

Research Article

Robust Individual-Cell/Object Tracking via PCANet Deep Network in Biomedicine and Computer Vision

**Bineng Zhong,¹ Shengnan Pan,¹ Cheng Wang,¹ Tian Wang,¹ Jixiang Du,¹
Duansheng Chen,¹ and Liujuan Cao²**

¹*Department of Computer Science and Engineering, Huaqiao University, Xiamen, Fujian Province 361021, China*

²*School of Information Science and Technology, Xiamen University, China*

Correspondence should be addressed to Bineng Zhong; bnzhong@hqu.edu.cn

Received 1 June 2016; Accepted 17 July 2016

Academic Editor: Qin Ma

Copyright © 2016 Bineng Zhong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Tracking individual-cell/object over time is important in understanding drug treatment effects on cancer cells and video surveillance. A fundamental problem of individual-cell/object tracking is to simultaneously address the cell/object appearance variations caused by intrinsic and extrinsic factors. In this paper, inspired by the architecture of deep learning, we propose a robust feature learning method for constructing discriminative appearance models without large-scale pretraining. Specifically, in the initial frames, an unsupervised method is firstly used to learn the abstract feature of a target by exploiting both classic principal component analysis (PCA) algorithms with recent deep learning representation architectures. We use learned PCA eigenvectors as filters and develop a novel algorithm to represent a target by composing of a PCA-based filter bank layer, a nonlinear layer, and a patch-based pooling layer, respectively. Then, based on the feature representation, a neural network with one hidden layer is trained in a supervised mode to construct a discriminative appearance model. Finally, to alleviate the tracker drifting problem, a sample update scheme is carefully designed to keep track of the most representative and diverse samples during tracking. We test the proposed tracking method on two standard individual cell/object tracking benchmarks to show our tracker's state-of-the-art performance.

1. Introduction

Individual-cell/object tracking is a fundamental problem in computational biology [1–4], drug treatment effects on cancer cells [2], high-content screening [5], and computer vision [6–8]. Therefore, it has attracted much attention due to the potential value for its theoretical challenges and practical applications. Although it has been investigated in the past decades, designing a robust cell/object tracker to cope with appearance changes of a cell/object is still a great challenging task. The appearance changes of a cell/object include intrinsic (e.g., pose changes, motion blur, scale variations, and nonrigid deformation) and extrinsic (e.g., illumination variations, cluttered scenes, and occlusions) factors. Such appearance changes may make a tracker drift away from the cell/object. Moreover, because a large number of manual

operations are required in existing cell tracking [9], how to design an accurate and automatic cell tracker with limited manual operations [10] is another challenge.

To capture appearance variations, most state-of-the-art trackers rely on handcrafted features to adaptively construct and update the generative or discriminative models of object appearances (e.g., principal component analysis [1, 2, 11, 12], Hough forest [13], support vector machine [14], and ensemble learning [15, 16]). By using various handcrafted features [16–26], these handcrafted feature-based tracking methods are developed for certain scenarios. Consequently, they are unable to capture the rich semantic information of a target as their generalization is not well. Therefore, they are prone to tracking failure in some challenging conditions.

Recently, deep learning [27–32] has attracted much attention in computational biology, cell biology, and computer

vision. Instead of using handcrafted features, deep learning aims to automatically learn hierarchical feature representation from raw data. With the impressive performance achieved by deep learning on speech recognition [28] and image recognition [29, 31, 32], a few of early researchers [33–39] have applied it to object tracking and achieved competitive performance. However, as only the annotation of a target object in the initial frames is available, the deep learning-based trackers usually use large-scale training data to prelearn deep structure and transfer the pretrained feature representation to the tracking tasks. Consequently, the large-scale pretraining is time-consuming and the pretrained feature representation may be less discriminative for tracking a specific cell/object. Moreover, they may be sensitive to partial occlusion and pose changes due to using a single global bounding box to delineate the entire cell/object.

In this paper, we propose a robust discriminative tracking method which automatically learns feature representation without large-scale pretraining and explicitly handles partial occlusion by fusing a global structure and local details in a cell/object. Specifically, in the initial frames, an unsupervised method is firstly used to learn the abstract feature of a cell/object by exploiting both classic principal component analysis (PCA) algorithms with recent deep learning representation architectures. We use learned PCA eigenvectors as filters and develop a novel algorithm to represent a target by composing of a PCA-based filter bank layer, a nonlinear layer, and a patch-based pooling layer, respectively. Then, based on the feature learned from the above unsupervised method, a neural network with one hidden layer is trained in a supervised mode to construct a discriminative target appearance model. By exploiting the advantage of deep learning architecture, our method is able to learn a generic and hierarchical feature representation while performing more efficiently without large-scale pretraining. Compared with holistic-based models, our method simultaneously maintains holistic and local appearance information and therefore provides a compact representation of the target object. Finally, to alleviate the tracker drifting problem, a simple yet effective sample update scheme is adopted to keep track of the most representative and diverse samples while tracking. The experiments on two standard individual-cell/object tracking benchmarks (i.e., the Mitocheck cell dataset [40] and the online tracking benchmark (OTB) [41]) show that our tracker achieves a promising performance.

The rest of the paper is organized as follows. Section 2 discusses the most related work to ours. The detailed overall framework of our tracking method is described in Section 3. The performance of our tracking method is demonstrated in Section 4. Finally, Section 5 summarizes our findings.

2. Related Work

Much work has been done in the area of cell/object tracking and the comprehensive review is beyond the scope of this paper. Please refer to [3, 4, 6–8] for more complete reviews on cell/object tracking and recent tracking benchmarks. In this section we briefly review some representative works on visual tracking and put our work in a proper context.

2.1. Individual-Cell and Object Tracking with Handcrafted Features. For decades, many tracking methods with handcrafted features have been proposed, which focus on constructing robust cell/object appearance models to handle the inevitable appearance changes of a cell/object. In [17], a mean shift-based tracking method using color histograms is proposed. Li et al. [18] propose a multiple nuclei tracking method with the intensity features for quantitative cancer cell cycle analysis. In [19], Danelljan et al. propose an adaptive color attribute-based tracking method under a coloration filtering framework. In [26], Lou et al. propose an active structured learning-based cell tracking method by combining multiple complementary features, such as position, intensity, and shape. In [12], an incremental principal component analysis-based tracking method is proposed for robust visual tracking. Recently, a variety of low-rank subspaces and sparse representations based tracking methods have been proposed [42–47] for cell/object tracking due to their robustness to occlusion and image noises. Zhong et al. [48] propose a weakly supervised learning-based tracking method, in which multiple complementary trackers are effectively fused to achieve robust tracking results. Zhou et al. [49] propose a similarity fusion-based tracking method, in which multiple features and context structure of unlabeled data are effectively utilized.

Coupled with designing handcrafted features, numerous advanced machine learning methods have been developed to further improve the tracking performances. The typical learning methods include support vector machine (SVM) classifiers [14], structured output SVM [21], online boosting [15, 20], P-N learning [50], multiple instance learning [51], and correlation filters [52–54]. In [55], for improving the tracking performance, Lou et al. incorporate a shape prior into a learning method to segment dense cell nuclei. Dzyubachyk et al. [56] utilize a level set-based method for cell tracking in time-lapse fluorescence microscopy.

Moreover, to explicitly deal with the occlusion problem, several part-based models have been proposed. In [13], Gall et al. propose a part-based voting schema via Hough forests for robust tracking. In [17], online latent structural learning is employed for a part-based object tracking method. However, the part-based tracking methods still rely on low-level features. Although tracking methods with handcrafted features usually produce more accurate results under less complex environments, they may be limited by using handcrafted features which cannot be simply adapted according to the new observed data obtained while tracking.

2.2. Single-Cell and Object Tracking with Deep Learning. Inspired by the success of deep learning in speech and visual recognition tasks [27–32], a few of deep learning-based tracking methods have been recently proposed [33–39] for robust cell/object tracking. In [35], based on a pretrained convolutional neural network, Fan et al. propose a tracking method for human. One of the limitations is that the pretrained convolutional neural network is fixed during the online tracking process. Wang and Yeung [34] propose an autoencoder based tracking method. Instead of using unrelated images

for pretraining, Wang et al. [57] propose a tracking method which prelearns features robust to diverse motion patterns from auxiliary video sequences. However, they only evaluate the method on 10 video sequences. In [58], Li et al. effectively combine multiple convolutional neural networks for robust tracking. Within a particle filtering framework, Carneiro and Nascimento [33] use deep learning architectures to cope with the left ventricle endocardium in ultrasound data. In [36], based on the deep network of VGG, a fully convolutional neural network is proposed for robust tracking. In [37], Hong et al. propose a tracking method by learning discriminative saliency map with convolutional neural network. In [38], Ma et al. fuse the correlation filters and pretrained VGG network for robust tracking. In [39], Nam and Han propose a multidomain convolutional neural network-based tracking method.

However, these tracking methods are time-consuming due to the large-scale pretraining. Moreover, the pretrained feature representation may be less discriminative for tracking specific target objects.

3. The Proposed Individual-Cell and Object Tracking Algorithm

In this section, we develop our discriminative tracking algorithm via a PCANet deep network [32]. Based on a particle filtering framework, the proposed PCANet-based tracking method for individual-cell/object is schematically shown in Algorithm 1.

Specifically, the proposed tracking algorithm works as follows: the target object is manually selected in the first frame by a bounding box. Then, an unsupervised method is used to learn the abstract feature of the target object by exploiting both classic principal component analysis (PCA) algorithms with recent deep learning representation architectures. Furthermore, based on the feature learned from the above unsupervised method, a neural network with one hidden layer is trained in a supervised mode to construct a discriminative object appearance model. Meanwhile, a set of particles with associated weights is initialized within a particle filtering framework. For one incoming video frame t , we first predict each particle using the dynamic model. Then, we compute weights for each particle using the observation model (i.e., the discriminative appearance model). According to the obtained weights, we determine the optimal object state as the particle with the maximum weight and resample particles. Finally, the pretrained feature is updated according to the new observed data. Meanwhile, the discriminative appearance model is also incrementally updated via a simple yet effective sample update scheme which keeps track of the most representative and diverse samples while tracking. The tracking procedure continues in this iterative fashion until the end of video.

Below we give a detailed description about each component of our method.

Algorithm 1. Overview of the proposed PCANet-based tracking method for individual-cell/object is shown below.

Input is as follows:

- (1) Get one initialized video frame with ground-truth bounding box on a cell/object.
- (2) Pretrain an abstract feature of a cell/object via an unsupervised method.
- (3) Build a neural network-based discriminative appearance model for the cell/object based on the feature learned from the above unsupervised method.
- (4) Initialize a set of particles with associated weights within a particle filtering framework.

Output is as follows:

- (1) Predict each particle using a Gaussian function-based motion model.
- (2) Compute weights for each particle using a PCANet-based discriminative appearance model.
- (3) Determine the optimal cell/object state as the particle with the maximum weight.
- (4) Resample particles based on their corresponding weights.
- (5) Update the pretrained feature and the PCANet-based discriminative appearance model according to the newly observed data.

3.1. Particle Filtering. The proposed tracking algorithm is carried out using the particle filtering framework which is a Markov model with hidden state variables. Supposing that we have observations of the target object $Z_t = [z_1, \dots, z_t]$ up to the t th frame, the hidden state variable x_t is estimated by the well-known two-step iteration (i.e., the prediction and the update steps):

$$p(x_t | Z_t) \propto p(z_t | x_t) \int p(x_t | x_{t-1}) p(x_{t-1} | Z_{t-1}) dx_{t-1}, \quad (1)$$

where $p(x_t | x_{t-1})$ is the dynamic (motion) model between two consecutive states and $p(z_t | x_t)$ is the observation model which estimates the likelihood of observing z_t at state x_t . The optimal object state x_t^* at time t can be determined by the maximum a posteriori estimation over N samples (particles) at the t th frame by

$$x_t^* = \arg \max_{x_t^i} \{p(z_t^i | x_t^i) p(x_t^i | x_{t-1}^i)\}, \quad (2)$$

where x_t^i is the i th sample of the state x_t and z_t^i is the image observation predicted by x_t^i .

Motion Estimation. In this paper, for simplicity and computational efficiency reasons, we choose to track only the location and size. Let $x_t = (l_t^x, l_t^y, w_t, h_t)$ denote the object state parameters including the horizontal coordinate, vertical coordinate, width, and height, respectively. We use a Gaussian

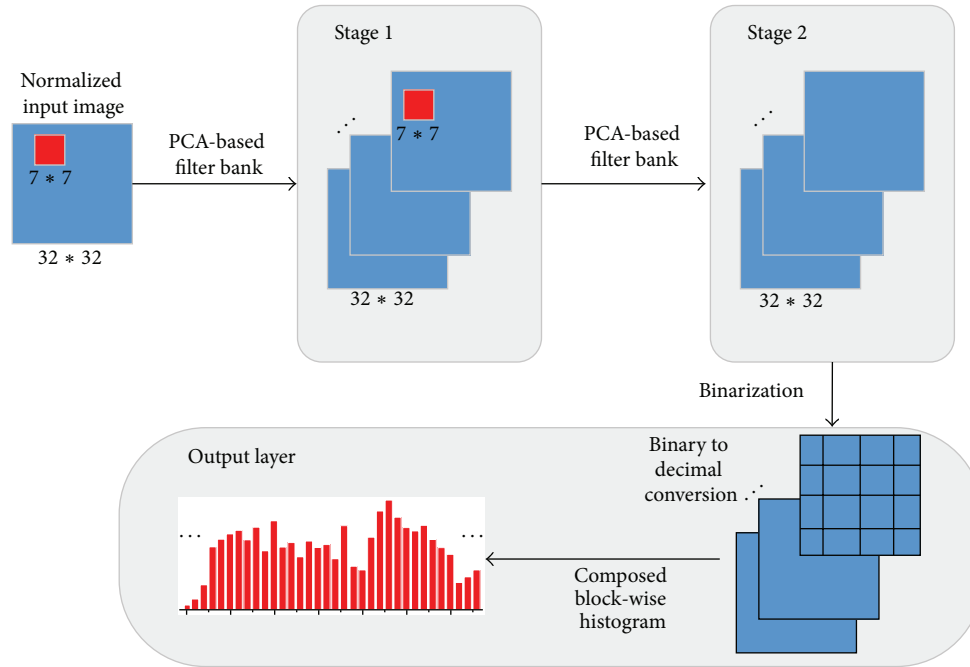


FIGURE 1: Illustration of the structure of the used PCANet deep network [32].

distribution to model the dynamic model between two consecutive frames.

Likelihood Evaluation. For each state x_t , there is a corresponding image patch that is normalized to 32×32 pixels by image scaling. The likelihood function is calculated based on the proposed discriminative appearance model; that is, $p(z_t | x_t) = d_t$, where d_t is an output score from the proposed discriminative appearance model.

3.2. The Proposed Discriminative Appearance Model from PCANet. In this section, we address the problem of how to learn a data-driven and discriminative appearance model without large-scale pretraining. In the first frame, an unsupervised method is firstly used to learn the abstract feature of a target object by exploiting both classic principal component analysis (PCA) algorithms with recent deep learning representation architectures. Then, based on the feature learned from the above unsupervised method, a neural network with one hidden layer is trained in a supervised mode to construct a discriminative target appearance model.

More specifically, we use the newly proposed PCANet deep network [32] to prelearn the abstract feature of a target object. The PCANet is a simple convolutional deep learning network composed of cascaded PCA, binary hashing, and block histograms. The work on PCANet shows that applying arbitrary nonlinearities on top of PCA projections of image patches can be surprisingly effective for image classification. Inspired by their work, we propose a PCANet-based unsupervised method to effectively learn the abstract feature of a target object and the discriminative structure between the target and background.

The PCANet model is illustrated in Figure 1, and only the PCA filters need to be learned from the training images. Following the notations of Han Chan et al. [32], we will briefly review the PCANet model.

The Cascaded PCA. Denote $\{I_i \in \mathbb{R}^{m \times n}\}_{i=1}^N$ as N input training images and $k_1 \times k_2$ as the 2D convolutional filter size. Around each pixel, PCANet takes $k_1 \times k_2$ patch and collects all (overlapping) patches of the i th image as the training data. Then, PCANet computes projection vectors in such a way that most variations in the training data can be retained. The PCA filters in the PCANet are expressed as the leading principal eigenvectors. Similar to deep neural network, PCANet can stack multiple stages of PCA filters to extract higher level features.

Binary Hashing and Block Histograms. Let L_1 and L_2 denote the number of PCA filters in the first and second stage of PCANet, respectively. For each of the L_1 input images I_i^l for the second state, each input image has L_2 real-valued outputs $\{I_i^l * W_\ell^2\}_{\ell=1}^{L_2}$ from the second stage. These outputs are binarized via a hashing function, in which an output value is one for positive entries and zero otherwise.

Around each pixel, the vector of L_2 binary bits is viewed as a decimal number. This converts L_2 outputs of the i th input image I_i^l back into a single integer-valued image T_i^l , where $i = 1, \dots, L_1$. Then, each of the L_1 images T_i^l is divided into m overlapping or nonoverlapping blocks. PCANet compute the histogram of the decimal values in each block and concatenate all m histograms into one vector and denote them as $H(T_i^l)$.

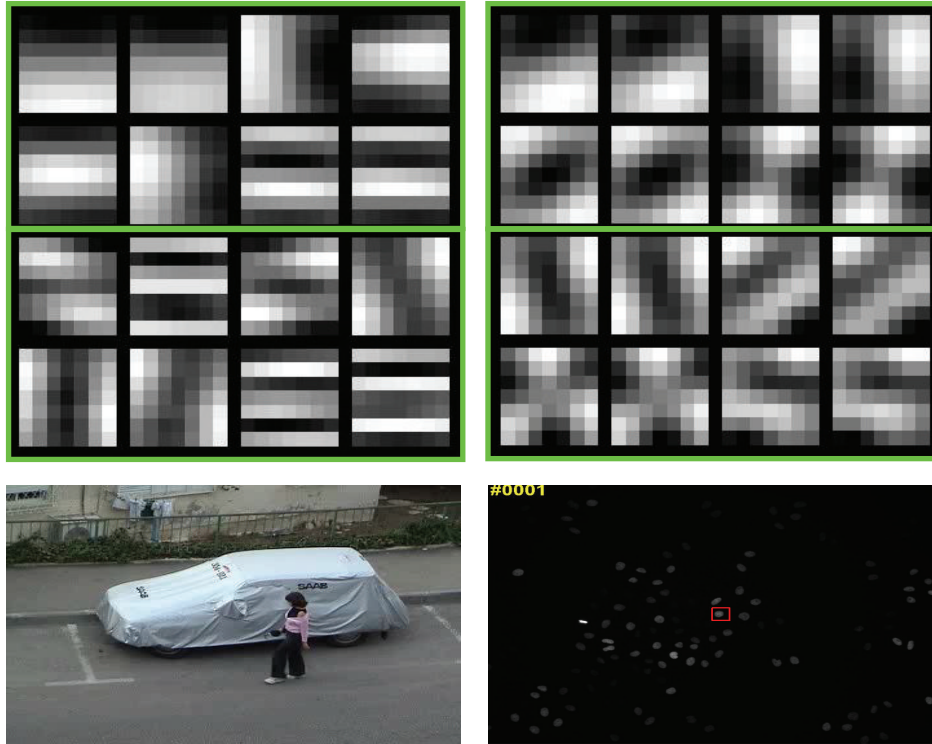


FIGURE 2: The PCA-based filters learned on the training data from the first frame of woman sequence from the online tracking benchmark (OTB) [41] and the Mitocheck cell dataset, respectively [40]. The top two rows show the eight PCA-based $7 * 7$ filters learned in first layer. The bottom two rows show the eight PCA-based $7 * 7$ filters in second layer.

After this encoding process, the feature of the input image I_i is then defined to be the set of block-wise histograms; that is, $f_i = [H(T_i^L), \dots, H(T_i^{L_1})]$.

To empirically illustrate the efficacy of the learned PCANet features, we check the fine-tuned filters trained on the training data from a specific tracking task. In Figure 2, we show the PCA-based filters learned on the training data from the first frame of woman sequence from the online tracking benchmark (OTB) [41] and the Mitocheck cell dataset, respectively [40]. The top two rows show the eight PCA-based $7 * 7$ filters learned in first layer. The bottom two rows show the eight PCA-based $7 * 7$ filters in second layer. It is obvious that the proposed PCANet-based model can effectively learn the useful information from the data, such as edge and corner and junction detectors.

4. Experiments Evaluation

This section presents our implemental details, experimental configurations, dataset, and evaluation setting. The effectiveness of our tracking algorithm (named ours-1) is then demonstrated by quantitative and qualitative analysis on the online tracking benchmark (OTB) [41] and the Mitocheck cell dataset, respectively [40]. For the sake of computational robustness, we further consider the effect of the different PCA layers in PCANet (i.e., a variety of different numbers of the PCA layers) on the tracking performance.

4.1. Implementation Details and Experimental Configurations.

To reduce computational cost, we simply consider the object state information in 2D translation and scaling in a particle filtering framework, where the corresponding variance parameters are set to 15, 15, 0.1, and 0.1, respectively. The proposed tracking method (i.e., ours-1) is implemented in Matlab without code optimization and runs on a PC with a 2.40 GHz processor and 12 G RAM. 1,000 samples are empirically drawn for particle filtering. For each particle, there is a corresponding image region normalized to a $32 * 32$ patch. The buffer size of a temporal sliding window is set as 25. The typical training time of PCANet-based deep network is about 10 seconds in Matlab without using GPUs. Our PCANet-based tracker takes about one second to process each video frame.

4.2. Datasets and Evaluation Settings

4.2.1. *Datasets.* To evaluate the performance of the proposed tracking method (i.e., ours-1) for tracking individual-cell/object, we use not only the Mitocheck cell dataset [40] but also the online tracking benchmark (OTB) [41]. The Mitocheck dataset is a time-lapse microscopic image sequence which contains higher cell density, larger intensity variability, and illumination variations. The online tracking benchmark (OTB) [41] is a collection of 50 video sequences tagged with 11 attributes which covers various challenging factors in visual tracking, such as deformation, fast motion,

background clutter, and occlusion. The 50 video sequences are defined with bounding box annotations.

4.2.2. Evaluation Settings. The OTB benchmark uses two different evaluation metrics: the precision plot and success plot. For the precision plot, a target object is considered to be successfully tracked on a video frame if the distance between the centers of the estimated box and the ground-truth bounding box is below a threshold. Thus, numerous precision plots can be obtained by varying the threshold values. Typically, the trackers are ranked based on the precision at threshold of 20 pixels for the precision plot. On the other hand, for the success plot, a target object is considered to be successfully located on a video frame if the predicted bounding box and the ground-truth bounding box have an intersection-over-union (IoU) overlap higher than a threshold. The success plot illustrates the percentage of frames considered to be successful. The area under curve (AUC) score is used to rank the tracking algorithms. Three different experiments are performed, that is, one-pass evaluation (OPE), temporal robustness evaluation (TRE), and spatial robustness evaluation (SRE). For TRE, the starting frame of the evaluation is randomized. For SRE, the initial bounding boxes are randomly perturbed. Please see the original paper [41] for more details. For the evaluation on the Mitocheck cell dataset [40], we just use the qualitative results to show the tracking performance due to the unavailability of ground-truth labeling.

4.3. Evaluation Results on the Online Tracking Benchmark (OTB)

Overall Performance. We quantitatively analysed the overall tracking performance, and Figure 3 shows the precision and success plots on all the 50 sequences of the top 10 tracking methods. In terms of both evaluation metrics, the proposed tracking method (i.e., ours-1) is able to obtain better results than any of the comparison methods due to the robust feature learning via online PCANet deep network. In the precision plot of OPE, the precision score of the proposed tracking method (i.e., ours-1) is 0.707, which is ranked the first place. Meanwhile, the other top four tracking methods are Struck (0.656), SCM (0.649), TLD (0.608), and VTD (0.576), respectively. In the success plot of OPE, the AUC score of the proposed tracking method (i.e., ours-1) is 0.566, which is also ranked the first place. Meanwhile, the other top four tracking methods are SCM (0.499), Struck (0.474), TLD (0.437), and ASLA (0.434), respectively. According to the precision and AUC scores, the proposed tracking method (i.e., ours-1) is comparable to the state-of-the-art tracking methods in both the precision and success plots.

Performance Analysis on 11 Different Attributes. To further analyse the proposed tracking method, we validate the performance of the proposed tracker on each attribute provided in the online tracking benchmark (OTB) [41]. In the OTB, there are 11 different attributes which describe a variety of tracking challenges. Each video sequence is annotated by

some attributes. We report the precision and success plots of one-pass evaluation (OPE) for trackers on the 11 attributes in Figures 4 and 5, respectively. According to Figures 4 and 5, it is easy to observe that the proposed tracking method (i.e., ours-1) provides sufficient robustness to the 11 attributes, and our tracker consistently outperforms the other trackers in most of the challenges.

Qualitative Results. In Figure 6, we illustrate the qualitative results of four typical image sequences. To facilitate more detailed analysis, we further report the curves of center distance error per frame in Figure 7. As our tracker can better capture major variations in the data, we can observe that the proposed tracking method demonstrates superior performance over other tracking methods.

4.4. Effect of Different PCA Layers in PCANet. In this subsection, we investigate how the number of PCA layers in PCANet affects the tracking performance of the proposed method. Specifically, we compare our tracker (i.e., ours-1) with one different structure. The new variation of ours-1 is denoted as ours-2. Different to ours-1 tracker which contains two PCA filtering layers, ours-2 tracker contains three PCA filtering layers. Figure 8 demonstrates the performance comparison of the proposed tracking method with different PCA layers in PCANet in terms of the success and precision plots of TRE on the online tracking benchmark (OTB) [41]. We observe that ours-2 tracker with three PCA filtering layers obtains a better result than that of ours-1 tracker with two PCA filtering layers. This indicates that the performance of the proposed tracking method can be further improved when the number of PCA layers in PCANet is increased. However, the improvement is not significant and is computationally inefficient.

4.5. Qualitative Results on the Mitocheck Cell Dataset. To evaluate the performance of the proposed tracking method on individual-cell tracking, we test the proposed tracking method on the Mitocheck cell dataset [40]. In Figure 9, we report the qualitative tracking results of four individual-cells from the Mitocheck dataset. We can observe that the proposed tracking method simultaneously maintains holistic and local appearance information and therefore provides a compact representation of the cells. Consequently, the proposed tracking method can achieve a good performance on individual-cell tracking.

4.6. Discussion. In this paper, we focus on learning a robust PCANet-based appearance model for individual-cell/object tracking. According to the above experimental results on challenging dataset, the proposed tracking method has achieved promising results. However, the performance of the proposed tracker may be deteriorated when a target object is occluded over a long period of time. The reason is that the PCANet-based appearance model is updated via a simple yet concrete schema which does not explicitly detect occlusion. To address the problem, more complicated occlusion detection and forgetting schemas should be incorporated into the proposed tracker to achieve effective model updating.

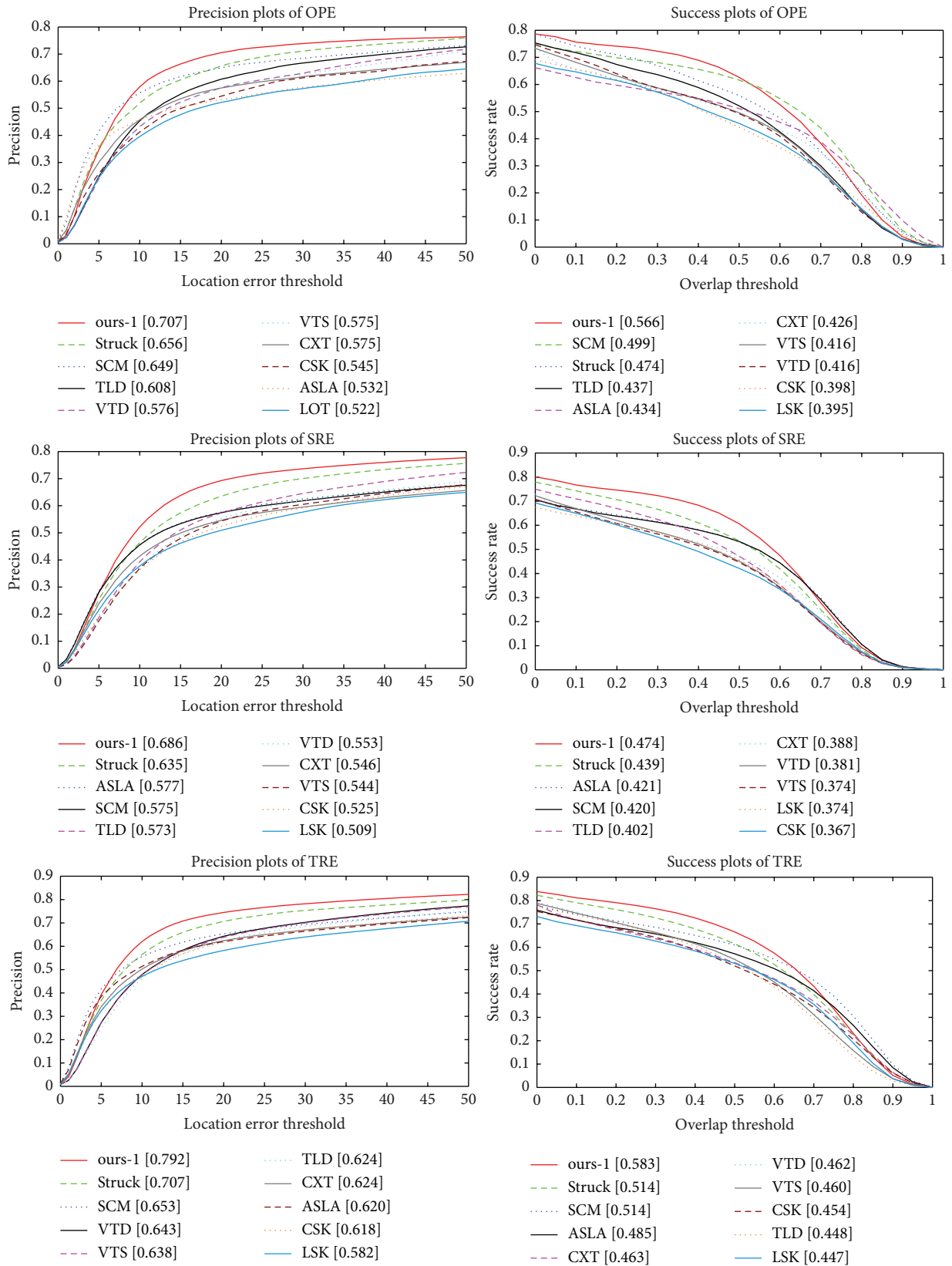


FIGURE 3: The precision and success plots of one-pass evaluation (OPE), temporal robustness evaluation (TRE), and spatial robustness evaluation (SRE) for the 50 sequences in the online tracking benchmark (OTB) [41], respectively. The legend lists the corresponding evaluation score for each tracking method. The proposed tracking method (i.e., ours-1 in red) is ranked first among the state-of-the-art trackers in both the precision and success plots.

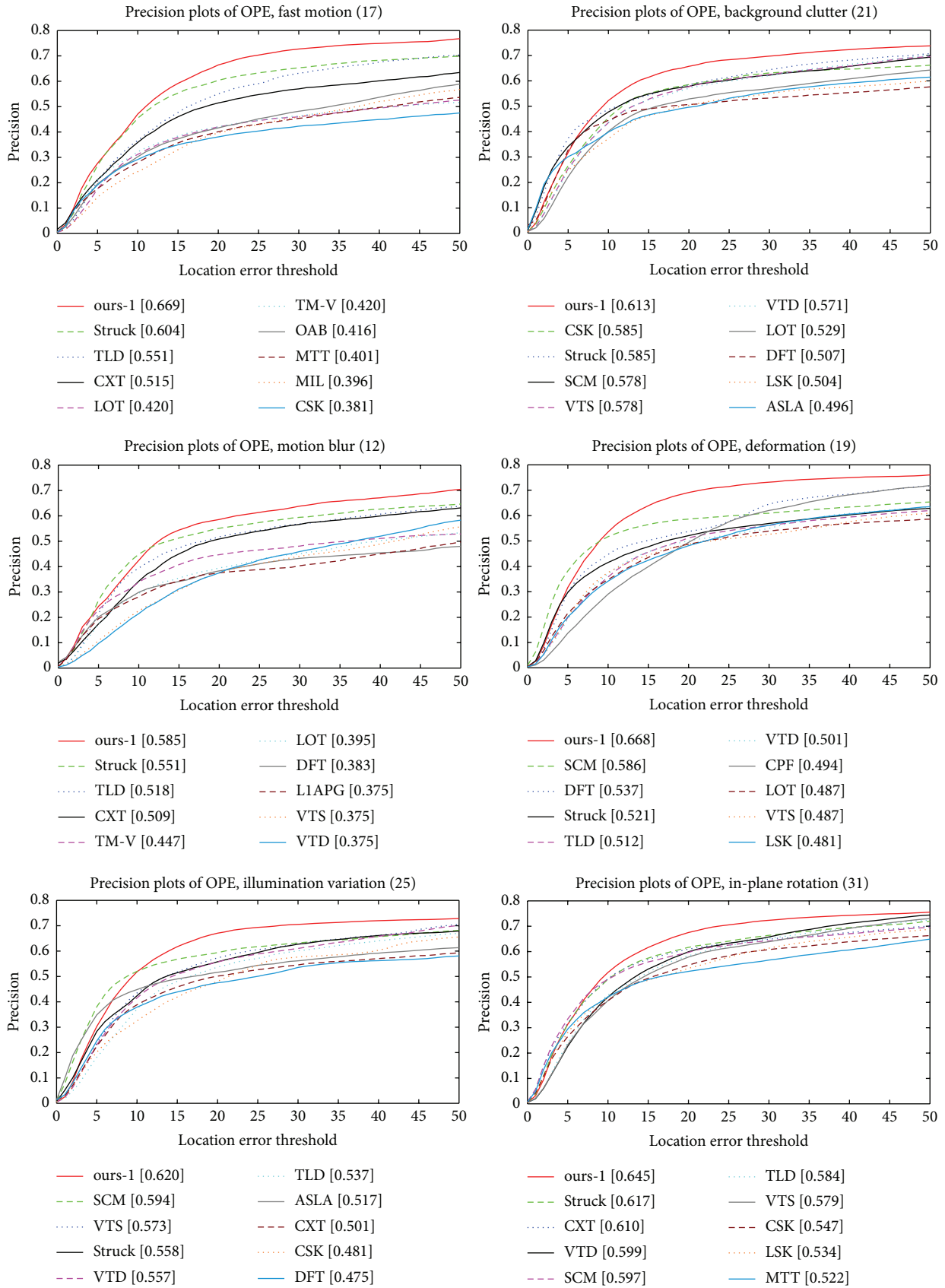


FIGURE 4: Continued.

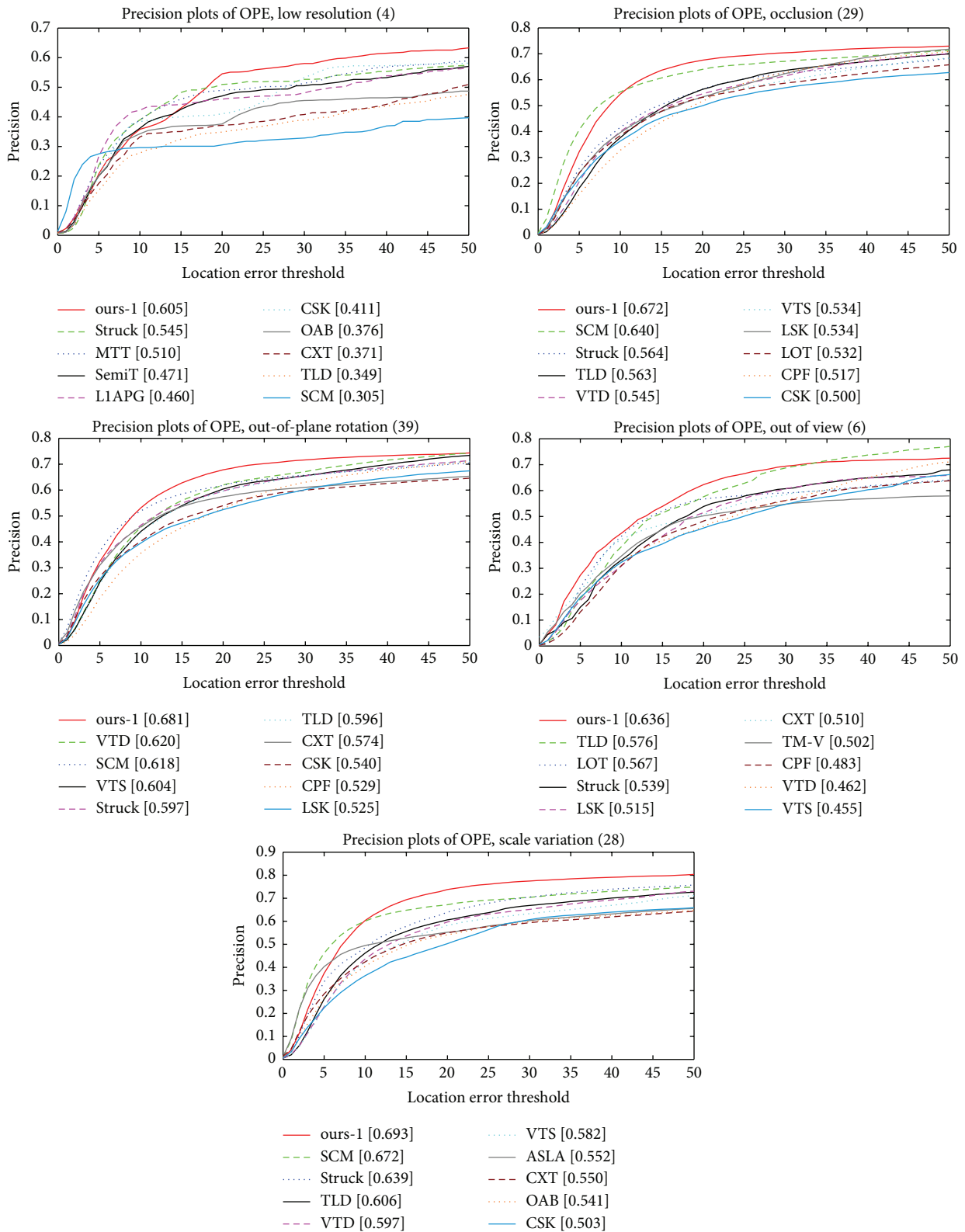


FIGURE 4: The precision plots of one-pass evaluation (OPE) for trackers on the 11 attributes. The values next to the attributes denote the number of video sequences involving the corresponding attribute.

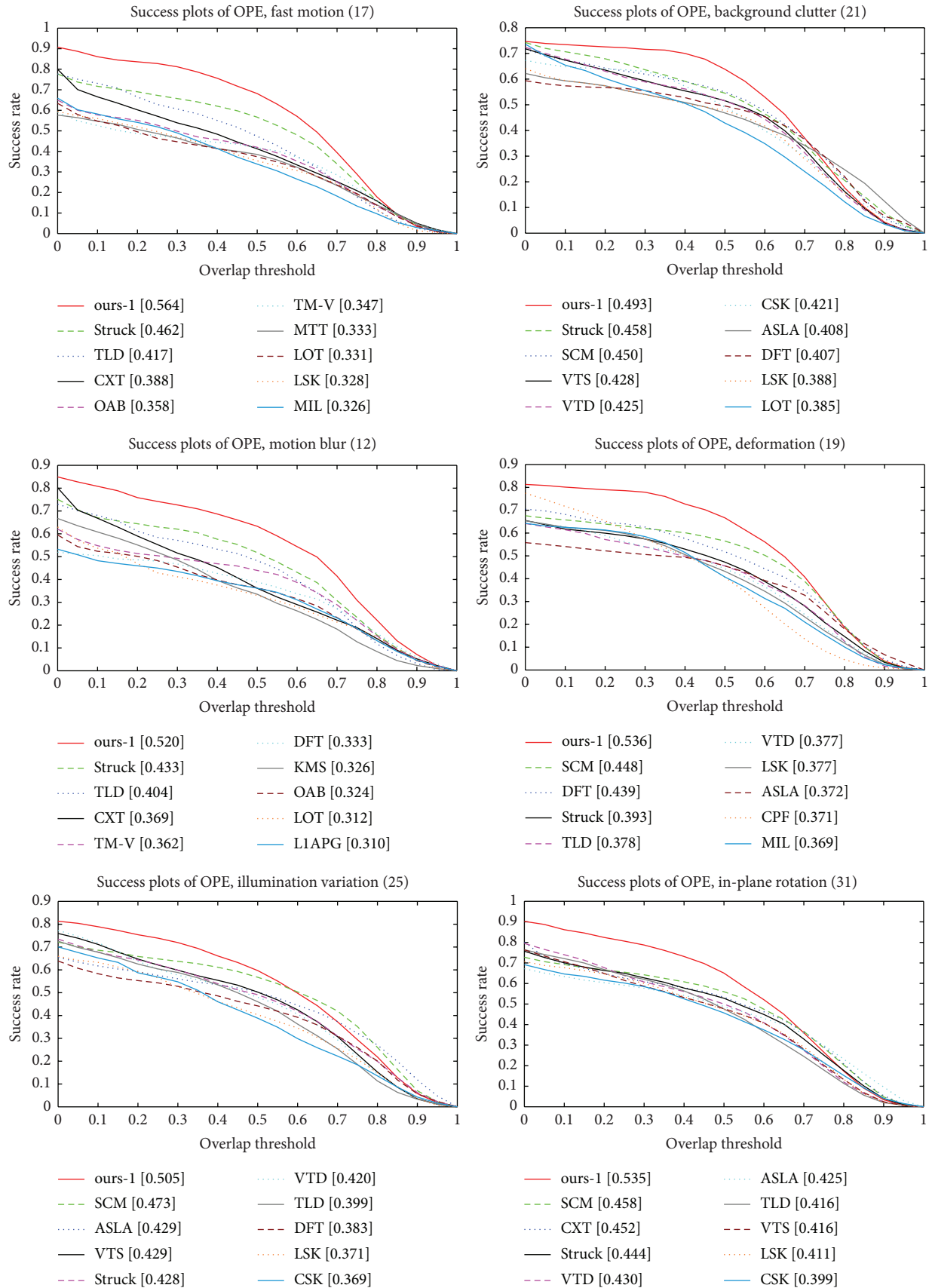


FIGURE 5: Continued.

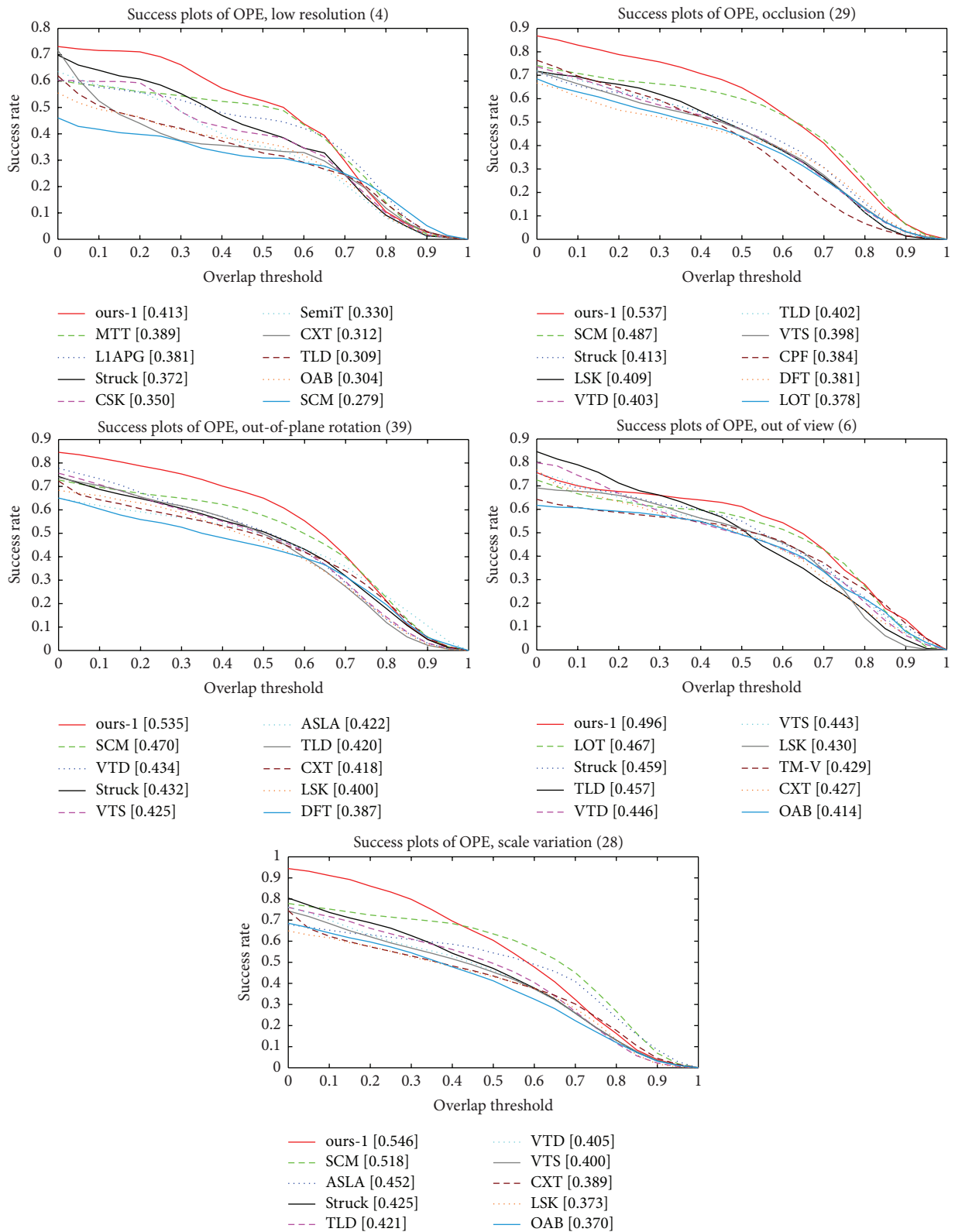
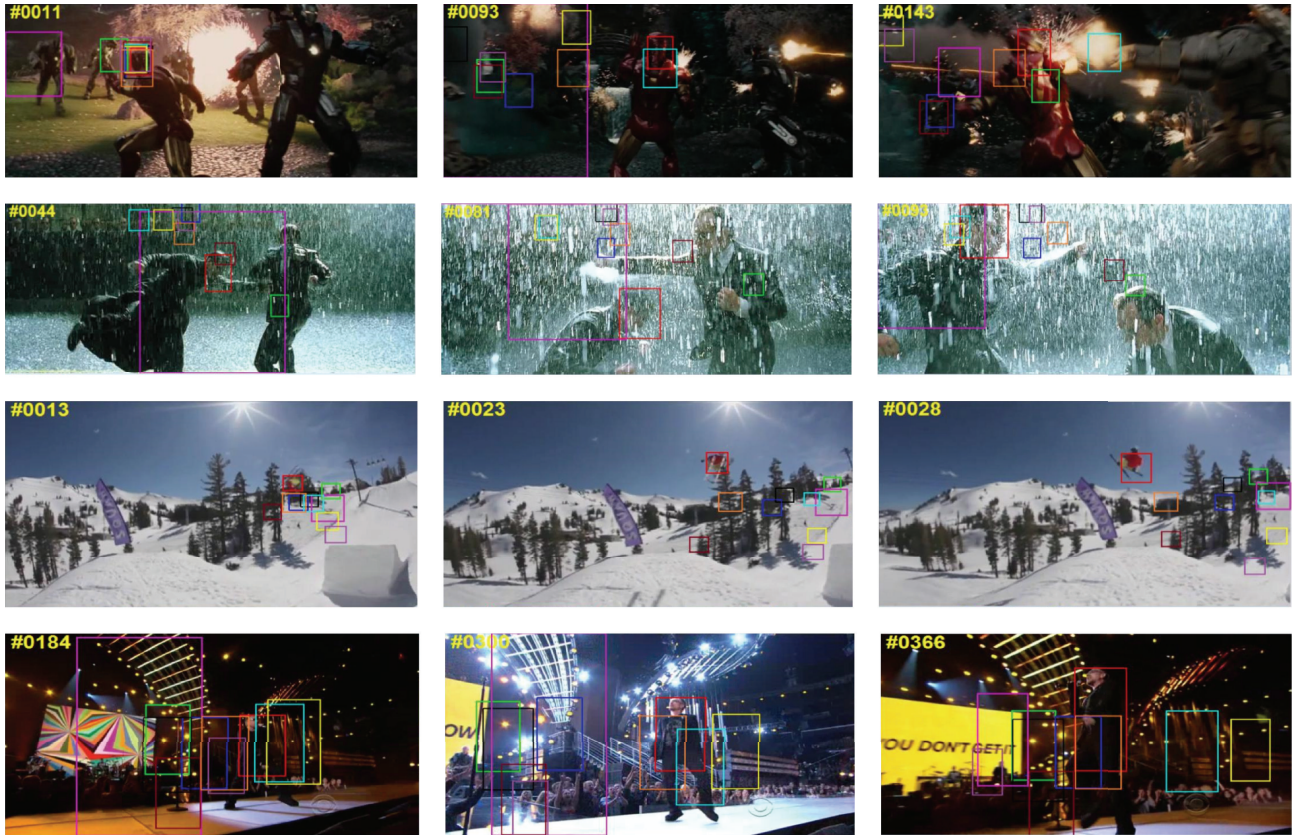


FIGURE 5: The success plots of one-pass evaluation (OPE) for trackers on the 11 attributes. The values next to the attributes denote the number of video sequences involving the corresponding attribute.



- ours-1
- SCM
- CXT
- VTS
- TLD
- Struck
- ASLA
- VTD
- CSK
- LSK

FIGURE 6: Qualitative results of the proposed tracking method (i.e., ours-1) on several challenging sequences from [41].

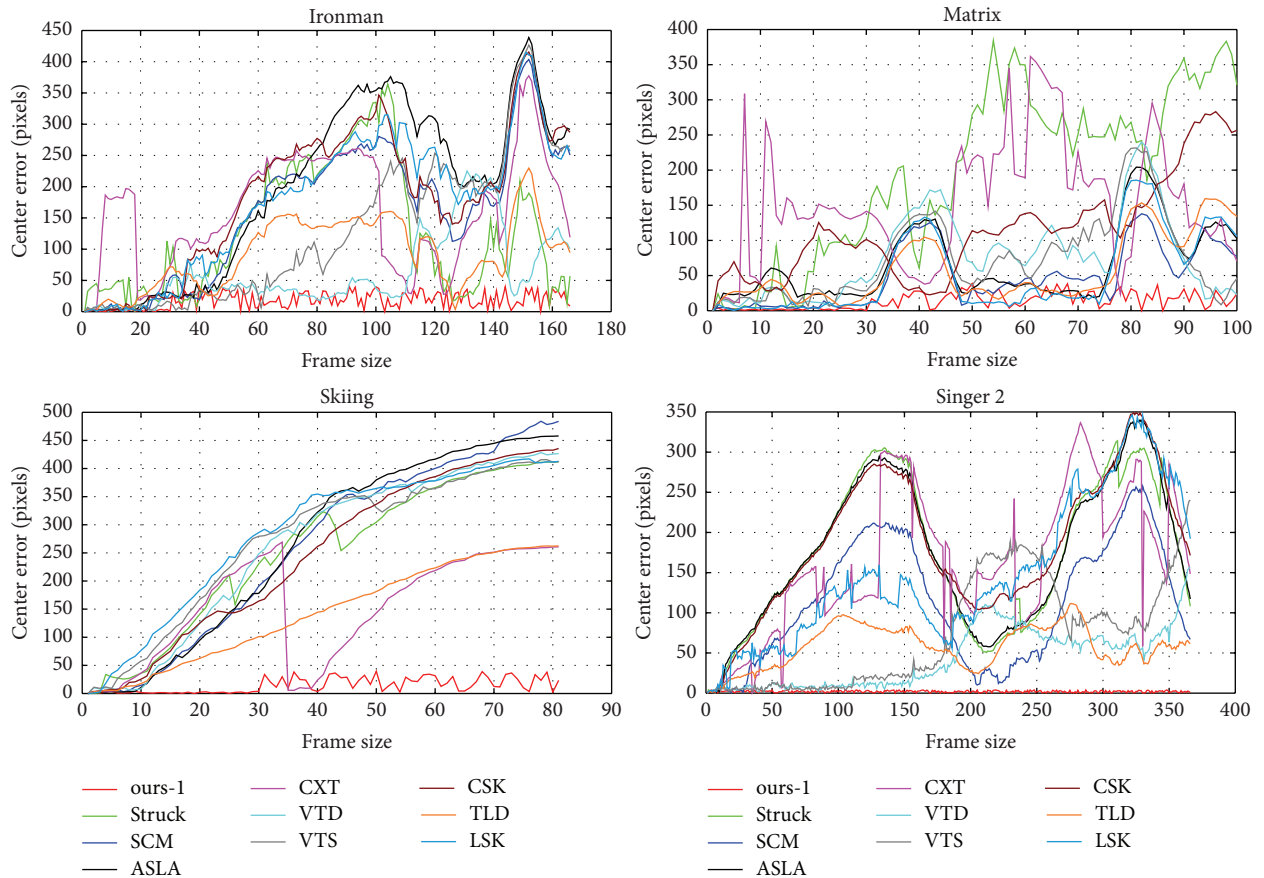


FIGURE 7: Quantitative results on the center distance error per frame for several challenging sequences from [41].

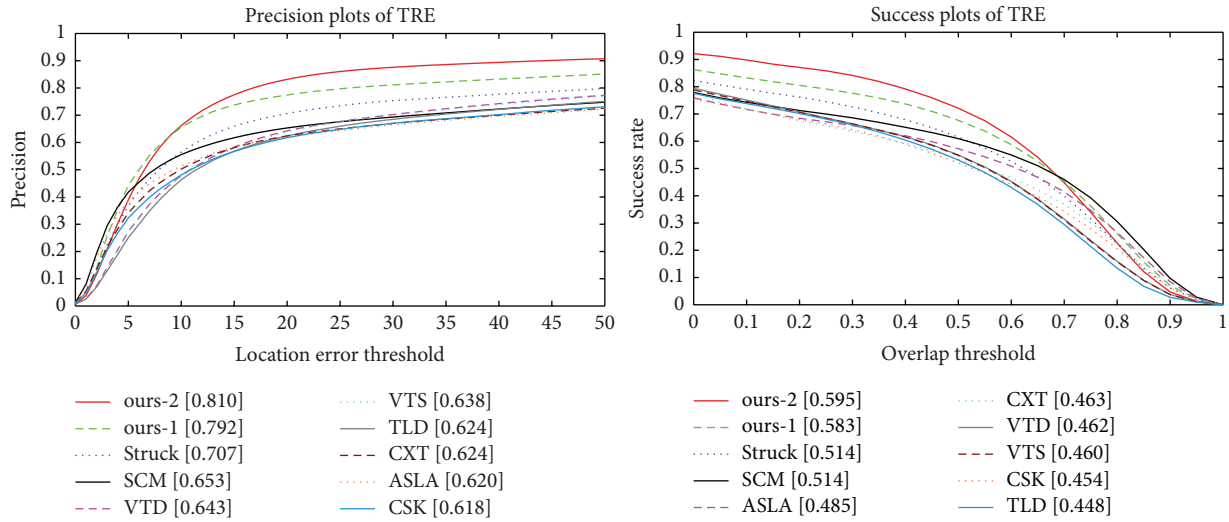


FIGURE 8: The precision and success plots of TRE for the proposed tracking method (e.g., ours-1 and ours-2) as the number of PCA filtering layers in PCANet grows. Please see the text for more details.

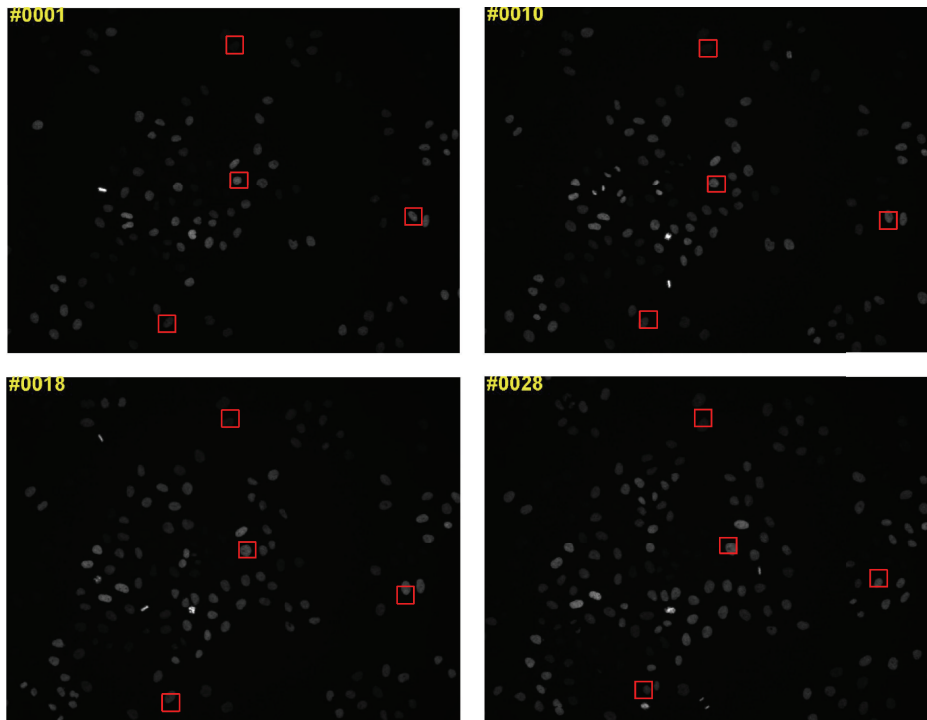


FIGURE 9: Qualitative results on individual-cell from the Mitocheck dataset [40].

5. Conclusion

We have proposed a robust feature learning method via PCANet deep network for robust individual-cell/object tracking in the time-lapse and 2D color imaging sequences. A cell/object is firstly effectively represented by composing of a PCA-based filter bank layer, a nonlinear layer, and a

patch-based pooling layer, respectively. Then, a discriminative target appearance model is constructed by training a neural network with one hidden layer. Finally, to alleviate the tracker drifting problem, a sample update scheme is carefully designed to keep track of the most representative and diverse samples while tracking. Extensive experiments on challenging image sequences from the Mitocheck cell dataset

and the online tracking benchmark (OTB) [41] validate the robustness and effectiveness of the proposed individual-cell/object tracking method.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work is supported by Natural Science Foundation of China (nos. 61572205, 61572206, 51305142, and 61175121), Natural Science Foundation of Fujian Province (nos. 2015J01257 and 2013J06014), Promotion Program for Young and Middle-Aged Teacher in Science and Technology Research of Huaqiao University (nos. ZQN-PY210 and ZQN-YX108), and 2015 Program for New Century Excellent Talents in Fujian Province University.

References

- [1] X. Chen, X. Zhou, and S. T. C. Wong, "Automated segmentation, classification, and tracking of cancer cell nuclei in time-lapse microscopy," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 4, pp. 762–766, 2006.
- [2] E. Meijering, O. Dzyubachyk, and I. Smal, "Methods for cell and particle tracking," in *Imaging and Spectroscopic Analysis of Living Cells*, vol. 504, pp. 183–200, Elsevier, 2012.
- [3] E. Meijering, O. Dzyubachyk, I. Smal, and W. A. van Cappellen, "Tracking in cell and developmental biology," *Seminars in Cell and Developmental Biology*, vol. 20, no. 8, pp. 894–902, 2009.
- [4] T. Kanade, Z. Yin, R. Bise et al., "Cell image analysis: Algorithms, system and applications," in *Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV '11)*, pp. 374–381, Kona, Hawaii, USA, January 2011.
- [5] V. C. Abraham, D. L. Taylor, and J. R. Haskins, "High content screening applied to large-scale cell biology," *Trends in Biotechnology*, vol. 22, no. 1, pp. 15–22, 2004.
- [6] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Computing Surveys*, vol. 38, no. 4, pp. 1–45, 2006.
- [7] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: an experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1442–1468, 2014.
- [8] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. van den Hengel, "A survey of appearance models in visual object tracking," *ACM Transactions on Intelligent Systems and Technology*, vol. 4, no. 4, pp. 1–58, 2013.
- [9] T. Schroeder, "Long-term single-cell imaging of mammalian stem cells," *Nature Methods*, vol. 8, no. 4, pp. S30–S35, 2011.
- [10] J. W. Young, J. C. W. Locke, A. Altinok et al., "Measuring single-cell gene expression dynamics in bacteria using fluorescence time-lapse microscopy," *Nature Protocols*, vol. 7, no. 1, pp. 80–88, 2012.
- [11] Q. Zou, J. Zeng, L. Cao, and R. Ji, "A novel features ranking metric with application to scalable visual and bioinformatics data classification," *Neurocomputing*, vol. 173, pp. 346–354, 2016.
- [12] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1, pp. 125–141, 2008.
- [13] J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky, "Hough forests for object detection, tracking, and action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2188–2202, 2011.
- [14] S. Avidan, "Support vector tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1064–1072, 2004.
- [15] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in *Proceedings of the 10th European Conference on Computer Vision*, Marseille, France, October 2008.
- [16] C. Lin, W. Chen, C. Qiu, Y. Wu, S. Krishnan, and Q. Zou, "LibD3C: ensemble classifiers with a clustering and dynamic selection strategy," *Neurocomputing*, vol. 123, pp. 424–435, 2014.
- [17] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.
- [18] F. Li, X. Zhou, J. Ma, and S. T. C. Wong, "Multiple nuclei tracking using integer programming for quantitative cancer cell cycle analysis," *IEEE Transactions on Medical Imaging*, vol. 29, no. 1, pp. 96–105, 2010.
- [19] M. Danelljan, F. S. Khan, M. Felsberg, and J. V. Weijer, "Adaptive color attributes for real-time visual tracking," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 1090–1097, Columbus, Ohio, USA, June 2014.
- [20] H. Grabner and H. Bischof, "On-line boosting and vision," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, pp. 260–267, IEEE, New York, NY, USA, June 2006.
- [21] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: structured output tracking with kernels," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 263–270, Barcelona, Spain, November 2011.
- [22] R. Yao, Q. Shi, C. Shen, Y. Zhang, and A. Van Den Hengel, "Part-based visual tracking with online latent structural learning," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 2363–2370, Portland, Ore, USA, June 2013.
- [23] V. Takala and M. Pietikäinen, "Multi-object tracking using color, texture and motion," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, pp. 1–7, Minneapolis, Minn, USA, June 2007.
- [24] Y. Lu, T. F. Wu, and S.-C. Zhu, "Online object tracking, learning, and parsing with and-or graphs," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 3462–3469, Columbus, Ohio, USA, June 2014.
- [25] J. Fan, X. Shen, and Y. Wu, "Scribble tracker: a matting-based approach for robust tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 8, pp. 1633–1644, 2012.
- [26] X. Lou, M. Schiegg, and F. A. Hamprecht, "Active structured learning for cell tracking: algorithm, framework, and usability," *IEEE Transactions on Medical Imaging*, vol. 33, no. 4, pp. 849–860, 2014.
- [27] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [28] G. E. Hinton, L. Deng, D. Yu et al., "Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

- [29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS '12)*, pp. 1097–1105, December 2012.
- [30] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: a review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
- [31] J. Donahue, Y. Jia, O. Vinyals et al., "DeCAF: a deep convolutional activation feature for generic visual recognition," in *Proceedings of the 31st International Conference on Machine Learning (ICML '14)*, pp. 988–996, June 2014.
- [32] T. Han Chan, K. Jia, S. H. Gao, J. W. Lu, Z. N. Zeng, and Y. Ma, "PCANet: a simple deep learning baseline for image classification," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5017–5032, 2015.
- [33] G. Carneiro and J. C. Nascimento, "Combining multiple dynamic models and deep learning architectures for tracking the left ventricle endocardium in ultrasound data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2592–2607, 2013.
- [34] N. Y. Wang and D. Y. Yeung, *Learning a Deep Compact Image Representation for Visual Tracking*, Advances in Neural Information Processing Systems, 2013.
- [35] J. Fan, W. Xu, Y. Wu, and Y. Gong, "Human tracking using convolutional neural networks," *IEEE Transactions on Neural Networks*, vol. 21, no. 10, pp. 1610–1623, 2010.
- [36] L. J. Wang, W. L. Ouyang, X. G. Wang, and H. C. Lu, "Visual tracking with fully convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '15)*, pp. 3119–3127, Santiago, Chile, December 2015.
- [37] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *Proceedings of the 32nd International Conference on Machine Learning (ICML '15)*, pp. 597–606, 2015.
- [38] C. Ma, J. B. Huang, X. K. Yang, and M. H. Yang, "Hierarchical convolutional features for visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '15)*, pp. 3074–3082, Santiago, Chile, December 2015.
- [39] H. S. Nam and B. Y. Han, "Learning multi-domain convolutional neural networks for visual tracking," <http://arxiv.org/abs/1510.07945>.
- [40] http://www.mitoccheck.org/cgi-bin/mtc?action=show_movie;query=243867.
- [41] Y. Wu, J. W. Lim, and M.-H. Yang, "Online object tracking: a benchmark," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 2411–2418, IEEE, Portland, Ore, USA, June 2013.
- [42] X. Mei and H. Ling, "Robust visual tracking using L1 minimization," in *Proceedings of the IEEE 12th International Conference on Computer Vision*, September 2009.
- [43] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust L1 tracker using accelerated proximal gradient approach," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 1830–1837, Providence, RI, USA, June 2012.
- [44] K. H. Zhang, L. Zhang, and M. H. Yang, "Real-time compressive tracking," in *Proceedings of the European Conference on Computer Vision (ECCV '12)*, Florence, Italy, October 2012.
- [45] T. Zhang, S. Liu, N. Ahuja, M.-H. Yang, and B. Ghanem, "Robust visual tracking via consistent low-rank sparse learning," *International Journal of Computer Vision*, vol. 111, no. 2, pp. 171–190, 2014.
- [46] X. Jia, H. C. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 1822–1829, IEEE, Providence, RI, USA, June 2012.
- [47] Z. Zhang and K. H. Wong, "Pyramid-based visual tracking using sparsity represented mean transform," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 1226–1233, Columbus, Ohio, USA, June 2014.
- [48] B. N. Zhong, H. X. Yao, S. Chen, R. R. Ji, T.-J. Chin, and H. Z. Wang, "Visual tracking via weakly supervised learning from multiple imperfect oracles," *Pattern Recognition*, vol. 47, no. 3, pp. 1395–1410, 2014.
- [49] Y. Zhou, X. Bai, W. Y. Liu, and L. J. Latecki, "Similarity fusion for visual tracking," *International Journal of Computer Vision*, vol. 118, no. 3, pp. 337–363, 2016.
- [50] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.
- [51] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1619–1632, 2011.
- [52] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.
- [53] Z. Chen, Z. Hong, and D. Tao, "An experimental survey on correlation filter-based tracking," <http://arxiv.org/abs/1509.05520>.
- [54] W. M. Zuo, X. H. Wu, L. Lin, L. Zhang, and M.-H. Yang, "Learning support correlation filters for visual tracking," <http://arxiv.org/abs/1601.06032>.
- [55] X. Lou, U. Koethe, J. Wittbrodt, and F. A. Hamprecht, "Learning to segment dense cell nuclei with shape prior," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 1012–1018, IEEE, Providence, RI, USA, June 2012.
- [56] O. Dzyubachyk, W. A. Van Cappellen, J. Essers, W. J. Niessen, and E. Meijering, "Advanced level-set-based cell tracking in time-lapse fluorescence microscopy," *IEEE Transactions on Medical Imaging*, vol. 29, no. 3, pp. 852–867, 2010.
- [57] L. Wang, T. Liu, G. Wang, K. L. Chan, and Q. Yang, "Video tracking using learned hierarchical features," *IEEE Transactions on Image Processing*, vol. 24, no. 4, pp. 1424–1435, 2015.
- [58] H. X. Li, Y. Li, and F. Porikli, "Deeptrack: learning discriminative feature representations by convolutional neural networks for visual tracking," in *Proceedings of the British Machine Vision Conference (BMVC '14)*, BMVA Press, Nottingham, UK, September 2014.