Visual Tracking Based on Multi-cue Proposals and Long Short-Term Features Learning

Jiaming Wei[1], Huimin Ma[1(✉)], Ruiqi Lu[1], and Xiong Luo[2]

[1] Department of Electronic Engineering,
Tsinghua University, Beijing 100084, China
{wjml6, lrql7}@mails.tsinghua.edu.cn,
mhmpub@tsinghua.edu.cn
[2] School of Computer and Communication Engineering & Institute of Artificial Intelligence, University of Science and Technology Beijing,
Beijing 100083, China
xluo@ustb.edu.cn

Abstract. Tracking-by-detection frameworks have made significant progress in recent years. However, proposal and classification of this framework could still be severely affected by occlusion and motion. In this paper, we propose a tracking algorithm MPLST to improve the accuracy and robustness of proposal and classification under challenges. First, we provide a multi-cue proposal method, which combines Gaussian sampling utilizing previous target state, and motion and appearance selective search making use of the motion and appearance features of the target respectively. Second, we provide a long short-term features learning approach for target classification and network updating. The long-term features are robust to occlusion, and the short-term features can keep up with the fast motion of the target. Experiments on the OTB100 [1] and VOT2015 [2] datasets demonstrate that our MPLST can effectively deal with occlusion and motion, and achieve competitive performance against state-of-the-art trackers.

Keywords: Visual tracking · Region proposal · Long short-term features

1 Introduction

Tracking-by-detection mainly contains two stages: proposals and classification. This framework has made significant progress in improving accuracy for regular tracking. However, it sometimes fails under the challenges like occlusion and motion, which would strongly influence both two stages of detection. Tracking algorithms require cautious updating under occlusion but need to be fast updated for following the fast motion of the target. As a result, tracking algorithms suffer from the challenges of dynamic occlusion and motion. Specifically, in the proposal stage, Gaussian sampling [3] relies too much on the previous target state and fails under camera motion and target deformation. Region proposal networks [7] and selective search [6] only utilize the appearance features of objects and usually leave out the small and occluded target. In the classification stage, the most important process is to learn the target features and

Y. Wang et al. (Eds.): IGTA 2019, CCIS 1043, pp. 633–644, 2019.
https://doi.org/10.1007/978-981-13-9917-6_60

online update a target classifier. However, robust classifiers [3–5] usually cautiously update the network and fail to follow the fast changes of the target. In contrast, other classifiers [10–13] which continuously keep up with the target are sensitive to occlusion and could fall into tracking drift easily.

In this paper, we propose a tracking algorithm MPLST based on multi-cue proposals and long short-term features learning. We balance fast learning and robust tracking, in order to solve dynamic occlusion and motion. Main contributions of our work are summarized as the following three folds:

(1) **Multi-cue proposals:** We provide a multi-cue proposal method to extract more accurate proposals by utilizing cues of previous target state, motion and appearance features. The Gaussian sampling is robust to occlusion. The motion-appearance selective search can generate high-quality proposals under motion and deformation.
(2) **Long short-term features learning:** We propose a feature learning and target classification method. Long-term features are robust to occlusion, while short-term features can keep up with fast motion of the target. The optimal representation of the target is obtained by combining long-term and short-term target features.
(3) **Accuracy and robustness balancing:** We evaluate our MPLST on two tracking benchmarks, the OTB100 [1] and VOT2015 [2] datasets. The experiments demonstrate that MPLST can deal with occlusion and motion, and achieve competitive performance on both accuracy and robustness against state-of-the-art trackers.

2 Related Work

2.1 Challenges Handling

Challenges of tracking include occlusion, scale change, deformation, illuminate variation and so on [1]. Occlusion is the dominant challenge followed by scale change and fast motion [2]. To handle fast motion, [22] utilizes the image template of the previous target and a regression network; [12–14, 30] use discriminative correlation filters to model the target. Those approaches can adapt to current target features quickly but suffer from wrong training samples caused by occlusion. As a contrast, [3–5, 23, 31] initialize and update the deep neural networks [24] using hundreds of training samples to resist the influence of occlusion. However, those approaches usually fail under fast motion because of the slow learning rate of the network.

2.2 Region Proposals

Proposals generation is the first stage of tracking-by-detection framework. [3–5] utilize Gaussian sampling to randomly extract proposals around the previous target by the Gaussian distribution. Gaussian sampling only relies on the previous location and size of the target and achieves robust performance under heavy occlusion. However, it

cannot deal with fast motion and deformation due to the lack of target motion features. [25] utilizes selective search [6] to make use of appearance features of the target. [26] generates flexible and tight proposals using deep features of the target extracted by region proposal network [7] and takes advantages of appearance contrasts between the target and background. However, those methods usually extract proposals on salient objects and ignore small and occluded target, leading to tracking drift as a result.

2.3 Classification and Updating

Classification and model updating also suffer from dynamic occlusion and motion. Classifiers used in state-of-the-art trackers can be roughly categorized into two types: convolutional neural network (CNN) based models [3–5, 23] and discriminative correlation filters (DCF) based models [10–13, 16]. CNN based models take advantages of stable deep semantic features of the target to achieve better performance under heavy occlusion, but suffer from fast motion because a large number of samples and learning iterations are needed to online update the model. On the other hand, DCF based models only need several samples to update the classifier and can follow the fast change of the target. However, those methods usually fail when update models with wrong samples under heavy occlusion. Classifiers for tracking have to deal with dynamic occlusion and motion at the same time to achieve better performance.

3 Overview of Tracking Framework

Our MPLST is a tracking-by-detection framework with four parts as shown in Fig. 1. The fundamental purpose of MPLST is making better proposals and classification to handle the challenge of dynamic occlusion and motion in one tracking framework.

The first part of MPLST is feature extraction. The inputs are the current frame and the previous target states. We compute the optical flow map and extract motion and appearance features of the whole image using convolutional layers. The motion and appearance features are further concatenated as the feature maps. We utilize a network with only three layers because the targets are usually small and a shallow network will increase location precision. The networks are the convolutional layers of the VGG-M [14] network pre-trained on the ImageNet [8] dataset.

Second, 256 candidates are extracted by the multi-cue proposal approach combining Gaussian sampling and motion-appearance selective search. The Gaussian sampling generates 128 candidates by randomly sampling around the previous target state, which is robust to occlusion. The motion-appearance selective search extract 64 samples on the optical flow map and the current image respectively by standard selective search, which is more flexible for motion and deformation. We limit the search region around the target and choose candidates through previous target state. ROI pooling is further performed to obtain features of the candidates with a size of $3 \times 3 \times 1024$.

Third, the target-background classification is performed by combining long-term and short-term features of the target. The long-term features extracted by fully connected layers trained using samples collected over a long time period (100 frames). In

contrast, the other path responses to the short-term features of the target which are learned using samples collected in a short time (10 frames). Those two paths also have different network architectures. We delete one fully connected layer for faster learning of the short-term features. The optimal representation of the target is obtained by training the classifier combining the long-term and short-term features. The reason for using two complementary features is that the long-term features are robust to occlusion and the short-term features can keep up with fast motion of the target.

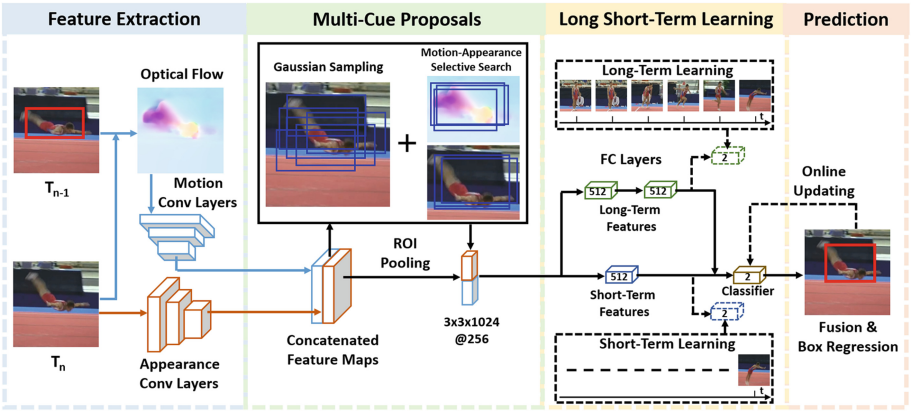

Fig. 1. Overview of our MPLST tracking frameworks which has four parts. The motion and appearance feature maps are extracted by the convolutional neural networks. The multi-cue proposal which combines Gaussian Sampling and motion-appearance selective search generates accurate proposals under occlusion and motion. The multi-path fully connected layers classify proposals utilizing the long-term and short-term features. Several high score proposals will be fused and a box regression is performed to refine the target state.

Finally, the boxes with the highest scores will be fused into one prediction box in the fourth part. We perform a linear bounding box regression to refine the target box. The regression model is trained by 1000 target samples collected in the first frame. Positive and negative samples are collected in each frame for online updating.

4 Multi-cue Proposals

The multi-cue proposal method combines three algorithms to generate better candidates. The Gaussian sampling makes use of the previous state of the target. Motion selective search utilizes optical flow map as motion features for sampling. Appearance selective search generates candidates using appearance contrasts between objects. Those three methods are complementary for handling multiple challenges.

4.1 Optical Flow Map

The states of proposals mainly depend on the motion of the target. The target motion usually contains two parts, the camera motion and the object motion. The camera

motion influences the absolute locations of objects and the search region of tracking algorithms. The object motion influences the relative positions of objects and the posture of the target. Hence, we first calculate the optical flow between two frames to obtain the motion features of the target. The Horn-Schunck method [9] is performed to calculate dense optical flow around the target. The visual optical flow map is further generated by a pseudo-color transformation approach [29]. We make use of the optical flow map in two ways. First, the average optical flow outside the previous target state is regarded as the camera motion. The search region of our tracking algorithm is relocated according to the camera motion. Second, the optical flow around the previous target is regarded as the object motion and utilized for motion selective search.

4.2 Gaussian Sampling

Gaussian sampling is widely used in state-of-the-art tracking algorithms [3–5]. We choose Gaussian sampling as the basic proposal method of our tracking framework. In each frame, 128 samples are generated by translation and scale dimension of the target state in the previous frame. The translation is subject to Gaussian distribution whose mean is previous target state and the covariance is $0.09r^2$, where r is the mean of the height and width of the target. The scale s_t is also subject to Gaussian distribution whose mean is the initial target scale and the covariance is 0.25. The scale of candidates is calculated by multiply 1.05^{s_t} to previous target scale. The camera motion has been excluded before the Gaussian sampling. Because the Gaussian sampling mainly utilizes the previous target state, it is a highly effective proposal method for small target or target whose motion is not too large.

4.3 Motion-Appearance Selective Search

In order to utilize the motion and appearance features of the target, we draw 64 samples with motion selective search and 64 samples with appearance selective search. We choose motion and appearance features because those features are complementary features. A target that remains stationary will not be influenced by motion blur and has clear appearance contrast with backgrounds. On the other hand, a target with large motion will be salient on the optical flow map. We first determine the search region because performing selective search on the whole image is a waste of computation and could extract too many background objects. A region that is two times larger than the previous target is regarded as the search region. Next, selective search [6] is performed on the optical flow map and the current image respectively. We generate 256 samples in total with the Gaussian sampling and the motion-appearance selective search. The results of the proposals are demonstrated in Fig. 1.

5 Long Short-Term Features Learning

The features of the target usually continuously change during tracking. The long-term features can resist interference and increase tracking robustness. The short-term features represent the recent states of the target and increase tracking accuracy. In order to

balance the accuracy and robustness, we propose a long short-term features learning method to make use of long-term and short-term features at the same time.

5.1 Motion and Appearance Features

The convolutional features of the current frame represent the appearance features of the target. Although they are widely used in tracking algorithms, the appearance features are not enough to deal with fast motion and scale change. Hence, we extract the motion features of the target on the optical flow map using pre-trained convolutional networks. The motion features and the appearance features are further concatenated as the target features and used to classify the target and the backgrounds. The features of candidates with a size of $3 \times 3 \times 1024$ are generated after ROI pooling on the concatenated feature maps.

5.2 Long-Term Features

The fully connected layers are used to further extract the feature vectors of the candidates and perform target/background classification. The single path of fully connected layers [3] is duplicated into two paths for learning long-term and short-term features respectively. The long-term features are robust to occlusion because the true target samples are much more than noisy occluded samples. The long-term path has two fully-connected layers with a size of $3 \times 3 \times 1024 \times 512$ and $1 \times 1 \times 512 \times 512$. This path extracts the long-term high-level features of the target using relatively deeper FC layers, in order to improve updating robustness under occlusion. The target samples for online updating are collected in a long time as shown in Fig. 1. We choose target samples in recent 100 frames, about 4 s, which are long enough during tracking. Finally, the long-term feature vector with 512 channels is generated.

5.3 Short-Term Features

The short-term path has shallower fully connected layers than the long-term path for quickly online updating. This path is used to learn fast feature change of the target under big motion and deformation. Hence, the short-term features focus on what the target exactly looks like recently. We force the network branch to learn the short-term features in two ways. First, the second fully-connected layers are deleted to learn the low-level target features. A shallow network is also easy to online update. Second, only target samples extracted in recent 10 frames are used to train the short-term path, in order to exclude the influence of the long-term target samples. The short-term feature vector also has 512 channels as the long-term vector.

5.4 Classification and Online Updating

The long-term and short-term features are concatenated into one feature vector with 1024 channels for target/background classification. The classifier is a fully connected layer with a size of $1 \times 1 \times 1024 \times 2$. The classifier generates a target score $P^{+}(x_i)$ and a background score $P^{-}(x_i)$ for each candidate x_i. The target scores are used to

evaluate the similarity between candidates and previous target features. The optimal target state x^* is predicted by Eq. 1.

$$x^* = \text{argmax}_{x_i} P^+(x_i). \tag{1}$$

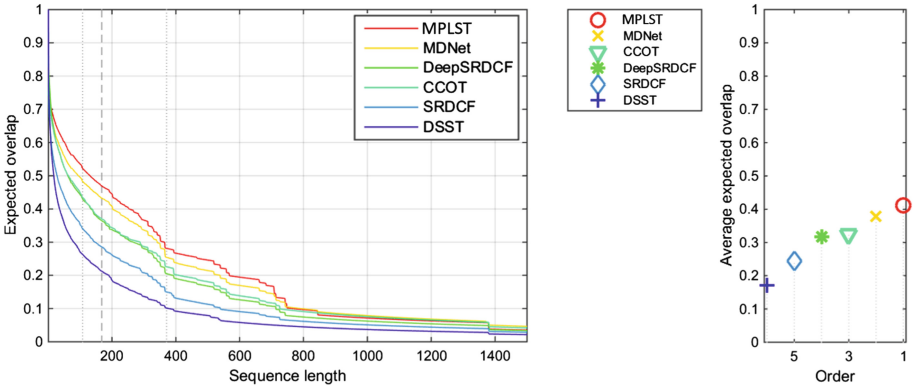

Fig. 2. Expect average overlap curves and ratios ranks from right to left.

The classifier and the long short-term paths are updated using a three-step method. First, positive and negative training samples are collected during tracking. We collect 50 positive samples and 200 negative samples in each frame. The positive samples have > 0.7 IoU overlap ratios with the predicted target. The negative samples have < 0.5 IoU overlap ratios. Second, the long-term path and the short-term path are trained using positive samples collected in recent 100 frames and 10 frames. The negative samples in recent 20 frames are used. The loss function is softmax loss as Eq. 2 with K = 2. Where f_j denotes the score of category j, f_{y_i} denotes the score of the ground-truth category. Stochastic gradient descent (SGD) is used for loss backpropagation.

$$L = -f_{y_i} + log \sum_{j=1}^{K} e^{f_j}. \tag{2}$$

Third, the target/background classifier is trained to make use of long-term and short-term features at the same time. The positive and the negative samples in recent 20 frames are used. The purpose of our network architecture is to obtain the optimal representation of the target using long short-term features. The loss function is also softmax loss as Eq. 2. The classifier and the long short-term paths are initialized in the first frame with 500 positive samples and 5000 negative samples.

Table 1. Accuracy and Robustness evaluation of our MPLST and state-of-the-art trackers on VOT2015 [2] dataset under challenges including camera motion, illumination change, motion change, occlusion and size change. Left of slash is accuracy and right of slash is robustness. Top performance is highlighted in bold.

	empty	camera	illume	motion	occlu	size	mean
DSST	61.1/37	55.8/69	67.3/7	49.1/61	38.4/28	52.2/33	54.0/39.2
SRDCF	58.0/16	53.6/43	69.2/8	48.6/36	42.1/22	47.2/21	53.1/24.3
CCOT	57.0/11	53.5/24	66.0/2	46.4/20	44.5/14	49.1/13	52.8/14.0
DeepSRDCF	62.8/9	56.4/25	66.5/**0**	49.7/23	45.2/26	53.1/**8**	55.6/15.2
MDNet	65.0/6	61.0/20	68.0/1	56.0/15	54.3/14	56.0/11	60.0/11.4
MPLST (Ours)	**65.0/4**	**61.9/17**	**70.6/0**	**56.6/13**	**56.3/13**	**58.1**/9	**61.4/9.3**

6 Experiments

Our challenges handling network (MPLST) is evaluated on two large-scale visual tracking benchmarks: the VOT2015 [2] and OTB100 [1] datasets. We compare our tracking algorithm with other state-of-art trackers under multiple challenges.

6.1 Evaluation on VOT2015

VOT2015 [2] is a commonly used dataset which contains 60 sequences with full annotation. The 60 videos provide different challenges, including camera motion, illumination change, motion change, occlusion and size change. Three main metrics including accuracy, robustness and expect average overlap (EAO) are utilized to evaluate the performance of trackers. We compare our tracker MPLST with state-of-art trackers including MDNet [3], DeepSRDCF [13], CCOT [12], SRDCF [11] and DSST [10]. Our tracker is pre-trained using sequences in OTB100 [1] datasets excluding sequences in VOT2015 [2].

Table 1 illustrates the accuracy (left) and robustness (right) evaluation of trackers under challenges. Our MPLST achieves better performance on most challenges, especially on occlusion and size change. We improve the accuracy by 2.0% on occlusion, 2.1% on size change and 1.4% in the average. And the robustness is also improved under most challenges, especially on camera motion and size change. Figure 2 plots the expect average overlap (EAO) curves and the ranks of trackers. EAO evaluates the expected accuracy of the results on long-term tracking. Our MPLST has better feature learning ability for long-term tracking as shown in Fig. 2.

6.2 Evaluation on OTB100

OTB100 [1] is consist of 100 fully annotated videos with different attributes including fast motion, occlusion, deformation, etc. The one-pass evaluation (OPE) is performed on our MPLST comparing with 14 state-of-art trackers including MDNet [3], CCOT [12], DeepSRDCF [13], HDT [15], SRDCFdecon [16], CF2 [17], CNN-SVM [18], SRDCF [11], staple [19], MEEM [20], SAMF [21], LCT [27], KCF [28] and DSST [10]. The main metrics for performance evaluation are precision and success plots. The

precision plot measures the accuracy of the target location. The success plot evaluates the bounding box overlap ratio. Our tracker is pre-trained on VOT2015 [2], excluding sequences in OTB100 [1].

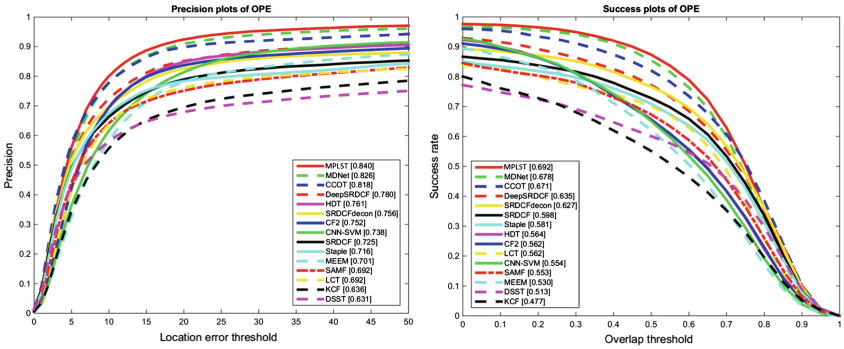

Fig. 3. Precision and success plots of OPE on OTB100 [1], comparing our tracker MPLST with state-of-the-art trackers.

Figure 3 illustrates the precision and success plots of OPE on the OTB100 [1] dataset. Our MPLST outperforms state-of-the-art trackers on both location accuracy and bounding box overlap. We make progress of 1.4% on precision and 1.4% on success. We further evaluate trackers under different attributes and plot the precision curves as shown in Fig. 4. Our MPLST performs better on most attributes, especially scale variation, out-plane rotation, and fast motion. The experiment results demonstrate that our MPLST can handle multiple dynamic challenges in one framework.

6.3 Experimental Analysis

The experiments mainly evaluate the accuracy and robustness of trackers under challenges. We summarize the results of experiments in three folds:

First, our MPLST achieves competitive performance against state-of-art trackers on both accuracy and robustness on the OTB100 [1] and VOT2015 [2] datasets. MPLST improves both proposal and classification stage of a tracking-by-detection framework. The multi-cue proposal method is utilized to generate more accurate candidates. The long short-term learning method learns features of the target in different time periods.

Second, our MPLST achieves better performance on multiple challenges, especially deformation, fast motion, and occlusion. The experiments results achieve the fundamental purpose of our tracking frameworks. Complementary approaches are effective to deal with dynamic occlusion and motion at the same time.

Third, some future work needs to be done to further improve our tracking performance. MPLST cannot deal with the extremely small target and similar background. The features of the small target will be lost after convolution and pooling in deep layers. Also, our tracker cannot distinguish the target and similar background because their features are almost the same (Fig. 5).

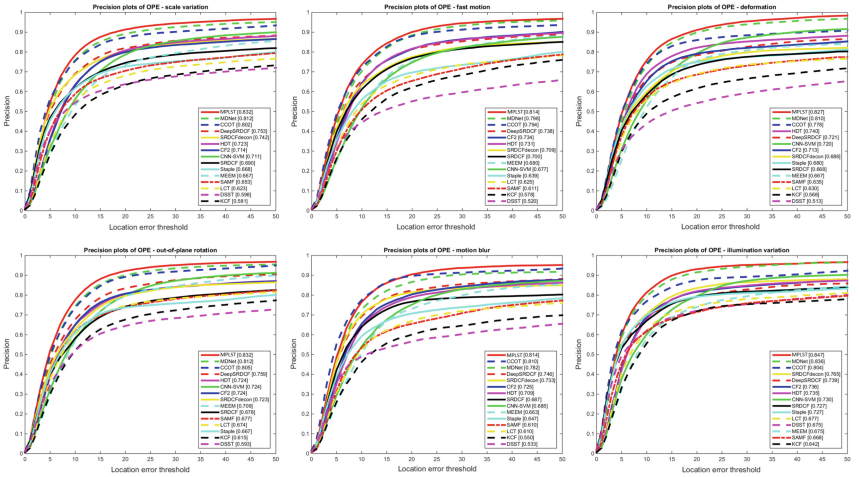

Fig. 4. Precision plots of OPE under different attributes on OTB100 [1]. The attributes include scale variation, fast motion, deformation, out-plane-rotation, motion blur, and illumination variation.

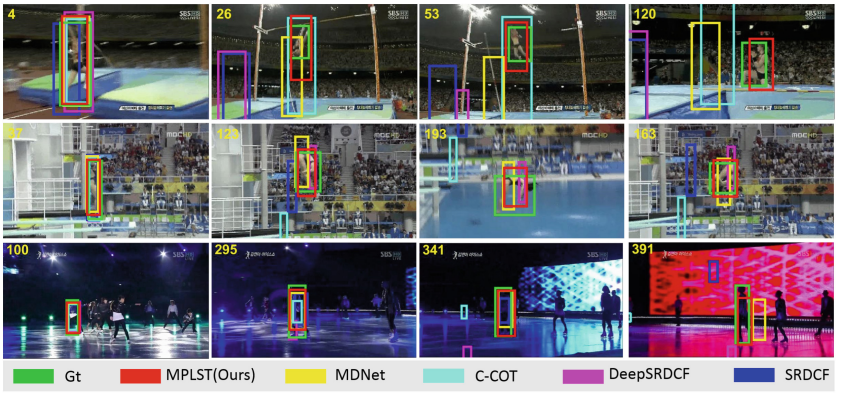

Fig. 5. Qualitative results of our MPLST comparing with other state-of-the-art trackers.

7 Conclusions

The proposed MPLST tracking framework handles the challenge of dynamic occlusion and motion simultaneously by multi-cue proposals and long short-term features learning. The multi-cue proposal method which combines Gaussian sampling and motion-appearance selective search improves the robustness of proposal under occlusion and the accuracy of proposals under fast motion. The long short-term features learning approach constructs the optimal representation of the target by combining long-term and short-term target features. Our MPLST resists occlusion using the long-term features and keeps up with fast motion of the target using short-term features. Our

tracking algorithm improves both tracking accuracy and robustness under challenges including occlusion, motion, deformation, and scale variation and achieves competitive performance against state-of-the-art trackers. In the future work, the feature pyramid networks could be utilized to learn low-level features and handle small targets. The training frequency should be adaptively adjusted to improve real-time performance.

Acknowledgment. This work was supported by the National Key R&D Plan (No. 2016YFB0100901), the National Natural Science Foundation of China (No. 61773231).

References

1. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1834–1848 (2015)
2. Kristan, M., et al.: The visual object tracking VOT2015 challenge results. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 1–23 (2015)
3. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4293–4302 (2016)
4. Han, B., Sim, J., Adam, H.: BranchOut: regularization for online ensemble tracking with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3356–3365 (2017)
5. Nam, H., Baek, M., Han, B.: Modeling and propagating CNNs in a tree structure for visual tracking. arXiv preprint arXiv:1608.07242 (2016)
6. Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W.: Selective search for object recognition. Int. J. Comput. Vis. **104**(2), 154–171 (2013)
7. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2017)
8. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
9. Horn, B.K., Schunck, B.G.: Determining optical flow. Artif. Intell. **17**(1–3), 185–203 (1981)
10. Danelljan, M., Häger, G., Khan, F., Felsberg, M.: Accurate scale estimation for robust visual tracking. In: British Machine Vision Conference, Nottingham, 1–5 September 2014. BMVA Press (2014)
11. Danelljan, M., Hager, G., Shahbaz Khan, F., Felsberg, M.: Learning spatially regularized correlation filters for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4310–4318 (2015)
12. Danelljan, M., Robinson, A., Shahbaz Khan, F., Felsberg, M.: Beyond correlation filters: learning continuous convolution operators for visual tracking. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9909, pp. 472–488. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46454-1_29
13. Danelljan, M., Hager, G., Shahbaz Khan, F., Felsberg, M.: Convolutional features for correlation filter based visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 58–66 (2015)
14. Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: delving deep into convolutional nets. arXiv preprint arXiv:1405.3531 (2014)
15. Qi, Y., et al.: Hedged deep tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4303–4311 (2016)

16. Danelljan, M., Hager, G., Shahbaz Khan, F., Felsberg, M.: Adaptive decontamination of the training set: a unified formulation for discriminative visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1430–1438 (2016)
17. Ma, C., Huang, J.B., Yang, X., Yang, M.H.: Hierarchical convolutional features for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3074–3082 (2015)
18. Niu, X.X., Suen, C.Y.: A novel hybrid CNN–SVM classifier for recognizing handwritten digits. Pattern Recogn. **45**(4), 1318–1325 (2012)
19. Bertinetto, L., Valmadre, J., Golodetz, S., Miksik, O., Torr, P.H.: Staple: complementary learners for real-time tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1401–1409 (2016)
20. Zhang, J., Ma, S., Sclaroff, S.: MEEM: robust tracking via multiple experts using entropy minimization. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 188–203. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10599-4_13
21. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8926, pp. 254–265. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16181-5_18
22. Held, D., Thrun, S., Savarese, S.: Learning to track at 100 FPS with deep regression networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 749–765. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_45
23. Fan, H., Ling, H.: SANet: structure-aware network for visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 42–49 (2017)
24. LeCun, Y., et al.: Backpropagation applied to handwritten zip code recognition. Neural Comput. **1**(4), 541–551 (1989)
25. Keuper, M., Tang, S., Zhongjie, Y., Andres, B., Brox, T., Schiele, B.: A multi-cut formulation for joint segmentation and tracking of multiple objects. arXiv preprint arXiv:1607.06317 (2016)
26. Ren, J., et al.: Robust tracking using region proposal networks. arXiv preprint arXiv:1705.10447 (2017)
27. Ma, C., Yang, X., Zhang, C., Yang, M.H.: Long-term correlation tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5388–5396 (2015)
28. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 583–596 (2015)
29. Jain, S.D., Xiong, B., Grauman, K.: FusionSeg: learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2117–2126. IEEE (2017)
30. Zhou, J., Wang, R., Ding, J.: Deep convolutional features for correlation filter based tracking with parallel network. In: Wang, Y., Jiang, Z., Peng, Y. (eds.) IGTA 2018. CCIS, vol. 875, pp. 461–470. Springer, Singapore (2018). https://doi.org/10.1007/978-981-13-1702-6_46
31. Wang, R., Zou, J., Che, M., Xiong, C.: Robust and real-time visual tracking based on single-layer convolutional features and accurate scale estimation. In: Wang, Y., Jiang, Z., Peng, Y. (eds.) IGTA 2018. CCIS, vol. 875, pp. 471–482. Springer, Singapore (2018). https://doi.org/10.1007/978-981-13-1702-6_47