Robust Visual Tracking Based On Convolutional Sparse Coding

Yun Liang, Dong Wang*
College of mathematics and Informatics, South China Agricultural University, Guangzhou, 510642, China
e-mail: wdngdng@163.com

Yijin Chen, Lei Xiao, Caixing Liu
College of mathematics and Informatics, South China Agricultural University, Guangzhou, 510642, China
e-mail: sdliangyun@163.com

Abstract—This paper proposes a target tracking algorithm based on convolutional sparse coding (CSC). This algorithm first uses the CSC to divide the interest region into a smooth image and four detail images with different fitting degrees. Then, the smooth image is tracked to produce the initial result based on the kernel correlation filtering (KCF). According to the initial value of the target area and the linear combination of four detail images, the appearance model of the details is constructed. The matching between samples and the appearance model is performed according to the overlap rate and Euclidean distance to determine the tracking results of the detail images. In the end, the two tracking results including the one based on the smooth image and the other one based on the detail images are linearly combined to determine the final position of the target in the new frame. Many experiments on the video sequences from Tracking Benchmark 2015 show that: our method produces much better result than most of present visual tracking methods.

Keywords: Target tracking; Convolutional Sparse Coding; Kernel Correlation Filtering; detail image; smooth image

I. RELATED WORK

The current tracking methods are mainly divided into two categories: generative tracking methods and discriminative tracking methods. The generative methods are proposed to directly describe the target and find the target sample with the maximum likelihood probability or posterior probability as the current estimate of the real target. The discriminative tracking methods make full use of target information and background. They train a classification model between the information of target and background to judge the target.

The generative tracking methods find the candidate target closest to the model of target as the current estimate in the high-dimensional space described by the target. For example, Black et al. [1] proposed a subspace-based method to calculate the radiation transformation between the current image containing the target and the image reconstructed with feature vector. Based on it, Ross et al. [2] updated the basis of feature space online with SKL. Mei et al. [3] solved the sparsity between the target template and the subspace compose of positive and negative trivial templates through the $l_1$ regularized least squares problem which can overcome problems such as illumination variations and occlusions effectively during the tracking progress. Subsequently, many algorithms [4]-[5] have emerged based on $L_1$ algorithm for continuous improvement and improvement. Although the generative tracking methods made a great breakthrough, they are limited by accurately separating the background and target.

The discriminative methods learn to distinguish between target and background. Such as Support Vector Machine [6], the target and background are distinguished by dividing positive and negative samples. Based on SVM, Hare et al. [7] proposed Structured Support Vector Machine (SSVM) to further enhance its discriminating ability. And based on SSVM, Ning et al. [8] proposed Dual Linear Structured Support Vector Machine (DLSSVM) to enhance the ability to progress high-dimensional features and sampling capabilities. For the discriminative tracking methods, the most famous one is based on correlation filtering [9]-[15]. Bolme et al. [9] proposed MOSSE tracking algorithm which first apply correlation filtering to the target tracking. It performs a convolution calculation on the Fourier domain between the target template and the sample. The larger the response value of the sample, the more similar it is to the target template. Based on MOSSE, Henriques et al. [10] introduced the cyclic matrix and kernel method to the target tracking field, and convolved the Dense Sampling sample in the tracking process with the cyclic matrix formed by the target template in the Fourier domain. And then he proposed the CSK tracking algorithm. Based on CSK, while guaranteeing the speed of tracking, Henriques et al. [11] introduced HOG feature to convert a single channel to multiple channels. And then, Bertinetto et al. [12] integrated HOG feature and color feature to further improve the accuracy of tracking. Based on Spatially Regularized Correlation Filters (SRDCF), Danelljan et al. [13] used the depth feature of single-layer convolution in CNN to replace the HOG feature in SRDCF, improving the accuracy of tracking. After that, Danelljan et al. [14] improved the speed and the stability of the algorithm based on C-COT by reducing model parameters, reducing the number of samples, and adopting a sparse update strategy. Valmadre [15] used an end-to-end approach to treat correlation filtering as a layer in CNN which guaranteed tracking efficiency.

Recently, tracking method with deep features [16]-[19] have become popular. Li et al. [16] proposed a method for learning target perception features, integrating the target perception features with Siamese matching network. This method identify targets with significant appearance changes. Wang et al. [17] proposed SiamFC-based tracker, and its structure is divided into two parts: “rough matching” and “fine matching”. Enhance robustness through generalized training in the rough matching stage and enhance discrimination power through distance learning network in fine matching. Du et al. [18] proposed a tracker to detect target corners. It first uses Siamese network to roughly judge the foreground and background to get the ROI. And then, it uses the relationship between the target template and ROI to highlight the corner regions and enhance features of the ROI for corner detection. In the end, it achieves more accurate bounding box estimation.
Guo et al. [19] proposed a fully convolutional Siamese network structure. The Siamese Subnetwork is first used for feature extraction, and then the classification regression subnetwork is used for the prediction of the target frame, and fast and more accurate segmentation results can be obtained.

All the above tracking methods establish an appearance model of target to achieve the tracking algorithm. So, when encountering the program of model drift, it is very difficult to correct itself in the subsequent tracking process. Especially, when updated the target appearance model, if the online update ability of the model is too strong, it is easy to take the background information around the target as the target, thereby causing overfitting. But, if the online update ability of the model is too weak, when the target is deformed or partially blocked, it leads to loss of tracking, which means underfitting.

This paper divides the target into the smooth image and the detail part based on CSC, and establishes appearance models for the two parts to implement the tracking algorithm. The visual tracking is achieved by combining the tracked results of two appearance models to improve the performance.

II. OVERALL FRAMEWORK

This paper first extracts the interest area based on the target area, which means that the target area is expanded 2.5 times around. Then divide the interest region into a smooth image and four detail images with different fitting degrees by CSC. For the smooth image, the model is initialized and tracked based on the KCF. And for the detail images, linearly combine four detail images with different fitting degrees to establish the appearance model of detail images according to the size of the standard area.

B. Target tracking phase

Based on the target position and the size of the previous frame, expand the 2.5 times area in the new frame of the image, and the area is divided into a smooth image and the detail images based on CSC. Then, we track the two parts separately with template matching. For the smooth image, we track it based on the KCF. The position of the target in the smooth image is determined by calculating the probability value of the target in each position of the interest region in the smooth image. For the detail images, randomly select N samples whose size is same as the appearance model of the detail images in the interest region. The target position of images about the details in the new frame of image is determined by the overlap rate and Euclidean distance value of each sample and the appearance model. Lastly, by linearly combining the result of the smooth image and the detail images, the final position of the tracking target in the new frame is determined.

C. Model update phase

The model update includes the appearance model update of smooth image and detail images. For the smooth image, first, its appearance model is established according to the KCF. And for the detail images, linearly combine four detail images with different fitting degrees to establish the appearance model of detail images according to the size of the standard area.

III. TARGET TRACKING BASED ON CSC

This section, we first introduce how to divide interest region into a smooth image and four detail images with different fitting degrees based on CSC, and establish the appearance model of smooth image and detail images respectively as shown in Fig. 2. For the smooth image, tracking based on the KCF. And for the detail images, first, randomly select N samples in the interest region.
calculate the overlap rate and Euclidean distance of each sample block and the appearance model of the detail images to match the template to get the tracking result of the detail images. Lastly, linearly combine the tracking result of the smooth image and the detail images to get the position of tracking target in a new frame.

A. Initialize the target model based on CSC

In this paper, based on the method proposed by Gu et al. [21], the image is divided into the smooth part and the detail parts by using N filters, as shown in Fig. 3. The smooth part contains the image color and shape features, and the detail parts represents the image edge and texture structure features, as shown in

\[ y = f^s \otimes Z^s_y + Y. \]  

Among them, \( y \) represents the original image, \( f^s \otimes Z^s_y \) represents the smooth part, and \( Y \) represents the detail part. \( f^s \) is a low-pass filter and \( Z^s_y \) is a characteristic diagram.

As shown in Fig. 2, the green frame is the target standard area, and the red frame is the interest region. Then the interest region is divided into a smooth image and four detail images with different fitting degrees based on CSC.

1) Initialize the target appearance model of the smooth image based on the KCF

The establishment of the appearance model of the smooth image is based on the KCF. So, the first step to initialize the tracking target is construct a cyclic matrix \( C(\bar{x}) \) by extracting target features. And then diagonalize the cyclic matrix to obtain the diagonal cyclic feature matrix \( X \) using the Discrete Fourier transform, as shown in

\[ X = F \text{diag}(\bar{x})F^H. \]  

Among them, \( F \) represents the Constant Fourier matrix, and \( FF^H = I \). \( \bar{x} \) is the generated vector after Fourier transform. Then, solve the least squares problem in the Fourier domain based on \( X \) to train the target detector \( \hat{\bar{a}} \), as shown in (3) where \( k^{xx} \) is the first row of the kernel matrix on the Fourier domain and \( \bar{y} \) is the regression target training based on Gaussian kernel function.

\[ \hat{\bar{a}} = \frac{\bar{y}}{k^{xx} + \lambda}. \]  

2) Initialize the target appearance model of the detail images based on the linear combination

As shown in Fig. 3, this paper constructs the model of images about the details by extracting the target area of the detail images. This paper uses 400 filters in the progress of CSC. So, it can get 400 detailed feature maps about tracking targets \((r_i \sim r_400)\). In order to prevent under-fitting or over-fitting, this paper overlap every 100 detail pictures, that are \( R_1 = \sum_{i=1}^{100} r_i, R_2 = \sum_{i=1}^{200} r_i, R_3 = \sum_{i=1}^{300} r_i, R_4 = \sum_{i=1}^{400} r_i \). So, there will be four detailed model \((R_1 \sim R_4)\) with different fitting degrees that track target. Lastly, linear combination of \( R_1, R_2, R_3 \) and \( R_4 \) to get the final appearance model of the detail images which called \( \bar{R} \), as shown in (4) where \( \sum_{i=1}^{4} A_i = 1 \).

\[ \bar{R} = \sum_{i=1}^{4} A_i R_i. \]  

B. Target tracking based on CSC

This section mainly introduces the target tracking based on CSC. For each frame of video sequence, first we use CSC to divide it into the smooth image and the detail images. Then, for the smooth image, we use KCF to track target. For the detail images, the tracking result is determined by calculating the overlap rate and Euclidean distance between the sample and the model. Lastly, by linearly combining the tracking result of the smooth image and the detail images, the final position of target in the new frame is determined.

1) Smooth image tracking based on the KCF

Using the target position of previous image as the center position of current frame, we expand the target region 2.5 times to form the sampling area \( \tilde{Z} \). Then the features of the sampling area to form a feature matrix \( \tilde{X} \). Using kernel functions to calculate the kernel correlation values of cyclic feature matrix \( \tilde{X} \) and \( \tilde{Z} \). It means that \( k^{xz} = C(k^{xz}) \) where \( k^{xz} \) is a row vector from \( X \) and \( \tilde{Z} \) through kernel function operation, and \( C \) is a function that construct a cyclic matrix. So that \( k^{xz} \) is a round function. After that, we use target detectors \( \hat{a} \) and \( k^{xz} \) to do dot multiplication to get the response value matrix \( \tilde{f}(z) \) to each position of the tracking target in the sampling area, as shown in

\[ \tilde{f}(z) = \tilde{a} \otimes k^{xz}. \]  

The larger the value in the response value matrix, the greater the possibility of being target. So, the position corresponding to the maximum value in the response value matrix is used as the target position of the smooth image.
2) Detail images based tracking

For the tracking target of the detail images, we randomly select 400 samples (s1 ~ s400) which size is the same as the target model in the detail images of the interest region Z. Firstly, we calculate the overlap rate OR of each sample with the model, as shown in

\[
\text{Width} = (W_t + W_i) - (\max(x_{sr}, x_{tr}) - \min(x_{sl}, x_{td})).
\]

\[
\text{High} = (H_t + H_i) - (\max(y_{su}, y_{tu}) - \min(y_{sd}, y_{td})).
\]

\[
OR = \frac{\text{Width} \times \text{High}}{W_t \times H_t} \times 100\%.
\]

Width is the width of the overlapping part and High is the high of the overlapping part. \(W_t\) and \(H_t\) are the size of the target model. \(W_i\) and \(H_i\) are the size of the sample. \((x_{tr}, y_{tr})\) and \((x_{sl}, y_{su})\) represent the coordinate values of the lower right corner and the upper left corner of the model. \((x_{sr}, y_{tu})\) and \((x_{sd}, y_{sd})\) represent the coordinate values of the lower right corner and the upper left corner of the sample. When the overlap rate is greater than the set threshold, the sample is retained, otherwise the sample is deleted directly.

For the remaining samples, we calculate the Euclidean distance between the sample and the model. Do the same as we did with the model, first, we extract the four sample detail maps with different fitting degrees superimposed by each 100 detail maps and perform linear combination to obtain the final sample. Then, we calculate the Euclidean distance as EU between \(S\) and the model of detail part \(R\) by

\[
EU = \sqrt{(R - S)^2}.
\]

The minimum value of EU is taken as follows: the sample of \(\min(EU)\) is the final tracking result of the detail images.

3) Determine the final tracking result position

We combine the tracking result of the smooth image called \(pos_s\) and the detail images called \(pos_t\) to get the final position of target called \(pos_{tar}\) in a new frame, as shown in (8) where \(\eta_1 = \eta_2 = 1\).

\[
pos_{tar} = \eta_1 pos_s + \eta_2 pos_t.
\]

So far, the tracking progress of the tracking target in the new frame is completed. Determine if it is the last frame, and if so, the tracking of the video sequence ends; otherwise, update the appearance model of the target.

C. The update of the appearance model of target

After completing the tracking of the current frame image, we need to update the appearance model of target according to the result of tracking. The update progress of the appearance model of target is divided into updating the appearance model of the smooth image and the detail images.

1) Update the appearance model of the smooth image

The update of the model of the smooth image is mainly to update the cyclic feature matrix \(X\) which represents the target, and the target detector \(\hat{a}\). After getting the position of the tracking target in a new frame, determine the cyclic feature matrix \(X_t\) and the target detector \(\hat{a}_t\) of the target in the new frame. Then we combine the cyclic feature matrix \(X_{T-1}\) with the target detector \(\hat{a}_{T-1}\) of target in the previous frame, and we use linear interpolation to combine them by

\[
\begin{align*}
X_T &= (1 - \eta)X_{T-1} + \eta X_t. \\
\hat{a}_T &= (1 - \eta)\hat{a}_{T-1} + \eta \hat{a}_t.
\end{align*}
\]

\(X_T\) and \(\hat{a}_T\) are the updated cyclic feature matrix and target detector, and \(\eta\) is the learning parameter.

2) Update the appearance model of the detail images

The update of the appearance model of the detail images is means that update the four models of images about the details with different fitting degrees. By the standard position of the target in a new frame, we extract the area corresponding to the standard position in the detail images of the four interest regions with different fitting degrees for linear combination to get a new appearance model of the detail images.

IV. ANALYSIS OF RESULTS

This paper is based on MATLAB 2014b on an ordinary PC configured with Windows 7 system, Intel i7-6700 3.4GHz processor, 12G video memory and 12G memory. 61 sets of video sequences on the Visual tracking Benchmark platform were used for experiments. They include several challenges such as complex background, illumination variation, rotation, etc. We do quantitatively and qualitatively analysis with the current mainstream 13 methods such as DLSSVM [8], KCF [11], Staple [12], ECO [21], CNN-SVM [22], CFNet [15], SINT [23], Struct [7], DeepSRDCF [14], RPT [24], TLD [25], VTD [26] and CT [27]. The results show that the algorithm in this paper is more effective for in-plane rotation, complex background, and illumination variations.

A. Quantitative analysis

We analyze the target tracking based on the accuracy of the center position error and the coverage success rate. The center accuracy is obtained by the center distance between the tracking result and the ideal target area, and the success rate is calculated based on the overlap rate of two. Compared with the other 6 target trackers, the center accuracy and the success rate of our algorithm performs better as shown in Fig.4. For complex backgrounds, the center accuracy and the success rate value are ranked first with 0.781 and 0.605. For in-plane rotation, the center accuracy is 0.745 and the success rate value is ranked first with 0.555. For illumination variations, the center accuracy is 0.712, ranking third, but relative to the first place KCF and the second DLSSVM differs only by 0.014 and 0.002. And the success rate value is ranked first with 0.523. See Table 1 and Table 2 for other details.

B. Qualitative analysis

This section mainly analyzes the effectiveness of this paper for the challenges of occlusion, illumination variations and Out-of-Plane Rotation, and compares the algorithm with other nine algorithms.

1) Occlusion

Fig. 5 shows the comparisons about occlusion. Take Coke video and Jogging.1 video as examples. The target is blocked, such as frame 39 in Coke and frame 75 in Jogging.1, due to environmental factors, such as the leaves in Coke and the telephone pole in Jogging.1, during the movement. Many trackers easily lose target such as Struct and DLSSVM. If the target is blocked completely, the occlusion will be regarded as the target and lead to failure. We construct appearance model of the smooth image and the detail images to track the target, avoiding the model drift problem caused by a single model.
TABLE I. TRACKING ACCURACY OF 11 CHALLENGES WHERE RED IS FIRST, BLUE IS SECOND AND GREEN IS THIRD

| Trackers | FM  | BC  | MB  | DEF | IV  | IPR | LR  | OCC | OPR | OV  | SV  |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Ours     | 0.658 | 0.781 | 0.652 | 0.681 | 0.712 | 0.745 | 0.636 | 0.677 | 0.713 | 0.648 | 0.708 |
| KCF      | 0.665 | 0.757 | 0.662 | 0.676 | 0.726 | 0.710 | 0.610 | 0.657 | 0.700 | 0.606 | 0.712 |
| DLSSVM   | 0.705 | 0.708 | 0.752 | 0.703 | 0.714 | 0.747 | 0.747 | 0.749 | 0.755 | 0.714 | 0.736 |
| Struck   | 0.684 | 0.596 | 0.719 | 0.574 | 0.626 | 0.640 | 0.841 | 0.639 | 0.649 | 0.606 | 0.672 |
| TLD      | 0.558 | 0.537 | 0.616 | 0.469 | 0.581 | 0.604 | 0.515 | 0.542 | 0.590 | 0.537 | 0.608 |
| VTD      | 0.346 | 0.449 | 0.306 | 0.451 | 0.482 | 0.583 | 0.559 | 0.509 | 0.594 | 0.434 | 0.584 |
| CT       | 0.345 | 0.411 | 0.324 | 0.443 | 0.394 | 0.448 | 0.327 | 0.400 | 0.430 | 0.260 | 0.459 |

TABLE II. THE COVERAGE OF THE SUCCESS RATE VALUE FOR 11 CHALLENGES

| Trackers | FM  | BC  | MB  | DEF | IV  | IPR | LR  | OCC | OPR | OV  | SV  |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Ours     | 0.520 | 0.605 | 0.523 | 0.494 | 0.523 | 0.555 | 0.369 | 0.504 | 0.520 | 0.566 | 0.462 |
| KCF      | 0.520 | 0.588 | 0.527 | 0.497 | 0.509 | 0.525 | 0.369 | 0.486 | 0.501 | 0.530 | 0.461 |
| DLSSVM   | 0.569 | 0.560 | 0.592 | 0.526 | 0.518 | 0.545 | 0.452 | 0.563 | 0.545 | 0.609 | 0.484 |
| Struck   | 0.549 | 0.483 | 0.566 | 0.430 | 0.458 | 0.474 | 0.482 | 0.479 | 0.468 | 0.522 | 0.442 |
| TLD      | 0.481 | 0.429 | 0.524 | 0.345 | 0.427 | 0.458 | 0.335 | 0.424 | 0.440 | 0.486 | 0.439 |
| VTD      | 0.294 | 0.341 | 0.252 | 0.345 | 0.361 | 0.428 | 0.350 | 0.381 | 0.443 | 0.417 | 0.410 |
| CT       | 0.287 | 0.316 | 0.233 | 0.352 | 0.282 | 0.323 | 0.126 | 0.292 | 0.308 | 0.280 | 0.310 |

2) Illumination variation
As shown in Fig. 6, it is a comparison chart of the tracking effects of various algorithms under illumination variation. We use the Human8 video and the Man video as examples. In the Human8 video, from frame 2 to frame 20, we can get it that people first walk from a bright place to a dark place, and from frame 86 to frame 113, people walk from a dark place to a bright place. When people walk pass the shadow, we can find that tracking failed when using CFNet or Struck, because they extract target gray features. The light gradually increases from dark to bright in the Man video sequence. The figure shows that the tracking effect of the algorithm in this paper is good.

3) Out-of-Plane Rotation
Fig. 7 shows the tracking results of various algorithms under Out-of-Plane Rotation factors. Take Liquor video sequence and SUV video sequence as examples. In the Liquor video, such as shown in frame 386 and frame 401, during the movement of the target around the brown bottle, part of the target is not in the image sensor, thereby Out-of-Plane Rotation. In the SUV video, such as shown in frame 28 and frame 47, the fast movement of the car makes the camera unable to keep up, so that part of the target exceeds the image range. As shown in frame 432 in the Liquor video sequence and frame 63 in the SUV video sequence, if the target shown in the image completely, this algorithm in this paper can track the target in time. This is because when the algorithm in this paper matches the appearance model of detail, it calculates the overlap rate between the samples and the models to match,
and in the tracking of the smooth image, it can distinguish the background and the target. Combining the advantages of the two parts to guarantee the robustness of the algorithm when the target exceeds the image range.

![Image](Image)

Figure 7. Tracking result when target exceeds the image range (First row is the Liquor video sequence and the second row is the SUV video sequence)

V. CONCLUSION

This paper designs a target tracking algorithm based on the CSC. First, it extracts an interest region based on the target area. Then, based on CSC, we divide it into a smooth image and four detail images with different fitting degrees. For the smooth image, based on the KCF, it is initially modeled and tracked. For the detail images, we first extract four detail images and combine them to initialize the appearance model of the detail images. In the process of tracking, we randomly collect 400 samples in the interest region, and then we calculate the overlap rate and Euclidean distance between each sample and the appearance model to determine the tracking result of the detail images. By combining the tracking result of the smooth image and the tracking result of the detail images, the final position of the tracking target in a new frame is obtained. Our method establishes the appearance models of both the smooth image and detail images to achieve target tracking and to avoid the problem of model drift easily caused by a single model to improve the accuracy and robustness. We do the quantitative and qualitative analysis with the current trackers on the Tracking Benchmark2015. Many experiments show our method performs better under challenges such as illumination variations and complex background.

Acknowledgements: This work was supported by the Natural Science Foundation of China (61772209), Science and technology planning project of guangdong province (2019A050510034, 2019B020219001), the Production Project of Ministry Education China (201901240030), and the College Students Innovations Special Project of China under Grant 201910564037,202010564026.

REFERENCES

[1] M. J. Black, A. D. Jepson, “EigenTracking: Robust Matching and Tracking of Articulated Objects Using a ViewBased Representation,” International Journal of Computer Vision, vol. 26, pp.63-84, 1998.

[2] D. A. Ross, J. Lim, R. S. Lin, and M. Yang, “Incremental Learning for Robust Visual Tracking,” International Journal of Computer Vision, vol.77(1-3), pp.125-141, 2008.

[3] X. Mei and H. Ling, “Robust visual tracking using l1 minimization,” 2009 IEEE 12th International Conference on Computer Vision. Kyoto, pp.1436-1443, September 2009.

[4] X. Mei, H. Ling, Y. Wu, E. Blasch and L. Bai, “Minimum error bounded efficient l1 tracker with occlusion detection,” Computer Vision and Pattern Recognition, pp.1257-1264, June 2011.

[5] D. Wang, H. Lu, and M. H. Yang, “Online object tracking with sparse prototypes,” IEEE Trans Image Process, vol. 22, pp.314-325, 2013.

[6] S. Avidan, “Support vector tracking,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 8, pp. 1064-1072, August 2004.

[7] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M. Cheng, and S. L. Hicks, “Struck: Structured Output Tracking with Kernels,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 10, pp. 2096-2109, October 2016.

[8] J. Ning, J. Yang, S. Jiang, L. Zhang, etc, “Object Tracking via Dual Linear Structured SVM and Explicit Feature Map,” Computer Vision and Pattern Recognition. Las Vegas, NV, pp. 4266-4274, June 2016.

[9] D. S. Bolme, J. R. Beveridge, B. A. Draper and L. Yui, “Visual object tracking using adaptive correlation filters,” Computer Vision and Pattern Recognition, pp. 2546-2550, June 2010.

[10] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “Exploiting the circulant structure of tracking-by-detection with kernels,” European conference on computer vision, pp. 702-715, October 2012.

[11] J. F. Henriques, R. Caseiro, P. Martins, J. Batista, “High-Speed Tracking with Kernelized Correlation Filters,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 37, 3, pp. 583-596, 2015.

[12] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, P. H. S. Torr, “Staple: Complementary Learners for Real-Time Tracking,” Computer Vision and Pattern Recognition. Las Vegas, NV, pp. 1401-1409, June 2016.

[13] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, “Convolutional Features for Correlation Filter Based Visual Tracking,” International Conference on Computer Vision, pp. 621-629, December 2015.

[14] M. Danelljan, A. Robinson, F. S. Khan and M. Felsberg, “Beyond Correlation Filters: Learning Continuous Convolution Operators for Visual Tracking,” European Conference on Computer Vision, pp.472-488, October 2016.

[15] J. Valmadre, L. Bertinetto, J. F. Henriques, A. Vedaldi, and P. H. Torr, “End-to-End Representation Learning for Correlation Filter Based Tracking,” Computer Vision and Pattern Recognition, pp.5000-5008, July 2017.

[16] X. Li, C. Ma, B. Wu, Z. He and M. Yang, “Target-Aware Deep Tracking,” Computer Vision and Pattern Recognition, Long Beach. CA. USA, pp.1369-1378, June 2019.

[17] G. Wang, C. Luo, Z. Xiong, W. Zeng, “SPM-Tracker: Series-Parallel Matching for Real-Time Visual Object Tracking,” Computer Vision and Pattern Recognition, Long Beach. CA. USA, pp. 3638-3647, June 2019.

[18] F. Du, P. Liu, W. Zhao and X. Tang, “Correlation-Guided Attention for Corner Detection Based Visual Tracking,” Computer Vision and Pattern Recognition, pp. 6836-6845, June 2020.

[19] D. Guo, J. Wang, Y. Cui, Z. Wang, S. Chen, “SiamCAR: Siamese Fully Convolutional Classification and Regression for Visual Tracking,” Computer Vision and Pattern Recognition, pp. 6269-6277, June 2020.

[20] S. Gu, W. Zuo, X. Qie, D. Meng, X. Feng, L. Zhang, “Convolutional Sparse Coding for Image Super-Resolution,” International conference on computer vision, pp.1823-1831, December 2015.

[21] M. Danelljan, G. Bhat, F. S. Khan and M. Felsberg, “ECO: Efficient Convolution Operators for Tracking,” Conference on Computer Vision and Pattern Recognition, Honolulu, HI, pp. 6931-6939, July 2017.

[22] S. Hong, T. You, S.Kwak, & B.Han, “Online Tracking by Learning Discriminative Saliency Map with Convolutional Neural Network,” International Conference on Machine Learning, pp. 597-606, July 2015.

[23] R. Tao, E. Gavves and A. W. M. Smolders, “Siamese Instance Search for Tracking,” Computer Vision and Pattern Recognition, Las Vegas, NV, pp. 1420-1429, July 2016.

[24] Y. Li, J. Zhu and S. C. H. Hoi, “Reliable Patch Trackers: Robust visual tracking by exploiting reliable patches,” Computer Vision and Pattern Recognition, Boston, MA, pp. 353-361, June 2015.

[25] Z. Kalal, J. Matas and K. Mikolajczyk, “P-N learning: Bootstrapping binary classifiers by structural constraints,” Computer Vision and Pattern Recognition, San Francisco, CA, pp. 49-56, June 2010.

[26] J. Kwon and K. M. Lee, “Visual tracking decomposition,” Conference on Computer Vision and Pattern Recognition, San Francisco, CA, pp. 1269-1276, June 2010.

[27] K. Zhang, L. Zhang, and M. Yang, “Real-time compressive tracking,” European Conference on Computer Vision, pp.864-877, November 2012.