

## Research Article

# Target-Aware Deep Feature Compression for Power Intelligent Inspection Tracking

Wei Jiang <sup>1</sup>, Zhimin Guo <sup>2</sup>, Huanlong Zhang,<sup>3</sup> Liyun Cheng,<sup>3</sup> and Yangyang Tian<sup>2</sup>

<sup>1</sup>State Grid Corporation of China, Beijing 100032, China

<sup>2</sup>State Grid Henan Electric Power Research Institute, Zhengzhou 450000, China

<sup>3</sup>College of Electric and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China

Correspondence should be addressed to Zhimin Guo; [zhimin.guo@163.com](mailto:zhimin.guo@163.com)

Received 26 September 2021; Revised 13 December 2021; Accepted 4 January 2022; Published 21 February 2022

Academic Editor: Yang Li

Copyright © 2022 Wei Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Deep learning has brought revolutionary progress to computer vision, so intelligent inspection equipment based on computer vision has developed rapidly. However, due to the large number of existing deep features, it is difficult to deploy it on mobile devices to achieve real-time tracking speed. This paper presents a target-aware deep feature compression for power intelligent inspection tracking. First, a negative balance loss function is designed to mine channel features suitable for the current inspection scene by shrinking the contribution of pure background negative samples and enhancing the impact of difficult negative samples. Based on this, the deep feature compression model is combined with Siamese tracking framework to achieve real-time and robust tracking. Finally, we evaluate the proposed method on real application scenarios and general data to prove the practicability of the proposed method.

## 1. Introduction

A stable and reliable power system is the key to ensuring people's livelihood and economic development [1, 2]. In recent years, deep learning has been widely used in power systems due to its powerful ability, including voltage stability assessment [3], wind farm control [4], solar power generation system control [5], intelligent inspection of power [6–8], and so on. In the intelligent inspection of electric power, the application of vision-based inspection technology in electric power inspection has been widely concerned. Robot-based detection technology [6] and UAV-based detection technology [7, 8] both need to track the determined target, so the robust tracking method is of great importance. Deep learning provides strong support for the development of more robust target tracking [9, 10]. However, the deep features provided by the deep neural network model have more parameters, so it is very expensive in computing and storage, which makes it difficult to meet the real-time requirements of intelligent inspection and difficult to deploy on mobile intelligent inspection devices.

Some methods hope to improve machine learning performance by using optimization algorithms, so as to monitor power system state more effectively and quickly [11]. In addition, many lightweight methods for deep networks have been proposed, mainly including five methods: (1) parameter pruning, (2) parameter sharing, (3) low-rank decomposition, (4) designing compact convolutional filters, and (5) knowledge distillation. Parameter pruning mainly removes redundant parameters by designing criteria to judge whether the parameters are important or not [12, 13]. Parameter sharing mainly explores the redundancy of model parameters and uses Hash or quantization techniques to compress weights [14, 15]. Low rank decomposition uses matrix or tensor decomposition techniques to estimate and decompose the original convolution kernel in the deep model [16, 17]. The designing compact convolutional filters mainly reduces the storage and computational complexity of the model by designing special structured convolution kernel or compact convolution computation unit [18, 19]. Knowledge distillation mainly utilizes the knowledge of large networks and migrates its knowledge into the compact

distillation model [20, 21]. These methods tend to compress from the perspective of the whole network, but for tracking, only the feature output of a certain layer is used. So how to compress the deep feature is very important to improve the tracking timeliness.

In order to compress features, Li et al. [22] proposed a BinJaya-based feature selection method to select the optimal feature subset from the whole feature space composed of a set of systemic-level classification features extracted from PMU data. *The Method ECO*. Efficient convolution operators for tracking [23] uses principal component analysis to simplify features, which is still very slow with online updates. The method context-aware deep feature compression for high-speed visual tracking (TRACA) [24] combines a context-aware network with multiple expert auto-encoder to compress feature graphs. However, its compression performance mainly depends on the pretrained context-aware network, which cannot well adapt to the current tracking scene. On the contrary, method target-aware deep tracking (TADT) [25] proposed regression loss and sequencing loss. Under the guidance of these two losses, only the target information given in the first frame can be utilized to effectively eliminate redundant channel features and achieve robust and high-speed tracking. It is observed that TADT uses a large number of negative samples of pure background in regression loss learning to occupy more contributions, which leads to the activated channel features paying more attention to the difference between the target and the pure background, while tracking requires more attention to the interference that is indistinguishable from the target.

Inspired by the previously mentioned work, we proposed a target perception deep feature compression method for intelligent detection target tracking. We designed a negative balance loss function to shrink the loss of pure negative samples and prevent them from dominating the learning process. At the same time, the contribution of hard negative samples was enhanced, so that the activated channel features were more focused on the difference between target and similar interference. Then, the compressed feature is combined with Siamese framework to achieve robust and real-time tracking.

## 2. Regression via Convolution Layer

We first review the learning process of convolutional regression networks. The convolutional regression network aims to return training samples to soft labels, which are usually Gaussian-like. Given an initial image with annotations, a large number of training samples  $X$  are obtained through intensive sampling to generate the corresponding Gaussian soft label  $Y$ . The weight coefficient  $W$  of the regression network is estimated by solving the minimization problem

$$\arg \min_W \|W * X - Y\|^2 + \lambda \|W\|^2, \quad (1)$$

where  $*$  is the convolution operation and  $\lambda$  is the regularization parameter controlling overfitting. The network weight  $W$  is usually optimized by minimizing the square loss:

$$L(W) = \|W * X - Y\|^2 + \lambda \|W\|^2, \quad (2)$$

$$W_t = W_{t-1} - \eta \frac{\partial L}{\partial W}.$$

Samples for learning convolutional networks can be extracted by sliding a window on the input search area image. The search area must contain a large number of context areas around the target. Because the surrounding background information contains a large area of valuable information, it contributes to the network's ability to identify targets from the background. However, the large context region contains most of the pure background, with only a small number of target-like regions. These pure background areas increase the number of simple negative samples, resulting in large losses and the loss of valuable and small positive samples being drowned out. Due to the gradient of easy negative samples being dominant, the learning process does not know the difference between the attentional target and similar disturbances; thus, the importance of each channel feature to the representation of the target cannot be assessed more accurately.

## 3. Negative Balance Loss

One way to solve the imbalance of training data is to design with reduced effects of simple negative samples. Chen et al. [26] proposed an automatic hard negative mining method, which eliminated easy negative and enhanced positive. Song et al. [27] used cost sensitive loss to reduce the influence of easy negative factors on the learning process. Lu et al. [28] proposed shrinkage loss, which penalizes the loss of easy sample while keeping the hard sample unchanged. Bhat et al. [29] proposed discriminant learning loss to enhance the contribution of positive samples. Although these methods can simplify the loss of negative samples to some extent, they can reduce the loss of difficult negative samples.

In this paper, we designed a negative balance loss function, which can more effectively evaluate the effectiveness of each channel feature for the current target representation by shrinking the loss of simple negative samples and enhancing the contribution of difficult negative samples. The designed negative balance loss function is as follows:

$$\arg \min_W \|T(W * X - Y)\|^2 + \lambda \|W\|^2, \quad (3)$$

where  $T(\cdot)$  is the function used to adjust the loss and

$$T(r) = \frac{1}{1 + e^{(m-nr)}}, \quad (4)$$

where  $m$  and  $n$  are the hyperparameters. It is worth noting that the proposed loss function introduces two additional parameters,  $m$  and  $n$ . Parameter  $m$  is mainly used to control the degree of loss compression. The larger the loss compression is, the smaller the loss compression is, and the impact of easy negative samples will be greatly reduced. However, it also reduces the loss of hard negative samples. Another parameter  $n$  can solve this problem well. It is

proportional to the degree of loss compression. An S-shaped curve can be formed when an appropriate combination of  $m$  and  $n$  is selected, which can reduce the loss of easy negative samples and enhance the loss of difficult negative samples, as shown in Figure 1. Through analysis and experiments in the validation set, we determine  $m = 6$  and  $n = 10$ .

#### 4. Intelligent Inspection via Deep Feature Compression

Visual power inspection relies on the expression ability of image features [30, 31]. The emergence of neural networks has brought about powerful deep features. The development of deep neural networks gradually tends to be larger and deeper models, while benefiting from large-scale training data sets such as ImageNet, the performance of neural networks has been greatly improved. However, the neural network has a very high requirement for memory and computing power, so it has been paid more and more attention in academic circles. In practical application, it is a hot research direction to keep its performance and compress it, so that it can run effectively on intelligent inspection equipment.

To solve the previously mentioned problems, we proposed a channel pruning method based on target perception. Our work is based on the following inspiration: the sensitivity of each convolution filter to the current object can be obtained by the gradient obtained by the back propagation network, and then, the importance of each channel feature to the current target representation can be determined by using the global average pooling. Based on this, we used the designed negative balance loss function to guide the assessment of the importance of each channel feature to the target representation. Given a search area image, pretrained CNN is used to extract features, and the importance  $\Delta_i$  of the  $i$ -th channel feature  $z_i$  can be calculated by the following formula:

$$\Delta_i = \text{GAP}\left(\frac{\partial L}{\partial z_i}\right), \quad (5)$$

where  $i$  is the channel index,  $\text{GAP}(\cdot)$  is the global average pooling function, and  $L$  is the designed negative balance loss. In order to improve the effectiveness of judging the features of each channel for the current target representation, we use the negative balance loss to represent the problem

$$L_{Nreg} = \|T(W * X_{i,j} - Y(i, j))\|^2 + \lambda \|W\|^2. \quad (6)$$

Here, we regress all samples aligned with the target center in the image to a soft label, where the soft label is a Gaussian label  $Y(i, j) = e^{-i^2+j^2/2\sigma^2}$ ,  $(i, j)$  is the offset against the target and  $\sigma$  is the kernel width. The importance of each channel feature to the target representation can be calculated in terms of its contribution to the fitting of the label, that is, taking the derivative of  $L_{Nreg}$  with respect to the input feature  $X_{in}$ . Using the chain rule and (7), calculate the gradient of regression loss:

$$\begin{aligned} \frac{\partial L_{Nreg}}{\partial X_{in}} &= \sum_{i,j} \frac{\partial L_{Nreg}}{\partial X_o(i, j)} \times \frac{\partial X_o(i, j)}{\partial X_{in}(i, j)} \\ &= \sum_{i,j} 2(Y(i, j) - X_o(i, j)) \times W, \end{aligned} \quad (7)$$

where  $X_o$  is the output prediction. Finally, after obtaining the importance weight of each channel feature, we conducted binarization operation, and those whose contribution value was higher than a certain threshold value were retained, while the others were all deleted. The importance weight generation process for each channel is shown in Figure 2.

#### 5. Mobile Intelligent Inspection Tracking

After the intelligent inspection device determines the target to be tracked, the tracing process is started. First, the initial features extracted from the target and its context region are sent to a single-layer network for training. Then, the filter gradient of each channel is calculated according to the convergence loss. Then, the contribution of each channel feature is calculated according to the method given in Section 4 to guide channel pruning. Given the initial feature space  $\theta$  is suppressed, the feature space after compression is

$$\theta' = \varphi(\theta; \Delta), \quad (8)$$

where  $\varphi$  is the channel pruning function and  $\Delta$  is the binarized channel importance parameter. In the tracking stage, the initial target and the search area of the next frame are known, and the position of the predicted target in the coordinate system is calculated as

$$\hat{p} = \arg \max_p \theta'(x_1) * \theta'(z_t), \quad (9)$$

where  $x_1$  is the target template feature,  $z_t$  is the search region in the next frame, and  $t$  is the frame index.

In addition, as the intelligent inspection device or camera moves, the scale of the target will change, so the tracking process needs to adapt to this change. In order to better evaluate scale variation, we added a scale sensitive pruning method based on ranking loss to target sensing channel pruning. We used smoothing to approximate the sorting loss defined by

$$L_{\text{rank}} = \log \left( 1 + \sum_{(x_i, x_j)} \exp(f(x_i) - f(x_j)) \right). \quad (10)$$

#### 6. Results and Discussion

**6.1. Experimental Setup.** In order to verify the effectiveness of the proposed method, we first conducted experiments on a general tracking dataset TC-128 dataset [32], OTB100 [33] and UAV123 [34] and then conducted experiments on the actual inspection video collected. The experiment was carried out on a PC with 316G memory, I7 3.00GHZ CPU, and GTX-2060 GPU, using MatConvNet toolbox

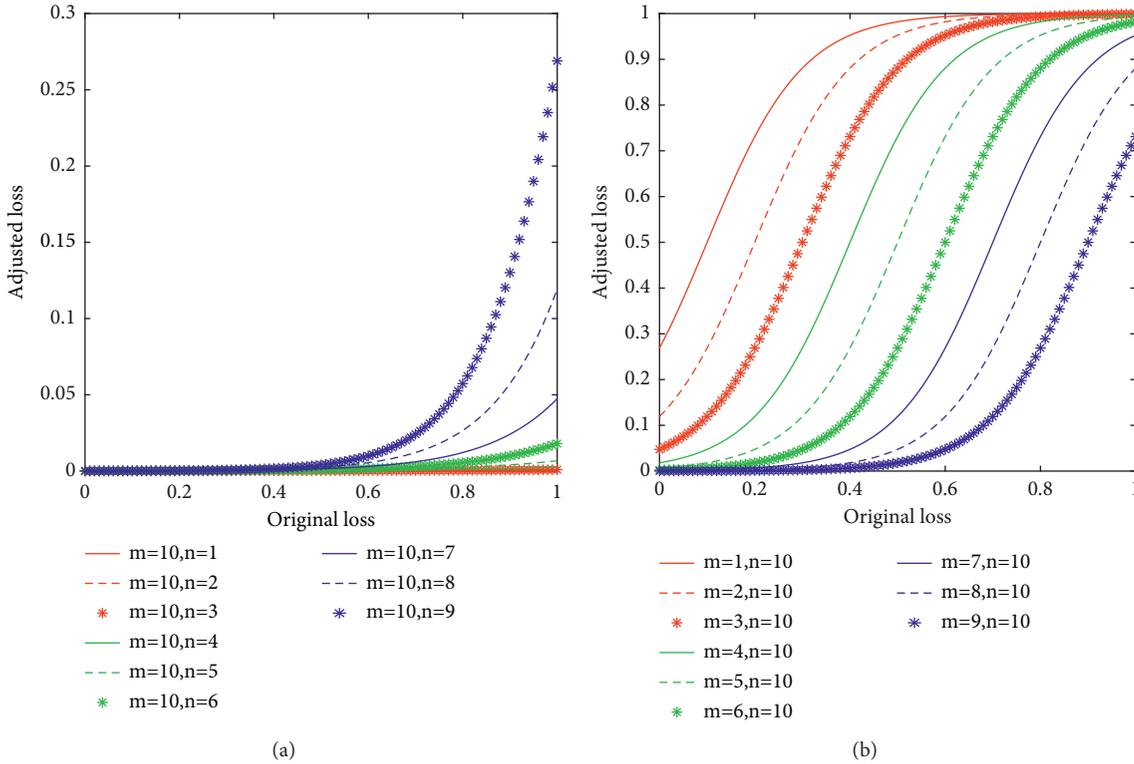


FIGURE 1: The relationship between loss and parameter.

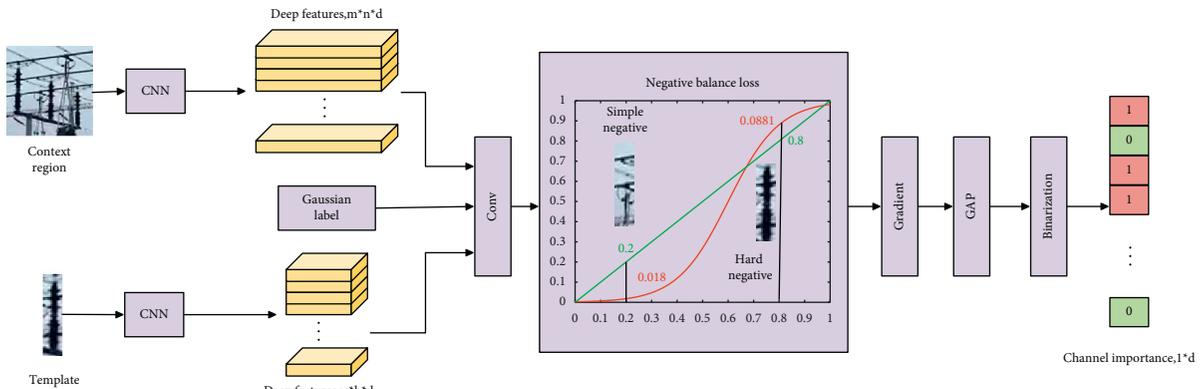


FIGURE 2: The importance weight assessment process for each channel.

[35] to implement the proposed method in Matlab2019a. The backbone network of the tracker was chosen as VGG-16 [36]. In order to obtain better accuracy and scale, the activation output of CONV4-3 and CONV4-1 layers was used as the base deep feature. After calculating the importance of each channel feature, the first 100 important channel features of Conv4-3 layer and the first 30 important channel features of CONV4-1 layer were retained. In the tracking stage, the given target is used as a template to cut out the search area image three times the size of the target from the current frame and then form the feature pyramid of 0.990, 1, and 1.005 times of the target scale, respectively.

**6.2. Overall Performance on Common Evaluation Benchmark.** We first evaluated the overall performance of the proposed method on two general tracking benchmarks (TC-128 and UAV123) and compared the precision and success rate plots of the tracking results under one-pass evaluation (OPE). *The Precision Plot.* First compute the center points of bounding box and artificially marked ground-truth objects, whose distance is less than the percentage of video frames with a given threshold. A curve obtained according to different percentages of different threshold values is the accuracy graph, and the precision of 20 pixels is the precision score (DP). *Success Plot.* First define coincidence rate score (overlap score, OS), the tracking method of bounding box (to a), and

groundway to box (to b). The coincidence rate is defined as  $OS = |a \cap b| / |a \cup b|$ , where  $|\cdot|$  in the said area is the number of pixels. When the OS of a frame is greater than the set threshold, the frame is regarded as success, and the percentage of the total number of successful frames in all frames is the success rate. A curve drawn according to the value range of OS is the success plot. Then, calculate the value of the area under the success plot, which is the AUC score.

*6.2.1. Overall Performance on the TC-128 Dataset.* We mainly compared the proposed method with the correlation filters based trackers, including method exploiting the circulant structure of tracking-by-detection with kernels (CSK) [37], method accurate scale estimation for robust visual tracking (DSST) [38], method high-speed tracking with kernelized correlation filters (KCF) [39], the deep feature-based methods include method hierarchical convolutional features for visual tracking (CF2) [40], ECO [23], TADT [25], and other trackers including method convolutional regression for visual tracking (CREST) [41]. The tracking results on the TC-128 dataset are shown in Figure 3. Through comparison, it can be found that the deep feature-based method is significantly better than the manual feature-based method. Compared with deep feature-based methods, our method also has greater advantages. It is worth noting that, compared with the traditional ridge regression used by TADT, the proposed method can effectively solve the problem of easy and difficult negative sample heterogeneity, thus obtaining more robust and accurate results.

*6.2.2. Overall Performance on the UAV123 Dataset.* All the video data in the UAV123 dataset were captured by unmanned aerial vehicles. Unmanned aerial vehicle (UAV) is widely used in intelligent inspection. Based on the similarity in this aspect, we further verify the effectiveness of the proposed method on UAV123 dataset. The overall tracking results are shown in Figure 4. Even the proposed ECO method, which is based on deep features, is no slouch when it comes to video taken by specialized unmanned aerial vehicles.

*6.3. Actual Inspection Performance.* We collected several actual inspection scene videos and made a small dataset for the experiment. The same evaluation metric as TC-128 was used to evaluate the overall performance of the proposed method. The precision plot and success plot are shown in Figure 5. We compared it with two classical methods, including the tracking methods KCF, CSK, and ECO, based on correlation filter, and the tracking methods TADT and high performance visual tracking with Siamese region proposal network (SiamRPN) [42] based on Siamese framework. It is worth noting that SiamRPN is a method combining detection and tracking, and our method is improved on the basis of TADT. The tracking methods KCF and CSK based on correlation filter used hand-crafted features, while ECO uses deep features. The correlation filter method using deep features is obviously better, which indicates that deep learning can effectively improve the robustness of tracking.

Thanks to more deep features and online updates, ECO has better performance, but its speed is only 3FPS, which cannot meet real-time requirements of intelligent inspection tracking. The proposed method achieves 30FPS, and the accuracy is close to that of ECO. SiamRPN regards tracking as a one-shot detection problem, but due to the complexity of inspection scenarios, it is not suitable to detect every frame, so its performance is limited.

Figure 6 shows the tracking results of different tracking methods in four real scenarios. As can be seen from the first column, the detection-based tracking method SiamRPN is not suitable for this tracing task. It can be seen from the second column that the tracking methods CSK and KCF based on correlation filtering are not only difficult to adapt to the problem of target scale change but also difficult to obtain higher tracking accuracy. The deep-based tracking method has higher accuracy. Similarly, the third and fourth columns also show that the deep-based tracking method is better.

*6.4. Analysis of Tracking Speed.* In addition to the comparison of tracking accuracy, another important evaluation is about the comparison of tracking speed, because the actual tracking scene needs real-time tracking speed. We compared the scores and tracking speed of different algorithms in several videos, and the tracking performance is shown in Table 1. The tracking speed of CSK and KCF based on hand-crafted feature correlation filtering is up to 2118FPS and 969FPS, respectively, but their tracking performance is limited. The deep-based ECO performs better, but at less than 3FPS. The tracking method TADT based on Siamese network achieves real-time speed and high performance. Our method is improved on the basis of TADT, the tracking accuracy is improved to ECO, and the speed is still real-time.

*6.5. Analysis of Compression Effect.* In order to prove the performance of the proposed target-aware deep feature compression method more clearly, we compared the number of parameters of the original deep feature and the compressed deep feature, the floating point operations during correlation operation, the number of channels, the DP and AUC scores on TC-128 dataset, and the tracking speed, as shown in Table 2.

As can be seen from Table 2, the original deep feature has 1024 channels, so the number of parameters is very large, and the floating point operations are very large during the correlation operation, which leads to the inability to realize real-time tracking. After compression using the proposed method, the numbers of parameters and floating point operations are greatly reduced, only 12.7% of the original, so the tracking speed is greatly improved, and real-time tracking is realized.

From the comparison of tracking results, the tracking effect does not benefit from the original deep feature with more channels. After the compression of deep features, the reduction of feature channels does not lead to a lower tracking score, but a higher score. This shows that the proposed method can effectively compress depth features to achieve robust tracking.

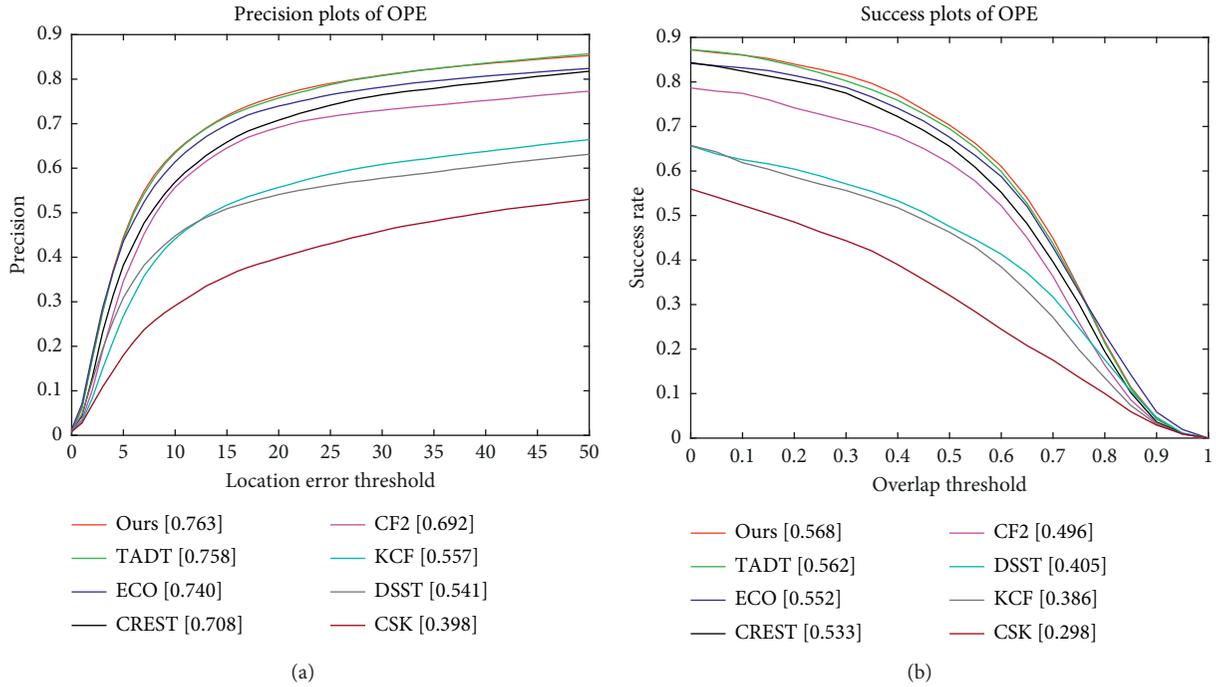


FIGURE 3: Success and precision plots on the TC-128 dataset.

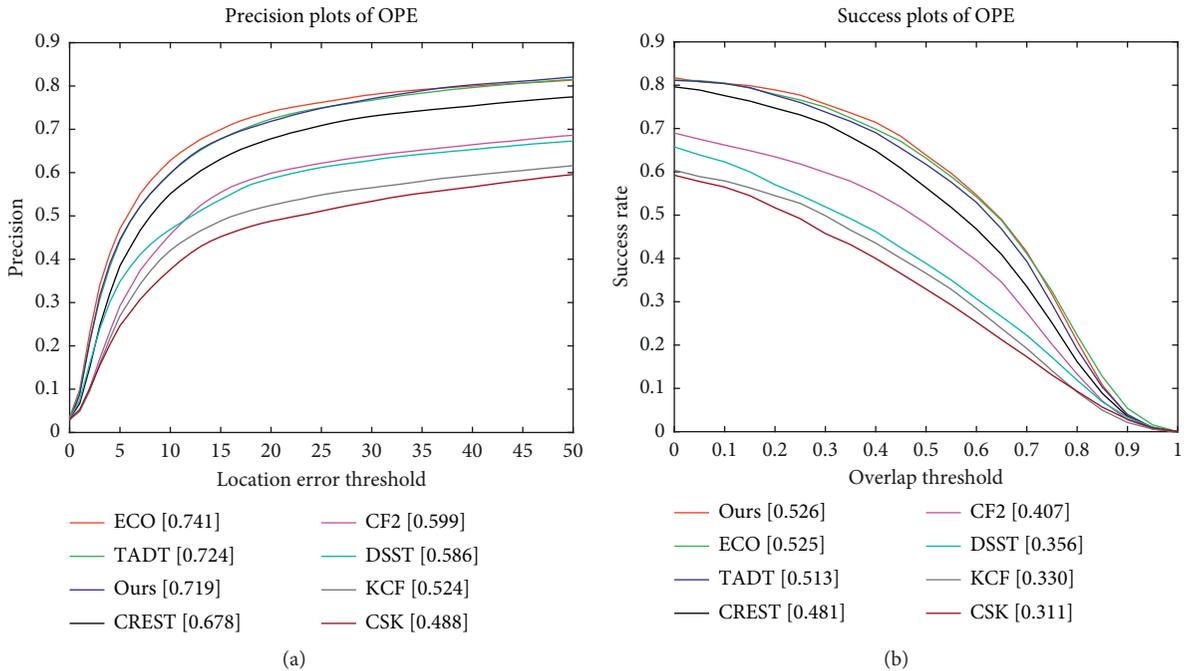


FIGURE 4: Success and precision plots on the UAV123 dataset.

### 7. Applicable Environment and Restrictions

In the intelligent inspection of power system based on mobile robot, the task of mobile robot is to read the indicator number of instrument panel using visual inspection technology. The whole task is divided into three parts. (1) The first part is to use detection technology to identify the instrument that needs to be read. (2) The second part is to use tracking technology to

continuously track the instrument until the instrument picture is clear. (3) The third part is using segmentation method to read the indicator number of instrument. The tracking method proposed by us is mainly used for instrument continuous tracking in the second part. After detecting the instrument to be tracked, the importance weight of each channel is calculated using the proposed compression method (that is, to determine which channel features are

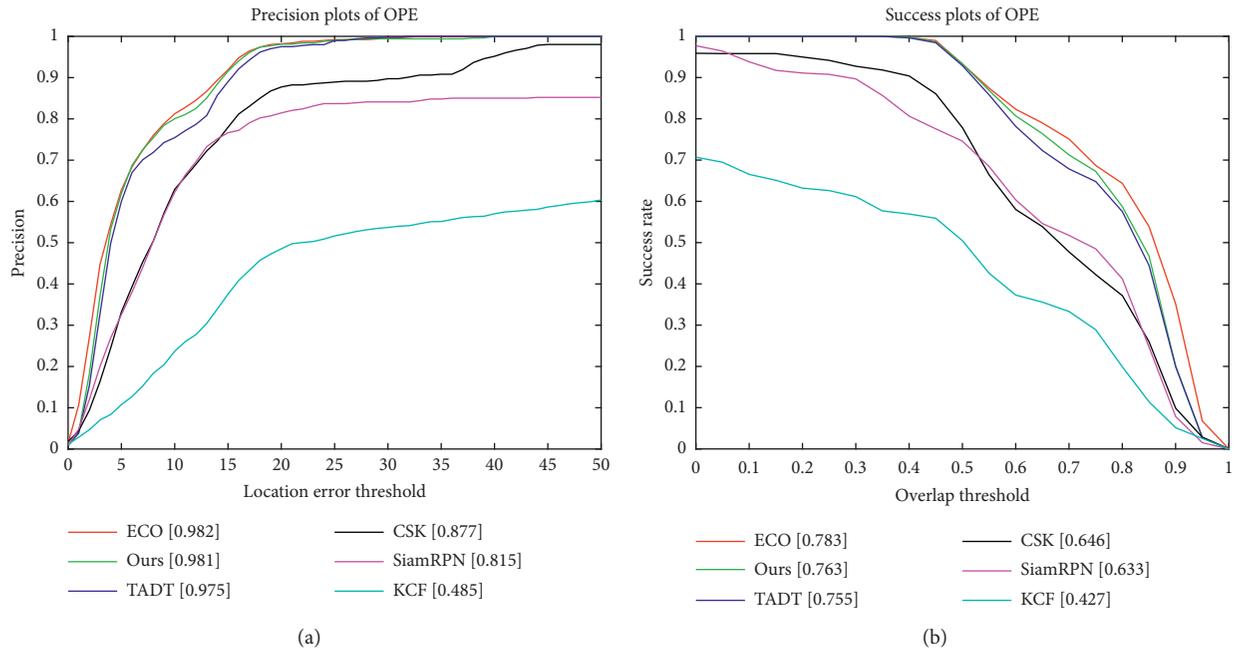


FIGURE 5: Overall performance on several actual inspection scene videos.

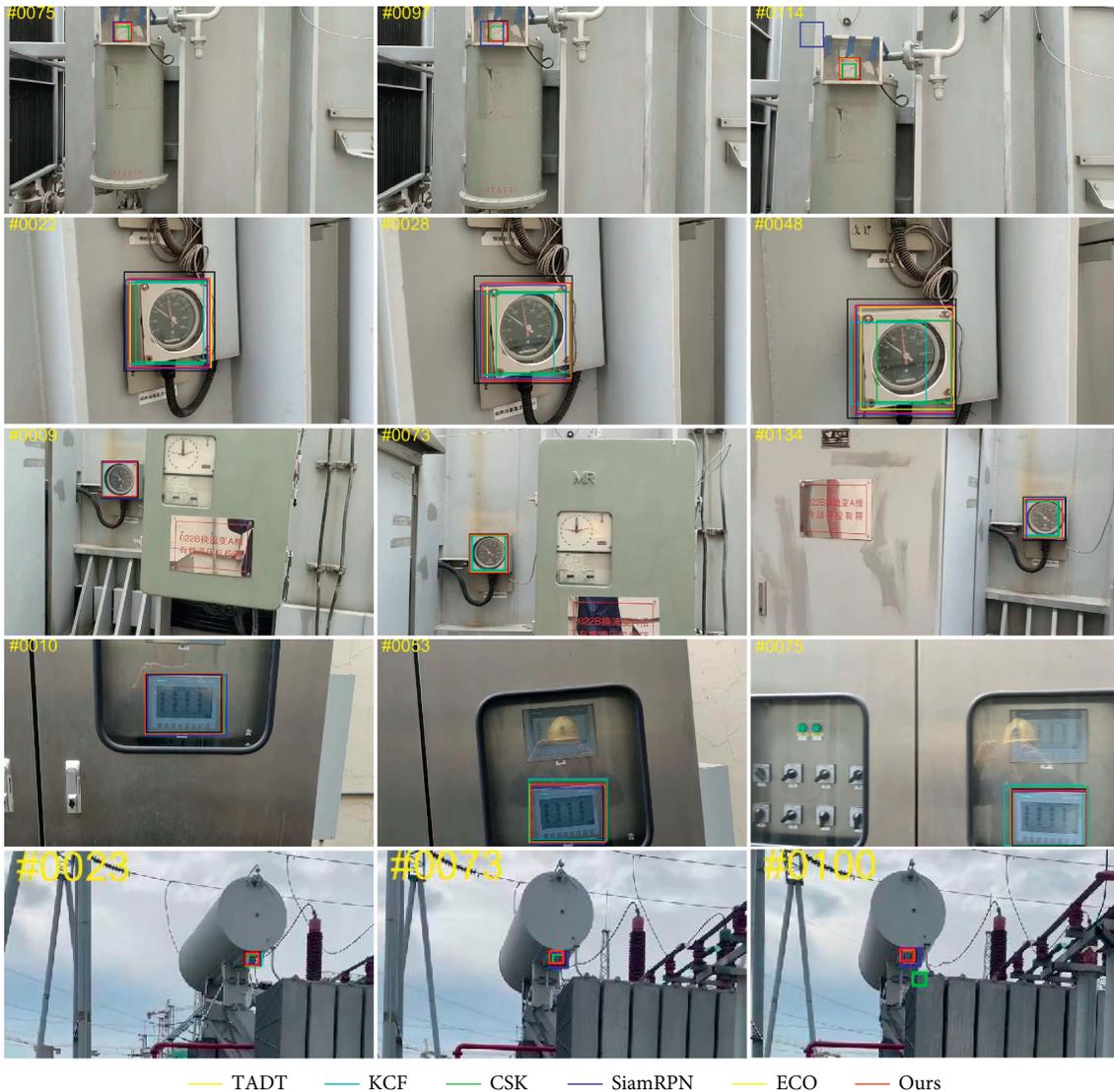


FIGURE 6: Tracking results of different tracking methods.

TABLE 1: Comparison of tracking speed and performance.

Sequences	Trackers	Score	FPS
Instrument1	CSK	0.596	357
	KCF	0.313	190.08
	ECO	0.913	2.338
	TADT	0.820	8.842
	SiamRPN	0.631	67.848
	Ours	0.829	31.066
Instrument2	CSK	0.567	119.761
	KCF	0.624	26.33
	ECO	0.814	1.44
	TADT	0.764	37.087
	SiamRPN	0.823	44.103
	Ours	0.816	31.894
Instrument4	CSK	0.776	499.397
	KCF	0.737	217.5
	ECO	0.885	1.431
	TADT	0.845	26.592
	SiamRPN	0.843	70.173
	Ours	0.845	27.635
Insulator1	CSK	0.417	610.642
	KCF	0.417	34.45
	ECO	0.428	2.366
	TADT	0.426	36.916
	SiamRPN	0.465	71.861
	Ours	0.425	34.809
Liquidometer	CSK	0.338	2118.214
	KCF	0.113	969.45
	ECO	0.791	2.4
	TADT	0.702	25.092
	SiamRPN	0.557	98.468
	Ours	0.572	28.871

TABLE 2: Comparison between original deep features and compressed deep features.

Features	Parameters	FLOPs	Channels	DP	AUC	FPS
Original	$1.46E+05$	$1.79E+08$	1024	0.719	0.546	14
Compressed	$1.26E+04$	$2.28E+07$	130	0.763	0.568	30

retained). Then, the importance weight of each channel is used to cut the original deep feature in the tracking process. Finally, the deep features after cutting are used for correlation operations to obtain the score map, and the maximum value on the score map is the predicted target position.

As can be seen from the previously mentioned experimental results, be it the general tracking dataset TC-128 or the dataset UAV123 for specific scenarios or for actual inspection scenarios, our method has excellent performance in tracking robustness and tracking speed, indicating that our method has good scalability.

The restriction of this method is that the three tasks are divided into three visual tasks to be processed separately, and there will be some errors in the communication between the three parts, and the coordination between the three parts needs to be carefully adjusted. However, the feature compression method proposed by us has good transferability and

can be applied in other tasks. In the future, we hope to integrate detection, tracking, and segmentation into the same network in the future, so as to achieve more efficient intelligent detection tasks.

## 8. Conclusion

It is very important to realize accurate and robust tracking in intelligent power inspection. The development of deep learning promotes the improvement of tracking performance. However, the tracking efficiency is very low due to the high dimensional data of deep features, so it cannot be applied to the intelligent inspection of electric power. In this paper, deep feature compression based on target-aware is proposed for power intelligent inspection system. This method retains important channel features more effectively and deletes unnecessary channel features through the

designed negative balance loss. The compressed feature is combined with the Siamese frame to realize tracking, and the effectiveness of the proposed method is proved by experiments on two general tracking datasets and power inspection video.

## Data Availability

The data used to support the findings of the study are available from the corresponding author upon request (email: zhimin.guo@163.com).

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (61873246, 62072416, 62006213, 6167241, and 61702462), Program for Science and Technology Innovation Talents in Universities of Henan Province (21HASTIT028), Natural Science Foundation of Henan (202300410495), and Zhongyuan Science and Technology Innovation Leadership Program (214200510026).

## References

- [1] M. Ombuki, O. Roy, and N. Julius, "Static voltage stability assessment of the Kenyan power network," *Journal of Electrical and Computer Engineering*, vol. 2021, Article ID 5079607, 16 pages, 2021.
- [2] M. M. Hadow, A. N. Abd Allah, and S. P. Abdul karim, "Reliability evaluation of distribution power systems based on artificial neural network techniques," *Journal of Electrical and Computer Engineering*, vol. 2012, Article ID 560541, 5 pages, 2012.
- [3] M. Zhang, J. Li, Y. Li, and R. Xu, "Deep learning for short-term voltage stability assessment of power systems," *IEEE Access*, vol. 9, pp. 29711–29718, 2021.
- [4] M. Madhiarasan, "Accurate prediction of different forecast horizons wind speed using a recursive radial basis function neural network," *Protection and Control of Modern Power Systems*, vol. 5, no. 1, 2020.
- [5] H. Hamdi, C. B. Regaya, and A. Zaafouri, "A sliding-neural network control of induction-motor-pump supplied by photovoltaic generator," *Protection and Control of Modern Power Systems*, vol. 5, no. 1, pp. 1–17, 2020.
- [6] S. Wang, E. Yang, X. Wang, Q. Deng, Q. Liang, and J. Mo, "Development of a novel live-line inspection robot system for post insulators at 220-kv substations," *Advanced Robotics*, vol. 24, no. 4, pp. 559–583, 2010.
- [7] X. Hui, J. Bian, X. Zhao, and M. Tan, "Vision-based autonomous navigation approach for unmanned aerial vehicle transmission-line inspection," *International Journal of Advanced Robotic Systems*, vol. 15, no. 1, Article ID 1729881417752821, 2018.
- [8] C.-a. Liu, R.-f. Dong, and H. Wu, "Flying robot based viewpoint selection for the electricity transmission equipment inspection," *Mathematical Problems in Engineering*, vol. 2014, Article ID 783810, 10 pages, 2014.
- [9] H. Chu, K. Wang, and X. Xing, "Target tracking via particle filter and convolutional network," *Journal of Electrical and Computer Engineering*, vol. 2018, Article ID 5381962, 9 pages, 2018.
- [10] M. Danelljan, G. Hager, F. Shahbaz, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 559–583, Santiago, Chile, December 2015.
- [11] Y. Zhang, T. Li, G. Na, G. Li, and Y. Li, "Optimized extreme learning machine for power system transient stability prediction using synchrophasors," *Mathematical Problems in Engineering*, vol. 2015, Article ID 529724, 8 pages, 2015.
- [12] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient transfer learning," in *Proceedings of the International Conference on Learning Representations*, San Juan, Puerto Rico, 2016.
- [13] J.-H. Luo, J. Wu, and W. Lin, "Thinet: A filter level pruning method for deep neural network compression," in *Proceedings of the IEEE international conference on computer vision*, pp. 5058–5066, Venice, Italy, October 2017.
- [14] Z. Cai, X. He, J. Sun, and N. Vasconcelos, "Deep learning with low precision by half-wave Gaussian quantization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5918–5926, Honolulu, HI, USA, July 2017.
- [15] I. Hubara, M. Courbariaux, D. Soudry, and Y. Ran, "Quantized neural networks: training neural networks with low precision weights and activations," *Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6869–6898, 2017.
- [16] Y. Ioannou, D. Robertson, J. Shotton, R. Cipolla, and A. Criminisi, "Training CNNs with low-rank filters for efficient image classification," *Journal of Asian Studies*, vol. 62, no. 3, pp. 952–953, 2015.
- [17] C. Tai, T. Xiao, Y. Zhang, W. Xiaogang, and E. Weinan, "Convolutional neural networks with low-rank regularization," in *Proceedings of the International Conference on Learning Representations*, San Juan, Puerto Rico, 2016.
- [18] F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1800–1807, Honolulu, HI, USA, July 2017.
- [19] F. N. Iandola, M. W. Moskewicz, and K. Ashraf, "Squeezenet: alexnetlevel accuracy with 50x fewer parameters and <1mb model size," in *Proceedings of the International Conference on Learning Representations*, Toulon, France, 2017.
- [20] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *Computer Science*, vol. 147, pp. 38–39, 2015.
- [21] S. Zagoruyko and N. Komodakis, "Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer," in *Proceedings of the International Conference on Learning Representations*, Toulon, France, 2017.
- [22] Y. Li and Z. Yang, "Application of EOS-ELM with binary jaya-based feature selection to real-time transient stability assessment using PMU data," *IEEE Access*, vol. 5, pp. 23092–23101, 2017.
- [23] M. Danelljan, G. Bhat, F. Shahbaz Khan, and M. Felsberg, "Eco: efficient convolution operators for tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6638–6646, Honolulu, HI, USA, July 2017.
- [24] J. Choi, H. J. Chang, T. Fischer et al., "Context-aware deep feature compression for high-speed visual tracking," in *Proceedings of the IEEE Conference on Computer Vision and*

- Pattern Recognition*, pp. 479–488, Salt Lake City, UT, USA, June 2018.
- [25] X. Li, C. Ma, and B. Wu, “Target-aware deep tracking,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1369–1378, Long Beach, CA, USA, June 2019.
- [26] K. Chen and W. Tao, “Convolutional regression for visual tracking,” *IEEE Transactions on Image Processing*, vol. 27, no. 7, pp. 3611–3620, 2018.
- [27] Y. Song, C. Ma, X. Wu, W. H. Lau, and Y. Ming-Hsuan, “Vital: visual tracking via adversarial learning,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8990–8999, Salt Lake City, UT, USA, June 2018.
- [28] X. Lu, C. Ma, B. Ni, Y. Xiaokang, R. Ian, and Y. Ming-Hsuan, “Deep regression tracking with shrinkage loss,” in *Proceedings of the European conference on computer vision*, pp. 353–369, Munich, Germany, January 2018.
- [29] G. Bhat, M. Danelljan, L. V. Gool, and R. Timofte, “Learning discriminative model prediction for tracking,” in *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6181–6190, Seoul, Korea (South), November 2019.
- [30] J. Jang, M. Shin, S. Lim, and J. Park, “Intelligent image-based railway inspection system using deep learning-based object detection and weber contrastbased image comparison,” *Sensors*, vol. 19, no. 21, 2019.
- [31] R. Jenssen, D. Roverso, and D. Roverso, “Automatic autonomous vision-based power line inspection: a review of current status and the potential role of deep learning,” *International Journal of Electrical Power & Energy Systems*, vol. 99, pp. 107–120, 2018.
- [32] P. Liang, E. Blasch, and H. Ling, “Encoding color information for visual tracking: algorithms and benchmark,” *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5630–5644, 2015.
- [33] Y. Wu, J. Lim, and M.-H. Yang, “Object tracking benchmark,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.
- [34] M. Mueller, N. Smith, and B. Ghanem, “A Benchmark and Simulator for UAV Tracking,” in *Computer Vision - ECCV 2016*, vol. 9905, pp. 445–461, Springer International Publishing, 2016.
- [35] A. Vedaldi and K. Lenc, “Matconvnet: convolutional neural networks for matlab,” in *Proceedings of the 23rd ACM international conference on Multimedia*, pp. 689–692, Queensland, Australia, October 2015.
- [36] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *Proceedings of the International Conference on Learning Representations*, San Diego, CA, USA, 2015.
- [37] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “Exploiting the circulant structure of tracking-by-detection with kernels,” *Computer Vision - ECCV 2012*, vol. 7575, pp. 702–715, 2012.
- [38] M. Danelljan, G. Hager, F. Khan, and M. Felsberg, “Accurate scale estimation for robust visual tracking,” in *Proceedings of the British Machine Vision Conference*, January 2014.
- [39] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “High-speed tracking with kernelized correlation filters,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2014.
- [40] C. Ma, J.-B. Huang, X. Yang, and Y. Ming-Hsuan, “Hierarchical convolutional features for visual tracking,” in *Proceedings of the IEEE international conference on computer vision*, pp. 3074–3082, Santiago, Chile, December 2015.
- [41] Y. Song, C. Ma, L. Gong, J. Zhang, R. W. H. Lau, and Y. Ming-Hsuan, “CREST: convolutional residual learning for visual tracking,” in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2574–2583, Venice, Italy, October 2017.
- [42] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, “High performance visual tracking with siamese region proposal network,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8971–8980, Salt Lake City, UT, USA, June 2018.