#### SPECIAL ISSUE ARTICLE



# Long-term real time object tracking based on multi-scale local correlation filtering and global re-detection

Qi Zhao  $^1\cdot Boxue\ Zhang ^1\cdot Wenquan\ Feng ^1\cdot Zhiying\ Du ^1\cdot Hong\ Zhang ^2\cdot Daniel\ Sun ^3$ 

Received: 26 April 2018 / Accepted: 19 March 2020 / Published online: 30 March 2020 © The Author(s) 2020

## Abstract

This paper investigates long-term visual object tracking which is a complex problem in computer vision community and big data analysis, due to the variation of the target and the surrounding environment. A novel tracking algorithm based on local correlation filtering and global keypoint matching is proposed to solve problems occurred during long-term tracking such as occlusion, target-losing, etc. The algorithm consists of two major components: (1) local object tracking module, and (2) global losing re-detection module. The local tracking module optimizes the conventional correlation filtering algorithm. Firstly, the Color Name feature is applied to increase the color sensitivity. Secondly, a scale traversal is employed to accommodate target scale changes. In the global losing re-detection module, the target losing judgment and global re-detection is realized by keypoint feature models of foreground and background. The proposed tracker achieves the 1st place in the VTB50 test set with 81.3% precision and 61.3% success rate, which outperforms other existing state-of-the-art trackers by over 10%. And it achieves the 2nd place in our Chasing-Car test set with a higher real-time performance 43.2 fps. The experimental results show that the proposed tracker has higher accuracy and robustness when dealing with situations like object deformation, occlusion and target-losing, etc.

Keywords Long-term tracking  $\cdot$  Global keypoint  $\cdot$  Correlation filtering  $\cdot$  Multi-scale tracking  $\cdot$  Computer vision

## Mathematics Subject Classification 68T45

Hong Zhang dmrzhang@buaa.edu.cn

<sup>2</sup> Image Processing Center, Beihang University, Beijing, China

<sup>3</sup> Data61, CSIRO, Canberra, Australia

This study was supported by the National Natural Foundation of China (Grant Number 61772052).

<sup>&</sup>lt;sup>1</sup> School of Electronic and Information Engineering, Beihang University, Beijing, China

## **1** Introduction

Visual object tracking, the problem of locating objects in a video sequence, is one of the central research topics in the field of computer vision [1] and Big Data Analytics (BDA). Recently more and more modern applications, such as precision guidance [2], intelligent surveillance [3], and auto-control system [4], etc., require object tracking result of high accuracy. According to the target changes in the process of tracking, object tracking can be divided into long-term tracking and short-term tracking. Most existing trackers focus on short-term tracking and has achieved excellent performance. However, the long-term tracking problem is still not a fully studied problem [5]. The following problems have been long-standing: First, the model update drift, which is caused by the changes of the appearance or scale of target. The target may have different shapes and color, and also, videos may suffer from noise; Second, target occlusion and losing, which refers to the phenomenon that the target may be blocked by other object, and it is a common situation in real video test; Finally, the complexity of the algorithm. The tracker must be simple and flexible enough, as most applications of target tracking always have very high real-time requirements.

In this paper, all these problems considered simultaneously with a correlation filtering and global keypoint matching based tracking algorithm. Combining the speed advantage of correlation filtering in local tracking with the scale and rotation invariance of keypoint in global detection, an accurate and robust long-term tracker is realized. Correlation filtering mainly plays the role of local tracking. It makes use of the circulant matrix's properties to reduce the computational complexity to ensure the real-time performance. As the target model and update mechanism are not sensitive enough to the detail, target occlusion and losing can't be judged well. To solve this problem, global keypoint features are introduced. Through building foreground/background keypoint model, target occlusion judgments and global target detection is realized. The tracker combines the results of both local and global algorithm to update the target model, and ensure the robustness of the long time tracking.

Experiments on VTB50 [6] and the collected Chasing-Car videos show that the proposed tracker: (i)has higher accuracy and robustness when dealing with situations like object deformation or object occlusion; (ii) outperforms the state-of-the-art trackers especially on the long-term video test; (iii) can achieve excellent real-time performance.

The rest of this paper is organized as follows. Section 2 reviews the recent long-term trackers. Section 3 introduces the proposed tracker in detail. Section 4 presents experimental results on VTB50 and the collected Chasing-Car videos. Section 5 revisits the major contribution of this paper.

## 2 Related work

Object tracking could be divided into short-term and long-term tracking according to the complexity of the tracking process. Short-time target tracking mainly studies the object representation model and the discriminant method. Long-time target tracking focus on the problem of the large-scale appearance change, occlusion and target losing, and also the updating mechanism of the corresponding target model. TLD [7] algorithm uses a tracker combined a detector. In each frame, the tracker and the detector judge the candidate target separately, and both are used for result. It combines different characteristics of the detector and tracker, but parameters has large quantity and need set by specific target. Another method is the multi-expert [8] system. A tracker snapshot is generated as an expert at regular intervals during tracking process. The target is tracked by multiple experts simultaneously to avoid model drift. This method improves the robustness of the tracker, but the computational complexity of the algorithm increased exponentially.

Correlation filters have been widely used in object detection and recognition. Since the operator is readily transferred into the Fourier domain as element-wise multiplication, correlation filters have attracted considerable attention recently to visual tracking due to its computational efficiency. LCT [9], a simple detector based on random forest is adopted to alleviate drift problem. MUSTer [10] develops an additional long-term tracker based on keypoints to re-locate the target object. DLT [11] introduces tracking resumption in correlation trackers using a detector mechanism that re-initializes the tracker upon a target loss identified using an adaptive threshold. PDCT [12] utilizes a superpixel optical flow to construct a predictor to estimate the target motion and internal scale variation.

In recent years, deep learning has made breakthrough and achievements in the field of computer vision [13]. However, it has some difficulties in target tracking. First, object tracking is a model-free problem without priori knowledge, therefore, offline training is not feasible. Second, the target tracking has high real-time requirements, and usually run on low-power computing platforms, such as handheld devices, which limit the complexity of deep learning algorithm. Although there are some quick deep learning trackers, such as GOTRUN [14], the robustness performance of these algorithms are unsatisfactory.

# 3 Method

## 3.1 Method overview

The framework of the tracker is shown in Fig. 1. First of all, the initial frame of the video is used to initialize the local and global model of the target. Then the target position of next frame will be calculated by the correlation filter based on local features, and global features will be used respectively to make a judgment whether the target is lost. If the judgment is 'Lost', the target will be re-detected by global keypoint detector. The model of the target will be updated by the current frame.

# 3.2 Local tracking based on optimized correlation filtering

The principle of the correlation filtering algorithm is the optimization of ridge regression. In the initial frame, different positions in the search area are marked with different probabilities to represent the possibilities of the positions to be a corresponding tar-

Initial Frame Model Local Global Correlation Filter Feature Feature Output Input Update Bounding box Model Video sequence sequence Bounding Key Point Detector Lost Box Re-Detection

Fig. 1 Tracker's framework

get. The probability of the center position is set to 1, and the probability decreases as the distance increases from the center position to the edge. The dense sampling is performed on the whole search area for training the ridge regression classifier:

$$f(x) = w^T x \tag{1}$$

where x represents the image features in the bounding box, and f(x) represents its probability. The optimization problem is constructed by ridge regression, the matrix form is:

$$\min_{w} \sum \|Xw - y\|^2 + \lambda \|w\|^2$$
(2)

Then, the optimum solution is:

$$w = \left(X^H X + \lambda I\right)^{-1} X^H \tag{3}$$

where *X* represents all of the samples in the search area. The candidate regions will overlap each other, so the solution of this problem can be optimized by using the properties of circulant matrix. The objective function is converted to the frequency domain by Fourier transformation:

$$\widehat{w} = \frac{\widehat{x}^* \bigodot \widehat{y}}{\widehat{x}^* \bigodot \widehat{x} + \lambda} \tag{4}$$

In order to avoid the high frequency noise caused by the edge effect in Fourier transformation, a two-dimensional Hamming window is employed to product the features, and smooth the results. During the tracking process, the image feature x of the search region will be transformed to frequency domain by Fourier transformation, and the frequency domain response  $\hat{y}$  will be calculated by the elementwise product between  $\hat{x}$  and  $\hat{w}$ .

$$\widehat{y} = \widehat{x} \bigodot \widehat{w} \tag{5}$$

The spatial response y is obtained through the inverse Fourier transformation, as shown in Fig. 2. The position where the response value is the largest is the position of target.



Fig. 2 The spatial response map of the target



Fig. 3 The feature combines the Color Name feature with the FHOG feature

Features of image intensity are commonly used in correlation filter trackers, such as HOG, FHOG [8,15,16], but these features cannot express the color characteristics of images. In our algorithm, the Color Name feature is employed as a supplement, which is proposed by Van et al [17], to achieve color sensitivity. The Color Name feature differs from other common HSV and LAB, while it contains a total of 11 basic colors which are learned from real-world images, as shown in Fig. 3.

Besides the expression of the color characteristics, another problem is that the scale of the target always changes in the video. In our algorithm, a multiscale traversal algorithm is used to find the changed target scale. First, a larger scale change coefficient  $\alpha_l$  is chosen to carry out the traversal of the scale space to roughly determine the target scale:

$$S_{\pm i} = S_0 \cdot \alpha_l^{\pm i}, i = 1, \dots, n \tag{6}$$

where  $S_0$  represents the initial scale in the model. The scale has the largest probability between the different response diagrams will be the current scale of the target:

$$S_{l} = S_{Index} \mid Index = \underset{j}{argmax} \left\{ max\left(y_{j}\right) \right\}, \ j = 0, 1, \dots, 2n$$
(7)

where  $y_i$  represents the probability response map in different scales.

Then, for a more accurate scale, changing the scale change coefficient  $\alpha_l$  to a smaller one and traverse again, and take  $S_l$  instead of  $S_0$ . With this method, the tracker can be well adapted to the change of the target scale. Meanwhile, the amount of calculation is much less than the traditional method with a constant  $\alpha$ , which guarantee the real-time performance.

#### 3.3 Global detection based on keypoint matching

The local tracking based on optimized correlation filter cannot deal with the problem of occlusion or lost, which are common situations in video target tracking.

Instead of the detection-based method, in our algorithm, the foreground and background model is constructed by the keypoint features and the classifier is replaced by keypoint matching. The advantage of this method is that the keypoints represent the features in local neighborhood of each image point, and the feature extraction will be finished in a full-image-traversal time, which avoids the high computational cost in the detection-based method. In addition, occlusion discrimination and global detection are all performed by keypoint matching without training any classifier model, which also reduces the computational complexity of the algorithm. In our method, ORB feature [18] is used for a better real-time performance.

To judge the occlusion/loss of the target, during a long time target tracking, the tracker extracts the global keypoints of the image, establishes the foreground and background model respectively. Then, both a forward and reverse keypoint matching are performed between the model and the current frame. According to the matching results, each point is marked foreground-matched or background-matched. Then the tracker determines whether the current result is missing or occlusion. If the number of matched points with the background model exceeds the threshold, the tracker will judge the target occluded or loss in this frame:

$$\frac{Number(M \cap B)}{Number(M)} > \sigma$$
(8)

where *M* represents the matching results, *Number*  $(M \cap B)$  refers to the number of matched points with background model,  $\sigma$  is the threshold.

When the target get lost, the global re-detection will be started. The position of target is located by the weight center of corresponding keypoints between the target keypoint model T and the current keypoints set  $C_t$ . The scale and rotation variation of the target should be estimated at first by corresponding keypoint pairs. As an instance shown in Fig. 4, k is a keypoint in T and k' is the corresponding keypoint in  $C_t$ . The scale and rotation variation can be estimated by:

$$S_{change} = \frac{1}{N^{k'}} \sum_{i=1}^{N^{k'}} \frac{s^{k'_i}}{s^{k_i}}, \ O_{change} = \frac{1}{N^{k'}} \sum_{i=1}^{N^{k'}} \left( o^{k'_i} - o^{k_i} \right)$$
(9)

where  $N^{k'}$  is the total number of keypoints.  $s^k$ ,  $s^{k'}$  refer to the scale of k and k',  $o^k$ ,  $o^{k'}$  refer to the orientation of k and k'.



Fig. 4 Left: keypoints of target model. Right: corresponding keypoint of current frame

In order to eliminate the effect of matching errors in the re-detection, when calculating the center of the target, different weights are assigned to different feature points to calculate the center of target. It is assumed that mismatched keypoints are usually far away from the previous frame target center, and changes between the two target centers of adjacent frames is in a certain range. The weights are set according to the distance from each current keypoint to the target center of the previous frame:

$$w^{k'} = g\left(\left\|p_{t-1}^t - p_t^{t,k'}\right\|\right) \mid g(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}}$$
(10)

where  $p_{t-1}^t$  is the target position in the previous frame and  $p_{t-1}^{t,k'}$  is the position of keypoint k'. In this method, mismatched keypoints are assigned smaller weights since their long distances to the target center. On the contrast, matched keypoints are assigned larger weights. Then, the target center is estimated by:

$$p_t^t = \sum_{i=1}^{N^{k'}} w^{k'_i} \cdot p_t^{t,k'_i}$$
(11)

where  $w^{k'_i}$  is the normalized weight.

#### 3.4 Model updating mechanism

After the local tracking and global detection, the tracker will fuse both the results as the final result. Meanwhile, the target model and the discrimination model will be updated according to the result of the occlusion judgment, as shown in Fig. 5. The tracker updates the local correlation filter using the incremental updating method while



Fig. 5 Flowchart of model updating mechanism

the global keypoint matching detector is updated by the frame-by-frame forgetting method.

The complete Long-term object tracking algorithm is presented below:

- 1. Input: Input each frame of the video sequence and the initial target position in the first frame.
- 2. Algorithm initialization: Extract FHOG and CN features of the area around the target in the first frame. The Fourier transformation is used to train the parameters of the correlation filter tracker in frequency domain. Then the model is constructed by the keypoints of foreground and background.
- 3. Correlation filter local tracking: In each frame of input sequence, FHOG and CN features are extracted in multiple scales. The frequency response probability map of each frame is calculated by Fourier transformation. The scale of the maximum point of the probability response is the changed scale of the target while its location is the new position of the target.
- 4. Judgment of occlusion/lost: Extract the keypoint set of foreground and background of each fame based on the position calculated by step 3. If the keypoint set match the model well, turn to step 6. If not, turn to step 5.
- 5. Global re-detection: If the judgment is lost or complete occlusion, the current frame will be marked lost and the next frame will be used to locate the target. If the judgment is part occlusion, the position, scale and rotation of the target will be calculated by keypoints to initialize the correlation filter tracker.
- 6. Model updating: If the target is well matched, update the correlation filter tracker and keypoint model. The parameters in the correlation filter will be updated incrementally, while the keypoint model will be replaced by the current frame model.
- 7. End of tracking: If it's not the last frame, repeat step 3-6.

# 3.5 Evaluation criterion

The performance evaluation of object tracking algorithm mainly includes several aspects: accuracy, robustness and real-time performance. The most common evaluation criterion is Location Error and Overlap Rate.

(a) Location error

The location error is employed to estimate the distance between tracking result and target's actual position. The tracking will be marked success, if the distance is shorter

than the distance threshold  $\sigma_d$ . And the score is calculated by:

$$Score = \frac{N_{success}}{N_{all}} \mid Distance = \sqrt{\left\|C_{alg} - C_{GT}\right\|^2}$$
(12)

where  $C_{alg}$  and  $C_{GT}$  represent coordinate of tracking result and actual target position. (b) Overlap rate

The overlap rate is defined by:

$$Ratio = \frac{|B_{alg} \bigcap B_{GT}|}{|B_{alg} \bigcup B_{GT}|}$$
(13)

where  $B_{alg}$  is the bounding box (top-left corner, width and height) of tracking result and  $B_{GT}$  represents the bounding box of actual target.  $\bigcap$  and  $\bigcup$  represent the intersection and union of two areas. If the ratio is larger than the overlap threshold  $\sigma_r$ , the tracking is marked success.

In our algorithm, the Precision Plot and Success Plot are both used for evaluation. The Precision Plot shows different location error under different distance thresholds  $\sigma_d$ , and the Success Plot shows overlap rates under different overlap thresholds  $\sigma_r$ .

## 4 Experiments

This section introduces the experimental setup and experimental results. In order to test the accuracy, robustness and real-time performance of our method, the experiment uses the public test set VTB50 and a collected Chasing-Car test set which includes five Chasing-Car videos taken on a helicopter. Those two test sets cover various situations that may occur in target tracking, such as illumination change, scale change, target deformation, fast motion, target occlusion, background clutter and so on. We use the test criteria presented in Visual Tracking Benchmark [6] for algorithmic comparison. Detailed test criteria are described in [6]. In the experiments, the padding of the searching area in correlation filtering was set 2.5, while the learning rate was set 0.05. The stride of the rough scale searching was 0.2, and it was set 0.1 in the accurate searching. The threshold in the keypoint matching was set 0.4, and the target will be marked lost if matched points are less than 5.

The proposed tracker is implemented in C++ with OpenCV library, and test on a PC with Intel I5-4460 3.20 GHz CPU and 8G RAM.

#### 4.1 Results on VTB50

The first test dataset is VTB50, shown in Fig. 6, which is the most commonly used test dataset in video object tracking. This dataset contains 50 videos with marked targets and the test include illumination change, scale change, occlusion, deformation, blur, rapid motion, rotation, out of view, low resolution etc. The targets include cars, pedestrians, faces, toys etc.



Fig. 6 VTB50 dataset



Fig. 7 Tracking results of different methods on VTB50. (Left) Lemming. (Middle) Skating. (Right) Deer

Figure 7 shows the tracking results of the proposed method and other state-of-art methods in three different videos of VTB50. Our algorithm performs well in all of the situations while other methods appear varying degrees of loss and shift. Figure 8 shows the precision and success rate tested on the VTB50 video, include the whole test set, occlusion and out-of-view situations. The legend in each plot list the top10 methods of each test. The details of precision and success rate comparisons are shown in Table 1. It can be seen that the proposed tracker achieves the best results in all of the tests. In the success plot on the overall test, the proposed tracker outperforms KCF, the second place, by over 10%. As for the occlusion and out-of-view tests, the success rates of other methods decreased a lot compared to the overall success rate, while our algorithm still maintains a high success rate. So it can conclude that the proposed tracker has an excellent performance in dealing with occlusion and target losing compared with other state-of-art trackers.

# 4.2 Results on Chasing-Car videos

In order to test the tracker's performance in long-term tracking, we collected an additional test set, which contains 5 long Chasing-Car videos, as shown in Fig. 9. All of those videos are taken on a helicopter, so there will be a lot of target occlusion, loss,



Fig. 8 Experiment results on VTB50. The first row is the precision plot and the second row is the success plot. From left to right are results for 3 attributes: the whole test set, occlusion and out-of-view. Our trackers provide consistent improvements compared to state-of-art trackers

and large-scale changes in target scale rotation. These conditions make the test a great challenge to the accuracy and robustness of the tracker. The resolution of the videos is  $480 \times 320$  and frame number of each is 920. We compare the proposed tracker with 4 state-of-art trackers [7,15,16,19].

Figure 10 shows the tracking results of different methods, it can be seen that there is a lot of rapid motion, scale change and target occlusion. Figure 11 shows the accuracy and success rate in this test dataset. Although the MDNet tracker is superior to our tracker in the accuracy, it is computationally complex and slow. Table 2 is a detailed algorithm comparison. Our tracker achieves 0.809 scores in the center location error score, and 0.610 scores in overlap score which behind the MDNet tracker but outperforms other three trackers by over 0.1 scores. Our tracker is about 60 times faster than the MDNet algorithm, while this deep learning method need online training and updating with high computational complexity. Therefore, the proposed tracker balances the performance and computational complexity, which is more feasible compared to other state-of-art trackers.

### **5** Conclusion

This paper presents a novel target tracker based on correlation filtering and keypoint matching. The correlation filter has been improved by the combined feature and a multi-scale detect algorithm. Meanwhile, a global keypoint matching algorithm is applied for the judgment of occlusion and target re-detection when the target lost. We perform comprehensive experiments on two data sets: VTB50 and our Chasing-Car data set. The proposed tracker achieves the 1st place in the VTB50 test set with 81.3% precision and 61.3% success rate outperforms other existing state-of-art trackers by

T										
	Proposed	KCF	SCM	ASLA	VTS	Struck	VTD	TLD	CXT	CSK
Precision										
Scale variation	0.758	0.668	0.672	0.552	0.582	0.639	0.597	0.509	0.550	0.503
Occlusion	0.808	0.740	0.640	0.460	0.534	0.564	0.545	0.554	0.494	0.500
Out of view	0.737	0.553	0.429	0.333	0.455	0.539	0.462	0.557	0.510	0.379
Illumination variation	0.757	0.712	0.594	0.517	0.573	0.558	0.557	0.537	0.501	0.481
Deformation	0.810	0.745	0.586	0.445	0.487	0.521	0.501	0.485	0.422	0.476
Overall	0.813	0.737	0.649	0.532	0.575	0.656	0.576	0.553	0.575	0.545
Success Rate										
Scale variation	0.582	0.413	0.518	0.452	0.400	0.425	0.405	0.421	0.389	0.350
Occlusion	0.598	0.500	0.487	0.376	0.398	0.413	0.403	0.416	0.369	0.365
Out of view	0.614	0.483	0.361	0.312	0.443	0.459	0.446	0.479	0.427	0.349
Illumination variation	0.581	0.478	0.473	0.429	0.429	0.428	0.420	0.399	0.365	0.369
Deformation	0.593	0.533	0.448	0.372	0.368	0.393	0.377	0.378	0.324	0.343
Overall	0.613	0.506	0.499	0.434	0.416	0.474	0.416	0.407	0.426	0.398

Table 1Precision comparison on VTB50



Fig.9 Chasing-Car test set. This test set contains 5 different Chasing-Car videos. In each video, the target car has undergone different situations including occlusion, target losing, significant scale and rotation change



Fig. 10 Cars tracking results



Fig. 11 Experiment results on Chasing-Car test set. Left: the precision plot. Right: the success plot

• •					
Tracker	Proposed	MDNet	KCF	TLD	СТ
Center location error score	0.809	0.894	0.690	0.673	0.553
Overlap score	0.610	0.619	0.513	0.487	0.407
FPS	43.2	0.70	67.7	14.4	41.3

Table 2 Algorithm comparison on Chasing-Car test set

over 10%. And it achieves the 2nd place in our Chasing-Car test set with a higher real-time performance 43.2 fps, which is 60 times faster than the first tracker MDNet. The experimental results show that the proposed tracker is much more robust in dif-

ferent situations like target scale change, target occlusion, target losing, and target deformation. It outperforms over existing state-of-art trackers, and also it proves that the combination of global and local information is a more feasible method to improve the tracker's performance.

#### **Compliance with ethical standards**

Conflict of interest The authors declare that they have no conflict of interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

## References

- 1. Yilmaz A, Javed O, Shah M (2006) Object tracking: a survey. ACM Comput Surv 38:13
- Wilson DB, Göktoğan AH, Sukkarieh S (2016) Vision-aided guidance and navigation for close formation flight. J Field Robot 33(5):661–686
- Filonenko A, Kurnianggoro L, Jo KH (2017) A fuzzy model-based integration framework for visionbased intelligent surveillance systems. In: 2017 IEEE international conference on mechatronics (ICM). IEEE, pp 358–361
- Zhang ZF, Zhai YS, Su YL et al (2016) Research on lamp auto-control system based on infrared image processing. Opt Int J Light Electron Opt 127(3):1144–1147
- Rochan M, Rahman S et al (2016) Weakly supervised object localization and segmentation in videos. Image Vis Comput 56:1–12
- Wu Y, Lim J, Yang M-H (2013) Online object tracking: a benchmark. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2411–2418
- Kalal Z, Mikolajczyk K, Matas J (2012) Tracking-learning-detection. IEEE Trans Pattern Anal Mach Intell 34(7):1409–1422
- Zhang J, Ma S, Sclaroff S (2014) Meem: robust tracking via multiple experts using entropy minimization. In: European conference on computer vision, Springer, pp 188–203
- Ma C, Yang X, Zhang C, Yang MH (2015) Long-term correlation tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5388–5396
- Hong Z, Chen Z, Wang C, Mei X, Prokhorov D, Tao D (2015) Multi-store tracker (muster): a cognitive psychology inspired approach to object tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 749–758
- Raju PM, Mishra D, Gorthi RKSS (2019) Detection based long term tracking in correlation filter trackers. Pattern Recogn Lett 122:79–85
- Liang N, Wu G, Kang W, Wang Z, Feng DD (2018) Real-time long-term tracking with predictiondetection-correction. IEEE Trans Multimedia 20(9):2289–2302
- He K, Zhang X, Ren S et al (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
- Held D, Thrun S, Savarese S (xxxx) Learning to track at 100 fps with deep regression networks. arXiv preprint arXiv:1604.01802
- Zhang K, Zhang L, Yang M-H (2012) Real-time compressive tracking. In: European conference on computer vision, Springer, pp 864–877
- Henriques JF, Caseiro R, Martins P, Batista J (2015) High-speed tracking with kernelized correlation filters. IEEE Trans Pattern Anal Mach Intell 37(3):583–596

- Van De Weijer J, Schmid C, Verbeek J, Larlus D (2009) Learning color names for real-world applications. IEEE Trans Image Process 18(7):1512–1523
- Rublee E, Rabaud V, Konolige K, Bradski G (2011) ORB: an efficient alternative to SIFT or SURF. In: 2011 International conference on computer vision, IEEE, pp 2564–2571
- Nam H, Han B (2016) Learning multi-domain convolutional neural networks for visual tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4293–4302

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.