DGT [7], and the proposed

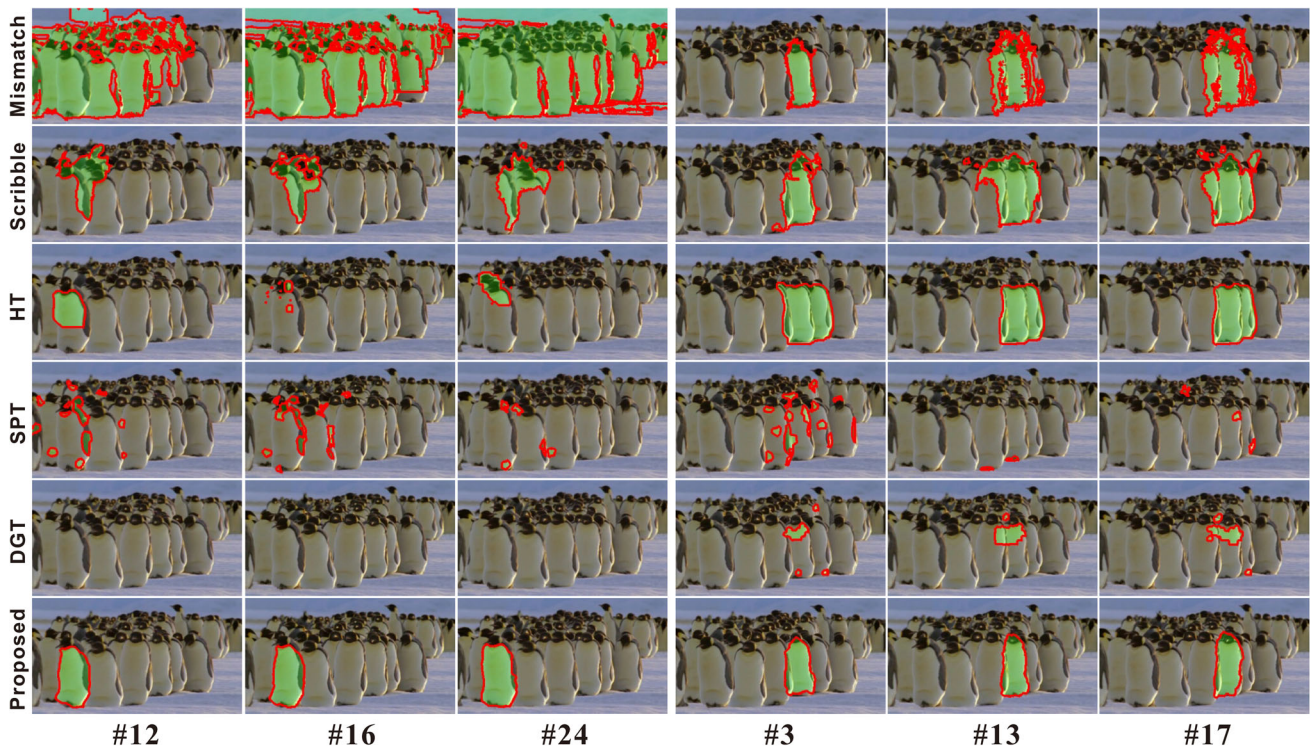


Fig. 11 Tracking and segmentation performance on sequences *Penguin-#1* (*left three columns*) and *Penguin-#4* (*right three columns*) of sex methods (from *top* to *bottom*): Mismatch [40], Scribble [13], HT [14], SPT [35], DGT [7], and the proposed

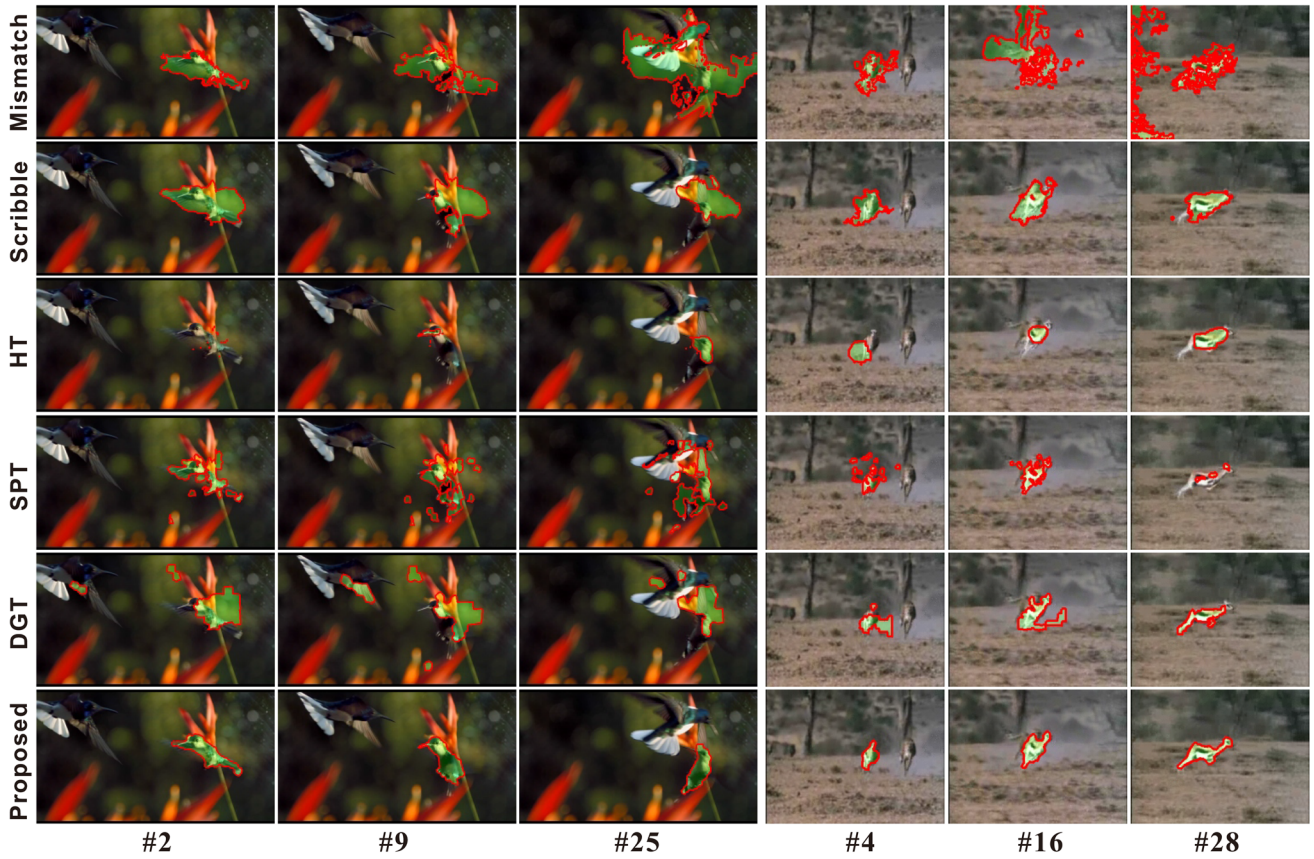


Fig. 12 Tracking and segmentation performance on sequences *Hummingbird-Right* (*left three columns*) and *Cheetah-Deer* (*right three columns*) of sex methods (from *top* to *bottom*): Mismatch [40], Scribble [13], HT [14], SPT [35], DGT [7], and the proposed

proposed method is able to obtain better and more stable performance on most tested sequences.

In Fig. 9, we also present the precision curves under four metrics. One can see that baseline methods with DRLSE and w/o shape perform significantly better than baseline method with GACV. However, without the shape information, these two methods are prone to be slightly effected by the false positive and negative regions generated from our appearance model. Overall, as shown in Fig. 9, by integrating with discriminative region, edge, and shape information, the proposed method has higher overlap rate and lower mean errors.

5.4 Quantitative comparison with video segmentation methods

To better demonstrate the effectiveness of the proposed method, we also compare our method with four offline video segmentation methods [15,20,22,28]. These four methods aim to segment the moving target by analysing the information of an entire video, such as motion and target appearance changes information. Compared to online target segmentation methods, video segmentation-based methods could obtain more target and video information; therefore, better segmentation performance is easier to be obtained. Although the offline processing limits their application, in our experiments, we still make some comparison with the proposed method.

The tracking and segmentation results of five implemented methods on SegTrack v2 dataset are presented in Table. 3. Although the proposed method could not always obtain best results, the proposed method is more stable than methods [15,20,28]. Method [22] performs best on the dataset; nevertheless, compared to [22], our method is able to generate very competitive results, as shown in Table 3.

5.5 Qualitative comparison

To more intuitively measure the comparisons, we show the tracking results on six video sequences including different challenges. Figure 10 shows two video sequences, *Frog* and *Worm*, which contain deformation and slow motion. In method Mismatch, the pixel-based flow model is difficult to capture the target deformation, which probably results in false segmentation, as shown in Fig. 10. Scribble, HT, and SPT perform better than Mismatch; nevertheless, these methods cannot accurately segment the specific detailed region

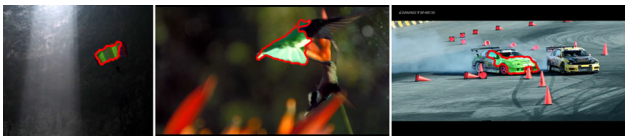


Fig. 13 Some false segmentation results of the proposed method

of the target. The graph-based method DGT could segment rough target region on sequence *Worm*, however, when strong deformation occurs (i.e. at frame 224 in *Frog*), DGT failed to segment the target. It is shown that with the help of the discriminative region and prior shape information, the proposed method could capture the target deformation and performs well on both sequences.

We show the comparative results of sequences *Penguin-#1* and *Penguin-#4* in Fig. 11, where we can see that both similar objects and occlusion occur in the sequences. When similar objects occur, SPT is hard to distinguish the target from background due to the shortage of local spatial information makes it difficult for. The same problem occurs to DGT, as shown in Fig. 11. Without the effective appearance model, Mismatch and Scribble also fail to segment the target. It is noteworthy that because the voting-based appearance model in HT contains the local spatial information of the target, thus the method is able to deal with similar objects in a way. However, without the shape information, HT could not accurately segment the target in most cases, as shown in frame #12 in *Penguin-#1* and frame #3 in *Penguin-#4*. In the proposed method, our appearance model and target shape model are able to provide rough target region information for the multi-cues-based active contour model, which makes our method perform more stable than other methods.

Figure 12 shows the experiment results of sequences *Hummingbird Right* and *Cheetah-Deer*. These two sequences contain complex background and appearance variation. Since Mismatch lacks of effective appearance model, the interference of the sophisticated background leads to false segmentation. SPT cannot obtain accurate segmentation results yet on both sequences. Scribble, HT, and DGT perform better on sequence *Cheetah-Deer*; however, without the shape restriction, all these three methods are prone to be interfered by the complex background in sequence *Hummingbird-Right* and, thus, result in inaccurate segmentation, as shown in Fig. 12. Overall, integrating with region, edge, and shape information, the proposed method generates better segmentation results on both sequences, which indicates that the method is robust to appearance variation and complex background.

6 Conclusion

In this paper, we propose a novel level set-based target contour tracking method based on multi-cues active contours by combing edge, region, and dynamic shape information to segment the target. Qualitative and quantitative results show that our method performs better than other state-of-the-art methods. Although the proposed method performs well on the most sequences, sometimes our method would be interfered by various conditions, as shown in Fig. 13. That is because our appearance model might not always generate good result, which would mislead the curve motion. Further work will aim at developing a more powerful appearance model to represent the target, which may improve the segmentation performance.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China under Grand (Nos. 61175096 and 61303245) and Specialized Fund for Joint Building Program of Beijing municipal Education Commission. The authors would also like to thank C. Li, J. Fan, M. Godec, S. Wang, Z. Cai, X. Jia, and M. Yang et al. for providing their source codes for comparisons in our experiments.

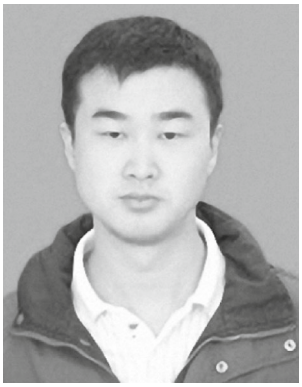
References

1. Allili, M.S., Ziou, D.: Active contours for video object tracking using region, boundary and shape information. Signal Image Video Process. **1**(2), 101–117 (2007)
2. Avidan, S.: Ensemble tracking. IEEE Trans. Pattern Anal. Mach. Intell. **29**(2), 261–271 (2007). doi:10.1109/TPAMI.2007.35
3. Bai, X., Wang, J., Simons, D., Sapiro, G.: Video snapcut: robust video object cutout using localized classifiers. ACM Trans. Gr. **28**(3), 70 (2009)
4. Bibby, C., Reid, I.: Robust real-time visual tracking using pixel-wise posteriors. In: Proc. Eur. Conf. Comput. Vision: Part II, ECCV '08, pp. 831–844 (2008)
5. Bibby, C., Reid, I.: Real-time tracking of multiple occluding objects using level sets. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1307–1314 (2010)
6. Cai, L., He, L., Yamashita, T., Xu, Y., Zhao, Y., Yang, X.: Robust contour tracking by combining region and boundary information. IEEE Trans. Circuits Syst. Video Technol. **21**(12), 1784–1794 (2011)
7. Cai, Z., Wen, L., Lei, Z., Vasconcelos, N., Li, S.: Robust deformable and occluded object tracking with dynamic graph. IEEE Trans. Image Process. **23**(12), 5497–5509 (2014). doi:10.1109/TIP.2014.2364919
8. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic active contours. In: Proc. IEEE Int. Conf. Comput. Vision, pp. 694–699 (1995)
9. Chen, L., Zhou, Y., Wang, Y., Yang, J.: Gacv: geodesic-aided c-v method. Pattern Recogn. **39**(7), 1391–1395 (2006)
10. Chiverton, J., Xie, X., Mirmehdi, M.: Tracking with active contours using dynamically updated shape information. In: Proc. British Mach. Vision Conf., pp. 253–262 (2008)
11. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Trans. Pattern Anal. Mach. Intell. **25**(5), 564–577 (2003). doi:10.1109/TPAMI.2003.1195991
12. Cremers, D.: Dynamical statistical shape priors for level set-based tracking. IEEE Trans. Pattern Anal. Mach. Intell. **28**(8), 1262–1273 (2006)
13. Fan, J., Shen, X., Wu, Y.: Scribble tracker: a matting-based approach for robust tracking. IEEE Trans. Pattern Anal. Mach. Intell. **34**(8), 1633–1644 (2012)

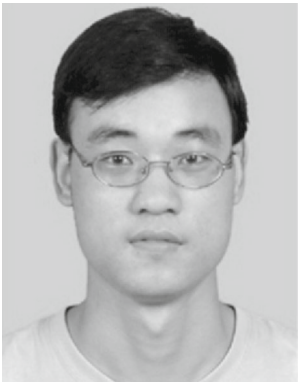
14. Godec, M., Roth, P.M., Bischof, H.: Hough-based tracking of non-rigid objects. Comput. Vision Image Underst. **117**(10), 1245–1256 (2013)
15. Grundmann, M., Kwatra, V., Han, M., Essa, I.: Efficient hierarchical graph-based video segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2141–2148 (2010). doi:10.1109/CVPR.2010.5539893
16. Hoch, M., Litwinowicz, P.C.: A semi-automatic system for edge tracking with snakes. Visual Comput **12**(2), 75–83. doi:10.1007/BF01782106
17. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: active contour models. Int. J. Comput. Vision **1**(4), 321–331 (1988). doi:10.1007/BF00133570
18. Khoreva, A., Galasso, F., Hein, M., Schiele, B.: Classifier based graph construction for video segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 951–960 (2015). doi:10.1109/CVPR.2015.7298697
19. Krotosky, S.J., Trivedi, M.M.: Person surveillance using visual and infrared imagery. IEEE Trans. Circuits Syst. Video Technol. **18**(8), 1096–1105 (2008)
20. Lee, Y.J., Kim, J., Grauman, K.: Key-segments for video object segmentation. In: IEEE International Conference on Computer Vision, pp. 1995–2002 (2011). doi:10.1109/ICCV.2011.6126471
21. Li, C., Xu, C., Gui, C., Fox, M.: Distance regularized level set evolution and its application to image segmentation. IEEE Trans. Image Process. **19**(12), 3243–3254 (2010)
22. Li, F., Kim, T., Humayun, A., Tsai, D., Rehg, J.: Video segmentation by tracking many figure-ground segments. In: IEEE International Conference on Computer Vision, pp. 2192–2199 (2013). doi:10.1109/ICCV.2013.273
23. Ma, T., Latecki, L.: Maximum weight cliques with mutex constraints for video object segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 670–677 (2012). doi:10.1109/CVPR.2012.6247735
24. Mahmoodi, S.: Shape-based active contours for fast video segmentation. IEEE Signal Process. Lett. **16**(10), 857–860 (2009). doi:10.1109/LSP.2009.2025924
25. Mansouri, A.R.: Region tracking via level set pdes without motion computation. IEEE Trans. Pattern Anal. Mach. Intell. **24**(7), 947–961 (2002)
26. Niethammer, M., Tannenbaum, A., Angenent, S.: Dynamic active contours for visual tracking. IEEE Trans. Autom. Control **51**(4), 562–579 (2006)
27. Paragios, N., Deriche, R.: Geodesic active contours and level sets for the detection and tracking of moving objects. IEEE Trans. Pattern Anal. Mach. Intell. **22**(3), 266–280 (2000)
28. Pirsiavash, H., Ramanan, D., Fowlkes, C.: Globally-optimal greedy algorithms for tracking a variable number of objects. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1201–1208 (2011). doi:10.1109/CVPR.2011.5995604
29. Ramakanth, S., Babu, R.: Seamseg: Video object segmentation using patch seams. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 376–383 (2014). doi:10.1109/CVPR.2014.55
30. Stenger, B., Thayananthan, A., Torr, P.H.S., Cipolla, R.: Model-based hand tracking using a hierarchical bayesian filter. IEEE Trans. Pattern Anal. Mach. Intell. **28**(9), 1372–1384 (2006)
31. Sun, X., Yao, H., Zhang, S.: A novel supervised level set method for non-rigid object tracking. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3393–3400 (2011). doi:10.1109/CVPR.2011.5995656
32. MP, Timo Ahonen Abdenour Hadid: Face description with local binary patterns: application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **28**(12), 2037–2041 (2006)
33. Tsai, D., Flagg, M., Nakazawa, A., Rehg, J.M.: Motion coherent tracking using multi-label MRF optimization. Int. J. Comput. Vision **100**(2), 190–202 (2012)
34. Vaswani, N., Rathi, Y., Yezzi, A., Tannenbaum, A.: Deform pf-mt: particle filter with mode tracker for tracking nonaffine contour deformations. IEEE Trans. Image Process. **19**(4), 841–857 (2010)
35. Wang, S., Lu, H., Yang, F., Yang, M.H.: Superpixel tracking. In: IEEE International Conference on Computer Vision, pp. 1323–1330 (2011). doi:10.1109/ICCV.2011.6126385
36. Wren, C., Azarbayejani, A., Darrell, T., Pentland, A.: Pfinder: real-time tracking of the human body. IEEE Trans. Pattern Anal. Mach. Intell. **19**(7), 780–785 (1997)
37. Xiao, C., Gan, J., Hu, X.: Fast level set image and video segmentation using new evolution indicator operators. Visual Comput. **29**(1), 27–39 (2013)
38. Yilmaz, A.: Kernel-based object tracking using asymmetric kernels with adaptive scale and orientation selection. Mach. Vision Appl. **22**(2), 255–268 (2011). doi:10.1007/s00138-009-0237-4
39. Yilmaz, A., Javed, O., Shah, M.: Object tracking: a survey. ACM Comput. Surv. **38**(4), 13 (2006)
40. Zhang, T., Freedman, D.: Improving performance of distribution tracking through background mismatch. IEEE Trans. Pattern Anal. Mach. Intell. **27**(2), 282–287 (2005)
41. Zou, D., Chen, X., Cao, G., Wang, X.: Video matting via sparse and low-rank representation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1564–1572 (2015)

Peng Lv received the BS degree in School of Mathematical Sciences, Shandong University, in 2011. He won bronze in the ACM International Collegiate Programming Contest in 2010 and 2011, respectively. Now, he is currently a Ph.D. candidate in School of Computer Science and Technology, Beijing Institute of Technology, China. His main research focused on target tracking, active contour model, computer vision, and image processing.

Qingjie Zhao received the BS and MS degrees in Control Science and Engineering, from China University of Petroleum in 1988, and Shandong University in 1996, respectively. She received the Ph.D. degree in Computer Science and Technology from Tsinghua University, in 2003. She was a visiting fellow in the School of Computer Science and Electronic Engineering, University of Essex, UK, from September 2008 to August 2009. She is currently a professor in the School of Computer Science and Technology, Beijing Institute of Technology, China. Her current research interests include computing intelligence, machine vision, system theory and application.



Yanming Chen received the BS and MS degrees in Signal and Information Processing, from Qiqihar University in 2006, and Harbin University of Science and Technology in 2009, respectively. He was a research assistant in the Institute of Computing Technology of the Chinese Academy of Sciences from September 2010 to August 2013. He is currently a Ph.D. candidate in the School of Computer Science and Technology, Beijing Institute of Technology, China. His current research interests are in the areas of distributed algorithms and sensor networks.



Liujun Zhao received his MS degrees in Computer Science from North China Electric Power University (Beijing), in 2011. He is currently a Ph.D. candidate of the School of Computer Science, Beijing Institute of Technology, China. His research interests include object tracking, image processing and machine vision.