Neurocomputing 83 (2012) 47–55

Contents lists available at SciVerse ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Tracking objects using shape context matching

Zhao Liu [a,*], Hui Shen [a], Guiyu Feng [b], Dewen Hu [a]

[a] Department of Automatic Control, College of Mechatronics and Automation, National University of Defense Technology, Changsha, Hunan 410073, China
[b] Institute of Computing Technology, Beijing Jiaotong University, Beijing 100029, China

ARTICLE INFO

Article history:
Received 9 July 2011
Received in revised form
4 November 2011
Accepted 18 November 2011
Communicated by L. Shao
Available online 27 December 2011

Keywords:
Object tracking
Shape context
Gaussian mixture model
Shape prior

ABSTRACT

In this paper we propose a novel tracking method, which provides accurately segmented object boundaries. The first step of the proposed method is to model the object and background using Gaussian mixture model (GMM), and extract a rough contour according to the object edge features. And then the states of the object, including translation, rotation and scale, are estimated using shape context matching. Finally, we take an elastic shape matching method to extract the exact contour. The proposed method is robust enough for tracking object with translation, rotation, scale change and partial occlusion, and it can also be used for real-time tracking applications. Experiments on both synthetic and real world video sequences demonstrate the effectiveness of the proposed method.

1. Introduction

Visual tracking is a fundamental problem in computer vision and has been applied to many aspects including robot vision, video surveillance, object-based compression, etc. Success has been declared in some limited cases, such as rigid objects or static cameras [1,2]. However, visual tracking remains a challenging and open problem in the case of non-rigid shape change, occlusion, rotation, scale change, complex scenes, etc.

To cope with such challenges, a number of elegant algorithms have been established. Generally, most tracking methods can be classified as either deterministic or probabilistic. Deterministic approaches solve an optimization problem under a prescribed cost function [3,4]. Probabilistic approaches estimate posterior distribution of the state of the object using a Kalman filter or particle filters [5–8].

Most of these approaches assume that the object-to-be-tracked has approximately a rectangular or elliptical shape. However, objects may have complex shapes, for example, hands, head and shoulders that cannot be well described by simple geometric shapes. Recently, some researchers [9,10] focused on providing accurate object segmentation instead of simple bounding boxes for tracking. The main issue of these methods is that the high computational complexity of performing figure/ground segmentation makes them unsuitable for real-time tracking. For example, Moreno-Noguer et al. [10] integrated multiple cues into a particle filtering framework and showed impressive results on challenging data sets. But the use of Fisher color space for segmentation made this algorithm require approximately a minute per frame for tracking.

In real world video sequences, rotation motion is another important factor to affect the tracking performance. A frequent measure to handle rotation is the particle filter method [7,8], which provides sufficient particles to estimate possible rotation states of the object. Another way is to rotate the prior model by specified amount and then the obtained model is matched to the image [11]. The optimal solution can be easily found by processing with all angles. However, this process will greatly increase the computational complexity and also make it unsuitable for real-time tracking applications.

In this paper, we propose an effective tracking framework, which is robust for tracking object with translation, rotation, scale change and partial occlusion. Furthermore, the proposed method has a low computational complexity and can be used for real-time tracking applications. Instead of directly providing an accurate contour solution, the first step of our method is to extract a rough object boundary. The simplified requirement of extracting rough boundary will decrease the computational complexity and make it easy to perform in cluttered background. The second step is to locate the object contour. To estimate the states of the object, including translation, rotation and scale, a fast algorithm of shape context matching is proposed. The estimated contour using shape context matching is very close to the object contour. To hold the model shape and get a more exact contour, we finally extract the exact boundary of the object by using elastic shape matching method.

The rest of the paper is organized as follows: Section 2 reviews related work. Section 3 describes the detailed tracking process steps of the proposed method, including figure/ground segmentation, extracting rough edge of the object, estimating the state

* Corresponding author.
E-mail address: zhaoyliu@sina.com (Z. Liu).

0925-2312/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2011.11.012

using shape context and extracting an exact contour. Experimental work is conducted in Section 4, and finally conclusions are given in Section 5.

2. Related work

This section mainly reviews the representative studies concerning the tracking techniques of active contour methods, figure/ground segmentation and shape matching.

The most popular contour tracking method is the active contour model, which was introduced by Kass et al. [12]. The objective of active contours is to get a tight contour enclosing the object. In early works [12,13], the contour energy is defined in the form of image gradient. However, active contours using image gradients are sensitive to noise and poor image contrast. To overcome these problems, some authors have incorporated region-based criteria into active contour models [14–18]. Yet edge-based and region-based methods cannot deal with occlusion problems or presence of strongly cluttered background. Therefore, prior shape knowledge has been integrated into contour representation to solve occlusion problem [19–21].

Recently, methods have become popular that treat object tracking as a segmentation problem [9,22–24]. Ren and Malik [9] employed figure/ground segmentation approach for object tracking. They repeatedly applied a conditional random field model of figure/ground segmentation, and obtained a figure mask in each frame. This process made tracking more robust and less susceptible to drifting. It has been successfully applied for tracking on long sports video with large variations in shape, appearance and scale. However, its high computational complexity makes it unsuitable for real-time tracking applications.

Kohli and Torr [22] proposed a fast graph cuts algorithm for video object tracking. This method can be used to perform real-time image segmentation in video sequences. However, it needs a clean object edge and is not suited for tracking object in cluttered background. Donoser and Bischof [23] proposed an efficient method of calculating color probability maps in an object-specific Fisher color space. This method provided robust tracking results and accurately segment object boundaries in short computation time. However, the method in [23] was mainly used for tracking object with similar color appearance.

For edge-based shape matching, Hausdorff distance [25] is the most widely used measure. However, Hausdorff measure using the maximum distance is sensitive to noise and occlusion. Hence, Park et al. [26] took a probabilistic Hausdorff measurement to locate the position of the object. This method produces good results under object shape deformation and is robust for partial occlusion. As the matching method using Hausdorff measure emphasize that parts of the edge map are not drastically affected by object motion, the method in Ref. [26] cannot deal with the case of rotation and scale change.

Quite recently, Schoenemann and Cremers [11] employed shape priors to find globally optimal solution for deformable shape. This method provides impressive results in the case of changing lighting condition and low contrast between object and background. However, for supporting rotation invariance, they sampled the prior shape of different angles and used these prior shapes for searching the most matched shape. This process greatly increases the computational complexity and is not suited for tracking object with rotation in real-time.

For supporting complete rotation invariance, in the shape context framework, Belongie et al. [27] suggested that one can use the tangent vector at each point as the positive x-axis instead of using the absolute frame for computing the shape context at each point. However, this reduces the discriminative power of the descriptor significantly. Especially for circular object, e.g. football, as the points on the circular edge have very similar feature, it is hard to find the corresponding points using shape contexts with relative coordinate. Since shape contexts are extremely rich descriptors, they are insensitive to small rotation of the shape. As almost all tracking algorithms assume that the object motion is smooth with no abrupt changes, it is unnecessary to use complete rotation invariance for object tracking.

In this paper, the purpose of figure/ground segmentation is to extract a rough boundary of the object. In contrast to other methods of providing exact segmentation, it is easy to perform in cluttered background. It will also be demonstrated that the proposed method using absolute frame for computing shape context is robust for tracking object with translation, rotation, scale change and partial occlusion.

3. Our method

3.1. Tracking framework

This section describes the detailed process of the proposed tracking method. The tracking framework mainly consists of four repeated steps. Firstly, the parameters of the Gaussian mixture models are computed using the known distributions of the object and background. Secondly, a coarse object contour is extracted using the edge pixel classification method [28]. Thirdly, shape context matching is used for object localization. Finally, an accurate contour is extracted using the elastic shape matching method [11]. Actually, the purpose of the first two steps is to extract a rough contour of the object and the purpose of the last two steps is to locate the position of the object and extract an exact contour. The entire algorithmic flowchart is illustrated in Fig. 1.

3.2. Object/background modeling

To evaluate the likelihood of each pixel belonging to the object or background, we take the Gaussian mixture model (GMM) to model the foreground and background. For the current frame I^t ($0 \le t < \infty$), the two mixture models $M(\Theta_O^t)$ and $M(\Theta_B^t)$ are learned from the previous frame I^{t-1}, where Θ_O^t and Θ_B^t denote the mixture parameters of the object and background in frame I^t, respectively. And it is supposed that the areas of the object and background of frame I^{t-1} are known. In the first frame of the video, the mixture models are calculated by manually designating the areas of the object and background. The parameters of the models are estimated by using the maximum likelihood method [29].

Let $z_i=(r,g,b)$ represent the pixel's color information. The likelihood of a pixel belonging to the object (O) or background (B) can be written as:

$$p(z|l) = \sum_{k=1}^{K_l} \omega_{l,k} G(z;\mu_{l,k},\Sigma_{l,k}) \tag{1}$$

where $l \in \{O,B\}$, representing foreground or background; the weight $\omega_{l,k}$ is the prior of the kth Gaussian component in the mixture model and fulfill $\sum_{k=1}^{K_l} \omega_{l,k} = 1$, and $G(z;\mu_{l,k},\sum_{l,k})$ is the kth Gaussian component as:

$$G(z;\mu_{l,k},\sum\nolimits_{l,k}) = \frac{1}{(2\pi)^{d/2}\left|\sum_{l,k}\right|^{1/2}} e^{-((z-\mu_{l,k})^T \sum_{l,k}^{-1}(z-\mu_{l,k}))/2} \tag{2}$$

where $d=3$ is the dimension of the GMM models, $\mu_{l,k}$ is the 3×1 mean vector and $\sum_{l,k}$ is the 3×3 covariance matrix of the kth component.

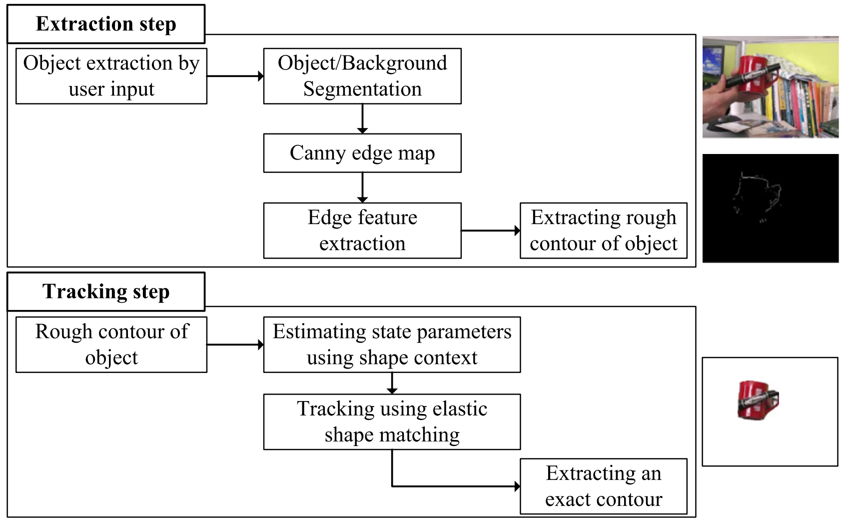

Fig. 1. The overview of the proposed method.

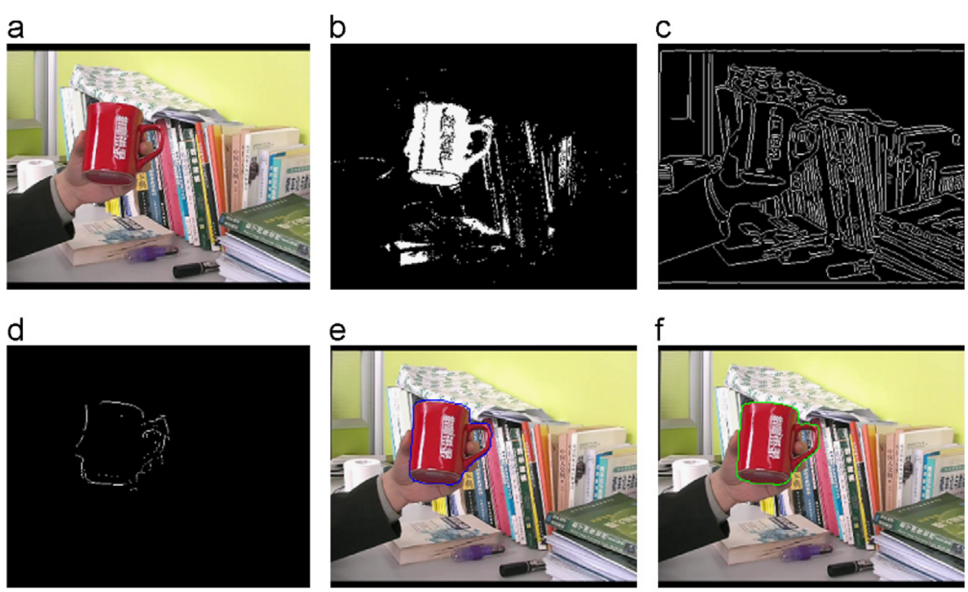

Fig. 2. The results of object/background modeling, edge pixel classification, shape context matching and elastic shape matching. (a) Input image. (b) Object/background classification ($p(z|O)>p(z|B)$). (c) Canny edge detection result. (d) Edge pixel classification. (e) Estimating the states of the object using shape context matching, including translation, rotation, scale change and partial occlusion. (f) Extracting the exact boundary.

As only color information is used for classification, the background noise cannot be fully eliminated. Fig. 2a is the original image. Fig. 2b shows the segmentation result using GMM models. The experimental results of the four steps are shown in Fig. 2.

3.3. Edge pixel classification

In Section 3.2, we have employed the GMM models to separate the object from the background, but the segmentation result is not ideal. To get a more accurate segmentation boundary, we extract the contour of the object by using edge pixel classification method which is similar to the method of [28].

To extract the boundary of the object, we first extract the edge by using edge detector (Fig. 3). Let $\vec{g}(u)=(g_x(u),g_y(u))$ denotes the spatial gradient at pixel u, $\theta(u)=\tan^{-1}(g_y(u)/g_x(u))$ is the direction of the gradient. Given the figure/ground segmentation result $I_{fg}(I_{fg}(u)=1$ if $p(z_u|O)>p(z_u|B)$, otherwise 0), we sample L pixels from point u along the direction $\theta(u)$. $N_\theta(u)$ denotes the number of the sampled object points. Similarly, $N_{\theta+\pi}(u)$ is the number of the object points which are sampled along the direction $\theta(u)+\pi$. Edge pixel u is regarded as object boundary pixel if

$$\min(N_\theta(u),N_{\theta+\pi}(u))<L/2 \quad \& \quad \max(N_\theta(u),N_{\theta+\pi}(u))>L/2 \tag{3}$$

The results of the edge pixel classification method are shown in Fig. 3 (third row). Although more complicated edge feature can be used for classification, this simple classifier is efficient and enough for our tracking framework. In this work, the parameter L is set to 10.

3.4. Fast shape context matching

Shape context was first proposed by Belongie et al. [30], and has been used for shape matching and object recognition [31,32]. The basic idea of shape contexts is shown in Fig. 4. A shape is represented by a discrete set of points sampled from the contours on the shape. These points can be on internal or external

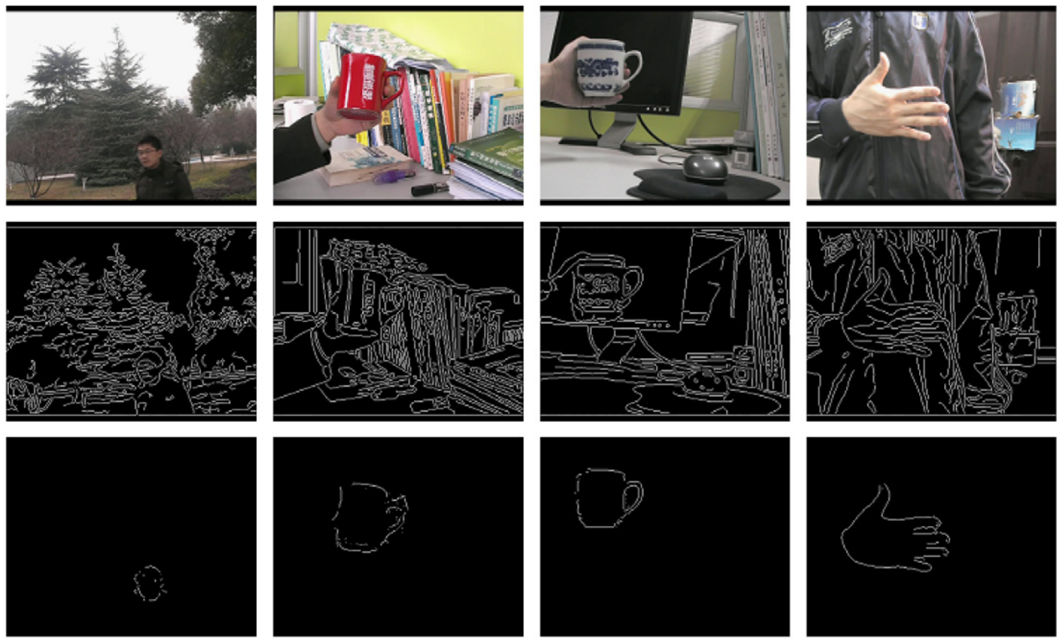

Fig. 3. Edge pixel extraction. Top row: first frame images of the tested videos in this paper. Middle row: canny edge detection results. Bottom row: edge pixel classification results.

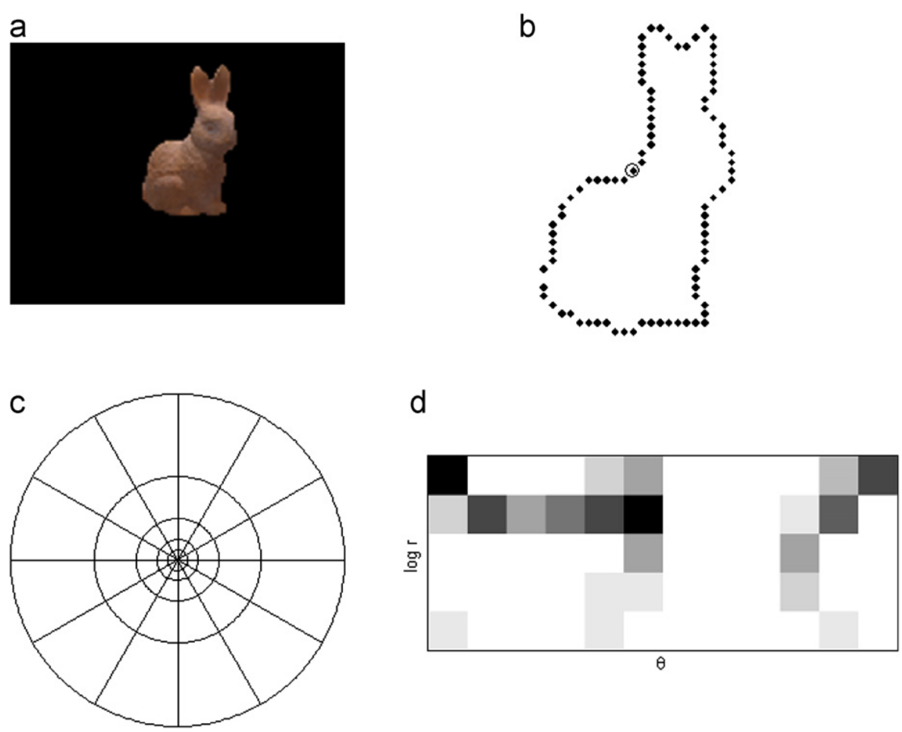

Fig. 4. Shape Context. (a) Object image. (b) Sampled edge points of a shape. (c) Diagram of log-polar histogram bins used in computing the shape contexts (5 bins for logr and 12 bins for θ). (d) Shape context for reference sample marked by $^\circ$.

contours. And each shape context is a log-polar histogram of the coordinates of the rest points measured using the reference point.

Suppose that the given shape contains n points. For a point p_i on the shape, the relevant histogram h_i of the relative coordinates of the remaining $n-1$ points can be computed as

$$h_i^k = \#\{q \neq p_i : (q-p_i) \in bin(k)\} \quad (4)$$

This histogram is defined to be the shape context of p_i. And the bins are uniform in log-polar space, which makes the descriptor more sensitive to differences in nearby pixels. Fig. 4c shows the example of the log-polar space.

Consider a point p_i on one shape and a point q_j on another shape. Let $C_{ij}=C(p_i,q_j)$ denote the cost of matching these two points. As shape contexts are distributions represented as histograms, the cost C_{ij} can be computed by using χ^2 test statistic:

$$C_{ij} = C(p_i, q_j) = \frac{1}{2}\sum_{k=1}^{K} \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)} \tag{5}$$

where $h_i(k)$ and $h_j(k)$ denote the K-bin normalized histogram at p_i and q_j, respectively.

Given the set of costs C_{ij} between all pairs of points p_i on one shape and q_j on another shape, Belongie et al. [31] found the matching by minimizing the total cost

$$H(\pi) = \sum_i C(p_i, q_{\pi(i)}), \tag{6}$$

where π is a permutation.

This is an instance of the square assignment problem, which can be solved in $O(N^3)$ time using the Hungarian method [33]. In [27], it has been solved by using the algorithm of [34]. However, this matching method has high computational complexity and cannot be used for real-time tracking. To lower the computational complexity, we take a simple and efficient method to find the matching results. For a point p_i on the first shape, the point q_j on the second shape is considered as its corresponding point, if

$$C_{ij} = \min_j C(p_i, q_j) \tag{7}$$

Given the matching point set $\{(p_i, q_{\pi(i)})\}$, let (x_i, y_i) and (x'_i, y'_i) be the coordinates of point p_i and $q_{\pi(i)}$, let r, θ and (x_c, y_c) be the scale, rotation and translation factors. Then, we have

$$r\begin{bmatrix} x_i & y_i \end{bmatrix}\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} + \begin{bmatrix} x_c & y_c \end{bmatrix} = \begin{bmatrix} x'_i & y'_i \end{bmatrix} \tag{8}$$

It can be described as

$$A_i\omega = B_i \tag{9}$$

where $A_i = \begin{bmatrix} x_i & y_i & 1 & 0; y_i & -x_i & 0 & 1 \end{bmatrix}$,
$\omega = \begin{bmatrix} r\cos\theta & r\sin\theta & x_c & y_c \end{bmatrix}^T$ and $B_i = \begin{bmatrix} x'_i & y'_i \end{bmatrix}^T$.

The solution ω can be obtained by solving the follow equation:

$$\min_{\omega} \|A\omega - B\|_2^2 \tag{10}$$

where $A = \begin{bmatrix} A_1^T & A_2^T & \cdots & A_{n-1}^T & A_n^T \end{bmatrix}^T$ and
$B = \begin{bmatrix} B_1^T & B_2^T & \cdots & B_{n-1}^T & B_n^T \end{bmatrix}^T$.

In order to handle the outliers robustly, we remove the pairs of the point that satisfy

$$\|A_i\omega^* - B_i\|_2^2 > \varepsilon_d \tag{11}$$

where ω^* is the solution of Eq. (10) and ε_d is a constant. After removing the outliers from A and B, we solve Eq. (10) again.

The proposed method is not the best way to handle outliers [27]. Given a relatively clear edge point set (Fig. 3 bottom row), however, it suffices for our purpose.

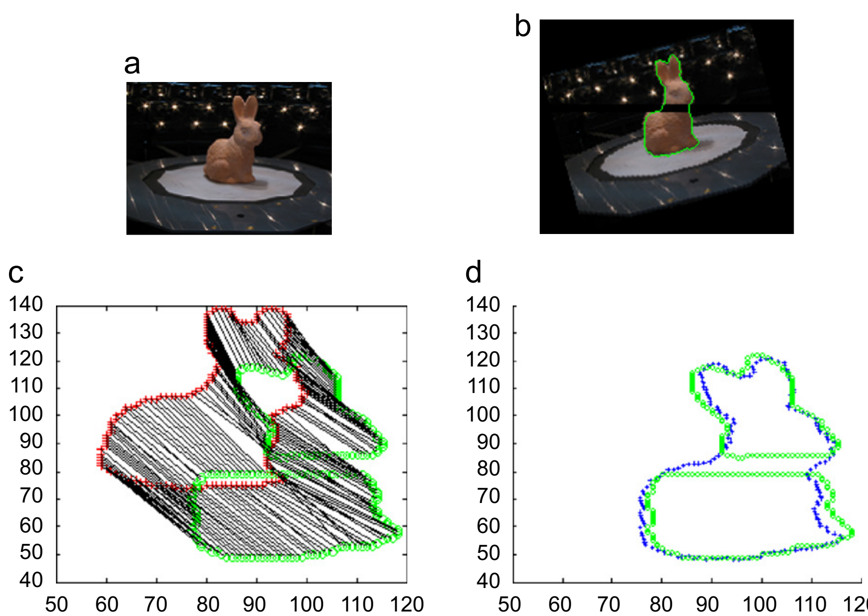

Fig. 5. Point correspondences using shape context. Shape context can be used to find corresponding points on similar shapes. (a) Original bunny image. (b) Bunny image with small rotation (10°) and partial occlusion. The green line is the final result. (c) Connections between corresponding points. Red contour is the model shape of (a) and the green contour is the test shape of (b). (d) Estimating the state parameters. The blue contour is the estimated contour. (For interpretation of the references to colour in this figure, the reader is referred to the web version of this article.)

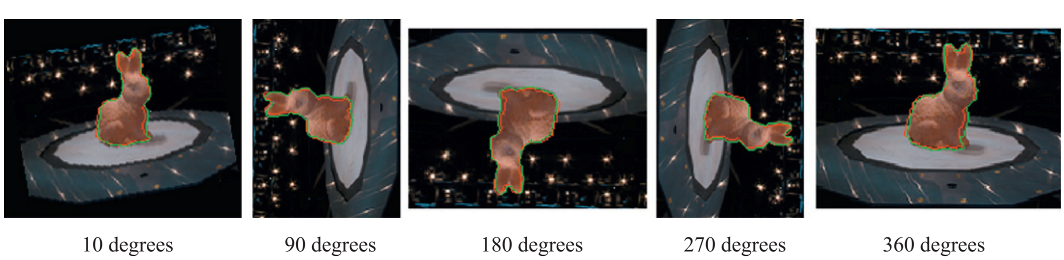

Fig. 6. Experimental results of tracking object with rotation. These images are obtained by rotating the original bunny image by specified amount. The red contours are the estimated results by using shape context matching. The greed contours are the final tracking results. (For interpretation of the references to colour in this figure, the reader is referred to the web version of this article.)

Since shape contexts are extremely rich descriptors, they are insensitive to small rotation and partial occlusion. As shown in Fig. 5, the result of point correspondences by using shape context is robust under small rotation and partial occlusion.

3.5. Elastic shape matching

To get a more accurate contour and hold the model shape, we employ the elastic shape matching method [11,35], which can find a globally optimal solution based on prior shape [11,36,37].

Given a contour S and an image I, the task is to find a contour C in the image that is located at image edges and similar to S. The optimal solution can be found by globally minimizing a ratio energy, which normalizes the integral of a data term and a shape consistency measure by the length of the desired contour. For real-time tracking performance in [11,35], the maximal velocity of D_{max} per frame is set to 15 and GPU implementation is need.

This method supports different amounts of invariance, including translational and rotational ones. However, for rotational invariance, it needs sufficient prior shapes of different angles

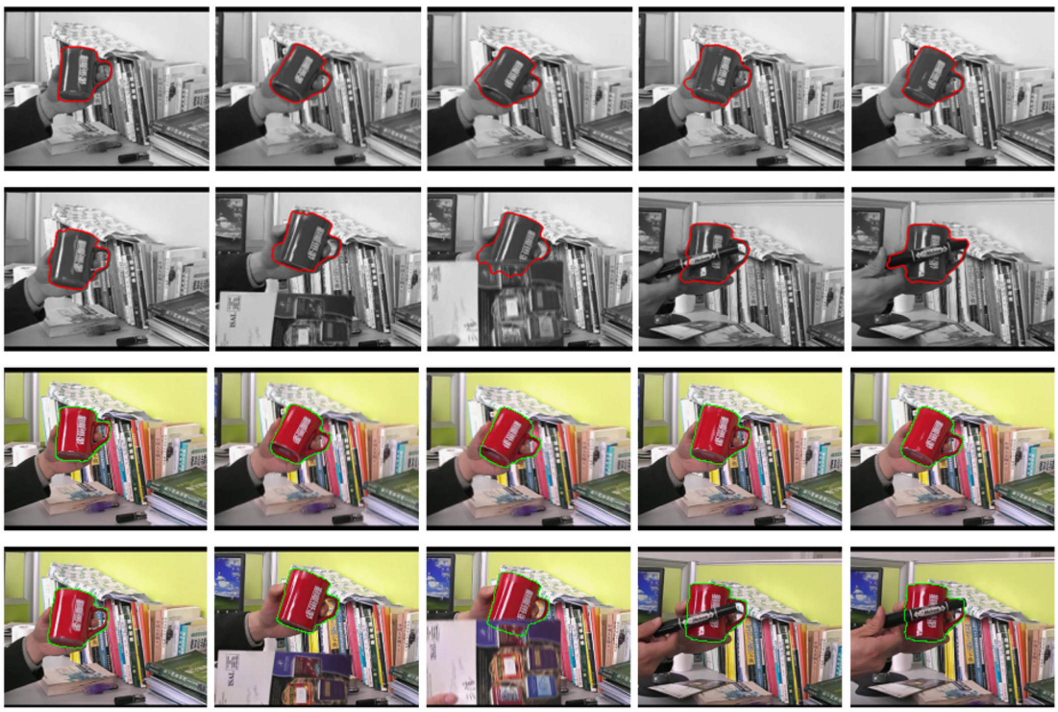

Fig. 7. Tracking cup with rotation and partial ocllusion. The first two rows are the results by using active contour method in [18]. The last two rows are the results by using the proposed method. From left to right and top to bottom, frames 1, 30, 40, 50, 60, 70, 100, 115, 150 and 155 are shown.

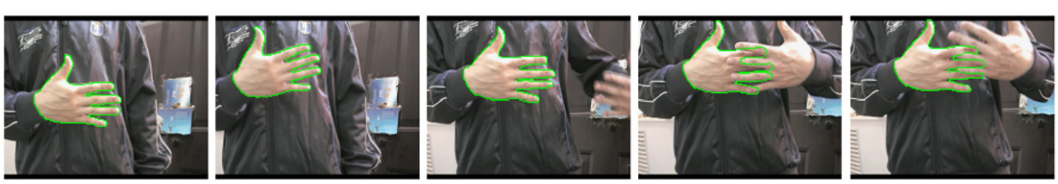

Fig. 8. Experimental results of tracking hand with rotation and occlusion. The frames 1, 40, 65, 105, 115 are shown.

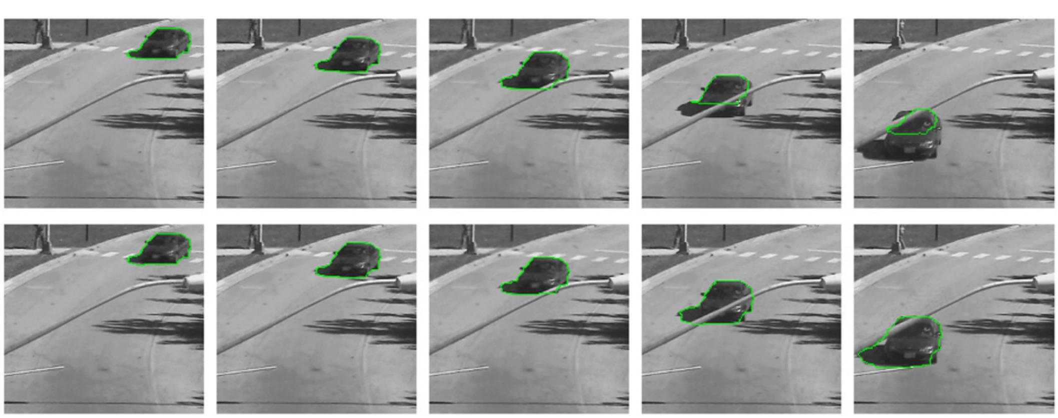

Fig. 9. Tracking a car through partial occlusion. The first row is the result by using the method of [11]. The second row is the result by using the proposed method. From left to right, frames 1, 13, 27, 46 and 68 are shown.

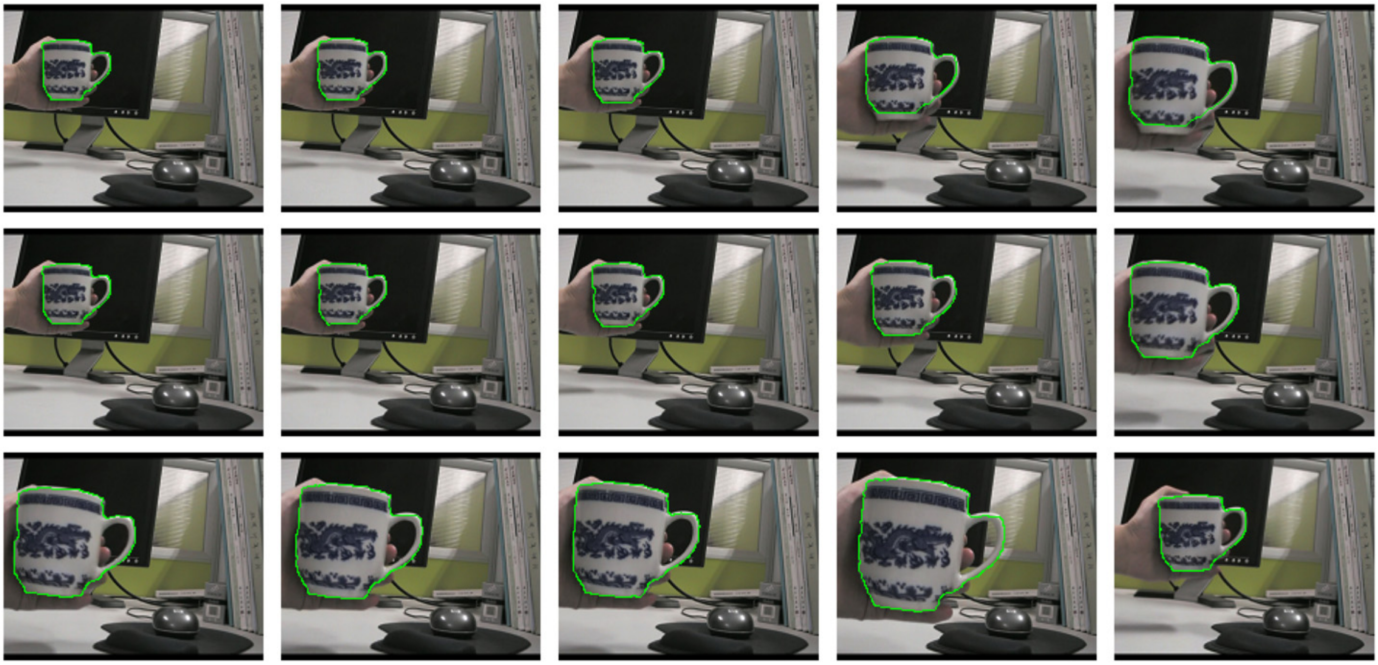

Fig. 10. Tracking cup with scale change. The first row shows the results by using the method of [11]. From left to right, frames 1, 10, 15, 31 and 35 are shown. The last two rows are the results by using the proposed method. From left to right and top to bottom, frames 1, 10, 15, 31, 35, 45, 65, 75, 95 and 120 are shown.

and these prior shapes can be sampled by rotating the original prior shape. The runtime of this process depends linearly on the number of sampled angle. This process will greatly increase the computational complexity and makes it not suited for tracking object with rotation in real-time.

In our tracking framework, the matched contour using shape context is very close to the object contour (Fig. 2e). Accordingly, we use the elastic shape matching method of [11] without considering velocity and rotation. The velocity parameter $D_{\max}$ of the elastic shape matching method in this work is set to 5. This makes our tracking approach still efficient with CPU implementation. The experimental results using the proposed tracking method are shown in Section 4.

4. Experimental results and analysis

We have applied the proposed tracking method on both synthetic data and several real world video sequences to evaluate the performance of our tracking scheme. Results on these sequences demonstrate that the proposed method is robust for tracking object with translation, rotation, scale change and partial occlusion. In addition, we compared our method with the active contour method [18] and the elastic shape matching method [11].

4.1. Tracking object with rotation

The test video sequence is a synthetic data. We got this sequence by rotating an bunny image [11] between 0° and 360° in steps of 10°. Fig. 6 shows the results by using our tracking framework. The red contours are the results using shape context and the green contours are the final tracking results. Compared with the method of [11], in which the authors get the prior shapes by sampling the rotation angle between -90° and $+90^\circ$ in steps of 2°, our method has lower computational complexity.

In other experiments, we have tested the proposed method on synthetic video sequences, which are formed by rotating the bunny image by specified amount. These similar experiments demonstrate that the proposed tracking method works well when the rotation angle of the object between two consecutive frames is no more than 10°.

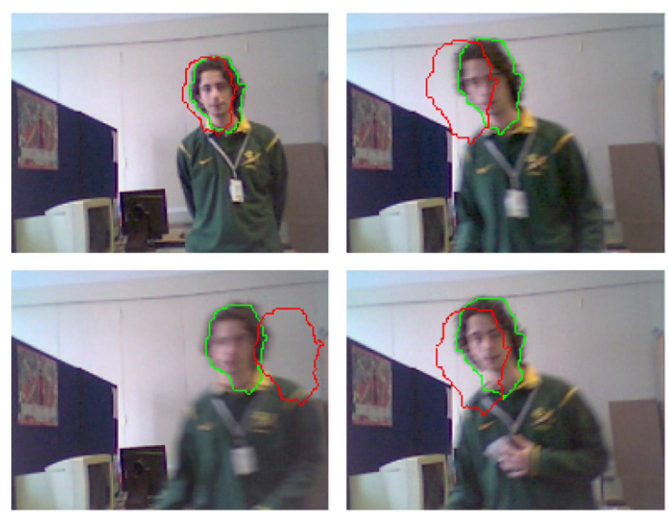

Fig. 11. Tracking fast moving objects. From left to right and top to bottom, frames 19, 42, 58 and 73 are shown.

4.2. Tracking object with rotation and occlusion

This video sequence (Fig. 7) mainly tests the performance of the proposed method for tracking with occlusion. When an object is occluded by a nearer object, then the tracking problem becomes harder. To test the performance of the proposed for handling occlusion, we define a simple measure, the occlusion rate *orate*(*f*) that denotes the percentage of the area occluded at frame *f*.

$$orate(f) = \frac{N_{occluded}(f)}{N_{object}(f)} \tag{12}$$

Fig. 12. Experimental results of tracking human head. From left to right and top to bottom, frames 1, 10, 20, 30, 40, 50, 60, 70, 80 and 105 are shown.

where $N_{occluded}(f)$ denotes the number of the occluded pixels and $N_{object}(f)$ denotes the number of the object pixels.

Some experiments have been done to test the performance of tracking ability with occlusion. These experiments demonstrate that the proposed method works well in the case that the occlusion rate $orate(f)$ is less than 25%.

We compared our method with the active contour method [18], which performed best in six different level-set methods by using the software [38] for the cup video sequence. The number of iteration is set to 200. In each frame, we used the estimated contour by shape context matching as the initial contour, which is close to the object boundary. The active contour method [18] brings imprecise results. However, our method tracks the object quite accurately under the condition of rotation and partial occlusion.

We also applied the proposed framework to hand tracking. In this video sequence (Fig. 8), the tracked hand is occluded by another hand. As the two hands have similar color and part of the tracked hand has been occluded, it is hard to track the object by using color appearance or contour evolution [12,23]. Experimental results on this sequence demonstrate that the proposed method is robust enough for tracking in this case. Another example for occlusion problem is shown in Fig. 9. A moving car is partially occluded by a street pole. The method of Ref. [11] exhibited gives poor results. However, the proposed method could successfully track the car.

4.3. Tracking object with scale change

In a natural setting, multiscale search could be performed, or scale-invariant interest point detection or segmentation could be used to estimate scale [32]. In this paper, we took the mean distance between points of the shape to estimate the scale [32]. Fig. 10 shows the result of tracking object with scale change. In the beginning, the cup is far away to the camera and then become close to the camera. At the end of the sequence it is far away from the camera. Compared with the elastic shape matching method [11], our method produced a more accurate result for tracking the cup with scale change.

4.4. Tracking fast moving object

Fig. 11 shows the results of the proposed method for tracking fast moving object. The maximal distance of the object between two consecutive frames is 52 (58 frame). The red line denotes the contour of the object from the previous frame and the green line represents the tracking result. The runtime of the method [11] depends on the size of the search window, fast moving object cannot yet be processed in real time. In contrast, the proposed method successfully tracks the object at 18 fps.

4.5. Tracking in cluttered background

We tested the proposed method for tracking object in a cluttered background (Fig. 12). In this video sequence, both the object and the camera are moving. In this case, we also obtained robust results using the proposed method.

4.6. Computational time

To obtain further speedups, several measures have been taken. In computing the parameters of the GMM, we used a relatively small number of data points and predefined the number of Gaussians (3 for object and 5 for background). In the shape context matching step, the number of the sampled points of the model shape was limited to 150 and the number of the sampled points in the test image was limited to 220. The experiments were performed on a 2.8 GHz PC. The images for video sequence are of resolution 288-by-360. The total processing for each frame took an average of 60 ms (cup and head sequences), which is acceptable for near real-time tracking.

5. Conclusions

In this paper, we introduced an approach for real-time tracking. The main idea is to employ shape context descriptor to locate the object contour. We have applied the proposed method on several synthetic and real world video sequences. Experimental results on these sequences demonstrate that the proposed method is robust enough for tracking objects with translation, rotation, scale change and partial occlusion.

The proposed tracking framework has some limitations, which we intend to overcome in future work. First, the use of fixed shape template may restrict the ability to track object with skew and aspect ratio changes. Second, the Gaussian mixture model performs poorly under severe and fast lighting condition changes. The online discriminative analysis method [39,40], which can effectively separate the foreground from the background under varying lighting condition, might overcome this problem.

Acknowledgements

The authors would like to thank the anonymous reviewers and the editor for their helpful comments and suggestions. This work is partially supported by the National Natural Science Foundation of China (60835005, 61003202, 61005084).

Appendix A. Supplementary material

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.neucom.2011.11.012.

References

[1] H. Yang, L. Shao, F. Zheng, L. Wang, Z. Song, Recent advances and trends in visual tracking: a review, Neurocomputing 74 (18) (2011) 3823–3831.
[2] A. Yilmaz, O. Javed, M. Shah, Object tracking: a survey, in: IPCV, (2006).
[3] V. Comaniciu, D. Ramesh, P. Meer, Kernel-based object tracking, IEEE Trans. Pattern Anal. Mach. Intell. 25 (5) (2003) 564–577.
[4] F. Porikli, O. Tuzel, P. Meer, Covariance tracking using model update based on lie algebra, Proc. IEEE Int. Conf. Comput. Vision Pattern Recognition 1 (2006) 725–735.
[5] T. Broida, S. Chandra, R. Chellappa, Recursive techniques for the estimation of 3d translation and rotation parameters from noisy image sequences, IEEE Trans. Aerosp. Electron. Syst. 26 (1990) 639–656.
[6] A. Doucet, N. Freitas, N. Gordon, Sequential Monte Carlo Methods in Practice, Springer Verlag, 2001.
[7] M. Isard, A. Blake, Condensation—conditional density propagation for visual tracking, Int. J. Comput. Vision 29 (1) (1998) 5–28.
[8] J. Kwon, K.M. Lee, F.C. Park, Visual tracking via geometric particle filtering on the affine group with optimal importance functions, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, (2009).
[9] X. Ren, J. Malik, Tracking as repeated figure/ground segmentation, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, (2007).
[10] F. Moreno-Noguer, A. Sanfeliu, D. Samaras, Dependent multiple cue integration for robust tracking, IEEE Trans. Pattern Anal. Mach. Intell. 30 (4) (2008) 670–685.
[11] T. Schoenemann, D. Cremers, A combinatorial solution for model-based image segmentation and real-time tracking, IEEE Trans. Pattern Anal. Mach. Intell. 32 (7) (2010) 1153–1164.
[12] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, Int. J. Comput. Vision 1 (4) (1988) 321–331.
[13] V. Caselles, R. Kimmel, G. Sapiro, Geodesic active contours, Int. J. Comput. Vision 22 (1) (1997) 61–79.
[14] S. Zhu, A. Yuille, Region competition: unifying snakes, region growing, and bayes/mdl for multiband image segmentation, IEEE Trans. Pattern Anal. Mach. Int. 18 (9) (1996) 884–900.
[15] A. Mansouri, Region tracking via level set pdes without motion computation, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 947–961.
[16] Y. Rathi, N. Vaswani, A. Tannenbaum, A. Yezzi, Tracking deforming objects using particle filtering for geometric active contours, IEEE Trans. Pattern Anal. Mach. Intell. 29 (8) (2007).
[17] T.F. Chan, L.A. Vese, Active contours without edges, IEEE Trans. Image Process. 10 (2) (2001) 266–276.
[18] S. Lankton, A. Tannenbaum, Localizing region-based active contours, IEEE Trans. Image Process. 17 (2008) 2029–2039.
[19] X. Bresson, P. Vandergheynst, J.-P. Thiran, A variational model for object segmentation using boundary information and shape prior driven by the Mumford–Shah functional, Int. J. Comput. Vision 68 (2) (2006) 145–162.
[20] M. Leventon, W. Grimson, O. Faugeras, Statistical shape influence in geodesic active contours, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, (2000), pp. 316–323.
[21] Y. Chen, H.D. Tagare, S. Thiruvenkadam, F. Huang, D. Wilson, K.S. Gopinath, R.W. Briggsand, E.A. Geiser, Using prior shapes in geometric active contours in a variational framework, Int. J. Comput. Vision 50 (3) (2002) 315–328.
[22] P. Kohli, P. Torr, Efficiently solving dynamic markov random fields using graph cuts, Int. Conf. Comput. Vision 2 (2005) 922–927.
[23] M. Donoser, H. Bischof, Fast non-rigid object boundary tracking, British Machine Vision Conference, (2008).
[24] Z. Yin, R.T. Collins, Shape constrained figure-ground segmentation and tracking, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, (2009).
[25] D. Huttenlocher, J. Noh, W. Rucklidge, Tracking nonrigid objects in complex scenes, International Conference on Computer Vision, (1993), pp. 93–101.
[26] S.-C. Park, S.-H. Lim, B.-K. Sin, S.-W. Lee, Tracking non-rigid objects using probabilistic hausdorff distance matching, Pattern Recognition 38 (12) (2005) 2373–2384.
[27] S. Belongie, J. Malik, J. Puzicha, Matching shapes, International Conference on Computer Vision, (2001), pp. 454–461.
[28] Z. Yin, R. Collins, Online figure-ground segmentation with edge pixel classification, British Machine Vision Conference, (2008).
[29] G. McLachlan, D. Peel, Finite Mixture Models, John Wiley and Sons, 2000.
[30] S. Belongie, J. Malik, Matching with shape context, IEEE Workshop on Context-based Access of Image and Video Libraries, (2000).
[31] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. Pattern Anal. Mach. Intell. 24 (4) (2002) 509–522.
[32] G. Mori, S. Belongie, J. Malik, Efficient shape matching using shape contexts, IEEE Trans. Pattern Anal. Mach. Intell. 27 (11) (2005) 1832–1837.
[33] C. Rapadimitriou, K. Stieglitz, Combinatorial Optimization: Algorithms and Complexity, Prenticice Hall, 1982.
[34] R. Jonker, A. Volgenant, A shortest augmenting path algorithm for dense and sparse linear assignment problems, Computing 38 (1987) 325–340.
[35] T. Schoenemann, D. Cremers, Globally optimal shape-based tracking in real-time, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, (2008).
[36] P.F. Felzenszwalb, Representation and detection of deformable shapes, IEEE Trans. Pattern Anal. Mach. Intell. 27 (2) (2005) 208–220.
[37] J. Coughlan, A. Yuille, C. English, D. Snow, Efficient deformable template detection and localization without user initialization, Comput. Vision Image Understanding 78 (3) (2000) 303–319.
[38] T. Dietenbeck, M. Alessandrini, D. Friboulet, O. Bernard, CREASEG: a free software for the evaluation of image segmentation algorithms based on level-set, Proceedings of the IEEE International Conference On Image Processing, (2010).
[39] R. Collins, Y. Liu, M. Leordeanu, On-line selection of discriminative tracking features, IEEE Trans. Pattern Anal. Mach. Intell. 27 (10) (2005) 1631–1643.
[40] S. Avidan, Ensemble tracking, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2) (2007) 261–271.

Zhao Liu received B.Sc. and M.Sc. degrees both from National University of Defense Technology in 2005 and 2007, respectively. He is presently a Ph.D. candidate in National University of Defense Technology. His research interests include image segmentation and video processing.

Hui Shen received B.Sc., M.Sc. and Ph.D. degrees from the National University of Defense Technology, China, in 1997, 2000 and 2003, respectively. He was promoted associate Professor in 2005. His research interests include neural image/signal processing, computer/biological vision, cognitive neuroscience. He has over 40 papers and 1 academic prize in the areas of his interests.

Guiyu Feng received B.Sc. and Ph.D. degrees from the National University of Defense Technology, China, in 2001 and 2007, respectively. Since 2008, he has been a Research Staff member at the Institute of Computing Technology, Beijing Jiaotong University. His research interests include biometrics, manifold learning, and biomedical image processing.

Dewen Hu received the B.Sc. and M.Sc. degrees from Xi'an Jiaotong University, China, in 1983 and 1986, respectively. From 1986, he was with the National University of Defense Technology. From October 1995 to October 1996, he was a Visiting Scholar with the University of Sheffield, UK. He got his Ph.D. degree from the National University of Defense Technology in 1999. He was promoted Professor in 1996. His research interests include image processing, system identification and control, neural networks and cognitive science. He is an action editor of Neural Networks.