Using Segmentation With Multi-Scale Selective Kernel for Visual Object Tracking

Feng Bao, Yifei Cao, Shunli Zhang, Beibei Lin, and Sicong Zhao

***Abstract*—Generic visual object tracking is challenging due to various difficulties, e.g. scale variations and deformations. To solve those problems, we propose a novel multi-scale selective kernel module for tracking, which contains small-scale and large-scale branches to model the target at different scales and attention mechanism to capture the more effective appearance information of the target. In our module, we cascade multiple small-scale convolutional blocks as an equivalent large-scale branch to extract large-scale features of the target effectively. Besides, we present a hybrid strategy for feature selection to extract significant information from features of different scales. Based on the current excellent segmentation tracking framework, we propose a novel tracking network that leverages our module at multiple places in the upsample phase to construct a more accurate and robust appearance model. Extensive experimental results show that our tracker outperforms other state-of-the-art trackers on multiple challenging benchmarks including VOT2018, TrackingNet, DAVIS-2017, and YouTube-VOS-2018 while achieves real-time tracking.**

***Index Terms*—Feature selection, object tracking, selective kernel, video segmentation.**

I. Introduction

VISUAL object tracking has been a fundamental topic of computer vision for decades and has extensive applications such as autonomous driving [1], human-computer interaction, and traffic video surveillance [2]. The objective of tracking is to estimate the location and the size of the object in each frame of the video according to the given initial target annotation.

Recently, deep learning has achieved great success in tracking and other tasks [3]–[6]. Specifically, the trackers based on Siamese and segmentation networks become popular. The trackers based on Siamese networks are trained on datasets with rectangular annotations while those based on segmentation networks are constructed based on video object segmentation datasets annotated by masks. Both types of trackers represent the tracking results in the form of the rectangle bounding box for evaluation.

The basic idea of Siamese trackers is to learn a similarity measurement between the exemplar and search image regions through depth-wise correlation and the generated similarity feature can be exploited for further classification and regression tasks [7], [8]. Different strategies [9], [10] with complicated network architectures, fancy modules [11], [12] or more constraints [13], [14] have been introduced into Siamese trackers, which greatly improves the representation ability and the running speed of the network. For example, the Binary Channel Manipulation (BCM) [15] is proposed as the search algorithm to automatically select and combine matching operators, and the unified unsupervised/weakly supervised framework [16] is introduced to dramatically reduce the burden of the annotation.

However, Siamese-like trackers take all the content of the annotation area containing some background as the object, which may interfere with the representation of the target. As a result, trackers based on segmentation pixel-level masks come into fashion. To improve tracking accuracy, Cheng *et al.* [17] proposed the modular interactive segmentation framework called MiVOS to decouple interaction-to-mask and mask prorogation. However, the interaction operation is required during the segmentation process. Yan *et al.* [18] adopt the refinement module Alpha-Refine (AR) and auxiliary segmentation heads, which can enhance the tracker's box estimation quality but introduce extra huge parameters.

Since only rectangular annotations are provided in most cases, the trackers have to view the whole area as the object and translate the mask prediction into a rotated or axis-aligned rectangle. The translation bridging these two misaligned representations is vulnerable to the inaccurate mask prediction because the network is prone to take the nearby background as the part of the target which changes in scale and shape frequently during tracking. Moreover, current segmentation tracking networks do not pay enough attention to adapting to the variations of the target, which may make mistakes to the segmentation and degrade the tracking performance as Fig. 1 shows.

To solve the above issues, we propose a novel multi-scale selective kernel module (MSKM), which contains small-scale and large-scale branches to take the different scales of the target into account and adaptively select effective scale information to improve the robustness of the network to the variations. The small-scale branch concentrates on the small context of the target, which is suitable for the target whose scale is reduced or remains constant, and the large-scale branch focuses on the large surroundings of the target, which adapts to the scale increase of the target. In our MSKM, we combine average and maximization strategies to select appropriate scale features for comprehensive

Manuscript received September 8, 2021; revised November 26, 2021; accepted November 28, 2021. Date of publication January 14, 2022; date of current version February 16, 2022. This work was supported in part by the National Natural Science Foundation of China under Grants 61976017 and 61601021, in part by the Beijing Natural Science Foundation under Grant 4202056, and in part by the Fundamental Research Funds for the Central Universities under Grant 2020JBM078. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mylene Q. Farias. *(F. Bao and Y. Cao contributed equally to this work and should be considered as co-first author.) (Corresponding author: Shunli Zhang.)*

Feng Bao, Yifei Cao, Shunli Zhang, and Beibei Lin are with the School of Software Engineering, Beijing Jiaotong University, Beijing 100044, China (e-mail: 18121721@bjtu.edu.cn; 20126289@bjtu.edu.cn; slzhang@bjtu.edu.cn; 18126289@bjtu.edu.cn).

Sicong Zhao is with Beijing Aerocim Technology Company, Ltd., China Aerospace Science and Industry Corporation (CASIC), Beijing 102308, China (e-mail: zhaosc_casic@163.com).

Digital Object Identifier 10.1109/LSP.2022.3143360

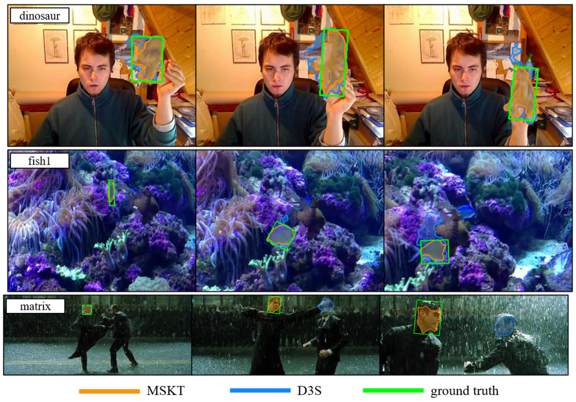

Fig. 1. Segmentation results of our MSKT, D3S and corresponding ground truth results on some sequences of VOT2018 benchmark. Sequence names are marked on the top left corner of each row.

representation. Besides, we propose a segmentation based tracking network that inserts the proposed MSKM into the refinement part to build a more accurate and robust appearance model for mask prediction and target location. Further, we construct a novel multi-scale selective kernel tracker (MSKT) based on the proposed network, which is comprehensively evaluated on popular tracking benchmarks.

The contributions of our work are summarized as follows:

1) We propose a novel multi-scale selective kernel module for scale feature selection and design a mixed selection strategy, which can effectively address the scale variations and deformations.
2) We propose a novel tracking network that integrates our proposed module to refine the up-sampled features, improving the accuracy and robustness.
3) Comprehensive experiments on multiple challenging benchmarks demonstrate the significant performance improvement of the proposed tracker.

II. Method

A. Network Architecture

The architecture of our network is shown in Fig. 3 which consists of three parts, i.e. feature extraction, coarse object segmentation, and feature refinement. The feature extraction part provides various multi-channel features that represent the object in both low-level and high-level forms. The segmentation part performs coarse segmentation and localization based on the high-level feature and generates the object-specific segmentation feature from the results. In the feature refinement part, the segmentation feature is first up-sampled by doubling its space dimension while halving the channel dimension through the Up block. Then our MSKM enhances the up-sampled feature on scale through attention mechanism and conducts point-wise addition with the corresponding low-level feature adjusted by one $Conv$ block for further refinement. We place our MSKM after 3 Up blocks to select the effective scales for feature fusion at different refinement phases. After the refinement part and softmax function, our network predicts the positive and negative masks of the target.

B. Multi-Scale Selective Kernel Module

In our method, we propose the MSKM to better adapt to the scale changes. The structure of MSKM is shown in Fig. 2. Different from [20], to reduce the complexity of computation, MSKM adopts two 3x3 convolutional kernels instead of a single 5x5 kernel. In addition, MSKT utilizes the hybrid pooling strategy including both AVG and MAX pooling for multi-scale feature fusion. Since the AVG pooling focuses on more image background while MAX pooling can retain more texture information, the proposed hybrid pooling strategy is helpful to obtain more comprehensive features, improving the representation ability.

In our formulation, the scale selection operation can be represented as follows

$$s = \mathrm{GAP}\sum_i \Phi^{(i)}(X) + \mathrm{GMP}\sum_i \Phi^{(i)}(X) \tag{1}$$

where X denotes input feature, $\Phi^{(i)}(\cdot)$ denotes the scale branch that has the number of i cascaded convolution blocks, GAP denotes global average pooling, GMP denotes global max pooling and s denotes the mixed channel feature of the two pooling strategies.

After obtaining the channel feature, we follow the pattern of the selective kernel to further extract channel weights for corresponding scale features via fully connected layers and softmax function. Then we apply element-wise addition between enhanced features of different scales which are weighted by corresponding channel weights. The final scale selected feature can be represented as

$$X' = \sum_i \underset{i}{\mathrm{softmax}}(\underset{j}{\mathrm{cat}}\{g_j(f(s))\}) \odot \Phi^{(i)}(X) \tag{2}$$

where $f(\cdot)$ denotes the fc layer for channel compression, $g_j(\cdot)$ denotes the fc layer for channel restoration of j-th scale, $\mathrm{cat}\{\cdot\}$ denotes concatenation for all j scale vectors, $\mathrm{softmax}$ denotes the softmax function and its subscript i denotes the channel weight of i-th scale, $\odot$ denotes channel-wise multiplication and X' denotes the scale selected feature.

C. Loss Function

To learn a more accurate and robust segmentation model, we set the network to predict both positive and negative masks of the target and train it with two types of labels that are complementary to each other. For the labels, 1 is used to represent the point belonging to the object and 0 corresponds to the background. Besides, we adopt the binary cross-entropy loss function to compute the errors on both positive and negative prediction outputs. The loss can be realized by taking the average

$$L = \frac{BCE(Y_P' - Y_P) + BCE(Y_N' - Y_N)}{2} \tag{3}$$

where L denotes total loss, BCE denotes the abbreviation of binary cross-entropy loss function computing the difference in the binary situation, Y' denotes the prediction mask, Y denotes the ground truth mask, the subscript P and N denote the positive and the negative, respectively.

III. Experiments

A. Experiment Settings

Our network is trained on the video object segmentation dataset YouTube-VOS-2018 [34] To avoid learning the bias that the object is always at the center, we apply a random offset to our training samples and corresponding annotations. We adopt 128 pairs of batches for 30 epochs with 1000 iterations due to the faster convergence in optimization. We test our MSKT comprehensively on multiple challenging object tracking benchmarks including, VOT2018 [36], TrackingNet [37], DAVIS-2017 [38]

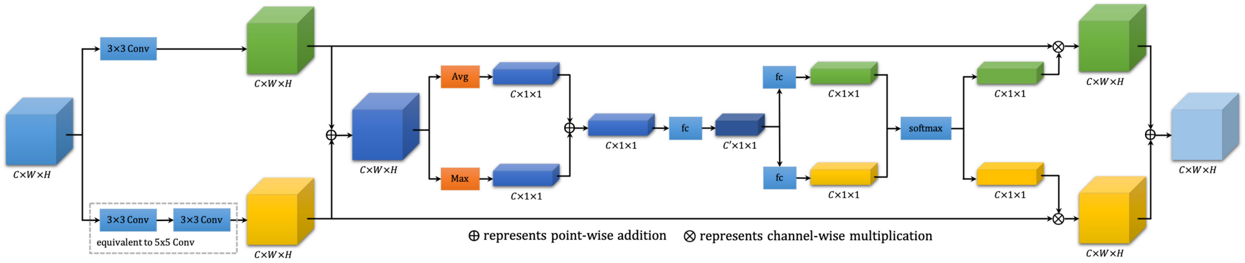

Fig. 2. Structure of our multi-scale selective kernel module. "3×3 Conv" denotes 3×3 convolutional block, "Avg" is global average pooling layer, "Max" is global max pooling layer and "fc" is fully connected layer.

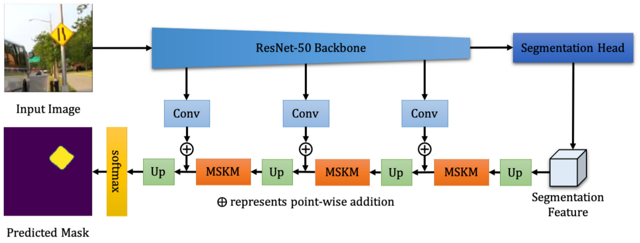

Fig. 3. Architecture of our tracking network. "Conv" means a block composed of a 3×3 convolutional layer, a batch normalization layer and a ReLU layer. "MSKM" means our proposed module. "Up" means a block composed of a up-sample layer and two "Conv" blocks. The overall network framework is based on [19], and the "MSKM" is inspired by [20].

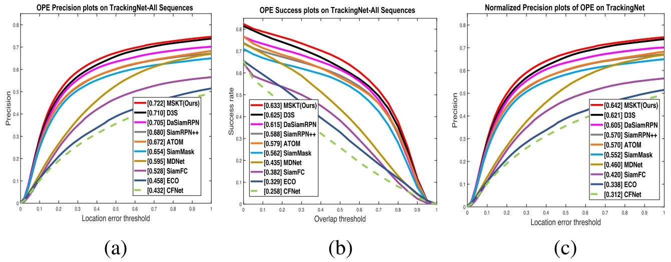

Fig. 4. Comparison results of the competing trackers on TrackingNet. (a) Precision plot. (b) Success plots. (c) Normalized Precision plots.

and YouTube-VOS-2018 [34]. Our MSKT is implemented on the Pytorch platform and runs at an average of 30fps on a single Nvidia GTX 1080Ti GPU.

B. Experiment on VOT2018 and TrackingNet

VOT is the abbreviation of visual object tracking challenge EAO (expected average overlap), robustness R (failure rate per video), and accuracy A (average overlap in successful tracking) are used for evaluation. The accuracy of frame T on video sequence vs is defined based on $\Phi_T = (BB_T^{Gt} \cap BB_T^{Pe})/(BB_T^{Gt} \cup BB_T^{Pe})$, where BB_T^{Gt} represents the bounding box corresponding to the ground truth in frame T, and BB_T^{Pe} represents the bounding box predicted by the tracker in frame T. The formulation of the average accuracy on all of the valid frames as below:

$$A = \frac{1}{N_{\text{val}}} \frac{1}{N_{rep}} \sum_{T=1}^{N_{\text{val}}} \sum_{k=1}^{N_{rep}} \Phi_T(k) \tag{4}$$

where $\Phi_T(k)$ denotes the accuracy of the tracker at frame T in the k-th repetition. N_{rep} and N_{val} denote the number of repetitions and valid frames, respectively. R is defined as:

$$R = \frac{1}{N_{rep}} \sum_{k=1}^{N_{rep}} F(k) \tag{5}$$

where $F(k)$ means the number of failure times in the k-th repetition.

$$EAO = \frac{1}{(N_{hi} - N_{lo})} \sum_{N_v=N_{lo}}^{N_{hi}} \Phi_{N_s} \tag{6}$$

where $\hat{\Phi}_{N_s}$ is computed as the average of the expected average overlap curve values Φ_{N_s} over an interval (N_{lo}, N_{hi}). Φ_{N_s} can be obtained by averaging the average overlaps on video sequences with N_s frames long.

On VOT2018, we compare MSKT with the following state-of-the-art trackers: SiamBAN [21], SiamAttn [11], DiMP-50 [22], SiamRPN++ [23], ATOM [24], SPM [27], SiamMask [26] and the LADCF [25]. Results are reported in Table I. MSKT outperforms the second-best tracker by 0.012 on EAO and by 10% on robustness, respectively. Among the trackers compared, D3S, SiamMask, and SPM are segmentation based trackers. MSKT outperforms them by a large margin because our MSKM may capture more effective scale information to obtain finer predictions.

TrackingNet is a large-scale dataset for object tracking. Trackers are ranked according to these criteria, including Area Under the Curve (AUC) which denotes the success at different thresholds, precision (Prec.), and normalized precision (Prec.$_{\text{N}}$) [37].

Following [37], our MSKT is compared with the top trackers including ECO [39], SiamFC [40], CFNet [40], MDNet [41], D3S [19], SiamRPN++[23], DaSiamRPN [42], and more recent trackers ATOM [24] and SiamMask [26]. The precision, AUC and normalized precision plots are shown in Fig. 4. It can be observed that the our MSKT obtains 0.722, 0.633 and 0.642 on precision, AUC and normalized precision, respectively, which outperforms most of the methods reported in [37] and achieves the best results.

C. Experiment on DAVIS-2017 and YouTube-VOS-2018

Both DAVIS-2017 and YouTube-VOS-2018 have dense annotations. We utilize the measurements [43] $\mathcal{J}_M$ and $\mathcal{F}_M$ to represent the region similarity and boundary accuracy on

TABLE I
COMPARISON RESULTS ON VOT2018

	Ours	D3S [19]	SiamAttn [11]	SiamBAN [21]	DiMP-50 [22]	SiamRPN++ [23]	ATOM [24]	LADCF [25]	SiamMask [26]	SPM [27]
EAO↑	**0.501**	**0.489**	0.470	0.452	0.440	0.414	0.401	0.389	0.380	0.338
R↓	**0.135**	**0.150**	0.160	0.178	0.153	0.234	0.204	0.159	0.276	0.300
A↑	**0.65**	**0.64**	0.63	0.60	0.60	0.60	0.59	0.51	0.61	0.58

TABLE II
COMPARISON RESULTS ON DAVIS-2017 AND YOUTUBE-VOS-2018

	DAVIS-2017		YouTube-VOS-2018				
	$\mathcal{J}_M$↑	$\mathcal{F}_M$↑	$\mathcal{J}_s$↑	$\mathcal{F}_s$↑	$\mathcal{J}_u$↑	$\mathcal{F}_u$↑	$\mathcal{O}$↑
Ours	**64.4**	**68.1**	**66.3**	**69.1**	59.7	**70.1**	**66.3**
GC [28]	**66.4**	66.9	**63.1**	**65.9**	**67.2**	**72.0**	**67.1**
MuG-W [29]	54.1	58.0	-	-	-	-	-
RGMP [30]	64.2	67.6	59.5	53.3	45.2	57.2	53.8
OnAVOS [31]	61.6	**69.1**	60.1	46.6	**62.7**	51.4	55.2
OSMN [32]	52.5	57.1	60.0	40.6	60.1	44.0	51.2
OSVOS [33]	56.6	63.9	59.8	54.2	60.4	60.7	58.8
SiamMask [26]	54.3	58.5	60.2	45.1	58.2	47.7	52.8
D3S [19]	57.8	63.8	-	-	-	-	-

TABLE III
RESULTS OF DIFFERENT VARIANTS OF MSKT ON VOT2018

	MSKT	MSKT-3	MSKT-4	MSKT-Avg	MSKT-Max	MSKT-3+5
EAO↑	**0.501**	0.490	0.480	0.457	**0.492**	0.484
R↓	**0.135**	0.152	0.173	0.174	0.162	**0.151**
A↑	**0.65**	**0.65**	0.63	**0.64**	**0.64**	**0.64**

DAVIS-2017, respectively.

$$\mathcal{J}_M = \frac{1}{N_{vs}} \sum_{i=1}^{N_{vs}} \frac{|M_i \cap G_i|}{|M_i \cup G_i|}, \mathcal{F}_M = \frac{2PR}{P+R} \tag{7}$$

where $\mathcal{J}_M$ denotes the IOU between the mask M_i and Ground truth G_i, $\mathcal{F}_M$ is the measure of contour based on accuracy P and recall R. N_{vs} is the number of sequences.

On YouTube-VOS-2018 benchmark, we adopt $\mathcal{J}_s$, $\mathcal{F}_s$, $\mathcal{J}_u$, $\mathcal{F}_u$, and $\mathcal{O}$ for evaluation. $\mathcal{J}_s$ and $\mathcal{F}_s$ correspond to the seen categories, while $\mathcal{J}_u$ and $\mathcal{F}_u$ correspond to the unseen categories. We report these index for accuracy assessment. Moreover, the overall score $\mathcal{O}$ is generated by averaging $\mathcal{J}_{(\cdot)}$ and $\mathcal{F}_{(\cdot)}$ scores on seen and unseen classes.

We compare our MSKT with several state-of-the-art trackers including GC [28], MuG-W [29], RGMP [30], OnAVOS [31], OSVOS [33], OSMN [32], SiamMask [26], and D3S [19], and show the results in Table II. It can be seen that our tracker achieves the optimal performance in DAVIS-2017. MSKT obtain 66.3% average scores of four measures in YouTube-VOS-2018, which outperforms the Siamese network-based trackers SiamMask and OSVOS by 13.5% and 7.5%, respectively. The results in Table II indicate that the proposed MSKT achieves that optimal performance in both DAVIS-2017 and YouTube-VOS-2018 benchmarks as well.

D. Ablation Study

MSKM plays an important role in our method. To further study the influence of the inner structure of our MSKM on mask prediction and investigate the influence of the number of scale features on scale selection, we construct two variant trackers, MSKT-3 and MSKT-4, which has 3 and 4 scale branches respectively for comparison on VOT2018 benchmark. Besides, we study the influence of channel weight generation strategies. We built the other two methods, MSKT-Avg and MSKT-Max, which adopt global average pooling and global max pooling respectively to generate channel weights from the summed scale feature. Moreover, we also study the influence of different ways to extract the large-scale features of the target. Here, we built the variant MSKT-3+5 which has two branches of 3×3 and 5×5 convolutional layers respectively to explain the advantages of multiple cascaded 3×3 layers in our MSKM.

From the Table III we can find that the EAO of MSKT is 0.011 greater and robustness is 0.017 lower than MSKT-3, which means too more scale branches may affect the segmentation performance and harm the accuracy and robustness of the network. Further experimental results on MSKT-4 confirm such inference because all the three indicators of MSKT-4 are worse than those of MSKT-3. Moreover, different channel feature extraction methods are influential to our MSKM. Either MSKT-Avg or MSKT-Max is worse than MSKT which adopts both global average pooling and global max pooling functions. The comparison results show that both global pooling functions have their own preferences for channel features and are complementary to each other in channel feature extraction where the max one provides more contribution to performance promotion. The last comparative variant is MSKT-3+5 which adopts one 5×5 convolutional kernel to extract a large scale feature and has the same size of the receptive field of two cascaded 3×3 kernels. The experimental results of MSKT-3+5 indicate that a network with two cascaded 3×3 convolutions outperforms that with 5×5 on all three indicators by 0.017, 0.016, and 0.01, respectively. It is due to the more nonlinearities provided by one more convolutional block which facilitates building a more accurate and robust model of the large target.

IV. CONCLUSION

In this letter, we have proposed a novel multi-scale selective kernel module that has small-scale and large-scale branches to adapt to targets of different scales to improve tracking performance on scale variations and deformations. In the module, we adopt the cascaded small-scale convolutional blocks which provide more nonlinearities and facilitate optimization to capture larger context. Besides, we adopt the mixture of global average and max pooling functions to select effective channel feature. We apply our module after each up-sample stage in the refinement part of the segmentation tracking architecture for more accurate and robust mask prediction. Experimental results demonstrate that our tracker outperforms most state-of-the-art trackers on multiple challenging benchmarks, which show its great potential in object tracking.

REFERENCES

[1] S. Tang, M. Andriluka, B. Andres, and B. Schiele, "Multiple people tracking by lifted multicut and person re-identification," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 3701–3710.

[2] K.-H. Lee and J.-N. Hwang, "On-road pedestrian tracking across multiple driving recorders," *IEEE Trans. Multimedia*, vol. 17, pp. 1429–1438, 2015.

[3] D. Yuan, X. Chang, P.-Y. Huang, Q. Liu, and Z. He, "Self-supervised deep correlation tracking," *IEEE Trans. Image Process.*, vol. 30, pp. 976–985, 2020.

[4] X. Chang, F. Nie, S. Wang, Y. Yang, X. Zhou, and C. Zhang, "Compound rank-k projections for bilinear analysis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 7, pp. 1502–1513, Jul. 2015.

[5] Z. Li, F. Nie, X. Chang, Y. Yang, C. Zhang, and N. Sebe, "Dynamic affinity graph construction for spectral clustering using multiple features," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6323–6332, Dec. 2018.

[6] Z. Li, F. Nie, X. Chang, L. Nie, H. Zhang, and Y. Yang, "Rank-constrained spectral clustering with flexible embedding," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6073–6082, Dec. 2018.

[7] A. He, C. Luo, X. Tian, and W. Zeng, "A twofold Siamese network for real-time object tracking," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 4834–4843.

[8] Z. Zhang and H. Peng, "Deeper and wider Siamese networks for real-time visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 4586–4595.

[9] X. Dong, J. Shen, W. Wang, L. Shao, H. Ling, and F. Porikli, "Dynamical hyperparameter optimization via deep reinforcement learning in tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1515–1529, May 2021.

[10] X. Dong, J. Shen, D. Yu, W. Wang, J. Liu, and H. Huang, "Occlusion-aware real-time object tracking," *IEEE Trans. Multimedia*, vol. 19, pp. 763–771, 2017.

[11] Y. Yu, Y. Xiong, W. Huang, and M. R. Scott, "Deformable Siamese attention networks for visual object tracking," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2020, pp. 6727–6736.

[12] Z. Li, B. Li, J. Gao, L. Li, and W. Hu, "Manipulating template pixels for model adaptation of Siamese visual tracking," *IEEE Signal Process. Lett.*, vol. 27, pp. 1690–1694, 2020.

[13] H. Fan and H. Ling, "Siamese cascaded region proposal networks for real-time visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 7944–7953.

[14] Y. Zhang, L. Wang, J. Qi, D. Wang, M. Feng, and H. Lu, "Structured Siamese network for real-time visual tracking," in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 351–366.

[15] Z. Zhang *et al.*, "Learn to match: Automatic matching network design for visual tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 13339–13348.

[16] Z. Zhou *et al.*, "Saliency-associated object tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9866–9875.

[17] H. K. Cheng, Y.-W. Tai, and C.-K. Tang, "Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2021, pp. 5559–5568.

[18] B. Yan, X. Zhang, D. Wang, H. Lu, and X. Yang, "Alpha-refine: Boosting tracking performance by precise bounding box estimation," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2021, pp. 5289–5298.

[19] A. Lukezic, J. Matas, and M. Kristan, "D3S-A discriminative single shot segmentation tracker," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2020, pp. 7133–7142.

[20] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 510–519.

[21] Z. Chen, B. Zhong, G. Li, S. Zhang, and R. Ji, "Siamese box adaptive network for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2020, pp. 6668–6677.

[22] G. Bhat, M. Danelljan, L. V. Gool, and R. Timofte, "Learning discriminative model prediction for tracking," in *Proc. IEEE Int. Conf. Comput. Vision*, 2019, pp. 6182–6191.

[23] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "SiamRPN: Evolution of Siamese visual tracking with very deep networks," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 4277–4286.

[24] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ATOM: Accurate tracking by overlap maximization," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 4655–4664.

[25] T. Xu, Z.-H. Feng, X.-J. Wu, and J. Kittler, "Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking," *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5596–5609, Nov. 2019.

[26] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 1328–1338.

[27] K. Dai, D. Wang, H. Lu, C. Sun, and J. Li, "Visual tracking via adaptive spatially-regularized correlation filters," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 4665–4674.

[28] Y. Li, Z. Shen, and Y. Shan, "Fast video object segmentation using the global context module," in *Proc. Eur. Conf. Comput. Vision*, Cham: Springer, 2020, pp. 735–750.

[29] X. Lu, W. Wang, J. Shen, Y.-W. Tai, D. J. Crandall, and S. C. Hoi, "Learning video object segmentation from unlabeled videos," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2020, pp. 8957–8967.

[30] S. W. Oh, J. Y. Lee, K. Sunkavalli, and S. J. Kim, "Fast video object segmentation by reference-guided mask propagation," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 7376–7385.

[31] P. Voigtlaender and B. Leibe, "Online adaptation of convolutional neural networks for the 2017 davis challenge on video object segmentation," in *Proc. DAVIS Challenge Video Object Segmentation-CVPR Workshops*, 2017, vol. 5, no. 6.

[32] R. Tang, H. Song, K. Zhang, and S. Jiang, "Video object segmentation via attention-modulating networks," *Electron. Lett.*, vol. 55, no. 8, pp. 455–457, 2019.

[33] S. Caelles, K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool, "One-shot video object segmentation," in *Proc.-30th IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, vol. 2017, pp. 5320–5329.

[34] N. Xu *et al.*, "Youtube-VOS: Sequence-to-sequence video object segmentation," in *Proc. Eur. Conf. Comput. Vision*, Switzerland, 2018, pp. 603–619.

[35] S. Hadfield, R. Bowden, and K. Lebeda, "The visual object tracking VOT2016 challenge results," *Lecture Notes Comput. Sci.*, vol. 9914, pp. 777–823, 2016.

[36] M. Kristan *et al.*, "The sixth visual object tracking VOT2018 challenge results," in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 0–0.

[37] M. Muller, A. Bibi, S. Giancola, S. Alsubaihi, and B. Ghanem, "TrackingNet: A large-scale dataset and benchmark for object tracking in the wild," in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 300–317.

[38] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool, "The 2017 Davis challenge on video object segmentation," 2017, *arXiv:1704.00675*.

[39] M. Danelljan, G. Bhat, F. Shahbaz Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 6931–6939.

[40] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional Siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vision*, Cham, Switzerland: Springer, 2016, pp. 850–865.

[41] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 4293–4302.

[42] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu, "Distractor-aware Siamese networks for visual object tracking," in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 101–117.

[43] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 724–732.