Robust Mean Shift Tracking Based on Refined Appearance Model and Online Update

Wangsheng Yu(✉), Zhiqiang Hou, Xiaohua Tian, and Dan Hu

Information and Navigation College, Air Force Engineering University, Xi'an, China
xing_fu_yu@sina.com

Abstract. In this paper, a robust mean shift tracking algorithm based on refined appearance model and online update strategy is proposed. The main idea of the proposed algorithm is to construct a more accurate appearance model and design an online update strategy. At the beginning of the tracking, the simple mean shift tracking algorithm is applied on the first few frames to collect a set of target templates, which contains both foreground and background of the target. During the model construction, simple linear iterative clustering (SLIC) algorithm is exploited to obtain the superpixels of the target templates, and the superpixels are further clustered to classify the background from foreground. A weighted vector is then obtained based on the classified background and foreground, which is utilized to modify the kernel histogram appearance model. The following frames are processed based on the mean shift tracking algorithm with the modified appearance model, and the stable tracking results with no occlusion will be selected to update the appearance model. The concrete operation of model update is the same as model construction. Experiment results on challenging test sequences indicate that the proposed algorithm can well cope with both appearance variation and background change to obtain a robust tracking performance.

1 Introduction

With the development of computer vision and multimedia technology, visual tracking has been widely applied in many civil and military fields. It plays more and more important roles in improving the efficiency of industrial and agricultural production, as well as the performances of weapons and equipment. In the past decade, visual tracking technology made much progress [1]. However, there exists limitation in most of the tracking methods because they are designed for the specific or relatively simple situations [2].

As one of the famous tracking methods, mean shift tracking attracts many attentions for the well-developed theory, simple course, outperformed performance and easy to implement. The mean shift algorithm was firstly proposed by Fukunaga *et al.* [3] to cope with the data analysis. Cheng [4] introduced it into the fields of image processing and computer vision. Bradski [5]

This research was supported by National Natural Science Foundation of China (No. 61175029 and No. 61473309)

H. Zha et al. (Eds.): CCCV 2015, Part I, CCIS 546, pp. 114–123, 2015.
DOI: 10.1007/978-3-662-48558-3_12

developed its application in face tracking and proposed a continuously adaptive mean shift (CAMSHIFT) algorithm. Comaniciu *et al.* summarized the mean shift as a robust approach to feature space analysis [6] and successfully applied it to visual tracking [7]. Collins [8] discussed the limitation of the scale adaptation in original mean shift tracking algorithm, and proposed a modified one in scale space. Zivkovic *et al.* [9] and Ning *et al.* [10] also discussed the scale and orientation estimation for mean shift tracking. Comaniciu *et al.* [7] proposed a background weighted kernel to increase the discriminative of foreground, which is widely applied in mean shift tracking [11]. Some other fruitful works have also promoted the development of mean shift tracking, such as online selection a discriminative feature [12], novel histogram for model representation [13], and cross-bin based similarity measure [14].

In this paper, we centralize on how to improve the robustness of the mean shift tracking algorithm. Wang *et al.* [15] proposed a superpixel method ro refine the appearance model, which improves the model precision. Zhuang *et al.* [16] proposed a discriminative sparse appearance model for visual tracking, which improves the tracking success rate in most cases. We based on our tracking algorithm partially on these two methods. Firstly, a clustering analysis method is introduced into classifying the background from foreground, which may distinctively improve the precision of the appearance model. Secondly, a simple but effective rule is proposed to select the stable results to update the appearance model. Finally, a set of challenging test sequences is exploited to verify the robustness of the proposed tracking algorithm.

2 Mean Shift Tracking Algorithm

In mean shift tracking algorithm, the target is usually defined as a rectangular or ellipsoidal region, and represented by a color histogram model. Given a target model, the main procedure of mean shift tracking is to iteratively search the best similar target candidate along the gradient ascent direction in feature space. Denote by $q = \{q_u | u = 1, 2, ..., B\}$ the target model, and $p(\mathrm{y}) = \{p_u(\mathrm{y}) | u = 1, 2, ..., B\}$ the target candidate. The similarity between them is defined by Bhattacharyya coefficient

$$\rho(\mathrm{y}) = \rho(p(\mathrm{y}), q) = \sum_{u=1}^{B} \sqrt{p_u(\mathrm{y}) q_u} \tag{1}$$

A Taylor series expansion around the target candidate $p_u(\mathrm{y}_0)$ yields a linear approximation to the coefficient

$$\rho(\mathrm{y}) \approx \frac{1}{2} \sum_{u=1}^{B} \sqrt{p_u(\mathrm{y}_0) q_u} + \frac{1}{2} \sum_{u=1}^{B} \sqrt{p_u(\mathrm{y})} \sqrt{\frac{q_u}{p_u(\mathrm{y}_0)}} \tag{2}$$

Searching the best target candidate is namely to find the $\mathrm{y} = \mathrm{y}_{opt}$ that maximizes the formula (2). By substituting $p_u(\mathrm{y})$ with the kernel-based histogram

representation, formula (2) is then translated into

$$\rho(\mathtt{y}) \approx \frac{1}{2}\sum_{u=1}^{B}\sqrt{p_u(\mathtt{y}_0)q_u} + \frac{C_h}{2}\sum_{i=1}^{N_h}\omega_i K(\|\frac{\mathtt{y}-\mathtt{x}_i}{h}\|^2) \tag{3}$$

where C_h is the normalization constant, B is the total number of the histogram bins, N_h is the total pixels in target candidate region, and $K(x)$ is the kernel profile with a bandwidth of h.

Denote by ω_i the weight of pixel $\mathtt{x}_i$. This weight is calculated by

$$\omega_i = \sum_{u=1}^{B}\sqrt{\frac{q_u}{p_u(\mathtt{y}_0)}}\delta[b(\mathtt{x}_i - u)] \tag{4}$$

where $\delta(x)$ is the Kronecker delta function, $b(\mathtt{x}_i)$ associates the pixel to the histogram bin.

Taking the derivation of Equation (3) with respect to $\mathtt{y}$ and setting $\frac{\partial\rho}{\partial\mathtt{y}} = 0$ yields

$$\mathtt{y}_1 = \frac{\sum_{i=1}^{N_h}\omega_i g(\|\frac{\mathtt{y}_0-\mathtt{x}_i}{h}\|^2)\mathtt{x}_i}{\sum_{i=1}^{N_h}\omega_i g(\|\frac{\mathtt{y}_0-\mathtt{x}_i}{h}\|^2)} \tag{5}$$

where $g(x) = -K'(x)$ is the negative derivative of the kernel profile. Equation (5) is the final iteration formula for standard mean shift tracking.

3 The Proposed Tracking Algorithm

The main idea of this proposed algorithm is to improve the robustness of mean shift tracking by refining the appearance model. The whole algorithm consists of (1) simple tracking, (2) model construction, (3) tracking with refined model, and (4) model update. Fig. 1 gives the tracking flowchart of the proposed algorithm. The following contents of this section will introduce these four parts.

3.1 Simple Tracking

Most of the traditional tracking algorithms construct the appearance model relying only on the target template of the initial frame. However, it is known that more prior knowledge of the target may produce a more accurate appearance model, which is very important to improve the tracking robustness. In this paper, we exploit the simple mean shift tracking algorithm to process the first few frames. During this course, a set of target templates are collected to construct target model. In order to ensure the templates contain both background and foreground, we set the target template four times as large as the predefined target area.

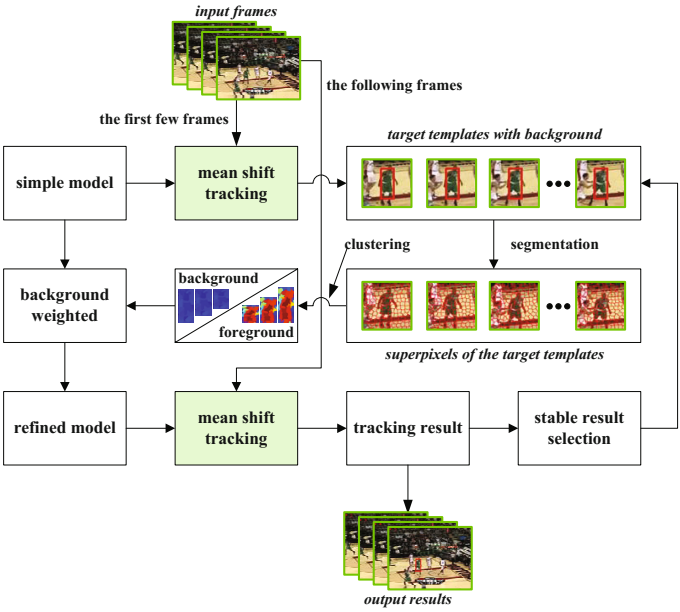

Fig. 1. The flowchart of the proposed tracking algorithm.

3.2 Model Construction

Based on the collected target templates, some image processing algorithms are introduced to construct the appearance model. Firstly, the simple linear iterative clustering (SLIC) [17] algorithm is exploited to obtain the superpixels of the target templates. Then, the mean shift clustering algorithm is applied to cluster the superpixels based on HSI space to classify the background from foreground. Finally, a background weighted vector is calculated from the background and foreground.

SLIC Segmentation. SLIC is a very efficient superpixel algorithm. It groups pixels into perceptually meaningful atomic regions which can be used to replace the rigid structure of the pixel grid. SLIC is very simple to use and understand. By default, the only parameter of this algorithm is N, the desired number of approximately equally sized superpixels. We refer to literature [17] for more details about SLIC segmentation.

Mean Shift Clustering. During the segmentation, the templates are grouped into lots of superpixels. In most cases, the superpixel contains only background or foreground. However, it is hard to classify the superpixels into background and foreground. So, we introduce the mean shift clustering algorithm to further

group the superpixels into background and foreground. The clustering feature is $8 \times 8 \times 8$ histogram in HSI color space, and we should preset the clustering bandwidth h.

Refined Appearance Model. Based on the clusters of background and foreground, it is easy to identify which superpixels belong to background. When the superpixels of target template input, we calculate the distance from each superpixel to the background cluster and foreground cluster, and then a probability whether the superpixel belongs to background (or foreground) is obtained. In this paper, we mark the background superpixel with -1 and foreground superpixel with 1. Based on the aforementioned processing, a probability map (with the same size of target template) valued from -1 to 1 is obtained. We calculate the histogram of the pixels with minus probability value as background histogram of the target template. Denote by $\{q_u|u = 1, 2, ..., B\}$ the normalized kernel histogram model of the target region and $\{o_u|u = 1, 2, ..., B\}$ the normalized background histogram. The minima positive number in $\{o_u|u = 1, 2, ..., B\}$ is marked as $\hat{o}$. The background weighted vector can be calculated by

$$v_u = min(\frac{\hat{o}}{o_u}, 1) \ , \quad u = 1, 2, ...B \tag{6}$$

and the final refined appearance model is $\{q'_u = C \cdot v_u q_u|u = 1, 2, ..., B\}$, where C is the normalization constant.

3.3 Tracking with Refined Model

With the refined model, the mean shift search will obtain a more accurate target location. So, we exploit the mean shift tracking algorithm with the refined appearance model to process the following frames to improve the tracking performance. More details about the tracking course please refer to subsection 3.1.

3.4 Model Update

During the tracking course, the target may undergo appearance variation and background change. In order to cope with these interfering to improve the robustness of long term tracking, we design an online strategy to update the appearance model. We set a rule to estimate whether the tracking result of the current frame is suitable for model update. The rule is as follows. We draw out a tracking result every U_1 frames to calculate the sum value of the weighted map W, if the value is not less than a preset ratio r to that of the last frame, we select it as a sample which will be exploited to update the target model.

The selected target template according to the tracking result is then grouped into superpixels using SLIC algorithm. When the number of the collected target templates equals the preset threshold U_2, the mean shift clustering algorithm is then exploited to renew the background cluster and foreground cluster. So, a new background weighted vector is obtained to refine the kernel histogram model, and the final appearance model is updated.

3.5 Summary of the Whole Algorithm

Based on the aforementioned introduction, we summarize the proposed tracking algorithm as follows.

Algorithm 1 Mean shift tracking based on refined appearance model and online update.

Input: frames with the ground truth of the initial frame.
For $i = 1$ to k (k is a preset parameter not greater than 5);
  Do the tracking using the basic mean shift tracking algorithm;
  Do the segmentation of the target template using SLIC algorithm;
End For.
Cluster the collected superpixels using mean shift clustering algorithm.
Refine the appearance model using the obtained background weighted vector.
For $i = k + 1$ to the last frame;
  Do the tracking using the mean shift algorithm with refined model;
  Pick out the tracking result if it is suitable for model update;
  Update the model if the number of the collected templates equals U_2;
End For.
Output: frames with tracking results marked by bounding boxes.

4 Experiments

To give an objective evaluation of the proposed tracking algorithm, we tested it on the dataset OTB2013 and compared the tracking results with standard mean shift tracking algorithm and its amelioration. The test dataset contains 50 challenging sequences with all kinds of variation from the target and background. Considering the paper length, we only selected 8 representative ones to give a qualitative comparison and quantitative analysis.

During the experiments, the target deformation (sequence *skiing* and *bolt*), occlusion (sequence *subway* and *crossing*), illumination variation (sequence *coke* and *tiger2*), and complicated background (*basketball* and *football1*) bring challenging to the tracking test. The referenced tracking algorithms consist of standard mean shift tracking (MS) [7], scale space mean shift tracking (SMS) [8], E-M shift tracking (EMS) [9], background weighted mean shift tracking (BWMS) [11], and two recent trackers, SPT [15] and DSST [16].

To be mentioned is that there are several parameters should be initialized before tracking. (1) The number of simple tracking frames, $k = 5$. (2) The maximum of superpixels in one target template, $N = 200$. (3) The bandwidth of mean shift clustering, $h = 0.1 \sim 0.3$. (4) The frequency of tracking result selection, $U_1 = 3$. (5) The threshold to select stable tracking result, $r = 0.7 \sim 0.9$. (6) The number of target templates to update model, $U_2 = 5$. The only two parameters need to adjust are h and r. In this paper, we run the tracker

many times to obtain relatively better values, and finally give an experimental evaluation as $h = 0.21$ and $r = 0.82$.

Fig. 2 shows some of the tracking results of the test tracking algorithms. We marked the results of different algorithms with different colored bounding boxes. For the sequences with target deformation, the proposed algorithm obtains relative better results by the refined appearance model. We refer to the frame # 40 in sequence *skiing* and frame # 150 sequence *bolt* for examples. When the target undergoes partial occlusion or fully occlusion, the model stops updating to prevent wrongly update. See the frame # 58 in sequence *subway* and the frame # 40 in sequence *crossing*. As the frame # 610 in sequence *basketball* and the frame # 68 in sequence *football1* show, the refined appearance model relies much more on the foreground of the target and suppresses the background. So, the background change during the tracking affects little to the tracking results. The online model update improves the tracking robustness when the target undergoes illumination change. We refer to the frame # 240 in sequence *coke* and frame # 310 sequence *tiger2* for more details.

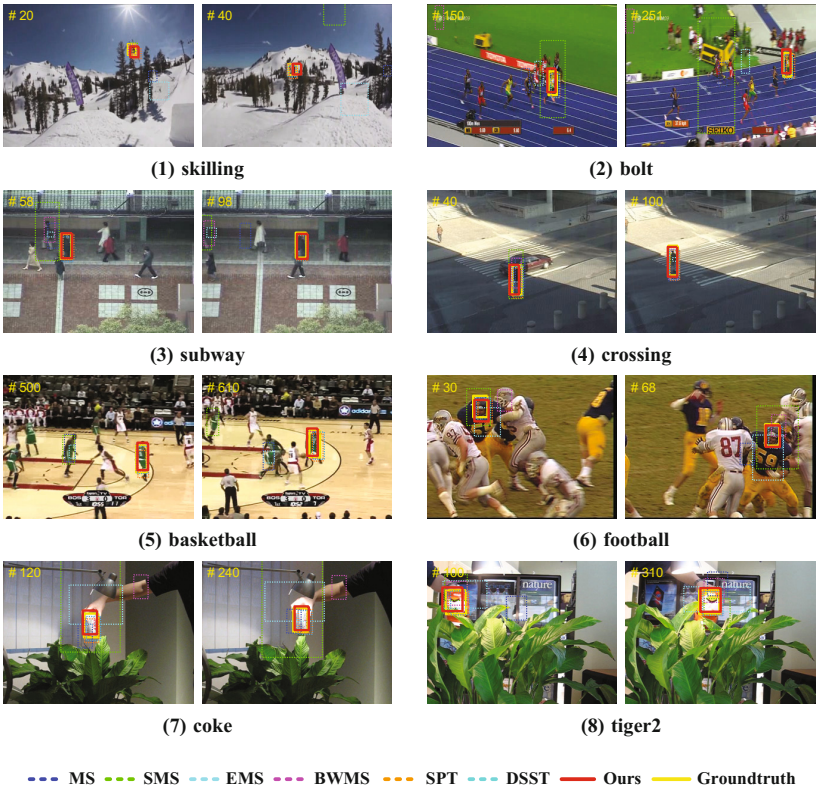

Fig. 2. Tracking results comparison of different algorithms.

We quantitative compared the proposed algorithm with the referenced algorithms using a popular metric, Center Location Error (CLE for short). It measures the error between the center location of tracking window and the ground truth. Certainly small errors in this metric are expected for an optimal tracker. We write down the test results for each algorithm and plot the error comparisons in Fig. 3. More statistic comparisons are detailed in Tab. 1.

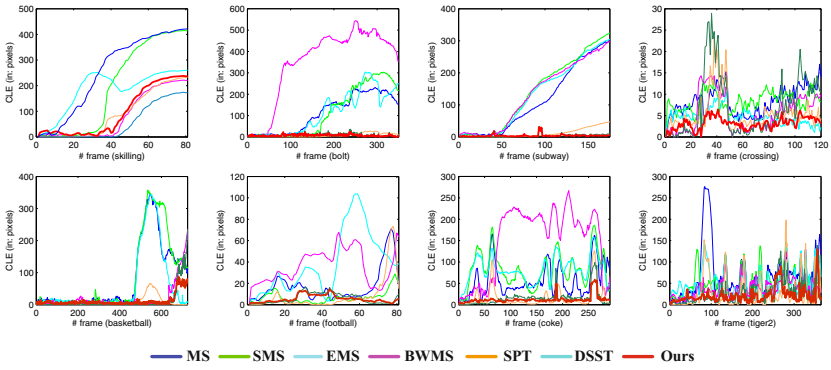

Fig. 3. The Center Location Error (CLE) comparison of different algorithms.

As shown in Tab. 1, the proposed algorithm obtains the smallest Mean Center Location Errors except for the sequence *skiing*, which indicates that the proposed algorithm improves the tracking performance in most cases. To be mentioned is that all the algorithms behave disappointed when tracking the sequence *skiing*.

Table 1. The *Mean Center Location Errors* (in pixels) of different algorithms. **Bold red** font indicates the best performance, green font indicates the second best.

sequences	MS	SMS	EMS	BWMS	SPT	DSST	Ours
skiing	254.8	195.6	185.0	82.0	81.1	**54.2**	96.4
bolt	109.9	115.9	112.1	349.5	**6.9**	8.8	7.3
subway	116.7	140.9	132.4	133.6	10.8	**3.9**	4.1
crossing	8.4	8.3	4.6	5.6	5.5	7.1	**3.2**
basketball	76.3	85.2	59.7	**6.7**	10.4	19.2	11.2
football1	17.1	6.8	38.5	35.8	8.2	5.9	**5.5**
coke	55.9	81.6	80.7	149.9	15.6	15.1	**12.1**
tiger2	71.4	53.6	52.4	40.6	26.3	24.0	**21.6**

5 Discussion

We introduced the proposed tracking algorithm in the former content. A set of tracking experiments on challenging sequences demonstrate that the proposed tracker obtains better performances than traditional trackers in most cases. We conclude the success of the proposed algorithm as three reasons.

Firstly, we exploited the mean shift algorithm as the tracking framework, which enhanced the tracking stability. Compared with the traditional mean shift tracker, we refined the tracking course to adjust to the appearance variation, which is very important for long term tracking in challenging situations.

Secondly, the suppixels based appearance model is validated as a feasible for visual tracking. We introduce this method to construct appearance model to improve the precision of target representation. This appearance model is discriminative between foreground and background, which is very important to suppress the background clutters. Based on this discriminative model, the tracking algorithm improved the performances in target localization.

Thirdly, we designed a strategy for appearance model update during the tracking course. We search the appropriate time for model update and renew the appearance model by clustering analysis of the suppixels of tracking results. This method improves the adaptability of the appearance model, especially in long term tracking tasks.

We keep the parameters unchanged as the former section described during the whole experiments, which indicates that the proposed tracking algorithm is effective in most cases with out parameters tuning. Further tuning of parameters may produce a slightly better performance in some given situations. However, the adopted parameters of the proposed algorithm are relatively better in most cases. The proposed tracker is an integration of mean shift tracking algorithm, suppixels appearance model and model update strategy. It improves the tracking performances at the cost of decreasing the tracking efficiency. The tracking experiments indicate that the proposed tracker runs at an average speed of 3.2 fps. Note that the clustering algorithm is time-consuming and an amelioration of clustering algorithm may improve the tracking speed.

6 Conclusion

We proposed a modified mean shift tracking algorithm with refined appearance model and online update. A background distribution is learned from the clustering analysis of the superpixels of the first few target templates. A background weighted vector is then calculated to refine the initial target model. During the tracking course, the stable tracking results are selected to online update the appearance model. The proposed algorithm improves the tracking performance in most of the cases. However, One of the main drawbacks is that the processing of model refining and updating increases the run time of the whole algorithm.

References

1. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: a benchmark. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2411–2418. IEEE Press, Portland (2013)
2. Smeulders, A., Chu, D., Cucchiara, R., Calderara, S., Dehghan, A., Shah, M.: Visual tracking: an experimental survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(7), 1442–1468 (2014)
3. Fukunaga, F., Hostetler, L.: The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Transactions on Information Theory **21**(1), 32–40 (1975)
4. Cheng, Y.: Mean shift, mode seeking and clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence **17**(8), 790–799 (1995)
5. Bradski, G.: Real time face and object tracking as a component of a perceptual user interface. In: Proceedings of the Fourth IEEE Workshop on Applications of Computer Vision, pp. 214–219. IEEE Press, Princeton (1998)
6. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence **24**(5), 603–619 (2002)
7. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(2), 564–577 (2003)
8. Collins, R.: Mean-shift blob tracking through scale space. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 234–240. IEEE Press, Wisconsin (2003)
9. Zivkovic, Z., Kröse, B.: An EM-like algorithm for color-histogram based object tracking. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 798–803. IEEE Press, Washington (2004)
10. Ning, J., Zhang, L., Zhang, D., Wu, C.: Scale and orientation adaptive mean shift tracking. IET Computer Vision **6**(1), 52–61 (2012)
11. Ning, J., Zhang, L., Zhang, D., Wu, C.: Robust mean-shift tracking with corrected background-weighted histogram. IET Computer Vision **6**(1), 62–69 (2012)
12. Collins, R., Liu, Y., Leordeanu, M.: Online selection of discriminative tracking features. IEEE Transactions on Pattern Analysis and Machine Intelligence **27**(10), 1631–1643 (2005)
13. Birchfield, S., Rangarajan, S.: Spatiograms versus histograms for region-based tracking. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1158–1163. IEEE Press, San Diego (2005)
14. Leichter, I.: Mean shift trackers with cross-bin metrics. IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(4), 695–706 (2012)
15. Wang, S., Lu, H., Yang, F., Yang, M.: Superpixel tracking. In: Proceedings of the International Conference on Computer Vision, pp. 1323–1330. IEEE Press, Barcelona (2011)
16. Zhuang, B., Lu, H., Xiao, Z., Wang, D.: Visual tracking via discriminative sparse similarity map. IEEE Transactions on Image Processing **23**(4), 1872–1881 (2014)
17. Achanta, R., Shaji, A., Smith, K., Pascal, F., Sabine, S.: SLIC Superpixels compared to state-of-the-art superpixel methods. IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(11), 2274–2281 (2012)