An Adaptive Superpixel Tracker Using Multiple Features

Jingjing Liu[1, 2], Bin Zhang[3], Xu Cheng[4], Ying Chen[5] and Li Zhao[1, *]

Abstract: Visual tracking is a challenging issue in the field of computer vision due to the objects' intricate appearance variation. To adapt the change of the appearance, multiple channel features which could provide more information are used. However, the low level feature could not represent the structure of the object. In this paper, a superpixel-based adaptive tracking algorithm by using color histogram and haar-like feature is proposed, whose feature is classified into the middle level. Based on the superpixel representation of video frames, the haar-like feature is extracted at the superpixel level as the local feature, and the color histogram feature is applied with the combination of background subtraction method as the frame feature. Then, local features are clustered and weighted according to the target label and the location center. Superpixel-based appearance model is measured by using the sum of the voting map, and the candidate with the highest score is selected as the tracking result. Finally, an efficient template updating scheme is introduced to obtain the robust results and improve the computational efficiency. The proposed algorithm is evaluated on eight challenging video sequences and experimental results demonstrate that the proposed method can get better performance on occlusion, illumination variation and transformation.

1 Introduction

Precise objection tracking is the most substantial component in a wide range of applications like activity analysis, visual surveillance, human computer interaction and medical imaging. With the development of high-powered computers and cheap high quality video cameras, the progress of object tracking techniques is fast in recent years. However, tracking objects is complex due to noise in images, partial and full object occlusions, scene illumination variation, real-time processing requirements and so on.

---

[1] School of Information Science and Engineering, Southeast University, Nanjing, 210096, China.

[2] College of Information Science and Engineering, Henan University of Technology, ZhengZhou, 450001, China.

[3] Enterprise and Information Technology, Henan Branch of China Telecom Co., Ltd, Zhengzhou, 450046, China.

[4] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, 210044, China.

[5] Department of Psychiatry, Columbia University in the City of New York and NYSPI, New York, NY 10032, USA.

[*] Corresponding Author: Li Zhao. Email: zhaoli@seu.edu.cn.

Zhenguo et al. represent the object by low dimensional compact shape and color feature [Gao, Xia, Zhang et al. (2018)]. In this paper, the target and its surroundings are represented by using small regions, called superpixels [Ren and Malik (2003)]. Superpixels are generally defined as contracting and clustering uniform pixels in the image, which are local, coherent. Superpixels have been broadly used in various vision applications such as image segmentation, object recognition, anomaly detection and object tracking. The predominant integrity of superpixels provides a more natural and meaningful representation. Numerous works have been applied superpixels known as middle level visual cue and pleasant consequences have been obtained. Image superpixel segmentation approach using Lazy random walk (LRW) algorithm has been presented by Shen, which achieves the better performance by relocating the center positions of superpixels and splitting the large superpixels into small ones [Shen, Du, Wang et al. (2014)]. Xuan constructs a video representation with a superpixel-based bag-of-features (BoF) model for action recognition which includes histograms of optical flow (HOF), histograms of oriented gradients (HOG), and motion boundary histograms [Dong, Tsoi and Lo (2014)]. Superpixel-based statistical anomaly detection for sense and avoidance was presented by Pappas, which is based on superpixel image segmentation combined with subsequent statistical analysis and anomaly detection, can be applied to solve the problem of visually detecting possible aircraft collisions [Giordano, Murabito, Palazzo et al. (2015)].

In terms of object tracking based superpixels, significant advances have been witnessed with the development of adaptive algorithms and productive applications. Robust superpixel tracking (SPT) adopts a discriminative model based on superpixels to distinguish the target from the background, which can obtain the best candidate by maximum a posterior estimation in the target-background confidence map [Yang, Lu and Yang (2014)]. Concerning to estimate the target position, Junqiu employs multiple hypotheses for superpixel matching and projects the matching results onto a displacement confidence map [Wang and Yagi (2014)]. In consideration of recovering the object from the drifting scene, Xu proposes robust superpixel tracking with weighted multiple-instance learning which achieves robust and accurate performance [Cheng (2015)]. Apart from making efforts on the discriminative appearance model, Yuxia proposes a superpixel tracking method via graph-based hybrid discriminative-generative appearance model to deal with occlusion and shape deformation [Wang and Zhao (2015)].

Though the aforementioned works can be efficient, these methods are hard to deal with the tracking adaptively. In this paper, our approach takes into account for the content and quality of the tracking objects and applies various update frequency according to the different sequences. In addition, the superpixel haar-like feature and the region of interest deviating from background subtraction are integrated to build the appearance model, which uses the local feature of the superpixel and the global feature of the region of interest. Moreover, the updating policy ensures the update frequency is adaptive.

The remainder of this paper is organized as follows. In Section 2, the proposed method is described. Section 3 presents the experiment which include experiment setups and experiment results. Then, the conclusion is made in Section 4.

2 Proposed algorithms

The proposed image representation scheme and tracking algorithm will be described in this section.

2.1 Superpixel based appearance model

In this paper, the mid-level cue with structural information captured in superpixels and saliency feature with global color information derived from background subtraction are utilized to represent the object. Each input video frame is partitioned into a set of superpixels, which usually have more regular and compact shape with better boundary adherence. Superpixels are utilized as the primitives for the superpixel-level haar-like feature and combined with the background subtraction.

To construct the superpixel-level feature, $N \cdot t$ superpixels are achieved by segmenting the surrounding region of the object from a set of n training frames using the LSC (Linear Spectral Clustering) algorithm [Li and Chen (2015)]. LSC produces compact and uniform superpixels with low computational costs, which terms as $sp(\mathrm{t,r})(\mathrm{t}=1,2,...\mathrm{m}, \mathrm{r}=1,2,...\mathrm{N}_t)$. Then haar-like feature is applied to represent the superpixels, and noted as $f_t^r = \{\mu_t^r, \Sigma_t^r\}$, where μ_t^r and Σ_t^r are the mean and the covariance of the region $r\text{'}s$ color features, respectively. After collecting the superpixels of all the training frames, mean shift clustering algorithm is employed on the feature pool $F = \{f_t^r \mid \mathrm{t}=1,2,...\mathrm{m}, \mathrm{r}=1,2,...\mathrm{N}_t\}$ and then n various clusters are obtained. Each cluster $clst(\mathrm{i})(\mathrm{i}=1,2,...\mathrm{n})$ has its unique cluster center $f_c(\mathrm{i})$, cluster radius $r_c(\mathrm{i})$ and its cluster members. Each cluster has its own scope to locate and corresponds to the particular place in the training frames. Among the clusters, weight W_i is necessary to assign which is defined as

$$W_i = \frac{S^+(\mathrm{i}) - \mathrm{S}^-(\mathrm{i})}{S^+(\mathrm{i}) + \mathrm{S}^-(\mathrm{i})}, (\mathrm{i}=1,2,...,\mathrm{n}) \tag{1}$$

where $S^+(\mathrm{i})$ and $\mathrm{S}^-(\mathrm{i})$ denote the size of cluster domain overlapping the target area and outside the target, respectively, which could be achieved from the prior label knowledge. Larger value of W_i indicates high confidence to assign to target and vice versa.

In consideration of acquiring more accurate and humanized weight for each superpixel, additional information is adopted, which is derived from the background subtraction. Although the background subtraction algorithms are huge enough, ViBe (visual background extractor) [Barnich and Droogenbroeck (2011)] is adopted because of its efficiency in terms of both computation speed and detection rate. Then, the interest region around the target is selected as the additional information for each superpixel. The selected interest region is processed as the binary image and labels the '0' area as Mark area. If the cluster center of a superpixel is located in the Mark area, its weight will be multiplied by a coefficient c which can be stated as:

$$W_i' = \begin{cases} W_i \bullet c & f_c(\mathrm{i}) \in \mathrm{Mark} \\ W_i & others \end{cases} \tag{2}$$

The clusters are sorted in descending order according to the cluster's weight value and then assemble their cluster center factor $f_c(\mathrm{i})$ as a dictionary D.

Color histograms are extracted at frame level as global feature for appearance model. We only use the Mark area to calculate color histogram rather than the whole frame to reduce the computing cost. The algorithm transfers the image from the RGB (red, green, blue) color space to HSV (hue, saturation, value) color space and quantifies Hue space to 16 bins noted as a vector. Then the difference between two frames is the Euclidean distance between the corresponding vectors named as V.

2.2 Combining vote maps

When the new frame reaches, a surrounding region of the target is extracted and segmented into superpixels, and then the Mark area is labeled by the ViBe algorithm. The superpixels in the same cluster are scored according to 3 factors which are the cluster it belongs to, the distance between this superpixel and its corresponding cluster center, and the cluster's label according to background or target. Additionally, clusters have weights in the feature space. The new frame's cluster center is the other factor to affect the superpixel's score. Therefore, the score of each superpixel is computed as follows:

$$C_r^s = \omega(\mathrm{r},\mathrm{i}) \times \mathrm{W}_t^{'} \times \lambda_l \quad \forall r = 1,2,...N_t, i = 1,2,...n, l = 0,1,2 \tag{3}$$

$$\text{And} \quad \omega(\mathrm{r},\mathrm{i}) = \exp\left(\frac{\left\| f_t^r - f_c(\mathrm{i}) \right\|_2}{r_c(\mathrm{i})}\right) \quad \forall r = 1,2,...N_t, \ i = 1,2,...n, \tag{4}$$

where $\omega(\mathrm{r},\mathrm{i})$ enotes the weighting factor based on the distance between the cluster center in the new superpixel and the cluster center in the feature tool. λ_l is a penalty factor of the cluster and its value is decided by the specified regions which are the target, background and Mark area. By taking the three factors into account, C_s^r is the score for superpixel r at the t-th frame $sp(t,r)$.

The confidence map for each pixel on the current frame is obtained based on the contribution. The pixel in the superpixel $sp(t,r)$ is assigned with the superpixel score C_s^r and the others are set to 0. After constructing the confidence map, the following will discuss the decision procedure of the likely location and the updating policy.

2.3 Motion model

Because of the short interval of the inter-frame, the displacement and the scope of the target in the current frame must move and change smoothly. Here the motion model is assumed as Gaussian distributed: $P(\mathrm{X}_t \mid \mathrm{X}_{t-1}) = \mathrm{N}(\mathrm{X}_t; \mathrm{X}_{t-1}, \phi)$, where Φ denotes a diagonal covariance matrix, which represents the standard deviations for target's location center and scale change. The most suitable candidate is obtained under the particle filter framework, and the mean of the random particles is the position of target's center in the

last frame. The candidate with the highest score is selected as the position of the current frame's target among all the candidates. Candidates' scores are measured using the weights for each pixel according to each superpixel and the difference between the histogram of color, which can described as following:

$$Sum(\text{k}) = \Sigma_r C_r^s \times n_t^r \times V, \ \forall r = 1, 2, ... N_t \tag{5}$$

where n_t^r is the number of the pixels for each superpixel, $Sum(\text{k})$ denotes the score for a certain k candidate and V represents as the difference of the color histogram between the current frame and the last frame. After all, the candidate with the highest score would be chosen and recorded its region and center as the tracking result.

2.4 Appearance model updated with occlusions

Updating the appearance mode is a significant and essential element in the tracking system, which can influence the efficiency, robustness and calculation speed. Updating the object appearance every U frame is frequently-used whose performance is affected by the amount of the frame number. In order to acquire a compromise between the tracking errors and computational cost, a toy experiment is performed and is shown in the Fig. 1. Based on the information in the Fig. 1, tracking error is the least in most situations when the update interval is 1. However, the tracking error is too high when the frame number is greater than 300 and the frame number is from 200 to 220. For the update interval with 10 and 30, the tracking errors fluctuate and their mean values are roughly the same especially when the frame number is greater than 200. However, the average per-frame run time is increasing with the growth of the update interval. In general, a constant update interval is set based on the experiment. However, constant update interval is silliness and adaptive updating scheme which is more effective is applied.
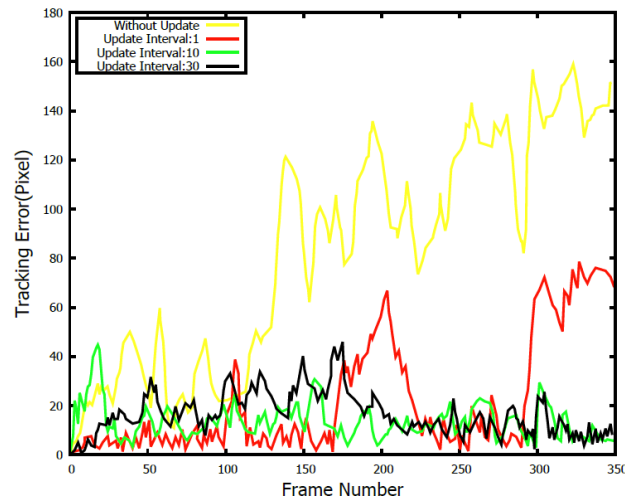
Figure 1: The relationship of tracking errors and update interval

The policy of updating the model is according to threefold factors. Firstly, the score of the candidate is an essential aspect and the score's tendency is noticed instead of the score itself. When the tendency of score is increasing or decreasing, the appearance model is updated. If the score fluctuates along with the arrival of the new frame, the model is kept. Secondly, the model is updated according to the deviation of the color histogram. If the difference is greater than 10% of the quadratic sum of the histogram for the origin model, the model updates and makes a caution as occlusion. If the difference is more than 20%, the object is considered as occluded. Thirdly, the updating frame is not accumulate the series of the adjacent frames, the update is applied using the discrete frames but the adjacent continues frames, which inspired by the coding of the MPEG (Moving Picture Experts Group). For example, the 40th frame need update, the information of the frame 33th, 35th, 37th, 38th, 39th, 40th are accumulated rather than the information from frame 33th to 40th. In this case, the information acquired from the discrete frames are more than the information from continues frames with the equal amount. Additionally, in order to guarantee the model's effectiveness, the model is forced to update every 30 frames, which is an empirical value. In addition, the origin model is reserved for all the tracking schemes.

2.5 The proposed tracking algorithm

The proposed whole tracker would be presented in this part, which is summarized in the Algorithm 1.

Algorithm 1. Main steps of the proposed tracking algorithm

Initialization:

For *t=1 to m (e.g., m=4)*

    1. Initialize arguments of the proposed algorithm in frame t.
    2. Segment the surrounding region and extract their features.

End

Discriminative appearance model is achieved by Eq. (1) and Eq. (2).

Tracking:

For *t=m+1* to the end of the sequence

    1. Segment a surrounding region and extract their features.
    2. Compute the voting map using Eq. (3) and Eq. (4).
    3. Sample N candidates with the voting map.
    4. Select the candidate with the highest score as the tracking result.
    5. Detect the occluded station.
    6. Update the appearance model using our specified update scheme.

End.

---

3 Experiments

The experimental setups and the exploratory results are depicted in this section. The algorithm is conducted in MATLAB and runs on Inter Core 2 Duo 2.93 GHz CPU with 2.96 GB memory.

3.1 Experimental setups

We utilize haar-like feature and information derived from background subtraction for each superpixel. The LSC algorithm is applied to segment frames into superpixels where the parameters are set as default values. The ViBe was called using the value in this paper: N=20, R=20, minimum number=2, Φ=16. To build the training dataset for initialization, the first 4 frames are processed using the simple tracker. The threshold for detecting occlusion is set to 0.2 and the forced updating frame is 30.

The proposed algorithm is evaluated on 8 challenging sequences with 8 different methods. The sequences chosen for evaluating the algorithms contain challenges like occlusion, the change of illumination and large movement of the target. The compared algorithms include IVT [Ross, Lim, Lin et al. (2008)], VTD [Kwon and Lee (2010)], MIL [Babenko, Yang and Belongie (2009)], L1 [Bao, Wu, Ling et al. (2012)], TLD [Kalal, Mikolajczyk and Matas (2012)], Frag [Adam, Rivlin and Shimshoni (2006)], SPT [Yang, Lu and Yang (2014)], and Cheng [Cheng (2015)], whose code can be obtained from the authors' homepage. Then, for completeness of the analysis, the proposed tracker is performed on the recent dataset VOT2014 [Kristan, Leonardis, Matas et al. (2015)], which contains 25 sequences. In the VOT2014, there are 38 trackers including generative trackers and discriminative trackers.

3.2 Experimental results

In the first experiment, 8 sequences are used to evaluate the proposed algorithm and these are: bird2, basketball, deer, David, faceoce2, shaking, singer1, and woman. For fair comparison, the parameters of each tracker remain consistent with the code provided by the authors. The average result of 3 runs are adopted as the comparative results, or the results are obtained straightforwardly from the published papers. The two criterions which are center location errors and successful tracking frames are used in this paper, which are shown in Tab. 1 and Tab. 2. Center location error is the average of Euclidean distance between the tracking result and the ground truth and successful tracking frames are the count of successful frame based on evaluation metric of the PASCAL VOC object detection [Everingham, Van, Williams et al. (2010)]. For presentation clarity, the top 2 tracking for each sequence are boldface. It is manifest from the tables that our algorithm achieves the better or similar performance comparing with the other trackers.

In the basketball sequence, the MIL, IVT and L1 methods drift to background area because of the deficient design of non-grid deformation. Our method achieves the second best results and deviates a little from the result of SPT. The Bird2 sequence has the non-grid deformation, pose change and occlusion, our algorithm and SPT works roughly the same. The algorithm from Cheng works the third best in the Bird2 sequence and we owe the success to the good attributes of Superpixels. In Fig. 2(c), DavidIndoor sequence

involves illumination and pose variations. L1 tracks the object in the small rectangle accurately because of its precise design for distance change. Although the LI and Cheng trackers achieve better result in this frame, our method achieves the minimum center location error which is shown in the Tab. 1.

Table 1: Tracking results: center location error (pixel)

sequence	IVT	Frag	MIL	VTD	L1	TLD	SPT	Cheng	Ours
Singer1	12.8	22.1	15.2	4.1	4.6	5.3	5	**3.2**	<u>3.4</u>
Woman	167.5	113.6	122.4	136.6	113	95	17.4	**15.3**	<u>15.6</u>
Faceoce2	31.9	16.7	14.1	10.4	11.1	16	18.7	**7.8**	<u>8</u>
Shaking	12.8	22.1	11.2	**4.1**	103.7	79	66.9	**6.5**	7.3
David	**3.1**	105.5	16.1	13.6	7.6	11.7	11.2	6.2	<u>4.6</u>
Deer	130.8	91.1	66.5	<u>12.3</u>	140.9	96	18.8	**11.2**	12.4
Bird2	119	28	18.3	48.5	19	86	**17**	20	**17**
Basketball	120	14.6	104	11.8	100	136	**6**	13.2	<u>8</u>

Table 2: Tracking results: successful tracking frames

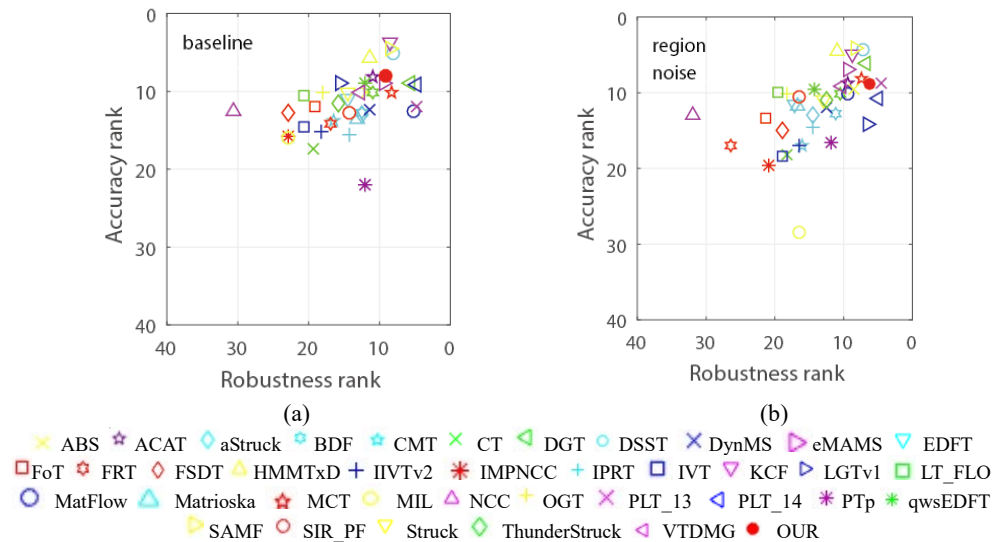| sequence	Total	IVT	Frag	MIL	VTD	L1	TLD	SPT	Cheng	Ours
Singer1	351	328	87	87	**351**	**351**	**351**	347	320	339
Woman	596	49	52	42	36	58	40	<u>298</u>	**330**	290
Faceoce2	820	320	568	580	572	620	631	559	<u>650</u>	**658**
Shaking	364	7	45	198	<u>281</u>	11	43	78	**293**	276
David	462	336	39	231	258	<u>326</u>	239	276	**354**	323
Deer	71	7	11	15	36	6	11	23	**41**	37
Bird2	103	9	44	<u>86</u>	9	76	12	**90**	78	<u>86</u>
Basketball	725	80	512	204	619	30	46	**695**	545	<u>664</u>

To the problem of occlusion and heavy motion in the sequences of woman, deer, and faceocc2 (Figs. 2(d)-2(f)), the algorithms from Cheng and ours perform well, whose results are almost equal. Yet VTD tracker acquires the best result in the Deer sequence. The reason is that it does not distinguish the target from the background and considers some background pixels as parts of the target, thereby getting good results because of the inherent attribute in the sequence. On the other hand, Cheng and our tracker build the appearance model with the target area and miss the similar information of the target. But the trackers are more robust in the other sequences.

Shaking sequence contains illumination variation and pose change with the head shaking. Singer1 clip includes illumination and scale variations which make the trackers drift. As it is shown in the Fig. 1, VTD, L1, and TLD can track the object quite well except for some errors in the Singer1, ours can also achieve the promising results and works almost in a bar with the design from Cheng. In the shaking sequence, our tracker is better than the traditional trackers which are less reliable except for VTD and Cheng's.

| (a) Basketball | (b) bird2 | (c) DavidIndoor | (d) woman |

| (e) Deer | (f) Faceocc2 | (g) Shaking | (h) singer1 |

IVT ——MIL——Frag——L1——TLD——SPT——Cheng——VTD——Ours
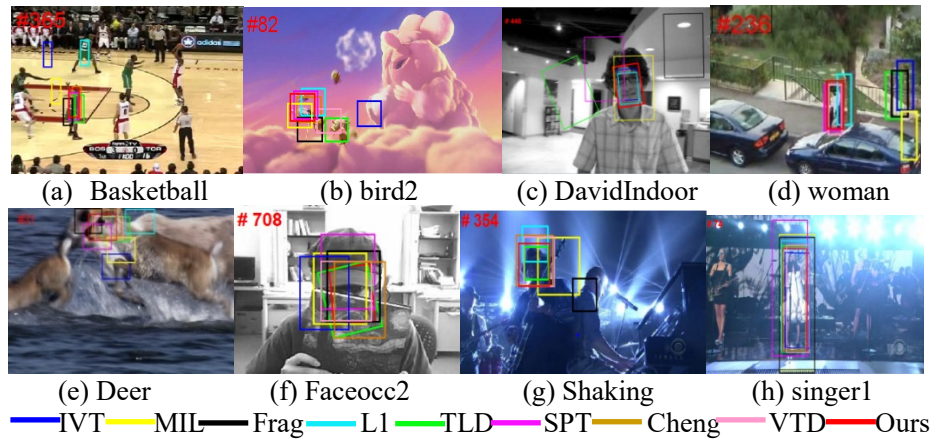
Figure 2: Tracking results with comparisons to the state of the art trackers

| (a) | (b) |

ABS ✩ ACAT ◇ aStruck ✿ BDF ✪ CMT ✕ CT ◁ DGT ○ DSST ✕ DynMS ▷ eMAMS ▽ EDFT
▢ FoT ✿ FRT ◇ FSDT △ HMMTxD ✚ IIVTv2 ✳ IMPNCC ✚ IPRT ▢ IVT ▽ KCF ▷ LGTv1 ▢ LT_FLO
○ MatFlow △ Matrioska ✩ MCT ○ MIL △ NCC ✚ OGT ✕ PLT_13 ◁ PLT_14 ✳ PTp ✳ qwsEDFT
▷ SAMF ○ SIR_PF ▽ Struck ◇ ThunderStruck ◁ VTDMG ● OUR

Figure 3: The accuracy-robustness ranking plots with respect to the two experiments

The comparison in the VOT2014 is displayed by visualizing the ranking results in the Fig. 3. Each tracker's AR rank is explained as a point which stands for its independent scores for accuracy and robustness. The experiments with the overall performance in the Fig. 3 contain the baseline and region noise experiments. A tracker is determined as good when it is close to the top-right corner. Except for the overall evaluations, per-visual-attribute experiments which contain camera motion, illumination change, motion change, occlusion, size change and no degradation are shown in the Fig. 4. The proposed tracker shows superior results under the situation of baseline and region noise, although it is not the top results in the comparison. The top trackers are correlation filter-based algorithm or deep learning algorithm while ours performs well in comparison with the state-of-the-art algorithms. For the videos with occlusion attribute, the proposed tracker ranks 1st. For

the motion change and camera motion attributes, our tracker rank 5th. Generally speaking, ours cannot reach the top but gives a comprehensive result and shows great in the traditional trackers.
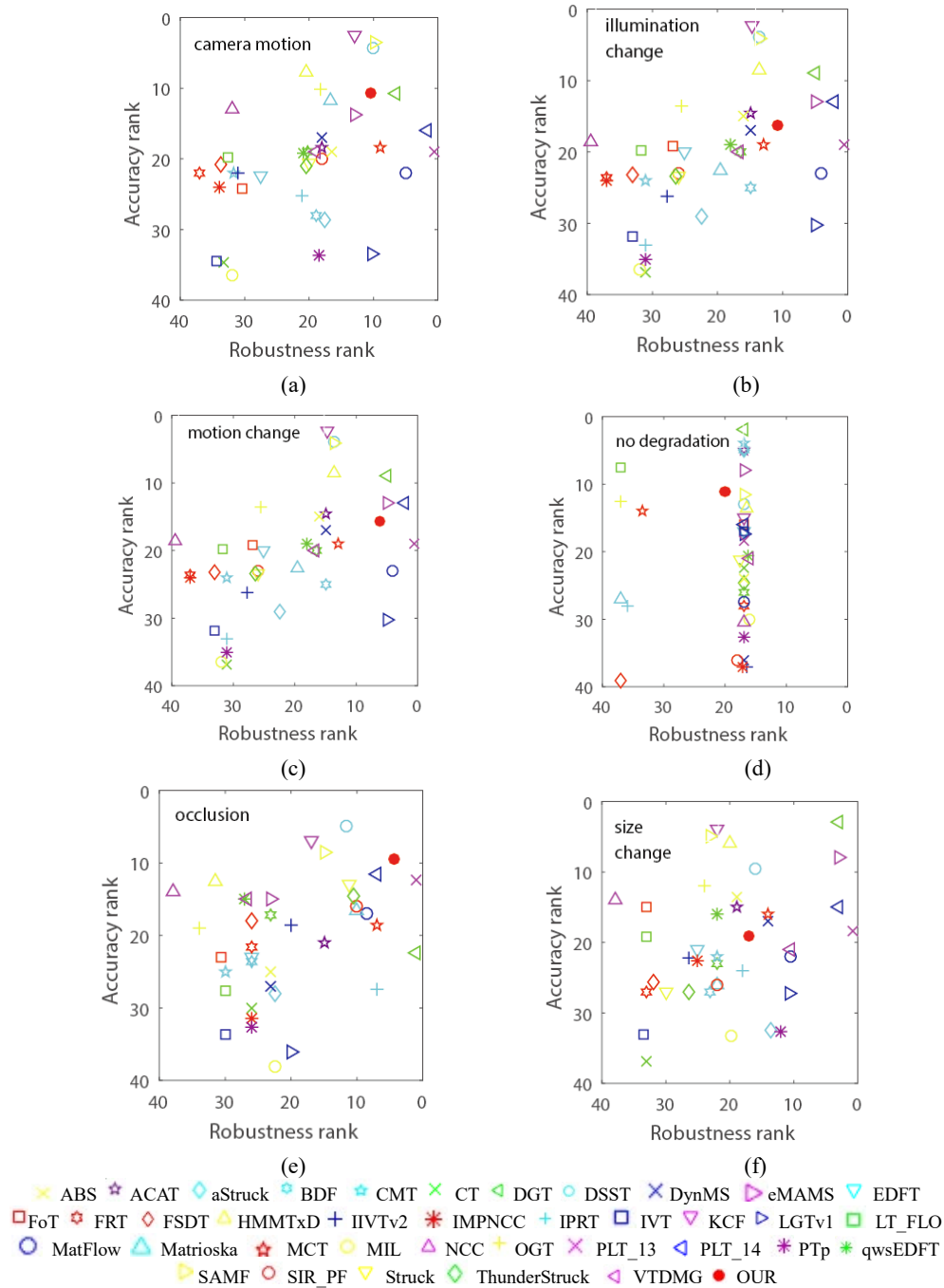
Figure 4: The accuracy-robustness ranking plots with respect to different attributes

4 Conclusion

In this paper, an adaptive superpixel-based tracking approach is proposed. We build an effective appearance model with superpixel which provides the flexible and effective middle level cues and background subtraction which supports the interesting area. It distinguishes the foreground target from the background. The unique updating policy guarantees the adaptive performance. The tracking algorithm could handle the difficulties like occlusion, transformation and illumination change. The comparison results and evaluations demonstrate the proposed tracker performs robustly comparing with the state-of-the-art trackers.

References

Adam, A.; Rivlin, E.; Shimshoni, I. (2006): Robust fragments-based tracking using the integral histogram. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 798-805.

Babenko, B.; Yang, M. H.; Belongie, S. (2009): Visual tracking with online Multiple Instance Learning. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 983-990.

Bao, C.; Wu, Y.; Ling, H.; Ji, H. (2012): Real time robust L1 tracker using accelerated proximal gradient approach. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1830-1837.

Barnich, O.; Droogenbroeck, M. V. (2011): Vibe: a universal background subtraction algorithm for video sequences. *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1709-1724.

Cheng, X. (2015): Robust superpixel tracking with weighted multiple-instance learning. *IEICE Transactions on Information and Systems*, vol. E98, no. D, pp. 980-984.

Dong, X.; Tsoi, A. C.; Lo, S. L. (2014): Superpixel appearance and motion descriptors for action recognition. *International Joint Conference on Neural Networks*, pp. 1173-1178.

Everingham, M.; Van Gool, L.; Williams, C. K. I.; Winn, J.; Zisserman, A. (2010): The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303-338.

Gao, Z. G.; Xia, S. X.; Zhang, Y. K.; Yao, R.; Zhao, J. Q. et al. (2018): Real-time visual tracking with compact shape and color feature. *Computers, Materials & Continua*, vol. 55, no. 3, pp. 509-521.

Giordano, D.; Murabito, F.; Palazzo, S.; Spampinato, C. (2015): Superpixel-based video object segmentation using perceptual organization and location prior. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4814-4822.

Kalal, Z.; Mikolajczyk, K.; Matas, J. (2012): Tracking learning detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409-1422.

Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M.; Pflugfelder, R. et al. (2015): The visual object tracking vot 2015 challenge results. *IEEE International Conference on Computer Vision Workshops*, pp. 1-23

Kwon, J.; Lee, K. M. (2010): Visual tracking decomposition. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1269-1276.

Li, N. Z.; Chen, N. J. (2015): Superpixel segmentation using linear spectral clustering. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1356-1363.

Ren, X.; Malik, J. (2003): Learning a classification model for segmentation. *Proceedings Ninth IEEE International Conference on Computer Vision*, vol. 1, pp. 10-17.

Ross, D. A.; Lim, J.; Lin, R. S.; Yang, M. H. (2008): Incremental learning for robust visual tracking. *International Journal of Computer Vision*, vol. 77, no. 1, pp. 125-141.

Shen, J.; Du, Y.; Wang, W.; Li, X. (2014): Lazy random walks for superpixel segmentation. *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1451-1462.

Wang, J.; Yagi, Y. (2014): Many-to-many superpixel matching for robust tracking. *IEEE Transactions on Cybernetics*, vol. 44, no. 7, pp. 1237-1248.

Wang, Y.; Zhao, Q. (2015): Superpixel tracking via graph-based semi-supervised SVM and supervised saliency detection. *IEEE International Conference on Multimedia and Expo*, pp. 1- 6.

Yang, F.; Lu, H.; Yang, M. H. (2014): Robust superpixel tracking. *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1639-1651.