# The seventh visual object tracking VOT2019 challenge results

# The Seventh Visual Object Tracking VOT2019 Challenge Results

Matej Kristan[1] and Jiří Matas[2]Aleš Leonardis[3]Michael Felsberg[4]Roman Pflugfelder[5,6]Joni-Kristian Kämäräinen[7]Luka Čehovin Zajc[1]Ondrej Drbohlav[2]Alan Lukežič[1]Amanda Berg[4,8]Abdelrahman Eldesokey[4]Jani Käpylä[7]Gustavo Fernández[5]Abel Gonzalez-Garcia[18]Alireza Memarmoghadam[50]Andong Lu[9]Anfeng He[52]Anton Varfolomieiev[37]Antoni Chan[17]Ardhendu Shekhar Tripathi[23]Arnold Smeulders[45]Bala Suraj Pedasingu[29]Bao Xin Chen[58]Baopeng Zhang[12]Baoyuan Wu[43]Bi Li[28]Bin He[10]Bin Yan[19]Bing Bai[20]Bing Li[16]Bo Li[40]Byeong Hak Kim[25,33]Chao Ma[41]Chen Fang[35]Chen Qian[40]Cheng Chen[38]Chenglong Li[9]Chengquan Zhang[10]Chi-Yi Tsai[42]Chong Luo[34]Christian Micheloni[55]Chunhui Zhang[16]Dacheng Tao[54]Deepak Gupta[45]Dejia Song[28]Dong Wang[19]Efstratios Gavves[45]Eunu Yi[25]Fahad Shahbaz Khan[4,30]Fangyi Zhang[16]Fei Wang[40]Fei Zhao[16]George De Ath[49]Goutam Bhat[23]Guangqi Chen[40]Guangting Wang[52]Guoxuan Li[40]Hakan Cevikalp[21]Hao Du[34]Haojie Zhao[19]Hasan Saribas[22]Ho Min Jung[33]Hongliang Bai[11]Hongyuan Yu[16,34]Houwen Peng[34]Huchuan Lu[19]Hui Li[32]Jiakun Li[12]Jianhua Li[19]Jianlong Fu[34]Jie Chen[57]Jie Gao[57]Jie Zhao[19]Jin Tang[9]Jing Li[26]Jingjing Wu[27]Jingtuo Liu[10]Jinqiao Wang[16]Jinqing Qi[19]Jinyue Zhang[57]John K. Tsotsos[58]Jong Hyuk Lee[33]Joost van de Weijer[18]Josef Kittler[53]Jun Ha Lee[33]Junfei Zhuang[13]Kangkai Zhang[16]Kangkang Wang[10]Kenan Dai[19]Lei Chen[40]Lei Liu[9]Leida Guo[59]Li Zhang[51]Liang Wang[16]Liangliang Wang[28]Lichao Zhang[18]Lijun Wang[19]Lijun Zhou[48]Linyu Zheng[16]Litu Rout[39]Luc Van Gool[23]Luca Bertinetto[24]Martin Danelljan[23]Matteo Dunnhofer[55]Meng Ni[19]Min Young Kim[33]Ming Tang[16]Ming-Hsuan Yang[46]Naveen Paluru[29]Niki Martinel[55]Pengfei Xu[20]Pengfei Zhang[54]Pengkun Zheng[38]Pengyu Zhang[19]Philip H.S. Torr[51]Qi Zhang Qiang Wang[16,31]Qing Guo[44]Radu Timofte[23]Rama Krishna Gorthi[29]Richard Everson[49]Ruize Han[44]Ruohan Zhang[57]Shan You[40]Shao-Chuan Zhao[32]Shengwei Zhao[16]Shihu Li[10]Shikun Li[16]Shiming Ge[16]Shuai Bai[13]Shuosen Guan[59]Tengfei Xing[20]Tianyang Xu[32]Tianyu Yang[17]Ting Zhang[14]Tomáš Vojíř[47]Wei Feng[44]Weiming Hu[16]Weizhao Wang[38]Wenjie Tang[14]Wenjun Zeng[34]Wenyu Liu[28]Xi Chen[60]Xi Qiu[56]Xiang Bai[28]Xiao-Jun Wu[32]Xiao-Jun Wu[32]Xiaoyun Yang[15]Xier Chen[57]Xin Li[26]Xing Sun[59]Xingyu Chen[16]Xinmei Tian[52]Xu Tang[10]Xue-Feng Zhu[32]Yan Huang[16]Yanan Chen[57]Yanchao Lian[57]Yang Gu[20]Yang Liu[36]Yanjie Chen[40]Yi Zhang[59]Yinda Xu[60]Yingming Wang[19]Yingping Li[57]Yu Zhou[28]Yuan Dong[13]Yufei Xu[52]Yunhua Zhang[19]Yunkun Li[32]Zeyu Wang Zhao Luo[16]Zhaoliang Zhang[14]Zhen-Hua Feng[53]Zhenyu He[26]Zhichao Song[20]Zhihao Chen[44]Zhipeng Zhang[16]Zhirong Wu[34]Zhiwei Xiong[52]Zhongjian Huang[57]Zhu Teng[12]Zihan Ni[10]

[1]University of Ljubljana, Slovenia
[2]Czech Technical University, Czech Republic
[3]University of Birmingham, United Kingdom
[4]Linköping University, Sweden
[5]Austrian Institute of Technology, Austria
[6]TU Wien, Austria
[7]Tampere University, Finland
[8]Termisk Systemteknik AB, Sweden
[9]Anhui University, China
[10]Department of Computer Vision Technology (VIS), Baidu Inc., China
[11]Beijing FaceAll Co., China

[12]Beijing Jiaotong University, China
[13]Beijing University of Posts and Telecommunications, China
[14]China National Electronics Import & Export Corporation, China
[15]China Science IntelliCloud Technology Co. Ltd, China
[16]Chinese Academy of Sciences, China
[17]City University of Hong Kong, Hong Kong
[18]Computer Vision Center, Spain
[19]Dalian University of Technology, China
[20]Didi Chuxing, China
[21]Eskisehir Osmangazi University, Turkey
[22]Eskisehir Technical University, Turkey
[23]ETH Zurich, Switzerland
[24]FiveAI, United Kingdom
[25]Hanwha Systems Co., South Korea
[26]Harbin Institute of Technology at Shenzhen, China
[27]Hefei University of Technology, China
[28]Huazhong University of Science and Technology, China
[29]IIT Tirupati, India
[30]Inception Institute of Artificial Intelligence, UAE
[31]INTELLIMIND LTD, China
[32]Jiangnan University, China
[33]Kyungpook National University, South Korea
[34]Microsoft Research, China
[35]Nanjing Normal University, China
[36]North China Electric Power University, China
[37]NTUU Igor Sikorsky Kyiv Polytechnic Institute, Ukraine
[38]Peking University, China
[39]SAC-ISRO, India
[40]SenseTime, China
[41]Shanghai Jiao Tong University, China
[42]Tamkang University, Taiwan
[43]Tencent AI Lab, China
[44]Tianjin University, China
[45]University of Amsterdam, The Netherlands
[46]University of California at Merced, USA
[47]University of Cambridge, United Kingdom
[48]University of Chinese Academy of Sciences, China
[49]University of Exeter, United Kingdom
[50]University of Isfahan, Iran
[51]University of Oxford, United Kingdom
[52]University of Science and Technology of China, China
[53]University of Surrey, United Kingdom
[54]University of Sydney, Australia
[55]University of Udine, Italy
[56]Xianan JiaoTong University, China
[57]Xidian University, China

[58]York University, Canada
[59]YouTu Lab, China
[60]Zhejiang University, China

## Abstract

*The Visual Object Tracking challenge VOT2019 is the seventh annual tracker benchmarking activity organized by the VOT initiative. Results of 81 trackers are presented; many are state-of-the-art trackers published at major computer vision conferences or in journals in the recent years. The evaluation included the standard VOT and other popular methodologies for short-term tracking analysis as well as the standard VOT methodology for long-term tracking analysis. The VOT2019 challenge was composed of five challenges focusing on different tracking domains: (i) VOT-ST2019 challenge focused on short-term tracking in RGB, (ii) VOT-RT2019 challenge focused on "real-time" short-term tracking in RGB, (iii) VOT-LT2019 focused on long-term tracking namely coping with target disappearance and reappearance. Two new challenges have been introduced: (iv) VOT-RGBT2019 challenge focused on short-term tracking in RGB and thermal imagery and (v) VOT-RGBD2019 challenge focused on long-term tracking in RGB and depth imagery. The VOT-ST2019, VOT-RT2019 and VOT-LT2019 datasets were refreshed while new datasets were introduced for VOT-RGBT2019 and VOT-RGBD2019. The VOT toolkit has been updated to support both standard short-term, long-term tracking and tracking with multi-channel imagery. Performance of the tested trackers typically by far exceeds standard baselines. The source code for most of the trackers is publicly available from the VOT page. The dataset, the evaluation kit and the results are publicly available at the challenge website*[1].


## 1. Introduction

Visual object tracking has consistently been a popular research area due to significant research challenges tracking offers as well as the commercial potential of tracking-based applications. Tracking research has been historically promoted by several initiatives like PETS [105], CAVIAR[2], i-LIDS [3], ETISEO[4], CDC [31], CVBASE [5], FERET [77], LTDT [6], MOTC [55, 83] and Videonet [7].

In 2013, the VOT[1] initiative has been formed with the objective of performance evaluation standardisation. The primary goal of VOT is establishing datasets, evaluation measures and toolkits as well as creating a platform for discussing evaluation-related issues through organization of tracking challenges. Since 2013, six challenges have taken place in conjunction with ICCV2013 (VOT2013 [53]), ECCV2014 (VOT2014 [54]), ICCV2015 (VOT2015 [51]), ECCV2016 (VOT2016 [50]), ICCV2017 (VOT2017 [49]) and ECCV2018 (VOT2018 [48]).

This paper presents the VOT2019 challenge, organized in conjunction with the ICCV2019 Visual Object Tracking Workshop, and the results obtained. The VOT2019 challenge covers two categories of trackers.

The first are single-camera, single-target, model-free, causal trackers, applied to short-term tracking. The *model-free* property means that the only training information provided is the bounding box in the first frame. The *short-term* tracking means that trackers are assumed not to be capable of performing successful re-detection after the target is lost and they are therefore reset after such an event. *Causality* requires that the tracker does not use any future frames, or frames prior to re-initialization, to infer the object position in the current frame.

The second category considers single-camera, single-target, model-free long-term trackers. *Long-term* tracking means that the trackers are *required* to perform re-detection after the target has been lost and are therefore *not* reset after such an event.

With respect to VOT2018, VOT2019 extends the set of challenges. It includes five challenges dedicated to either short-term or long-term tracking in RGB, RGB+thermal and RGB+depth sequences. In the following, we overview the most closely related works and point out the contributions of VOT2019.

### 1.1. Short-term tracker evaluation

The most widely-used methodologies originate from the "Online Tracking Benchmark" (OTB) [98] and the "Visual Object Tracking challenge" (VOT) [53, 52]. The OTB [98] methodology applies a no-reset experiment in which the tracker is initialized in the first frame and it runs unsupervised until the end of the sequence. Performance is summarized by a curve showing the percentage of frames where the overlap of the predicted and the ground truth bounding boxes exceeds a series of predefined thresholds. The area under the plot is the major performance score. This score has been shown in [89, 91] to be an average overlap (AO) computed over the entire sequence of frames. A downside of the AO is that all frames after the first failure receive a zero overlap, which increases bias and variance of the estimator [52]. To increase interpretability and reduce the bias, VOT [53, 52] applies a reset-based methodology in which the tracker is reset upon drifting off the target. Accuracy and robustness are defined as two measures for probing the tracking performance and the expected average overlap (EAO) is proposed as the primary measure that combines the two aspects of tracking performance in a principled way. VOT introduceed the so-called state-of-the-art

[1] http://votchallenge.net
[2] http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1
[3] http://www.homeoffice.gov.uk/science-research/hosdb/i-lids
[4] http://www-sop.inria.fr/orion/ETISEO
[5] http://vision.fe.uni-lj.si/cvbase06/
[6] http://www.micc.unifi.it/LTDT2014/
[7] http://videonet.team

bound (*SotA* bound) on all their benchmarks. Any tracker exceeding *SotA* bound is considered state-of-the-art by the VOT standard. By introducing the *SotA* bound, the VOT initiative aimed at providing an incentive for community-wide exploration of a wide spectrum of well-performing trackers and to reduce the pressure for fine-tuning to benchmarks with the sole purpose of reaching the number one rank on particular test data.

Most tracking datasets [98, 57, 82, 62, 73, 38, 24, 74] have partially followed the trend in computer vision of increasing the number of sequences. In contrast, the VOT [53, 54, 51, 52, 50, 49, 48] datasets have been constructed with diversity in mind and were kept sufficiently small to allow fast tracker development-and-testing cycles. In VOT2017 [49] a sequestered dataset was introduced to reduce the influence of tracker over-fitting without requiring to increase the public dataset size.

Several datasets for measuring short-term tracking performance have been proposed recently. UAV123 [73] and [110] datasets focus on drone tracking. Lin et al. [104] proposed a dataset for tracking faces by mobile phones. Galoogahi et al. [28] introduced a high-frame-rate dataset to analyze trade-offs between tracker speed and robustness. VOT2017 [49] argued that proper real-timer tracking performance should be measured on standard frame-rate datasets by limiting the tracker available processing time. Čehovin et al. [93] proposed a dataset with an active camera view control using omni directional videos. Mueller et al. [74] recently re-annotated selected sequences from YouTube bounding boxes [78] to consider tracking in the wild. Similarly, Fan et al. [24] and Huang et al., [38] introduced new large datasets for training and testing visual object trackers.

It is clear that several recent datasets [38, 24] have adopted elements of the VOT dataset construction principles. Despite significant activity in dataset construction, the VOT dataset remains unique for its carefully chosen and curated sequences guaranteeing relatively unbiased assessment of performance with respect to attributes.

### 1.2. Long-term tracker evaluation

A major difference between short-term (ST) and long-term (LT) trackers is that LT trackers are required to handle situations in which the target may leave the field of view for a longer duration. This means that a natural evaluation protocol for LT trackers is a no-reset protocol.

The set of performance measures in long-term tracking is quite diverse and has not been converging like in the short-term tracking. Early work [43, 76] has thus directly adapted precision, recall and F-measure computed at 0.5 IoU (overlap) threshold. Several authors [86, 72] propose a modification of the average overlap measure by specifying an overlap equal to 1 when the tracker correctly predicts the target absence. Since such a measure does not clearly separate tracking accuracy from a re-detection ability, Lukežič et. al. [67] proposed *tracking* precision, *tracking* recall and *tracking* F-measure that do not depend on specifying the IoU threshold. They have shown that their primary measure, the tracking F-measure, reduces to a standard short-term measure (average overlap) when computed in a short-term setup. They further showed in their extended report [68] that the measure is extremely robust and allows using a very sparse temporal target annotation. Recently, [39] introduced a measure that also directly addresses the evaluation of the re-detection ability.

The first LT dataset, introduced by the LTDT challenge [6], offered a collection of specific very long videos from [43, 76, 56, 82]. Mueller et al. [73] proposed an UAV20L dataset containing twenty long sequences with many target disappearances. Recently, three benchmarks that propose datasets with many target disappearances have almost concurrently appeared [72, 67, 39]. The benchmark [72] primarily analyzes performance of short-term trackers on long sequences, and [39] proposes a huge dataset constructed from YouTube bounding boxes [78]. The authors of [67] argue that long-term tracking does not just refer to the sequence length, but more importantly to the sequence properties, like the number and the length of target disappearances, and the type of tracking output expected. Their dataset construction approach follows these guidelines. For these reasons VOT2017 based their first long-term challenge on [67].

### 1.3. Beyond RGB-only tracking

Despite the significant application potential, tracking with non-RGB and mixed modalities has received significantly less attention than pure RGB tracking. Most related works consider tracking in infrared and thermal imagery and the combination of RGB with depth. The following overview thus focuses on these two areas.

The earliest thermal and infrared (TIR) tracking comparisons were organized by the Performance Evaluation of Tracking and Surveillance (PETS) [105] in 2005 and 2015. These challenges addressed multi-camera and long-term tracking, and behavior (threat) analysis. In 2015, the VOT initiative introduced the VOT-TIR [25] challenge that focused on short-term tracking [25, 26] and adopted the well-established VOT short-term tracking performance evaluation methodology. The challenge used the LTIR [3] dataset, which was the most advanced and diverse TIR general object tracking dataset at the time. In 2016, the challenge was repeated with an updated dataset, which was refreshed with more challenging sequences. Since the dataset did not saturate in the results of 2016, the same dataset was re-used in the VOT-TIR2017 challenge.

The participants of the VOT-TIR challenges have expressed growing interest in multi-modal variants, in par-

ticular a combination of RGB and thermal data. The advantage of multi-modal data is the larger variety of possible solutions, such as applying single modality trackers and investigating early and late fusion variants similar to [46] for increasing tracking performance due to complementary information in RGB and thermal images, novel aspects of the problem formulation, such as registration and synchronization issues [47], and novel types of applications, such as cross-modality learning [4]. In order to design a novel challenge on joint RGB+thermal (RGBT) tracking, the VOT-TIR challenge paused 2018. The VOT committee decided to base the VOT-RGBT-challenge on the existing RGBT-dataset published by [60]. In contrast to VOT-TIR and in agreement with the other VOT-challenges, this dataset has been complemented with annotations for rotated bounding boxes and the VOT attributes (frame-wise and sequence-wise). The annotation process has been performed semi-automatically based on the video object segmentation method [41] and further details are given in [5]. This methodology is generic and can be applied to other modalities, such as depth, as well.

A number of datasets in RGB+depth (RGBD) tracking focus on pedestrian and hand tracking [23, 85, 14, 30] and object pose estimation for robotics [13, 80]. These datasets use pre-computed object models and only a few datasets address general object tracking. The most popular is Princeton Tracking Benchmark (PTB) [84], which contains 95 RGB-D video sequences of rigid and non-rigid objects recorded with Kinect. The choice of sensor constrains the dataset to only indoor scenarios and has limited diversity. PTB addresses long-term tracking, in which the tracker has to detect target loss and perform re-detection. The primary performance measure is the percentage of frames in which the bounding box predicted by a tracker exceeds a $0.5$ overlap with the ground truth. The overlap is artificially set to 1 when the tracker accurately predicts target absence. Spatio-Temporal Consistency dataset (STC) [99] was recently proposed to address the drawbacks of the PTB. The dataset is recorded by Asus Xtion RGB-D sensor and contains only 36 sequences, but some of these are recorded outdoor in low light. The sequences are relatively short and the short-term performance evaluation methodology is used. Most recently, Color and Depth Visual Object Tracking Dataset and Benchmark (CDTB) [65] was introduced. CDTB contains 80 sequences recorded by several RGBD sensors and contains both indoor and outdoor sequences recorded under various lighting conditions. The sequences are approximately six times longer than those in PTB and STC and contain many more occlusions and target disappearances to simulate realistic tracking conditions. CDTB focuses on long-term tracking and adopts the long-term tracking methodology from [67].

### 1.4. The VOT2019 challenge

VOT2019 considers short-term as well as long-term trackers in separate challenges. We adopt the definitions from [67] which are used to position the trackers on the short-term/long-term spectrum:

- **Short-term tracker** ($\mathrm{ST}_0$). The target position is reported at each frame. The tracker does not implement target re-detection and does not explicitly detect occlusion. Such trackers are likely to fail at the first occlusion as their representation is affected by any occluder.

- **Short-term tracker with conservative updating** ($\mathrm{ST}_1$). The target position is reported at each frame. Target re-detection is not implemented, but tracking robustness is increased by selectively updating the visual model depending on a tracking confidence estimation mechanism.

- **Pseudo long-term tracker** ($\mathrm{LT}_0$). The target position is not reported in frames when the target is not visible. The tracker does not implement explicit target re-detection but uses an internal mechanism to identify and report tracking failure.

- **Re-detecting long-term tracker** ($\mathrm{LT}_1$). The target position is not reported in frames when the target is not visible. The tracker detects tracking failure and implements explicit target re-detection.

The evaluation toolkit and the datasets are provided by the VOT2019 organizers. The challenge officially opened on April 17th 2019 with approximately a month available for results submission. The VOT2019 challenge contained five challenges:

1. **VOT-ST2019 challenge**: This challenge was addressing short-term tracking in RGB images and has been running since VOT2013 with annual updates and modifications.

2. **VOT-RT2019 challenge**: This challenge addressed the same class of trackers as VOT-ST2019, except that the trackers had to process the sequences in real-time. The challenge was introduced in VOT2017.

3. **VOT-LT2019 challenge**: This challenge was addressing long-term tracking in RGB images. The challenge was introduced in VOT2018.

4. **VOT-RGBT challenge**: This challenge was addressing short-term tracking in RGB+thermal imagery. This is a new challenge in VOT2019 and can be viewed as the next step in evolution of the VOT-TIR challenge introduced in VOT2015.

5. **VOT-RGBD challenge**: This challenge was addressing long-term tracking in RGB+depth imagery. This is a new challenge in VOT2019.

The authors participating in the challenge were required to integrate their tracker into the VOT2019 evaluation kit, which automatically performed a set of standardized experiments. The results were analyzed according to the VOT2019 evaluation methodology. Upon submission of the results, the participants were required to classify their tracker along the short-term/long-term spectrum.

Participants were encouraged to submit their own new or previously published trackers as well as modified versions of third-party trackers. In the latter case, modifications had to be significant enough for acceptance. Participants were expected to submit a single set of results per tracker. Changes in the parameters did not constitute a different tracker. The tracker was required to run with fixed parameters in all experiments. The tracking method itself was allowed to internally change specific parameters, but these had to be set automatically by the tracker, e.g., from the image size and the initial size of the bounding box, and were not to be set by detecting a specific test sequence and then selecting the parameters that were hand-tuned for this sequence.

Each submission was accompanied by a short abstract describing the tracker, which was used for the short tracker descriptions in Appendix 5. In addition, participants filled out a questionnaire on the VOT submission page to categorize their tracker along various design properties. Authors had to agree to help the VOT technical committee to reproduce their results in case their tracker was selected for further validation. Participants with sufficiently well-performing submissions, who contributed with the text for this paper and agreed to make their tracker code publicly available from the VOT page were offered co-authorship of this results paper.

To counter attempts of intentionally reporting large bounding boxes to avoid resets, the VOT committee analyzed the submitted tracker outputs. The committee reserved the right to disqualify the tracker should such or a similar strategy be detected.

To compete for the winner of VOT2019 challenge, learning on specific datasets (OTB, VOT, ALOV, UAV123, NUSPRO, TempleColor and RGBT234) was prohibited. In the case of GOT10k, a list of 1k prohibited sequences was created, while the remaining 9k+ sequences were allowed for learning. The reason was that part of the GOT10k was used for VOT-ST2019 dataset update.

The use of class labels specific to VOT was not allowed (i.e., identifying a target class in each sequence and applying pre-trained class-specific trackers was not allowed). An agreement to publish the code online on VOT webpage was required. The organizers of VOT2019 were allowed to participate in the challenge, but did not compete for the winner titles. Further details are available from the challenge homepage[8].

VOT2019 goes beyond previous challenges by updating the datasets in VOT-ST, VOT-RT and VOT-LT challenges and introducing the two new challenges: VOT-RGBT and VOT-RGBD. The VOT2019 toolkit has been updated to allow seamless use of short-term, long-term, 3 channel (RGB) and 4 channel (RGB-T/D) images.

## 2. Performance evaluation protocols

Since 2018 VOT considers two classes of trackers: short-term (ST) and long-term (LT) trackers. These two classes primarily differ on the target presence assumptions, which affects the evaluation protocol as well as performance measures. These are outlined in following two subsections.

### 2.1. ST performance evaluation protocol

In a short-term setup, the target remains within the camera field of view throughout the sequence, but may undergo partial short-lasting occlusions. The tracker is required to report the target position at each frame. The main focus of ST tracking is designing robust trackers that can track throughout significant visual appearance changes without drifting off the target. Tracking sequences are typically relatively short. The ST performance measures should thus analyze the accuracy of the target localization and drifting.

As in VOT2018 [48], three primary measures were used to analyze the short-term tracking performance: accuracy ($A$), robustness ($R$) and expected average overlap (EAO). In the following, these are briefly over-viewed and we refer to [51, 52, 91] for further details.

The VOT short-term challenges apply a reset-based methodology. Whenever a tracker predicts a bounding box not overlapping with the ground truth, a failure is detected and the tracker is re-initialized five frames after the failure. Accuracy and robustness [91] are the basic measures used to probe tracker performance in the reset-based experiments. The accuracy is the average overlap between the predicted and ground truth bounding boxes during successful tracking periods. The robustness measures how many times the tracker loses the target (fails) during tracking. The potential bias due to resets is reduced by ignoring ten frames after re-initialization in the accuracy measure (note that a tracker is reinitialized five frames after failure), which is quite a conservative margin [52].

The third primary measure, called the expected average overlap (EAO), is an estimator of the average overlap a tracker is expected to attain on a large collection of short-term sequences with the same visual properties as the given dataset. The measure addresses the problem of increased

[8] http://www.votchallenge.net/vot2019/participation.html

variance and bias of AO [98] measure due to variable sequence lengths. Please see [51] for further details on the average expected overlap measure.

**Evaluation protocol.** A tracker is evaluated on a dataset by initializing on the first frame of a sequence and reset each time the overlap between the predicted and ground truth bounding box drops to zero. Accuracy, robustness and EAO measures are then computed. Average accuracy and failure-rates are reported for stochastic trackers, which are run 15 times. For reference, the toolkit also ran a no-reset experiment and the AO [98] was computed (available in the online results).

### 2.2. LT performance evaluation protocol

In a long-term (LT) tracking setup, the target may leave the camera field of view for longer duration before re-entering it, or may undergo long-lasting complete occlusions. The tracker is thus required to report the target position only for frames in which the target is visible and is required to recover from tracking failures. Long-term sequences are thus much longer than short-term sequences to test the re-detection capability. LT measures should therefore measure the target localization accuracy as well as target re-detection capability.

In contrast to the ST tracking setup, the tracker is not reset upon drifting off the target. To account for the most general case, the tracker is required to report the target position at every frame and provide a confidence score of target presence. The evaluation protocol [67] first used in the VOT2018 is adapted.

Three long-term tracking performance measures proposed in [67] are adopted: tracking precision ($Pr$), tracking recall ($Re$) and tracking F-score. These are briefly described in the following.

The $Pr$ and $Re$ are derived in [67] from the counterparts in detection literature with important differences that draw on advancements of tracking-specific performance measures. In particular, the bounding box overlap is integrated out, leaving both measures $Pr(\tau_\theta)$ and $Re(\tau_\theta)$ depend directly on the tracker prediction certainty threshold $\tau_\theta$, i.e., the value of tracking certainty below which the tracker output is ignored. Precision and accuracy are combined into a single score by computing the tracking F-measure

$$F(\tau_\theta) = 2Pr(\tau_\theta)Re(\tau_\theta)/(Pr(\tau_\theta) + Re(\tau_\theta)). \quad (1)$$

Long-term tracking performance can thus be visualized by tracking precision, tracking accuracy and tracking F-measure plots by computing these scores for all thresholds $\tau_\theta$ [67]. The final values of $Pr$, $Re$ and $F$-measure are obtained by selecting $\tau_\theta$ that maximizes tracker-specific $F$-measure. This avoids all manually-set thresholds in the primary performance measures.

**Evaluation protocol.** A tracker is evaluated on a dataset of several sequences by initializing on the first frame of a sequence and run until the end of the sequence without re-sets. A precision-recall graph is calculated on each sequence and averaged into a single plot. This guarantees that the result is not dominated by extremely long sequences. The F-measure plot is computed according to (1) from the average precision-recall plot. The maximal score on the F-measure plot (tracking F-score) is taken as the long-term tracking primary performance measure.

## 3. Description of individual challenges

In the following we provide descriptions of all five challenges running in the VOT2019 challenge.

### 3.1. VOT-ST2019 challenge outline

This challenge addressed RGB tracking in a short-term tracking setup. The performance evaluation protocol and measures outlined in Section 2.1 were applied. In the following, the details of the dataset and the winner identification protocols are provided.

#### 3.1.1 The dataset

Results of the VOT2018 showed that the dataset was not saturated [48]. But since the same dataset has been used in VOT2017, it has been decided to refresh the public dataset by replacing 20% of the sequences (see Figure 1). In addition, 5% of the sequestered dataset has been updated as well.

A review of the published datasets showed that currently the largest dataset with carefully selected and annotated sequences is the GOT-10k [38] dataset. The dataset was analyzed and a list of 1000 diverse sequences[9] was created (by random selection from the training set of GOT-10k). This has been the pool of sequences used to replace part of the VOT-ST challenge dataset.

The sequence selection and replacement procedure was as follows. (i) All sequences in the VOT2018 public dataset were ranked according to their difficulty, using robustness measure averaged over a subset of trackers. Out of 20 least difficult sequences, 12 had been selected for replacement such that the diversity of the dataset has been maintained. (ii) Around 150 sequences have been selected at random from the update pool of 1000 sequences collected from the GOT-10k dataset. The tracking difficulty measure for each sequence has been computed using the same procedure as for the VOT2018 sequence ranking. Out of these sequences, 30 most difficult ones became the candidates for VOT2019. Of these, 12 were selected, again maintaining diversity. Figure 1 shows the sequences removed from VOT2018 public dataset and their replacement.

[9] http://www.votchallenge.net/vot2019/res/list0_prohibited_1000.txt

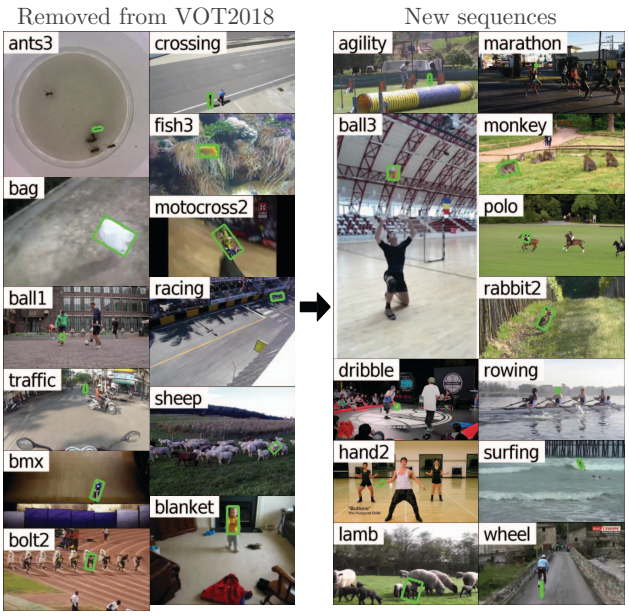


Figure 1. Sequences of VOT2018 public dataset (left column) that were replaced by new sequences in VOT2019 (right column).

Segmentation masks were manually created for tracking targets in all frames of the new sequences, and rotated bounding boxes were fitted to these segmentation masks using optimization formulation similar to the previous challenges. Per-frame visual attributes were semi-automatically assigned to the new sequences following the VOT attribute annotation protocol. In particular, each frame was annotated by the following visual attributes: (i) occlusion, (ii) illumination change, (iii) motion change, (iv) size change, (v) camera motion.

The sequestered dataset has been updated using an analogous procedure.

#### 3.1.2 Winner identification

The VOT-ST2019 winner was identified as follows. Trackers were ranked according to the EAO measure on the public dataset. Top five ranked trackers were then re-run by the VOT2019 committee on the sequestered dataset. The top ranked tracker on the sequestered dataset not submitted by the VOT2019 committee members was the winner of the VOT-ST2019 challenge.

### 3.2. VOT-RT2019 challenge outline

This challenge addressed *real-time* RGB tracking in a short-term tracking setup. The dataset was the same as in the VOT-ST2019 challenge, but the evaluation protocol was modified to emphasize the real-time component in tracking performance. In particular, the VOT-RT2019 challenge requires predicting bounding boxes faster or equal to the video frame-rate. The toolkit sends images to the tracker via the Trax protocol [88] at 20fps. If the tracker does not respond in time, the last reported bounding box is assumed as the reported tracker output at the available frame (zero-order hold dynamic model). As in VOT-ST2018, a reset-based evaluation protocol with post-reset frame skipping is applied.

#### 3.2.1 Winner identification protocol

All trackers are ranked on the public RGB short-term tracking dataset with respect to the EAO measure. The winner was identified as the top ranked tracker not submitted by the VOT2019 committee members.

### 3.3. VOT-LT2019 challenge outline

This challenge addressed RGB tracking in a long-term tracking setup and is a continuation of the VOT-LT2018 challenge. As in VOT-LT2018, we adopt the definitions from [67], which are used to position the trackers on the short-term/long-term spectrum. A long-term performance evaluation protocol and measures from Section 2.2 were used to evaluate tracking performance on VOT-LT2019. Compared to VOT-LT2018, a significant change is the new dataset described in the following.

#### 3.3.1 The dataset

The VOT-LT2019 trackers were evaluated on the LTB50 [67], which is an extension of the LTB35 [67] used in VOT-LT2018. LTB35 contains 35 sequences, carefully selected to obtain a dataset with long sequences containing many target disappearances. The LTB50 dataset contains 50 challenging sequences of diverse objects (persons, car, motorcycles, bicycles, boat, animals, etc.) with the total length of 215294 frames. Sequence resolutions range between $1280 \times 720$ and $290 \times 217$. Each sequence contains on average 10 long-term target disappearances, each lasting on average 52 frames. An overview of the dataset is shown in Figure 2. For additional information, please see [67].

The targets are annotated by axis-aligned bounding boxes. Sequences are annotated by the following visual attributes: (i) Full occlusion, (ii) Out-of-view, (iii) Partial occlusion, (iv) Camera motion, (v) Fast motion, (vi) Scale change, (vii) Aspect ratio change, (viii) Viewpoint change, (ix) Similar objects. Note this is per-sequence, not per-frame annotation and a sequence can be annotated by several attributes.

#### 3.3.2 Winner identification protocol

The VOT-LT2019 winner was identified as follows. Trackers were ranked according to the tracking F-score on the LTB50 dataset (no sequestered dataset available). The top

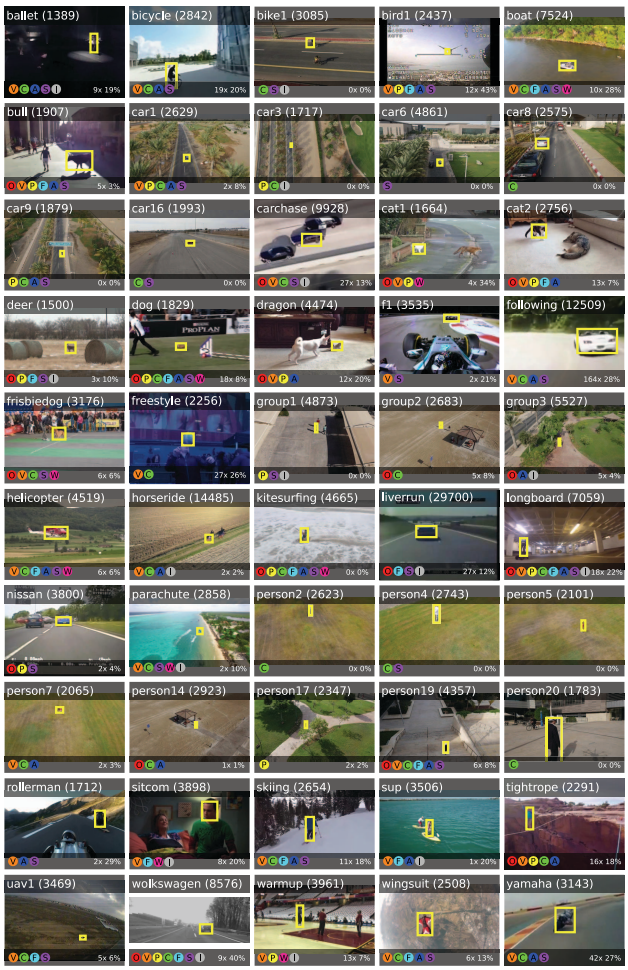


Figure 2. The LTB50 dataset – a frame selected from each sequence. Name and length (top), number of disappearances and percentage of frames without target (bottom right). Visual attributes (bottom left): (O) Full occlusion, (V) Out-of-view, (P) Partial occlusion, (C) Camera motion, (F) Fast motion, (S) Scale change, (A) Aspect ratio change, (W) Viewpoint change, (I) Similar objects. The dataset is highly diverse in attributes, target types and contains many target disappearances. Image reprinted with permission from [68].

ranked tracker on the dataset not submitted by the VOT2019 committee members was the winner of the VOT-LT2019 challenge.

## 3.4. VOT-RGBT2019 challenge outline

This challenge addressed short-term trackers using RGB and a thermal channel. The performance evaluation protocol and measures outlined in Section 2.1 were applied. In the following the details of the dataset and the winner identification protocols are provided.

### 3.4.1 The dataset

The community-driven move from the pure thermal infrared challenge VOT-TIR to VOT-RGBT (see section 1.3) requires a completely new dataset. The VOT committee decided to base the VOT-RGBT-challenge on the existing RGBT-dataset published by [60]. This dataset contains in total 234 sequences with an average length of 335 frames and all sequences have been clustered in the 11-dimensional global attribute space according to the VOT sequence clustering protocol [50]. From these clusters, 60 sequences for each dataset, the public dataset and the sequestered dataset, have been sampled. All frames in the two selected sets of sequences have been annotated with the attributes (i) occlusion, (iii) motion change, (iv) size change, (v) camera motion. The attribute (ii), illumination change, has not been used in the VOT-RGBT dataset, due to too scarce occurrences.

The original dataset contains axis-aligned annotations, but in order to achieve a higher significance of results, also the RGBT-dataset has been annotated with rotated bounding boxes. Similar to the ST-dataset, the annotation process has been performed in two steps. First, segmentation masks have been generated semi-automatically based on the video object segmentation method [41]. Manually generated segmentation masks in the respectively first and last frame of the RGB and the thermal stream are used as starting points for video object segmentation. Whenever the propagated segmentation mask disagreed significantly, additional manually generated masks have been added. In about 10% of the sequences, objects were too tiny so that segmentation masks have been generated manually or the original axis-aligned bounding boxes have been used. The rotated bounding boxes are generated from the masks using the approach proposed in [94]. The original axis-aligned bounding boxes have been used to reduce drift during the automatic procedure. Further details are given in [5]. An example for the new annotations is given in Figure 3, left.

Figure 3. Examples of images from the VOT-RGBT2019 dataset including annotations. Left: thermal and RGB image (frame 324 from sequence GREEN) illustrating the original axis aligned annotation (red), the automatically generated segmentation mask (white), and the final rotated bounding box (green). Right: thermal and RGB image (frame 5 from sequence MAN-WITHBASKETBALL) illustrating the inconsistent bounding boxes from the thermal channel (red) and the RGB channel (green), resulting from inaccurate synchronization.

VOT-RGBT is, besides VOT-RGBD, the only multi-

modal tracking challenge in VOT2019. Multi-modal tracking adds two difficulties compared to all other challenges: a) since the different sensors cannot be placed at the same physical location, the image registration will never be perfect. Thus, bounding boxes in the two modalities will never align exactly. b) since the two sensors are in separate devices and thus synchronized by software, frames with the same index might be subject to fixed or even varying relative delays. Also this will lead to inconsistent bounding boxes, see figure 3, right. Methods not considering a) and b) properly, will suffer from degraded performance due to reduced EAO in the RGB or T modality. For addressing the synchronization issue, we defined the thermal channel as the primary modality, so all ground truth is temporally aligned with it and the RGB channel is considered an auxiliary modality. Another consequence of the inconsistencies a) and b) is the upper bound of performance for multi-modal tracking that is reduced. In case of the VOT-RGBT challenge, the EAO between the two annotations is about $0.75$, thus no method can achieve a higher EAO.

#### 3.4.2 Winner identification protocol

The VOT-RGBT2019 winner has been identified as follows. Trackers were ranked according to the EAO measure on the public VOT-RGBT2019 dataset. The top five trackers have then been re-run by the VOT2019 committee on the sequestered VOT-RGBT dataset. The top ranked tracker on the sequestered dataset not submitted by the VOT2019 committee members was the winner of the VOT-RGBT2019 challenge.

### 3.5. VOT-RGBD2019 challenge outline

This challenge addressed long-term trackers using RGB and a depth channel (D). Evaluation of the long-term RGBD trackers was the same as for the long-term RGB trackers and therefore the long-term tracking evaluation protocol and measures from Section 2.2 were used.

#### 3.5.1 The dataset

The VOT-RGBD2019 trackers were evaluated on a new *Color and Depth Visual Object Tracking Dataset and Benchmark* (CDTB) [65]. CDTB contains 80 sequences acquired with three different setups: 1) a Kinect v2 RGBD sensor, 2) a pair of Time-of-Flight (Basler tof640) and an RGB camera (Basler acA1920), and 3) a stereo-pair (Basler acA1920). Kinect was used in 12 indoor sequences, RGB-ToF pair in 58 indoor sequences and the stereo-pair in 10 outdoor sequences. For all sequences CDTB provides RGB frames and dense depth frames which are aligned. The alignment is based on stereo pair calibration using the Cal-

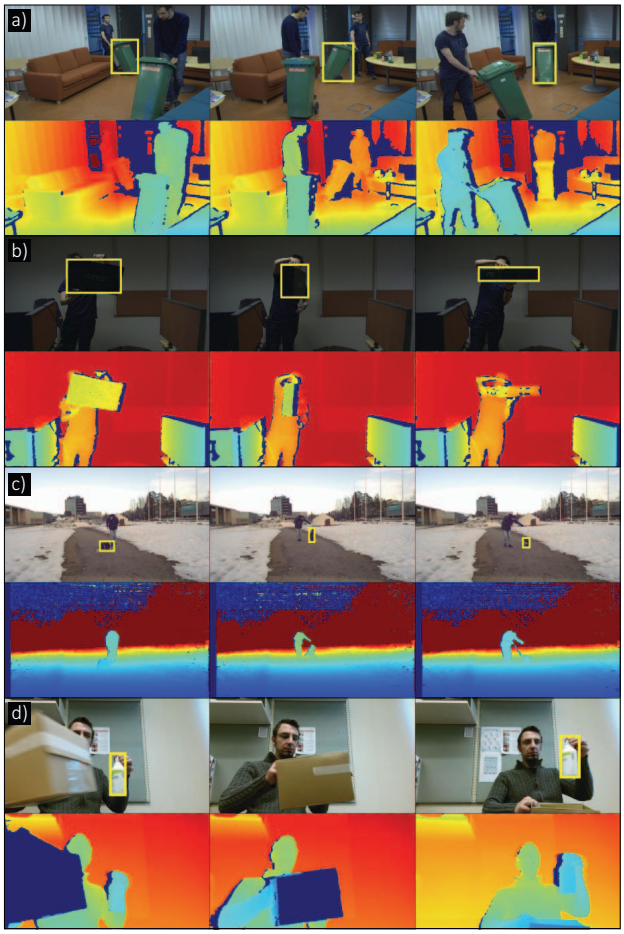


Figure 4. RGB and depth frames from the CDTB dataset [65] that contains eighty sequences captured outdoors by a stereo pair or indoors by a ToF-RGB pair or a Kinect sensor. Image reprinted with permission from [65].

Tech camera calibration toolbox[10] and the missing depth values are added by interpolation. The dataset contains tracking of various household and office objects (Figure 4). The sequences contain in-depth rotations, occlusions and disappearances that are challenging for RGB and, in particular, depth based trackers. The total number of frames is 101,956 in various resolutions. For more details, see [65].

#### 3.5.2 Winner identification protocol

The VOT-RGBD2019 winner was identified as follows. Trackers were ranked according to the F-score on the public VOT-RGBD2019 dataset (no sequestered dataset available). The top ranked tracker not submitted by the VOT2019 committee members was the winner of the VOT-RGBD2019 challenge.

[10] http://www.vision.caltech.edu/bouguetj/calib_doc

## 4. The VOT2019 challenge results

This section summarizes the trackers submitted, results analysis and winner identification for each of the five VOT2019 challenges.

### 4.1. The VOT-ST2019 challenge results

#### 4.1.1 Trackers submitted

In all, 46 valid entries were submitted to the VOT-ST2019 challenge. Each submission included the binaries or source code that allowed verification of the results if required. The VOT2019 committee and associates additionally contributed 11 baseline trackers. For these, the default parameters were selected, or, when not available, were set to reasonable values. Thus in total 57 trackers were tested on VOT-ST2019. In the following we briefly overview the entries and provide the references to original papers in the Appendix A where available.

Of all participating trackers, 37 trackers (65%) were categorized as $ST_0$, 19 trackers (33%) as $ST_1$ and 1 as $LT_1$. 79% applied discriminative and 21% applied generative models. Most trackers (84%) used holistic model, while 16% of the participating trackers used part-based models. Most trackers applied either a locally uniform dynamic model[11] (83%), a random walk dynamic model (12%) or a nearly-constant-velocity dynamic model (5%).

The trackers were based on various tracking principles: 3 (5%) trackers were based on recurrent neural network (A3CTD A.1, MemDTC A.28, ROAMpp A.34), 21 (37%) trackers applied Siamese networks (ALTO A.3, ARTCS A.5, Cola A.10, gasiamrpn A.19, iourpn A.21, MPAT A.30, RSiamFC A.35, SA-SIAM-R A.36, SiamCRF A.37, SiamCRF-RT A.38, SiamDW-ST A.39, Siamfcos A.40, SiamFCOSP A.41, SiamFCOT A.42, SiamMargin A.43, SiamMask A.44, SiamMsST A.45, SiamRPNpp A.46, SiamRPNX A.47, SPM A.48, TADT A.52), 24 trackers (42%) applied CNN or classical discriminative correlation filters (ACNT A.2, ANT A.4, ARTCS A.5, ATOM A.7, ATP A.8, CISRDCF A.9, Cola A.10, CSRDCF A.11, CSRpp A.12, DCFST A.13, DiMP A.14, DPT A.15, DRNet A.16, FSC2F A.18, KCF A.23, LSRDFT A.26, M2C2F A.27, SSRCCOT A.49, STN A.51, TCLCF A.53, TDE A.54, Trackyou A.55, UInet A.56, WSCFST A.57), 2 trackers (4%) applied ranking-based classifier learning (RankingR A.32, RankingT A.33), 4 trackers (7%) were based on classical discriminative and generative subspaces (IVT A.22, L1APG A.24, MIL A.29, Struck A.50), 2 trackers (4%) applied histogram-based similarity maximization (ASMS A.6, PBTS A.31), 3 trackers (5%) applied optical flow (ANT A.4, FoT A.17, LGT A.25), and 1 tracker was based on a combination of multiple basic classical trackers (HMMTxDT A.20).

Many trackers used combinations of several features. CNN features were used in 69% of trackers – these were either trained for discrimination (26 trackers) or localization (13 trackers). Hand-crafted features were used in 25% of trackers, keypoints in 4% of trackers, color histograms in 18% and grayscale features were used in 16% of trackers.

#### 4.1.2 Results

The results are summarized in the AR-raw plots and EAO curves in Figure 5 and the expected average overlap plots in Figure 6. The values are also reported in Table 2. The top ten trackers according to the primary EAO measure (Figure 6) are DRNet A.16, Trackyou A.55, ATP A.8, DiMP A.14, Cola A.10, ACNT A.2, SiamMargin A.43, DCFST A.13, SiamFCOT A.42, and SiamCRF A.37.

All trackers apply CNN features for target localization. Seven top trackers (DRNet, Trackyou, ATP, DiMP, Cola, ACNT, DCFST) apply a discriminative correlation-filter (DCF) for target localization, followed by bounding box regression. Most of these trackers do not apply a classical DCF, but rather a CNN formulation of the discriminative correlation filter from ATOM [16] and the bounding box prediction inspired by [40]. Three trackers (SiamMargin, SiamFCOT and SiamCRF) apply a siamese correlation filter (i.e., template-based correlation) followed by a bounding box regression. It appears that, similarly to VOT2018, the top performers contain discriminative correlation filters, but the formulation has shifted to a learning strategy that involves Gauss-Newton updated implemented via back-prop computations in standard CNN toolboxes. A strong commonality is the bounding box prediction module [40], which appears to increase the tracking accuracy. The most popular backbone used appears to be the Resnet-family [36] (in particular Resnet50 and Resnet18) – most of the top trackers apply it. None of the top performers use hand-crafted features, which is a stark contrast to VOT2018.

The top performer on public dataset is DRNet (A.16). This tracker applies a two-stage tracking framework: target position estimation, followed by the bounding box regression. The localization stage is aimed at increased robustness at a cost of potentially reduced accuracy by applying a discriminative correlation filter (DCF). The DFC learning is formulated as a CNN layer. Robustness is increased by a distractor-aware loss. Target localization is implemented by DCFs on multiple branches of the backbone networks, followed by a bounding box regression branch [40]. The backbone are ResNet50 and ResNetSE pre-trained on ImageNet [22], while the bounding box regression head is trained on LaSOT [24], TrackingNet [74] and COCO [63]

[11] The target was sought in a window centered at its estimated position in the previous frame. This is the simplest dynamic model that assumes all positions within a search region contain the target have equal prior probability.

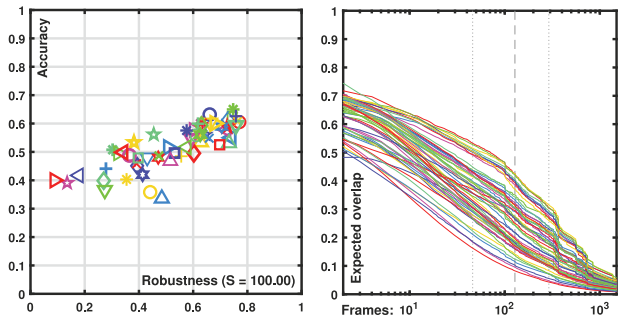


Figure 5. The VOT-ST2019 AR-raw plots generated by sequence pooling (left) and EAO curves (right).

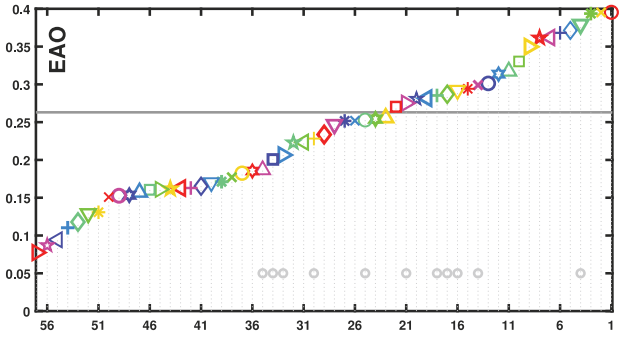


Figure 6. The VOT-ST2019 expected average overlap graph with trackers ranked from right to left. The right-most tracker is the top-performing according to the VOT-ST2019 expected average overlap values. The dashed horizontal line denotes the average performance of ten state-of-the-art trackers published in 2018 and 2019 at major computer vision venues. These trackers are denoted by gray circle in the bottom part of the graph.

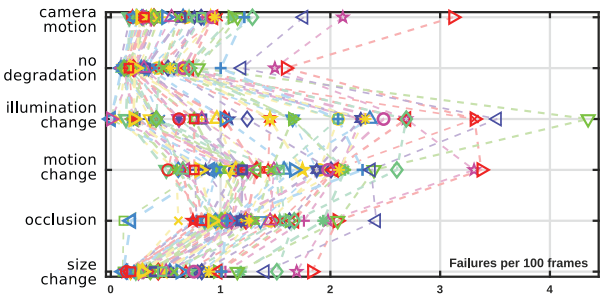


Figure 7. Failure rate with respect to the visual attributes.

datasets.

The second-best ranked tracker is Trackyou (A.55). This tracker is an extension of ATOM [16] by a dynamic optimization approach. A triplet loss is introduced to learn more discriminative features during offline training. Online updates are modulated by fuzzy combination of several target properties estimated during tracking.

The third top-performing tracker (ATP A.8) is also based on ATOM [16] for target localization, but followed by SiamMask [96] for target segmentation. A bounding box is fitted to the segmentation result. Feature pyramid network is used for fusing low-to-high-level features and an adaptive search window size is used. Hyperparameter optimization was applied to tune the performance.

The top-three trackers stand out slightly from the rest in EAO measure. These three trackers have very high accuracy, with ATP obtaining the highest among all 57 trackers. On the other hand, DRNet and Trackyou result in most robust tracking with the least number of failures.

While the top-ten trackers share many architectural similarities, we observe that these are shared among other trackers which are ranked significantly lower. One reason for this is that implementation plays a very important part in performance. Another reason is that training procedures seem to play an important role as well. From the tracker descriptions it is clear that different trackers apply slightly different sampling strategies in training as well as different datasets, arriving at features with varying discrimination properties and contain slight tracking architectural differences that might lead to significant performance differences.

The trackers which have been considered as baselines or state-of-the-art four years ago are positioned at the lower part of the AR-plots and at the tail of the EAO rank list. This speaks of the significant quality of the trackers submitted to VOT-ST2019. In fact, 11 tested trackers (19%) have been recently (2018/2019) published in major computer vision conferences and journals. These trackers are indicated in Figure 6, along with their average performance (EAO= 0.263), which constitutes the VOT2019 state-of-the-art bound. Approximately 49% of submitted trackers exceed this bound, which speaks of significant pace of advancements made in tracking within a span of only a few years.

| | CM | IC | MC | OC | SC |
|---|---|---|---|---|---|
| Accuracy | 0.53 | 0.48 ③ | 0.51 | 0.44 ① | 0.48 ② |
| Robustness | 0.63 | 1.18 ③ | 1.44 ① | 1.20 ② | 0.56 |

Table 1. VOT-ST2019 tracking difficulty with respect to the following visual attributes: camera motion (CM), illumination change (IC), motion change (MC), occlusion (OC) and size change (SC).

The number of failures with respect to the visual attributes is shown in Figure 7. The overall top performers remain at the top of per-attribute ranks as well, but none of the trackers consistently outperforms all others with respect to each attribute.

According to the median robustness and accuracy over each attribute (Table 1) the most challenging attributes in terms of failures are occlusion and motion change. Illumination change, motion change and scale change are challenging, but comparatively much better addressed by the submitted trackers. Tracking accuracy is most strongly affected by motion change and camera motion.

#### 4.1.3 The VOT-ST2019 challenge winner

Top five trackers from the baseline experiment (Table 2) were re-run on the sequestered dataset. Their scores obtained on sequestered dataset are shown in Table 3. The top tracker according to the EAO is ATP A.8 and is thus the VOT-ST2019 challenge winner.

### 4.2. The VOT-RT2019 challenge results

#### 4.2.1 Trackers submitted

The trackers that entered the VOT-ST2019 challenge were also run on the VOT-RT2019 challenge. Thus the statistics of submitted trackers was the same as in VOT-ST2019. For details please see Section 4.1.1 and Appendix A.

#### 4.2.2 Results

The EAO scores and AR-raw plots for the real-time experiments are shown in Figure 8, Figure 9 and Table 2. The top ten real-time trackers are SiamMargin A.43, SiamFCOT A.42, DiMP A.14, DCFST A.13, SiamDWST A.39, ARTCS A.5, SiamMask A.44, SiamRPNpp A.46, SPM A.48, SiamCRF-RT A.38.

Seven of the top ten realtime trackers (SiamMargin, SiamFCOT, SiamDWST, SiamMask, SiamRPNpp, SPM and SiamCRF-RT) are based on siamese correlation combined with bounding box regression. Most of these apply region-proposal-like bounding box prediction (e.g., akin to [59, 87]). SiamFCOT and SiamMask apply target segmentation for improved localization. Three of the top ten realtime trackers (DiMP, DCFST and ARTCS) apply a discriminative correlation filter embedded within a CNN. In particular, DCFST and ARTCS apply the formulation from [16], while DiMP applies an end-to-end trainable architecture for predicting and tuning the filter.

The top performer, SiamMargin, is derived from SiamRPNpp [58]. The features are trained offline in a standard siamese framework, but with added discriminative loss to encourage learning of a discrimintative embedding. During tracking, ROI-align is applied to extract features from

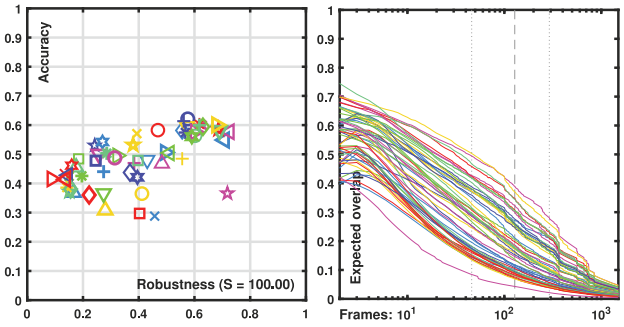

Figure 8. The VOT-RT2019 AR plot (left) and the EAO curves (right).

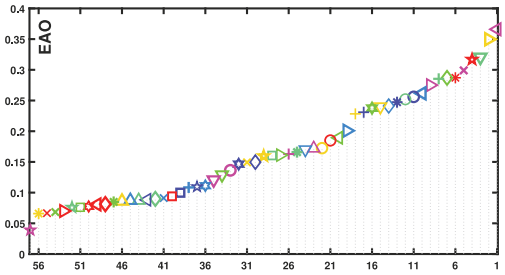

Figure 9. The VOT-RT2019 EAO plot (right).

the estimated target region. The template is updated by these features using a moving average.

SiamMargin is closely followed in EAO by SiamFCOT, which is in principle a classical Siamese correlation network running on Alexnet backbone. The estimated target position is further refined by a FCOS detector bounding box predictor [87] and the search region size is adapted during tracking with respect to the estimated target localization certainty. This tracker applies a Unet-like segmentation network to predict a binary mask in the estimated target region – the final bounding box is fitted to this mask. Figure 9 indicates that SiamMargin and SiamFCOT evidently stand out from the rest with respect to the EAO measure, indicating a clear state-of-the-art.

#### 4.2.3 The VOT-RT2019 challenge winner

According to the EAO results in Table 2, the top performer and the winner of the real-time tracking challenge VOT-RT2019 is SiamMargin (A.43).

### 4.3. The VOT-LT2019 challenge results

#### 4.3.1 Trackers submitted

The VOT-LT2019 challenge received 8 valid entries. The VOT2019 committee contributed an additional baseline, thus 9 trackers were considered in the challenge. In the following we briefly overview the entries and provide the references to original papers in Appendix B where avail-

| | | baseline | | | realtime | | | unsupervised | |
|---|---|---|---|---|---|---|---|---|---|
| | **Tracker** | **EAO** | **A** | **R** | **EAO** | **A** | **R** | **AO** | **Implementation** |
| 1. | DRNet | 0.395 ① | 0.605 | 0.261 ① | 0.185 | 0.583 | 0.757 | 0.511 ③ | D P G |
| 2. | Trackyou | 0.395 ② | 0.609 | 0.270 ② | 0.149 | 0.571 | 0.933 | 0.477 | S P G |
| 3. | ATP | 0.394 ③ | 0.650 ① | 0.291 | 0.085 | 0.426 | 1.630 | 0.502 | S P G |
| 4. | DiMP | 0.379 | 0.594 | 0.278 ③ | 0.321 ③ | 0.582 | 0.371 | 0.508 | S P C |
| 5. | Cola | 0.371 | 0.613 | 0.316 | 0.241 | 0.582 | 0.587 | 0.508 | S P G |
| 6. | ACNT | 0.368 | 0.626 ③ | 0.278 ③ | 0.231 | 0.613 ② | 0.577 | 0.513 ② | S P G |
| 7. | SiamMargin | 0.362 | 0.578 | 0.326 | 0.366 ① | 0.577 | 0.321 ① | 0.411 | D P G |
| 8. | DCFST | 0.361 | 0.589 | 0.321 | 0.317 | 0.585 | 0.376 | 0.446 | S P G |
| 9. | SiamFCOT | 0.350 | 0.601 | 0.386 | 0.350 ② | 0.601 | 0.386 | 0.470 | D P G |
| 10. | SiamCRF | 0.330 | 0.625 | 0.296 | 0.076 | 0.484 | 1.690 | 0.557 ① | S P G |
| 11. | LSRDFT | 0.317 | 0.531 | 0.312 | 0.087 | 0.455 | 1.741 | 0.413 | S P G |
| 12. | STN | 0.314 | 0.589 | 0.349 | 0.111 | 0.542 | 1.309 | 0.481 | S P G |
| 13. | MPAT | 0.301 | 0.632 ② | 0.414 | 0.256 | 0.621 ① | 0.552 | 0.448 | S P G |
| 14. | SiamDW_ST | 0.299 | 0.600 | 0.467 | 0.299 | 0.600 | 0.467 | 0.452 | S M G |
| 15. | ARTCS | 0.294 | 0.602 | 0.456 | 0.287 | 0.602 ③ | 0.482 | 0.466 | D P G |
| 16. | ATOM | 0.292 | 0.603 | 0.411 | 0.240 | 0.596 | 0.557 | 0.493 | S P G |
| 17. | SiamMask | 0.287 | 0.594 | 0.461 | 0.287 | 0.594 | 0.461 | 0.415 | D P G |
| 18. | SiamRPNpp | 0.285 | 0.599 | 0.482 | 0.285 | 0.599 | 0.482 | 0.482 | D P G |
| 19. | SiamCRF_RT | 0.282 | 0.550 | 0.301 | 0.262 | 0.549 | 0.346 ③ | 0.474 | S P G |
| 20. | ROAMpp | 0.281 | 0.561 | 0.438 | 0.110 | 0.530 | 1.420 | 0.416 | S P G |
| 21. | SPM | 0.275 | 0.577 | 0.507 | 0.275 | 0.577 | 0.507 | 0.449 | D P C |
| 22. | RankingT | 0.270 | 0.525 | 0.360 | 0.094 | 0.297 | 0.908 | 0.400 | S M G |
| 23. | TDE | 0.256 | 0.534 | 0.465 | 0.086 | 0.308 | 1.274 | 0.382 | S M C |
| 24. | UInet | 0.254 | 0.561 | 0.468 | 0.238 | 0.560 | 0.527 | 0.417 | S P G |
| 25. | SA_SIAM_R | 0.253 | 0.559 | 0.492 | 0.252 | 0.563 | 0.507 | 0.392 | D P G |
| 26. | RankingR | 0.252 | 0.548 | 0.417 | 0.091 | 0.288 | 0.783 | 0.435 | S M G |
| 27. | SiamMsST | 0.252 | 0.575 | 0.552 | 0.247 | 0.574 | 0.567 | 0.424 | D P G |
| 28. | gasiamrpn | 0.247 | 0.548 | 0.522 | 0.121 | 0.501 | 1.359 | 0.418 | D P G |
| 29. | SSRCCOT | 0.234 | 0.495 | 0.507 | 0.081 | 0.360 | 1.505 | 0.380 | S M C |
| 30. | MemDTC | 0.228 | 0.485 | 0.587 | 0.228 | 0.485 | 0.587 | 0.376 | D P G |
| 31. | SiamRPNX | 0.224 | 0.517 | 0.552 | 0.189 | 0.504 | 0.672 | 0.363 | D P G |
| 32. | Siamfcos | 0.223 | 0.561 | 0.788 | 0.076 | 0.372 | 1.891 | 0.378 | S P G |
| 33. | TADT | 0.207 | 0.516 | 0.677 | 0.201 | 0.506 | 0.702 | 0.386 | S P G |
| 34. | CSRDCF | 0.201 | 0.496 | 0.632 | 0.100 | 0.478 | 1.405 | 0.320 | D C C |
| 35. | CSRpp | 0.187 | 0.468 | 0.662 | 0.172 | 0.468 | 0.727 | 0.321 | D C C |
| 36. | FSC2F | 0.185 | 0.480 | 0.752 | 0.077 | 0.461 | 1.836 | 0.367 | S M C |
| 37. | ALTO | 0.182 | 0.358 | 0.818 | 0.172 | 0.365 | 0.888 | 0.252 | D P G |
| 38. | M2C2F | 0.177 | 0.486 | 0.747 | 0.068 | 0.424 | 1.896 | 0.349 | S M C |
| 39. | SiamFCOSP | 0.171 | 0.508 | 1.194 | 0.166 | 0.503 | 1.254 | 0.241 | S P G |
| 40. | TCLCF | 0.170 | 0.480 | 0.843 | 0.170 | 0.480 | 0.843 | 0.338 | D M C |
| 41. | A3CTD | 0.165 | 0.451 | 0.933 | 0.150 | 0.437 | 0.998 | 0.271 | D P G |
| 42. | RSiamFC | 0.163 | 0.470 | 0.958 | 0.163 | 0.470 | 0.958 | 0.285 | D P G |
| 43. | HMMTxD | 0.163 | 0.499 | 1.073 | 0.081 | 0.414 | 1.981 | 0.347 | D C C |
| 44. | WSCF_St | 0.162 | 0.534 | 0.963 | 0.160 | 0.532 | 0.968 | 0.332 | D M C |
| 45. | iourpn | 0.161 | 0.495 | 1.129 | 0.161 | 0.495 | 1.129 | 0.265 | S P G |
| 46. | ASMS* | 0.160 | 0.479 | 0.923 | 0.160 | 0.479 | 0.923 | 0.289 | D C C |
| 47. | PBTS | 0.157 | 0.336 | 0.725 | 0.087 | 0.368 | 1.796 | 0.217 | S P C |
| 48. | CISRDCF | 0.153 | 0.420 | 0.883 | 0.146 | 0.421 | 0.928 | 0.242 | D M C |
| 49. | DPT | 0.153 | 0.488 | 1.008 | 0.136 | 0.488 | 1.159 | 0.289 | D C C |
| 50. | ANT* | 0.151 | 0.458 | 0.938 | 0.067 | 0.434 | 2.017 | 0.239 | D M C |
| 51. | LGT* | 0.131 | 0.403 | 1.038 | 0.066 | 0.386 | 1.951 | 0.206 | S M C |
| 52. | FoT | 0.129 | 0.366 | 1.294 | 0.129 | 0.366 | 1.294 | 0.135 | D C C |
| 53. | MIL* | 0.118 | 0.398 | 1.309 | 0.090 | 0.380 | 1.861 | 0.166 | D C C |
| 54. | KCF | 0.110 | 0.441 | 1.279 | 0.108 | 0.440 | 1.294 | 0.206 | D C C |
| 55. | Struck* | 0.094 | 0.417 | 1.726 | 0.088 | 0.428 | 1.926 | 0.174 | D C C |
| 56. | IVT* | 0.087 | 0.391 | 2.002 | 0.039 | 0.366 | 0.331 ② | 0.110 | D M C |
| 57. | L1APG* | 0.077 | 0.400 | 2.420 | 0.070 | 0.415 | 2.428 | 0.122 | S M C |

Table 2. The table shows expected average overlap (EAO), as well as accuracy and robustness raw values (A,R) for the baseline and the realtime experiments. For the unsupervised experiment the no-reset average overlap AO [97] is used. The last column contains implementation details (first letter: (D)eterministic or (S)tohastic, second letter: tracker implemented in (M)atlab, (C)++, or (P)ython, third letter: tracker is using (G)PU or only (C)PU). A dash "-" indicates that the realtime experiment was performed using an outdated version of the toolkit and that the results are invalid.

| | Tracker | EAO | A | R |
|---|---|---|---|---|
| 1. | ATP | 0.2747 ① | 0.6692 ① | 0.4046 ② |
| 2. | DiMP | 0.2489 ② | 0.6110 | 0.3896 ① |
| 3. | DRNet | 0.2371 ③ | 0.6437 ② | 0.4465 ③ |
| 4. | Cola | 0.2218 | 0.2080 | 0.5133 |
| 5. | Trackyou | 0.2035 | 0.6358 ③ | 0.5301 |

Table 3. The top five trackers from Table 2 re-ranked on the VOT-ST2019 sequestered dataset.

able.

All participating trackers were categorized as $LT_1$ according to the ST-LT taxonomy from Section 1.4 in that they explicitly implemented explicit target re-detection. Eight out of nine trackers were based on CNNs. Six of these (CLGS B.2, CooSiam B.3, LT-DSE B.5, SiamDW-LT B.7, Siamfcos-LT B.8, SiamRPNsLT B.9) applied Siamese matching architectures akin to SiamFc [7] and SiamRpn [58]. ASINT B.1 applied CNN-based template matching trained in a siamese setup and mbdet B.6 applied online trained CNN classifier for target localization. One tracker was based purely on discriminative correlation filters on top of hand-crafted features (FuCoLoT B.4).

Four trackers (ASINT, FuCoLoT, LT-DSE, mbdet) updated the long-term visual model only when confident, Siamfcos-LT and SiamDW-LT applied a constant exponential forgetting, CLGS and SiamRPNsLT never updated the model and CooSiam applied mixed temporal updating akin to [70] with multiple visual model.

#### 4.3.2 Results

The overall performance is summarized in Figure 10. Three trackers stand out from the rest: LT-DSE, CLGS and SiamDW-LT. These trackers follow a short-term tracker long-term detector interaction approach [70], but differ in architectural details. LT-DSE applies a DCF [16] short-term tracker on top of extended ResNet18 features for initial target localization. The target position is refined by a SiamMask [96] run on the target initial position. The target presence is then verified by RT-MDNet [42]. If the target is deemed absent, an image-wide re-detection using a region proposal network MBMD [108] is applied. The region proposals are verified by the online trained verifier. CLGS applies a siamese short-term tracker for between-frame tracking. When the target is deemed absent, an RCNN region proposal is activated for generating potential target candidates, which are subsequently verified by a model similar to MDNet [42]. SiamDW-LT applies a deep-and-wide backbone from [109] to construct a short-term frame-to-frame tracker and a global re-detection module is activated whenever the confidence of the short-term tracker drops significantly. A model ensemble is applied to further improve the tracking accuracy and robustness.

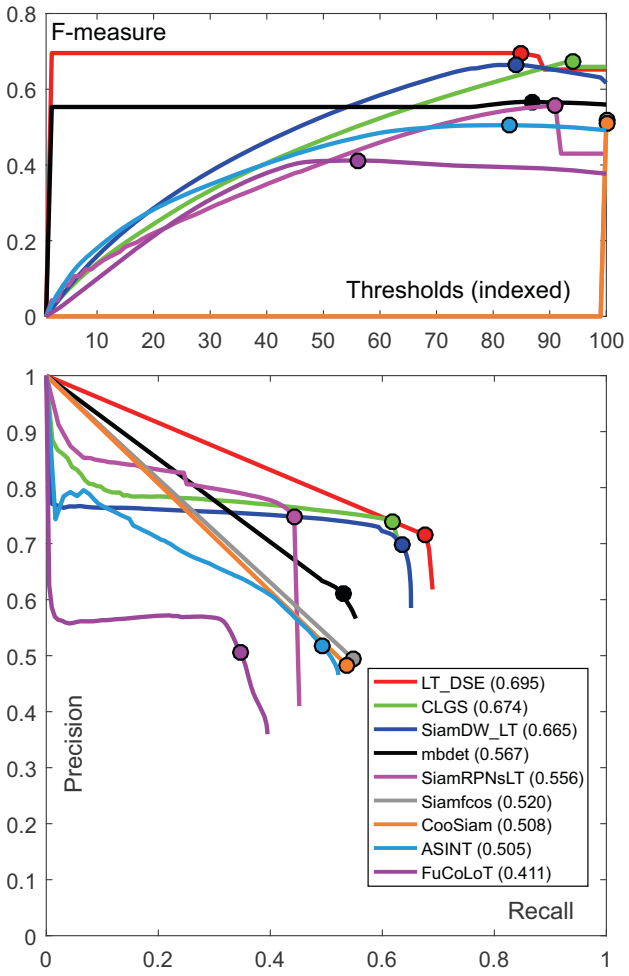


Figure 10. VOT-LT2019 challenge average tracking precision-recall curves (left), the corresponding F-score curves (right). Tracker labels are sorted according to maximum of the F-score.

LT-DSE achieves the best tracking F-score and best tracking Recall. This means that it recovers much more correct target positions that the other trackers (see Figure 10). This comes at a cost of slightly reduced tracking Precision. Performance of the baseline tracker FuCoLoT is the lowest across all attributes, which is likely due to the fact that this is the only tracker that applies hand-crafted features.

Figure 11 shows tracking performance with respect to nine visual attributes from Section 3.3. The most challenging attributes are out of view, viewpoint change, similar objects and partial occlusion. Performance across the attributes appears to be fairly stable for all trackers except SiamRPNsLT, whose performance significantly drops on the viewpoint change attribute.

| | Tracker | F-score | Pr | Re | ST/LT |
|---|---|---|---|---|---|
| 1. | LT_DSE | 0.695 ① | 0.715 ③ | 0.677 ① | $LT_1$ |
| 2. | CLGS | 0.674 ② | 0.739 ② | 0.619 ③ | $LT_1$ |
| 3. | SiamDW_LT | 0.665 ③ | 0.697 | 0.636 ② | $LT_1$ |
| 4. | mbdet | 0.567 | 0.609 | 0.530 | $LT_1$ |
| 5. | SiamRPNsLT | 0.556 | 0.749 ① | 0.443 | $LT_1$ |
| 6. | Siamfcos-LT | 0.520 | 0.493 | 0.549 | $LT_1$ |
| 7. | CooSiam | 0.508 | 0.482 | 0.537 | $LT_1$ |
| 8. | ASINT | 0.505 | 0.517 | 0.494 | $LT_1$ |
| 9. | FuCoLoT | 0.411 | 0.507 | 0.346 | $LT_1$ |

Table 4. List of trackers that participated in the VOT-LT2019 challenge along with their performance scores (F-score, Pr, Re) and ST/LT categorization.

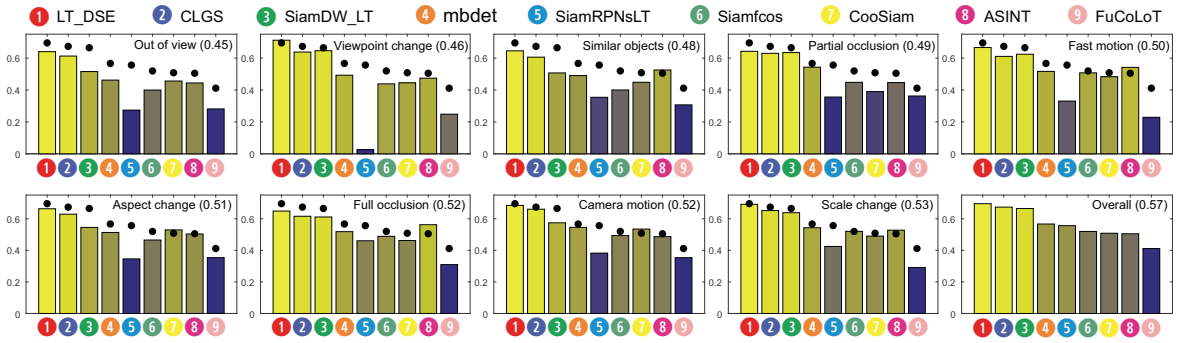


Figure 11. VOT-LT2019 challenge maximum F-score averaged over overlap thresholds for the visual attributes. The most challenging attributes are fast motion, out of view, aspect ratio change and full occlusion.

#### 4.3.3 The VOT-LT2019 challenge winner

According to the F-score, LT-DSE is well ahead of the rest of the trackers, it also achieves top tracking Recall score and is third-best tracker in tracking Precision score. Thus, according to the VOT2019 rules, LT-DSE B.5 is the winner of the VOT-LT2019 challenge.

### 4.4. The VOT-RGBT2019 challenge results

#### 4.4.1 Trackers submitted

In all, 10 entries were submitted to the VOT-RGBT2019 challenge. All but one submission included the binaries or source code that allowed verification of the results if required. One submission was an earlier version of another. No additional trackers were contributed by the VOT committee. Thus in total 8 valid trackers were tested on VOT-RGBT2019. In what follows we briefly overview the entries and provide the references to original papers in the Appendix A where available.

All participating trackers use discriminative models with a holistic representation. 5 trackers (62.5%) were categorized as $ST_1$ and 3 trackers (37.5%) as $ST_0$. 7 trackers (87.5%) applied a locally uniform dynamic model and 1 tracker (12.5%) a random walk dynamic model.

The trackers were based on various tracking principles: 4 trackers (50%) are based on discriminative correlation filters (CISRDCF C.1, GESBTT C.3, JMMAC C.4, and mfDiMP C.6), 4 trackers (50%) are based on multiple CNNs (MANet C.5, mfDiMP C.6, MPAT C.7, and SiamDW_T C.8), 4 trackers (50%) make use of Siamese CNNs (FSRPN C.2, mfDiMP C.6, MPAT C.7, and SiamDW_T C.8), 2 trackers (25%) apply a Kalman filter (GESBTT C.3 and JMMAC C.4), and respectively 1 tracker (12.5%) makes use of optical flow (GESBTT C.3) and ransac (JMMAC C.4).

5 trackers (62.5%) used combinations of several features. 6 trackers (75%) used CNN features and 3 trackers (37.5%) used hand-crafted features. Respectively 2 trackers (25%) used keypoints and grayscale features.

#### 4.4.2 Results

The results are summarized in the AR-raw plots and EAO curves in Figure 12 and the expected average overlap plots in Figure 13. The values are also reported in Table 5. The top five trackers according to the primary EAO measure (Figure 13) are JMMAC C.4, SiamDW_T C.8, mfDiMP C.6, FSRPN C.2, and MANet C.5.

All trackers apply CNN features for target localization. This is in contrast to the earlier VOT-TIR challenges, where

| | Tracker | EAO | A | R |
|---|---|---|---|---|
| 1. | JMMAC | 0.4826 ① | 0.6649 ① | 0.8211 ① |
| 2. | SiamDW_T | 0.3925 ② | 0.6158 | 0.7839 ③ |
| 3. | mfDiMP | 0.3879 ③ | 0.6019 | 0.8036 ② |
| 4. | FSRPN | 0.3553 | 0.6362 ② | 0.7069 |
| 5. | MANet | 0.3463 | 0.5823 | 0.7010 |
| 6. | MPAT | 0.3180 | 0.5723 | 0.7242 |
| 7. | CISRDCF | 0.2923 | 0.5215 | 0.6904 |
| 8. | gesbtt | 0.2896 | 0.6163 ③ | 0.6350 |

Table 5. Numerical results of VOT-RGBT2019 challenge on the public dataset.

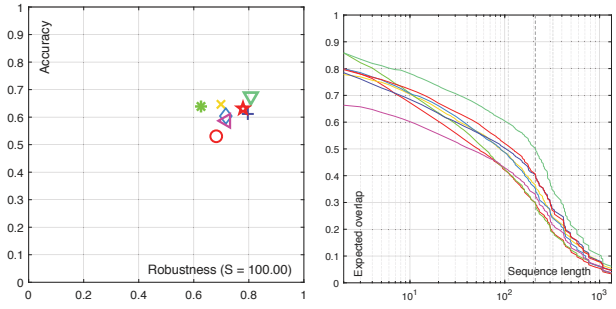


Figure 12. The VOT-RGBT2019 AR plot (left) and the EAO curves (right). The legend is given in figure 13.

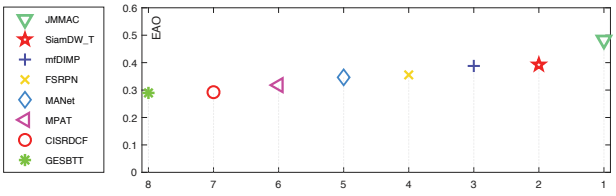


Figure 13. The VOT-RGBT2019 EAO plot.

| | CM | MC | OC | SC |
|---|---|---|---|---|
| Accuracy | 0.61 | 0.61 | 0.44 | 0.62 |
| Robustness | 0.87 | 0.62 | 0.79 | 0.90 |

Table 6. VOT-RGBT2019 tracking difficulty with respect to the following visual attributes: camera motion (CM), motion change (MC), occlusion (OC), and size change (SC).

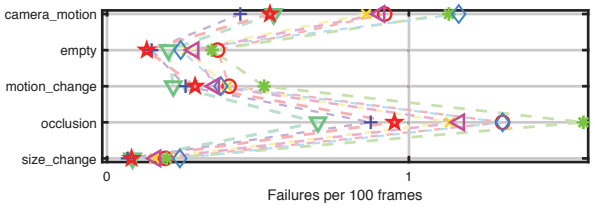


Figure 14. Failure rate with respect to the visual attributes. The legend is given in figure 13.

hand-crafted features still dominated [25, 26, 49]. Respectively 3 out of 5 trackers apply discriminative correlation-filters (DCF), multiple CNNs, and Siamese CNNs. Most trackers are combinations of these methods with few exceptions: JMMAC is only using DCFs, but still performing best on the public dataset. This is remarkable as the other two trackers solely relying on DCF are ranked lowest. Here, the use of RANSAC seem to lift the JMMAC performance. The use of a Kalman filter approach in JMMAC and gesbtt shows varying outcome, performing both strongest and weakest. Another exception is FSRPN, solely relying on a Siamese CNN, but without a significant performance difference compared to combinations of methods.

The top performer on the public dataset is JMMAC (C.4). This tracker applies a two-component approach, combining motion and appearance cues. The motion cue is inferred from key-point based camera motion estimation and a Kalman filter applied to object motion. The appearance cues are generated by an extension of the ECO model [15].

The second-best ranked tracker is SiamDW_T (C.8). This tracker is method-wise a complement to JMMAC and applies multiple CNNs and Siamese CNNs.

The third top-performing position is taken by mfDiMP (C.6). mfDiMP combines all approaches named in the paragraphs above. It is a multi-modal extension of the Discriminative Model Prediction (DiMP) tracker [8].

Only the top-ranked tracker stands out from the rest in EAO measure, otherwise the results are quite similar. Similar to earlier challenges, the EAO correlates stronger with robustness than with accuracy: The ranks 1-3 on robustness and EAO are shared among the three trackers above. Conversely, rank 2 and 3 for accuracy are ranked 4th and 8th in EAO.

Since this has been the first RGBT-challenge within VOT and due to the small number of participants, we have not introduced a state-of-the-art bound as for the VOT-ST challenge. However, similar to VOT-ST, we analyzed the number of failures with respect to the visual attributes (excluding illumination change), see Figure 14. The overall top performers remain at the top of per-attribute ranks as well, and JMMAC consistently outperforms all others with respect to each attribute.

According to the median robustness and accuracy over each attribute (Table 6) the most challenging attributes in terms of failures are occlusion and motion change. Occlusion strongly affects accuracy whereas motion change affects mostly robustness. Scale change and camera motion are significantly less challenging attributes.

#### 4.4.3 The VOT-RGBT2019 challenge winner

Top five trackers from the baseline experiment (Table 5) were selected to be re-run on the sequestered dataset. Their scores obtained on sequestered dataset are shown in Table 7.

The top tracker according to the EAO is mfDiMP (C.6) and is thus the VOT-RGBT2019 challenge winner.

| | Tracker | EAO | A | R |
|---|---|---|---|---|
| 1. | mfDiMP | 0.2347 ① | 0.6133 | 0.3160 ① |
| 2. | SiamDW_T | 0.2143 ② | 0.6515 ② | 0.2714 ② |
| 3. | MANet | 0.2041 ③ | 0.5784 | 0.2592 ③ |
| 4. | JMMAC | 0.2037 | 0.6337 ③ | 0.2441 |
| 5. | FSRPN | 0.1873 | 0.6561 ① | 0.1755 |

Table 7. Numerical results of VOT-RGBT2019 challenge on the sequestered dataset.

## 4.5. The VOT-RGBD2019 challenge results

### 4.5.1 Trackers submitted

The VOT-RGBD2019 challenge received 4 valid entries: ATCAIS (D.1), LTDSEd (D.2), SiamDW-D (D.3), SiamM_Ds (D.4). The VOT2019 committee contributed additional 8 baselines: MDNet [75], MBMD [108], FuCoLoT (B.4) [70], OTR [45], SiamFC [7], CSRDCF-D [44], ECO [15] and CADMS [64]; thus 12 trackers were considered in the challenge. In the following we briefly overview the entries and provide the references to the original papers in Appendix D where available.

Two of the baseline trackers were RGBD trackers, OTR [45] and CSRDCF-D [44], while the remaining six baseline trackers were well-performing long-term RGB trackers (MDNet, MBMD, ECO, FuCoLot, SiamFC and CADMS [64]) which omitted the depth channel. The main reason to include pure RGB trackers was to evaluate the additional value of the depth channel for tracking.

Three of the RGB-only baseline trackers (MDNet, MBMD and FuCoLoT) outperformed the best baseline RGBD tracker (OTR), but all four valid entries outperformed all baseline trackers. SiamDW-D (D.3) is a variant of the recent long-term Siamese network tracker [109]. ATCAIS (D.1) is based on the ATOM tracker [16] which implements a discriminative correlation filter loss for deep matching. ATCAIS also adopts Chen et al. [12] deep architecture for instance segmentation. LTDSEd (D.3) uses two different trackers (inc. ATOM) and visibility of the target is judged by the outputs of the both. The SiamM_Ds (D.4) tracker is a modified version of SiamMask [96]. The four entries do not define special processing of the depth channel beyond using it as an additional feature dimension. All trackers are based on deep features.

### 4.5.2 Results

The overall performances are summarized in Figure 15 and Table 8. The highest ranked tracker is the Siamese network based tracker *SiamDW-D*. Variants of the same tracker were ranked 14th in VOT-ST2019 (SiamDW-ST), 3rd in VOT-LT2019 (SiamDW-LT) and 3rd in VOT-RGBT2019 (SiamDW-T).

| | Tracker | F-score | Pr | Re | ST/LT |
|---|---|---|---|---|---|
| 1. | SiamDW_D | 0.681 ① | 0.677 ① | 0.685 ② | $LT_1$ |
| 2. | ATCAIS | 0.676 ② | 0.643 ③ | 0.712 ① | $LT_1$ |
| 3. | LTDSEd | 0.658 ③ | 0.674 ② | 0.643 ③ | $LT_1$ |
| 4. | SiamM_Ds | 0.455 | 0.516 | 0.406 | $LT_1$ |
| 5. | MDNet | 0.455 | 0.463 | 0.447 | $ST_1$ |
| 6. | MBMD | 0.441 | 0.454 | 0.429 | $LT_1$ |
| 7. | FuCoLoT | 0.391 | 0.459 | 0.340 | $LT_1$ |
| 8. | OTR | 0.336 | 0.364 | 0.312 | $LT_1$ |
| 9. | SiamFC | 0.333 | 0.356 | 0.312 | $ST_1$ |
| 10. | CSRDCF-D | 0.332 | 0.375 | 0.297 | $ST_0$ |
| 11. | ECO | 0.329 | 0.317 | 0.342 | $ST_1$ |
| 12. | CADMS | 0.271 | 0.284 | 0.259 | $LT_0$ |

Table 8. List of trackers that participated in the VOT-RGBD2019 challenge along with their performance scores (F-score, Pr, Re) and ST/LT categorization.

The second highest ranked tracker is *ATCAIS*. ATCAIS is based on the ATOM tracker [16] and the HTC method for instance segmentation [12]. ATCAIS does not particularly handle target loss (see the two last attributes in Figure 16).

The third tracker, LTDSEd, is from the same authors as ATCAIS. It contains two tracker components, one using ATOM for tracking with Wang et al. [96] method for foreground segmentation, and another using RT-MDNet [42]. The variant of this method, LT-DSE, won the VOT-LT2019 long-term tracking challenge.

The strength of the three best trackers is likely in their occlusion recovery handling as their other variants performed well in the long-term tracks of VOT2019. It is unclear how extensively these methods exploit the depth channel besides using it as an additional feature channel. The three best trackers behave similarly for all annotated attributes (Figure 16), except SiamDW-D which was particularly good on "Similar objects" and ATCAIS that completely failed on "Full occlusion" and "Out-of-frame". All three best trackers were distinctly better than the rest of the evaluated RGBD trackers making them good seeds for future work on long-term RGBD tracking.

### 4.5.3 The VOT-RGBD2019 challenge winner

It should be noted that there are only minor differences among the three best RGBD trackers. They all achieve the maximum F-measure near the same Precision-Recall region $Pr, Re \in [0.64, 0.71]$ (Figure 15). Their performances are almost comparable also within the different attributes (Figure 16) (except ATCAIS that does not handle target disappearance).

The winner is selected based on the best F-score and is SiamDW-D (F-score 0.681). For the winning F-score SiamDW-D also obtains the best precision (0.677) and the

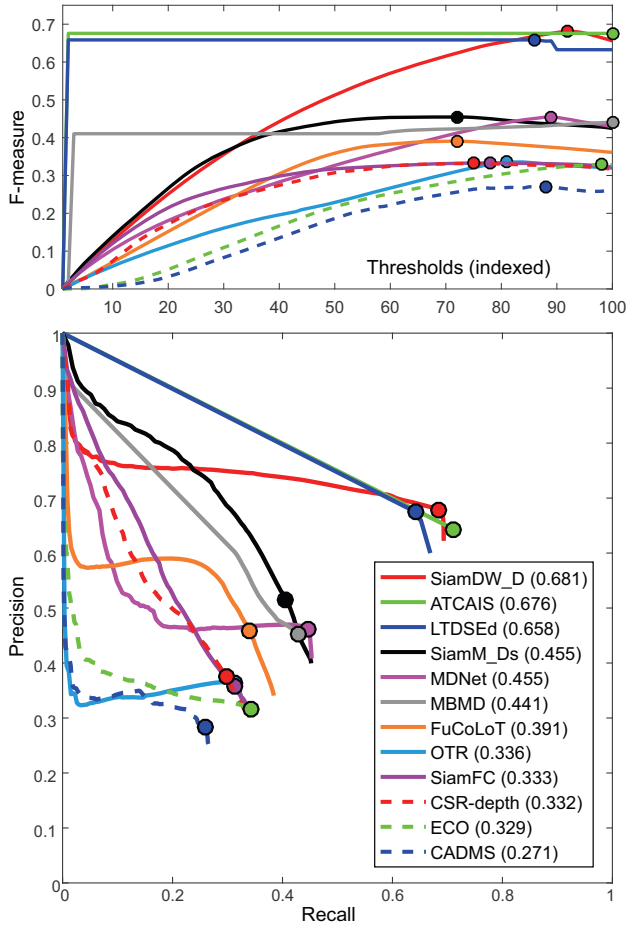


Figure 15. VOT-RGBD2019 challenge average tracking precision-recall curves (bottom), the corresponding F-score curves (top). Tracker labels are sorted according to maximum of the F-score.

second best recall (0.685) which indicate its very strong performance. Interestingly, SiamDW-D also uses ATOM-type detection refinement, but does not utilize any foreground segmentation network such as ATCAIS and LTDSEd do. According to the VOT winner rules, the VOT-RGBD2019 challenge winner is therefore SiamDW-D (D.3).

## 5. Conclusion

Results of the VOT2019 challenge were presented. The challenge is composed of the following five challenges focusing on various tracking aspects and domains: (i) the VOT2019 short-term RGB tracking challenge (VOT-ST2019), (ii) the VOT2019 short-term real-time RGB tracking challenge (VOT-RT2019), (iii) the VOT2019 long-term RGB tracking challenge (VOT-LT2019), (iv) the VOT2019 short-term RGB and thermal tracking challenge (VOT-RGBT2019) and (v) the VOT2019 long-term RGB and depth (D) tracking challenge (VOT-RGBD2019).

The overall results of the challenges indicate that top performers on VOT-ST and VOT-RGBD draw heavily on the recently proposed framework ATOM [16], which combines a robust DCF-like localizer with a IoUNet bounding box estimation. In contrast, the top performers of VOT-RT challenge are from the class of classical siamese correlation trackers [7] and siamese trackers with region proposals [58]. The VOT-LT challenge top performers apply the short-term localization and long-term re-detection tracker structure [70], but differ in design – the dominant methodologies are CNN DCF [16] and siamese correlation [7], region proposals and online trained CNN classifiers [75]. The top performers in VOT-RGBT challenge apply classical [15] or CNN-based [8] DCFs or siamese [7] approaches.

The top performer on the VOT-ST2019 *public dataset* is DRNet A.16, which is based on the recent CNN-based DCF [16] and adds a distractor-aware loss to increase the robustness. The top performer on the *sequestered dataset* and the VOT-ST2019 challenge winner is ATP A.8. This tracker combines the recent ATOM [16] and SiamMask [96] trackers, applies an improved bounding box estimation algorithm and feature pyramids for extraction of rich features.

The top performer and the winner of the VOT-RT2019 challenge is SiamMargin A.43, which is based on SiamRpn++ [58] with added discriminative loss in feature pre-training stage.

The top performer and the winner of the VOT-LT2019 challenge isLT-DSE B.5, wich combines a CNN-based DCF [16] with a siamese segmentation tracker [96] and an fast version of an online trained CNN classifier [75].

The top performer on the VOT-RGBT2019 *public dataset* is JMMAC (C.4), an approach that combines DCF-based appearance cues with motion cues derived from keypoint and a Kalman filter. The top performer on the sequestered dataset and the VOT-RGBT2019 challenge winner is mfDiMP (C.6), an end-to-end tracking framework for fusing the RGB and TIR modalities based on the DiMP (Discriminative Model Prediction) tracker [8].

The top performer and the winner of the VOT-RGBD2019 challenge is SiamDW-D (D.3) – a variant of the recent Siamese network tracker [109]. The tracker consists of two parts, a region proposal network and a proposal refinement network. The proposal refinement network is the same as in ATOM [16]. Since also the second and third best RGBD trackers were both based on ATOM and their variants performed well in the VOT-LT2019 challenge the challenge results indicate that good failure recovery and the ATOM network structures make a strong combination for RGBD tracking. However, none of these trackers particularly utilized the depth information which raises the question whether depth needs special attention at all - or do deep tracker architectures learn to exploit depth from training data.

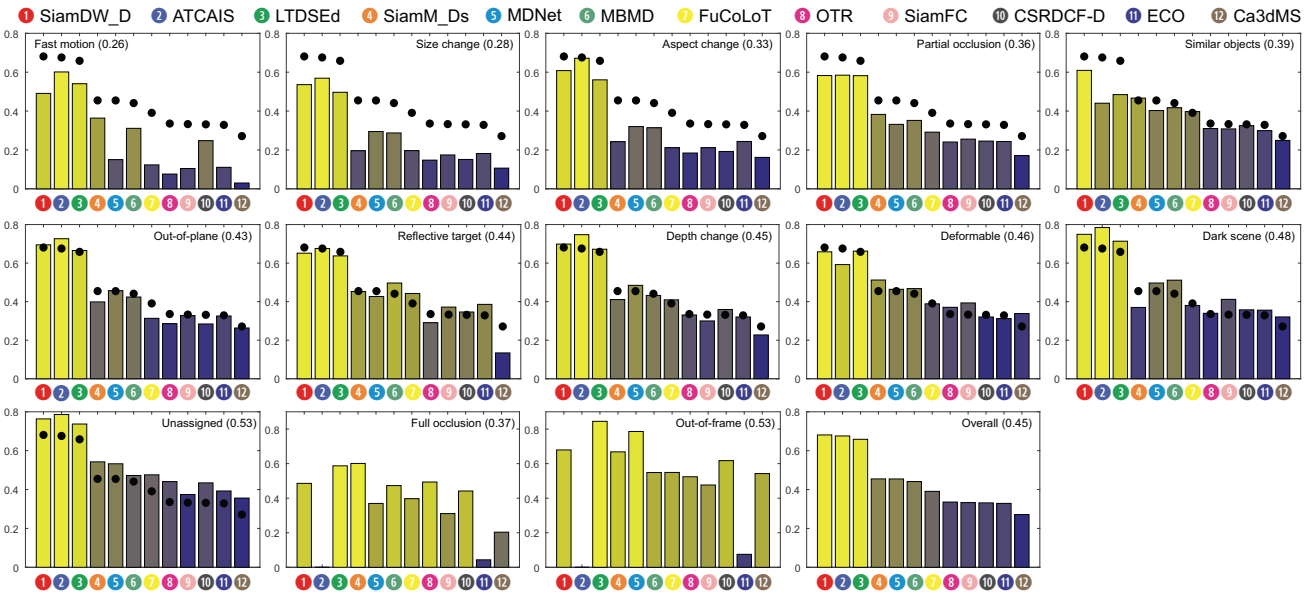


Figure 16. VOT-RGBD2019 challenge: tracking performance w.r.t. visual attributes. The first eleven attributes correspond to scenarios with a visible target (showing F-measure). The overall tracking performance is shown in each graph with black dots. The attributes full occlusion and out of view represent periods when the target is not visible and true negative rate is used to measure the performance.

In all years that we have been using sequestered dataset for winner identification, we have consistently observed that the ranks of the top performers change compared to the results on the public dataset. What is more, the performance consistently drops for all trackers on the sequestered dataset by a non-negligible amount. This supports the argument that the state-of-the-art should not be decided by forcing top rank on datasets, especially if they are public (which is the case for *all existing benchmarks* apart from VOT challenge). This has been vocalized by VOT for several years now, and that a more appropriate state-of-the-art identification approach is to consider a reasonably high sota-bound. Trackers well exceeding this bound surely exhibit state-of-the-art performance.

The VOT primary objective is to establish a platform for discussion of tracking performance evaluation and contributing to the tracking community with verified annotated datasets, performance measures and evaluation toolkits. The VOT2019 was a sevents effort toward this, following the very successful VOT2013, VOT2014, VOT2015, VOT2016, VOT2017 and VOT2018. Similarly to the previous challenges, the VOT2019 made several steps beyond, opening new challenges on new tracking domains. Our future work will follow this line of advancements.

## Acknowledgements

This work was supported in part by the following research programs and projects: Slovenian research agency research programs P2-0214, P2-0094, Slovenian research agency project J2-8175. Jiři Matas and Ondrej Drbohlav were supported by the Czech Science Foundation Project GACR P103/12/G084. Aleš Leonardis was supported by MURI project financed by MoD/Dstl and EPSRC through EP/N019415/1 grant. Michael Felsberg, Amanda Berg, and Abdelrahman Eldesokey were supported by WASP, VR (ELLIIT, LÄST, and NCNN), and SSF (SymbiCloud). Roman Pflugfelder and Gustavo Fernández were supported by the AIT Strategic Research Programme 2019 Visual Surveillance and Insight. The challenge was sponsored by the Faculty of Computer Science, University of Ljubljana, Slovenia.

## A. VOT-ST2019 and VOT-RT2019 submissions

This appendix provides a short summary of trackers considered in the VOT-ST2019 and VOT-RT2019 challenges.

### A.1. Visual Tracking by means of Deep Reinforcement Learning and an Expert Demonstrator (A3CTD)

*M. Dunnhofer, N. Martinel, C. Micheloni*
*dunnhofer.matteo@spes.uniud.it,*
*{niki.martinel, christian.micheloni}@uniud.it*

A3CTD is a novel real-time tracker built on a deep recurrent regression network architecture. It is trained offline using a reinforcement learning based framework that takes advantage of the demonstrations of an expert tracker. After training, the proposed tracker is capable of producing

bounding box estimates through the learned policy or by exploiting the demonstrator. Through the learned state value function, A3CTD is in fact able to evaluate the quality of its current tracking policy and of the expert's one, and to consequently decide if to output its own bounding box estimate or the one proposed by the demonstrator.

### A.2. Adaptive Correction Network based tracker (ACNT)

*T. Xu, Z.-H. Feng, S.-C. Zhao, X.-J. Wu, J. Kittler*
*tianyang_xu@163.com, z.feng@surrey.ac.uk,*
*zsc960813@163.com, wu_xiaojun@jiangnan.edu.cn,*
*j.kittler@surrey.ac.uk*

In the Adaptive Correction Network based Correlation Filter tracker a correlation filter is employed to predict the centre location while an IoU net is established to perform adaptive correction. We modified the loss function in IoU net of ATOM tracker [16] to jointly consider the bounding box overlap and centre location error.

### A.3. Adversarial Learning for Tracking Objects (ALTO)

*N. Paluru, B. Pedasingu, L. Rout, R. Gorthi*
*ee17ms004@iittp.ac.in, surajpedasingu@gmail.com,*
*lr@sac.isro.gov.in, rkg@iittp.ac.in*

The tracker ALTO is a novel framework based on adversarial learning to enhance the predictions given by a generative tracker, by leveraging the powers of a discriminative classifier and a regressor for effective tracking.

### A.4. ANT (ANT)

*Submitted by VOT Committee*

The ANT tracker is a conceptual increment to the idea of multi-layer appearance representation that is first described in [90]. The tracker addresses the problem of self-supervised estimation of a large number of parameters by introducing controlled graduation in estimation of the free parameters. The appearance of the object is decomposed into several sub-models, each describing the target at a different level of detail. The sub models interact during target localization and, depending on the visual uncertainty, serve for cross-sub-model supervised updating. The reader is referred to [92] for details.

### A.5. Accurate and Robust Tracking based on Correlation filter and SiamRPN (ARTCS)

*B. Yan, H. Zhao, D. Wang, H. Lu, X. Yang*
*yan_bin@mail.dlut.edu.cn, zhaohj1995@gmail.com,*
*{wdice, lhchuan}@dlut.edu.cn,*
*xiaoyun.yang@intellicloud.ai*

The tracker ARTCS consists of a robust CorrelationFilter-based localization module [16] and an accurate SiamRPN-based estimation module [58]. During the tracking process, ATOM robustly locates the target's rough position and a search region is cropped around it. In a second step, SiamRPN++ predicts the accurate position and the size of the target.

### A.6. Scale Adaptive Mean-Shift Tracker (ASMS)

*Submitted by VOT Committee*

The mean-shift tracker optimizes the Hellinger distance between template histogram and target candidate in the image. This optimization is done by a gradient descend. ASMS [95] addresses the problem of scale adaptation and presents a novel theoretically justified scale estimation mechanism which relies solely on the mean-shift procedure for the Hellinger distance. ASMS also introduces two improvements of the mean-shift tracker that make the scale estimation more robust in the presence of background clutter – a novel histogram colour weighting and a forward-backward consistency check. Code available at https://github.com/vojirt/asms.

### A.7. ATOM: Accurate Tracking by Overlap Maximization (ATOM)

*G. Bhat, M. Danelljan, F. Khan, M. Felsberg*
*{goutam.bhat, martin.danelljan}@vision.ee.ethz.ch,*
*fahad.khan@inceptioniai.org, michael.felsberg@liu.se*

ATOM separates the tracking problem into two subtasks: i) target classification, where the aim is to robustly distinguish the target from the background; and ii) target estimation, where an accurate bounding box for the target is determined. Target classification is performed by training a discriminative classifier online. Target estimation is performed by an overlap maximization approach where a network module is trained offline to predict the overlap between the target object and a bounding box estimate, conditioned on the target appearance in first frame. See [16] for more details.

### A.8. ATP: Accurate Tracking by Progressively refining (ATP)

*B. Li, D. Song, L. Wang, X. Tang, C. Zhang, Y. Liu, Z. Ni,*
*S. Li, K. Wang, Y. Zhou, X. Bai, W. Liu, B. He, J. Liu*
*{libi, djsong, makalo}@hust.edu.cn,*
*{tangxu02, zhangchengquan}@baidu.com,*
*987752424@qq.com,*
*{nizihan, lishihu, wangkangkang}@baidu.com,*
*{yuzhou, xbai, liuwy}@hust.edu.cn,*
*{hebin04, liujingtuo}@baidu.com*

We improve the tracking accuracy by progressively refining the target estimation. The process consists of three stages. At the first stage, we adopt the ATOM tracker [16] to estimate the axis-aligned target position. At the second stage, we crop out an image patch centred on the estimated

target and feed it into a segmentation network. Specifically, SiamMask [96] is used. At the third stage, a rotated bounding box is generated from the segmentation.

### A.9. Channel Independent Spatially Regularized Discriminative Correlation Filter Tracker (CISRDCF)

*A. Varfolomieiev*
*a.varfolomieiev@kpi.ua*

The method is based on SRDCF formulation [17], which defers from the original one in two main points: 1) it calculates the filter channels for each feature channel independently, and 2) the regularization in the filter is performed iteratively using the ADMM approach [29]. To suppress the information outside the object's bounding-box, a rectangular regularization window with slightly blurred edges is applied. The method uses the HOG features augmented with additional channel, which represents the backprojection of object histogram. This channel is equivalent to the per-pixel scores used in histogram-related part of Staple tracker [6]. The current version of the tracker extracts the object histogram from grayscale images and thus does not employ any colour information.

### A.10. Online Update Tracking Model for Discriminant Feature Learning (Cola)

*C. Chen, Q. Zhang*
*755062190@qq.com, labyrinth7x@gmail.com*

Our tracker is based on ATOM [16]. In order to enable the network to learn more discriminant features and eliminate the deviation of information during supervised learning, we use the features extracted from the ground truth in the test branch to modulate the features extracted from the proposals. In order to better combine the features in the reference and the test branches, we limit the interval between these two frames in the video sequence with a maximum gap of 15 frames. The implementation proves that our network can learn more discriminant features after such optimization processing, and get better results on OTB and VOT dataset.

### A.11. Discriminative Correlation Filter with Channel and Spatial Reliability (CSRDCF)

*Submitted by VOT Committee*

The CSRDCF [69] improves discriminative correlation filter trackers by introducing two concepts: spatial reliability and channel reliability. It uses colour segmentation as spatial reliability to adjust the filter support to the part of the object suitable for tracking. The channel reliability reflects the discriminative power of each filter channel. The tracker uses HoG and colour-names features.

### A.12. Discriminative Correlation Filter with Channel and Spatial Reliability - C++ (CSRpp)

*Submitted by VOT Committee*

The CSRpp tracker is the C++ implementation of the Discriminative Correlation Filter with Channel and Spatial Reliability (CSR-DCF) tracker A.11.

### A.13. Learning Features with Differentiable Closed-Form Solvers for Tracking (DCFST)

*L. Zheng, M. Tang, J. Wang*
*{linyu.zheng, tangm, jqwang}@nlpr.ia.ac.cn*

The tracker DCFST focuses on learning feature embeddings in an end-to-end way. DCFST employs ResNet-18 as its backbone and takes both images training and test as its input. Sample RoIs with target size in each image are obtained by uniform sampling and their feature maps are obtained by using PrPool [40] layer. We train a ridge regression model to fit the samples in the training image employing the trained model to predict the regression values of samples in the test image. Shrinkage loss function is employed to calculate the error between the predicted values and the labels of the test samples. In the online inference we locate the target with the location of the maximum response value in the search region. Finally, the target bounding box is refined by applying ATOM [16] algorithm.

### A.14. Learning Discriminative Model Prediction for Tracking (DiMP)

*G. Bhat, M. Danelljan, L. Van Gool, R. Timofte*
*{goutam.bhat, martin.danelljan, vangool, timofter}@vision.ee.ethz.ch*

DiMP is an end-to-end trainable tracking architecture, capable of fully exploiting both target and background appearance information for target model prediction. The architecture is derived from a discriminative learning loss by designing a dedicated optimization process that is capable of predicting a powerful model in only a few iterations. Furthermore, key aspects of the discriminative loss are themselves learned during offline training. See [8] for more details.

### A.15. Deformable part correlation filter tracker (DPT)

*Submitted by VOT Committee*

DPT is a part-based correlation filter composed of a coarse and mid-level target representations. Coarse representation is responsible for approximate target localization and uses HOG as well as colour features. The mid-level representation is a deformable parts correlation filter with fully-connected parts topology and applies a novel formulation that threats geometric and visual properties within a single convex optimization function. The mid level as well

as coarse level representations are based on the kernelized correlation filter from [37]. The reader is referred to [66] for details.

### A.16. High Accuracy Visual Tracking with Deep Regression Networks (DRNet)

*S. Bai, J. Zhuang, Y. Dong, H. Bai*
*baishuai@bupt.edu.cn, junfei.zhuang@faceall.cn,*
*yuandong@bupt.edu.cn, hongliang.bai@faceall.cn*

DRNet tracker consists of two phases, namely box selection and scale regression. The first phase predicts the location of the target, learning target-specific information by online updating a discriminative correlation filters (DCF) module [16]. A distractor-aware loss is designed for online learning by adaptively penalizing the interference peaks. During this phase, the fusion of ResNet50 and SE-ResNet50 with two independent DCF module branches, which use backbone features from the Block4 of ResNet50 and SE-ResNet5 as input, is introduced. The second phase uses the position predicted in the first phase and the size of the previous frame as the proposal box, extracts features using PrRoIPooling [40] and estimates the scale applying box regression. The box regression takes backbone features from the Block3 and Block4 of ResNet50 as input. The backbone network uses the pre-trained model of ImageNet [22] and the network parameters are fixed. In the offline training process, the scale regression sub-network is trained to predict the offset between the proposal box and the target with Large-scale Single Object Tracking (LaSOT) [24], TrackingNet [74] and COCO [63] datasets. During the online updating, the Conjugate Gradient is applied to update the DCF module.

### A.17. Flock of Trackers (FoT)

*Submitted by VOT Committee*

The Flock of Trackers (FoT) is a tracking framework where the object motion is estimated from the displacements or, more generally, transformation estimates of a number of local trackers covering the object. Each local tracker is attached to a certain area specified in the object coordinate frame. The local trackers are not robust and assume that the tracked area is visible in all images and that it undergoes a simple motion, e.g. translation. The FoT object motion estimate is robust if it is from local tracker motions by a combination which is insensitive to failures.

### A.18. Fast Saliency-guided Continuous Correlation Filter-based tracker (FSC2F)

*A. Memarmoghadam*
*a.memarmoghadam@yahoo.com*

FSC2F further enhances the robustness of the efficient ECOhc [15] by adaptively applying the motion-aware saliency map [32] on the contaminated confidence map. Moreover, to maintain computational complexity in a reasonable range for real-time tracking, the FSC2F tracker employs a faster scale estimation technique that improves the baseline fDSST [18] via jointly learning of the sparsely-sampled scale-spaces.

### A.19. SiamRPN with adaptive anchors and proposals (gasiamrpn)

*X. Li, J. Li, C. Ma, Z. He, M.-H. Yang*
*xinlihitsz@gmail.com, lijing@stu.hit.edu.cn,*
*chaoma@sjtu.edu.cn, zhenyuhe@hit.edu.cn,*
*mhyang@ucmerced.edu*

The gasiamrpn tracker aims to adapt the tracking model to changes of object states. The tracker exploits the sequential state information within the Siamese tracking framework and infers the target state. It adaptively generates anchor for the RPN model instead of using pre-defined anchors with fixed parameters. The adaptive anchors and proposals contribute to accurate bounding box regression and robust classification of the RPN model. The final target states are estimated as a Bayesian inference model constructed on top of a Siamese-based state prediction model.

### A.20. Online Adaptive Hidden Markov Model for Multi-Tracker Fusion (HMMTxD)

*Submitted by VOT Committee*

The HMMTxD method fuses observations from complementary out-of-the box trackers and a detector by utilizing a hidden Markov model whose latent states correspond to a binary vector expressing the failure of individual trackers. The Markov model is trained in an unsupervised way, relying on an online learned detector to provide a source of tracker-independent information for a modified Baum-Welch algorithm that updates the model w.r.t. the partially annotated data.

### A.21. SiamRPN with giou loss (iourpn)

*J. Li, Z. Teng, B. Zhang*
*{18125219, zteng, bpzhang}@bjtu.edu.cn*

The tracker iourpn is an end-to-end tracker based on SiamRPN [59], but aims to improve the accuracy of object localization beyond 50%. Furthermore, it can simultaneously suppress the background clutters and distractors. The proposed tracker brings the Generalized Intersection over Union (GIOU) [79] constraint into the tracking network, which guides the network with more accurate bounding box predictions.

### A.22. Incremental Learning for Robust Visual Tracking (IVT)

*Submitted by VOT Committee*

The idea of the IVT tracker [81] is to incrementally learn a low-dimensional sub-space representation, adapting on-

line to changes in the appearance of the target. The model update, based on incremental algorithms for principal component analysis, includes two features: a method for correctly updating the sample mean, and a forgetting factor to ensure less modelling power is expended fitting older observations.

### A.23. Kernelized Correlation Filter (KCF)

*Submitted by VOT Committee*

This tracker is a C++ implementation of Kernelized Correlation Filter [37] operating on simple HOG features and Colour Names. The KCF tracker is equivalent to a Kernel Ridge Regression trained with thousands of sample patches around the object at different translations. It implements multi-thread multi-scale support, sub-cell peak estimation and replacing the model update by linear interpolation with a more robust update scheme. Code available at https://github.com/vojirt/kcf.

### A.24. L1APG (L1APG)

*Submitted by VOT Committee*

L1-APG [2] considers tracking as a sparse approximation problem in a particle filter framework. To find the target in a new frame, each target candidate is sparsely represented in the space spanned by target templates and trivial templates. The candidate with the smallest projection error after solving an $\ell_1$ regularized least squares problem. The Bayesian state inference framework is used to propagate sample distributions over time.

### A.25. Local-Global Tracking tracker (LGT)

*Submitted by VOT Committee*

The core element of LGT is a coupled-layer visual model that combines the target global and local appearance by interlacing two layers. By this coupled constraint paradigm between the adaptation of the global and the local layer, a more robust tracking through significant appearance changes is achieved. The reader is referred to [90] for details.

### A.26. Learning Spatially Regularized correlation filters with Deep Features for Tracking (LSRDFT)

*X.-F. Zhu, X.-J. Wu, J. Kittler, T. Xu, H. Li, Y. Li*
*{xuefeng_zhu95, xiaojun_wu_jnu}@163.com,*
*j.kittler@surrey.ac.uk, tianyang_xu@163.com,*
*{lihui, 6171910026}@stu.jiangnan.edu.cn*

LSRDFT utilizes UPDT [9] as baseline, equipped with deep features from VGG16 and ResNet50, which are more robust and can be easily accelerated by GPU. In contrast with UPDT, the updating interval of the correlation filters is shortened in LSRDFT. For both VGG16 and ResNet50, augmentation is adopted using flip, rotation, blur and shift.

### A.27. Multi-Model Continuous Correlation Filter for visual tracking (M2C2F)

*A. Memarmoghadam*
*a.memarmoghadam@yahoo.com*

Inspired by ECO tracker [15], our efficient yet robust M2C2F tracker adaptively utilizes multiple representative models of the tracked object thereby estimating the object position every frame by weighted cumulative fusion of their respective regressors via a ridge regression optimization problem [71]. To further accelerate tracking performance, M2C2F enhances the baseline fDSST approach [18] by exploiting a faster scale estimation method in which the target scale filter is learned jointly via sparsely sampled scale-spaces. To suppress unwanted samples mostly belong to the occlusion or other non-object data, the M2C2F tracker conservatively updates every model on-the-fly in non-uniform time intervals.

### A.28. MemTrack with Distractor Template Canceling (MemDTC)

*T. Yang, A. Chan*
*tianyyang8-c@my.cityu.edu.hk, abchan@cityu.edu.hk*

This tracker extends MemTrack [101], which uses a dynamic memory network to maintain the appearance variations of the object as tracking proceeds, by introducing a Distractor Template Canceling (DTC) [102] scheme to cancel out wrong responses from the object template.

### A.29. Multiple Instance Learning tracker (MIL)

*Submitted by VOT Committee*

MIL tracker [1] uses a tracking-by-detection approach, more specifically Multiple Instance Learning instead of traditional supervised learning methods and shows improved robustness to inaccuracies of the tracker and to incorrectly labelled training samples.

### A.30. More precise box and accurate object tracking (MPAT)

*L. Zhou*
*lijun13548184189@outlook.com*

MPAT tracker is based on ATOM [16] and it uses three effective mechanisms. The first mechanism is to use the predicted target mask for more accurate positioning of the target position of the regression. In order to have have less background information and more accurate target information in the target, the scale of the target bounding box is reduced. The second mechanism consists of changing the method of data augmentation in the process of online learning. Finally, the scale update of the target sets the learning rate.

### A.31. Part-Based Tracking by Sampling (PBTS)

*G. De Ath, R. Everson*
*{g.de.ath, r.m.everson}@exeter.ac.uk*

PBTS [20] describes objects with a set of image patches represented by pairs of RGB pixel samples and counts of how many pixels in the patch are similar to them. This empirically characterises the underlying colour distribution of the patches and allows use of the Bhattacharyya distance. Candidate patch locations are generated by applying non-shearing affine transforms to the patches' previous locations. The best of these are locally optimised in a small region around each patch. PBTS uses an alpha matting-based patch initialisation technique [21] to place patches in regions of the bounding box that most likely contain the object.

### A.32. Ranking based tracking using CNNs and optical flow (RankingR)

*H. Saribas, H. Cevikalp*
*{hasansaribas48, hakan.cevikalp}@gmail.com*

The tracker RankingR uses a light weight deep neural network. The major novelty of the method is a novel ranking loss used by the network. We extract CNN features from both RGB and optical flow images. Ranking loss provides a fine-tuning of the target object position and returns more precise bounding boxes framing the target object. As a result, risk of tracking error accumulation and drifts are largely mitigated.

### A.33. Ranking based tracker using CNNs (RankingT)

*H. Cevikalp, H. Saribas*
*{hakan.cevikalp, hasansaribas48}@gmail.com*

This tracker uses a light weight deep neural network that uses a novel ranking loss especially designed for tracking. We extract CNN features from RGB images. The major novelty of the method is the proposed ranking loss. Ranking loss provides a fine-tuning of the target object position and returns more precise bounding boxes framing the target object. As a result, risk of tracking error accumulation and drifts are largely mitigated and the object is tracked with more successfully. This tracker differs from our other submitted tracker 'RankingR' A.32 in the way that it does not use optical flow images and it uses different cache models and heuristics for updating tracker models.

### A.34. ROAM++: Tracking via Resizable Response Generator and Bounding Box Regressor (ROAMpp)

*T. Yang, Y. Gu, T. Xing, Z. Song, B. Bai, P. Xu, A. Chan*
*tianyyang8-c@my.cityu.edu.hk, {guyangdavid, xingtengfei, songzhichao, baibing, xupengfeipf}@didiglobal.com, abchan@cityu.edu.hk*

This tracker extends ROAM (Recurrently Optimizing trAcking Model) [103] by introducing a resizeable bounding box regressor.

### A.35. Robust Siamese Fully Convolutional Tracker (RSiamFC)

*C. Fang*
*753317249@qq.com*

RSiamFC tracker is an extended SiamFC tracker [7] with a robust training method which puts a transformation on training sample to generate a pair of samples for feature extraction.

### A.36. SA-SIAM-R: A Twofold Siamese Network for Real-Time Object Tracking With Angle Estimation (SA-SIAM-R)

*A. He, C. Luo, X. Tian, W. Zeng*
*heanfeng@mail.ustc.edu.cn, cluo@microsoft.com, xinmei@ustc.edu.cn, wezeng@microsoft.com*

SA-SIAM-R is a variation of the Siamese network-based tracker SA-Siam [35]. SA-SIAM-R adopts three simple yet effective mechanisms, namely angle estimation, spatial mask, and template update. First, the framework includes multi-scale multi-angle candidates for search region. The scale change and the angle change of the tracked object are implicitly estimated according to the response maps. Second, spatial mask is applied when the aspect ratio of the target is apart from $1:1$ to reduce background noise. Lastly, moving average template update is adopted to deal with sequences with large target deformation.

### A.37. Cascade Siamese Conditional Random Fields Tracker (SiamCRF)

*F. Zhao, T. Zhang, Z. Zhang, W. Tang, J. Wang, M. Tang*
*fei.zhao@nlpr.ia.ac.cn,*
*{zhangting, zhangzhaoliang, tangwenjie}@ceiec.com.cn,*
*{jqwang, tangm}@nlpr.ia.ac.cn*

Unlike the previous works which divide the Convolutional Neural Network (CNN) and CRF individually, or optimize the CRF iteratively, we formulate and approximate the CRF as a siamese CNN which can be trained end-to-end with only one forward pass in the inference phase. The unary terms are modelled by one stream of the siamese CNN and the pairwise terms are modelled by the dense relationships between the features of both streams. Based on the weighted terms, SiamCRF predicts a probability for each position within the search area which measures how likely this position belongs to the target. To further improve the performance of SiamCRF, we save multiple target template and we create multiple proposals. Meanwhile, we propose the Proposal Refine Network (PRN), which can regress the bounding box in a cascade procedure [10] and select the

best proposal. The PRN consists of two fully connected layers. PRN outputs the bounding box regression offsets and predicts which proposal is the best.

### A.38. Fast Siamese Conditional Random Field Tracker (SiamCRF-RT)

*F. Zhao, T. Zhang, Z. Zhang, W. Tang, J. Wang, M. Tang*
*fei.zhao@nlpr.ia.ac.cn,*
*{zhangting, zhangzhaoliang, tangwenjie}@ceiec.com.cn,*
*{jqwang, tangm}@nlpr.ia.ac.cn*

We formulate and approximate the CRF as a Siamese CNN which can be trained end-to-end with only one forward pass in the inference phase.

### A.39. Online Deeper and Wider Siamese Networks for Real-Time Visual Tracking (SiamDW-ST)

*Z. Zhang, H. Peng*
*zhipeng.zhang2017@outlook.com,*
*houwen.peng@microsoft.com*

SiamDW-ST is a variant of [109]. The tracker consists of two parts, named region proposal network (RPN) and a proposal refinement network (PRN). In RPN network, we further increase the depth of the backbone network in SiamDW [109], and replace it with a much deeper one, i.e. MobileNetV2. The MobileNet-based backbone is lightweight, which guarantees the tracker can run at the real-time speed. Padding cropping operation is conducted on the backbone to alleviate perceptual inconsistency problem [109]. The PRN network has the similar architecture with IOU network [16]. It is appended after RPN to further refine the estimated bounding boxes of target objects. In PRN, we use the gradient of predicted IOU as a guidance to refine the predicted proposal of RPN. The refinement process is conducted only one time to guarantee the real-time speed.

### A.40. Siamese Fully Convolutional One-Stage Network for Short-Term Tracking (Siamfcos)

*X. Chen, Y. Lian, Y. Li, Y. Chen*
*{xechen, yclian}@stu.xidian.edu.cn,*
*{18792687583, 15764395531}@163.com*

Siamfcos tracker is based on the structure of SiamRPN++ [58]. The original anchor-based regression branch is replaced with an anchor-free regression branch and regress the distances from each location to the four sides of the bounding box. We also add a center-ness sub-branch to the classification branch to infer the center of the target. When tracking, the confidence score of the bounding box is obtained by multiplying the classification score and center-ness score. We train our network end-to-end on COCO, ILSVRC and partial data of the GOT-10k. During the inference phase, a long-term memory model (LMM) is employed to save and update templates instead of using only the first frame as a template image.

### A.41. Fully Convolutional One-Stage Siamese Network (SiamFCOSP)

*Z. Huang, J. Zhang*
*huangzj@stu.xidian.edu.cn, 2622786022@qq.com*

SiamFCOSP is an anchor-free tracker. ResNet50 is used as backbone to extract and to correlate multi-branch features, while FCOS is used as predict strategy. In SiamRPN++ algorithm [58] the multi-level features of the template image and the multi-level features of the search region are put into the cross correlation module to get the multi-channel correlation features. The Siamese Region Proposal Network (SiamRPN) head is replaced by the Siamese Fully Convolutional One-Stage Network (FCOS) head proposed in [87]. The Siamese FCOS head uses the multi-channel correlation features obtained by three branches: (i) the classification branch is used to predict the classification of each pixel, (ii) the centre-ness branch is used to predict the probability of being a target centre, and (iii) the regression branch is used to predict the offset of centre relative to bounding box. Finally, a single convolution layer is used to fusion all predictions.

### A.42. Siamese Fully Convolutional Object Tracking (SiamFCOT)

*W. Wang, X. Chen, X. Chen, Y. Xu, Z. Wang*
*weizhao.wang@pku.edu.cn, chenxingyu2015@ia.ac.cn,*
*xichen_zju@zju.edu.cn, yinda_xu@zju.edu.cn,*
*wangzeyu0408@outlook.com*

We propose an anchor-free technique for tracking task, namely SiamFCOT. Firstly, it performs a feature matching via cross-correlation operation. Next, the fused feature maps are sent to the head network to outputs on each feature-level pixel a regressed bounding box with its confidence score. After a penalization process, the box with the highest score is chosen and the image patch is processed by a mask branch to further refine the localization result. Adaptive search region size and template update are adopted to assure the robustness of the tracker.

### A.43. Discriminative Siamese Embedding for Object Tracking (SiamMargin)

*G. Chen, L. Chen, G. Li, Y. Chen, F. Wang, S. You, C. Qian*
{chenguangqi, chenlei, liguoxuan, chenyanjie, wangfei, youshan, qianchen}@sensetime.com

SiamMargin is based on the SiamRPN++[58] algorithm and it learns discriminative embedding features in Siamese networks for object tracking. In the training stage, a discrimination loss is added to the embedding layer which imposed a margin to the decision boundary to encourage learn-

ing discriminative embeddings. The discriminative embedding is offline learned in the training phase, so it keeps the high speed of Siamese RPN. In the inference stage we exploit an online updating method with ROIAlign for Siamese networks based trackers. The template feature of the object in current frame is obtained by ROIAlign from features of the current search region. Then, the template feature is updated via a moving average strategy. The discriminative embedding features are leveraged to accommodate the appearance change with properly online updating.

### A.44. SiamMask (SiamMask)

*Q. Wang, L. Zhang, L. Bertinetto, W. Hu, P. Torr*
*wangqiang2015@ia.ac.cn, {lz, luca}@robots.ox.ac.uk, wmhu@nlpr.ac.cn, philip.torr@eng.ox.ac.uk*

Our method, dubbed SiamMask, improves the offline training procedure of popular fully-convolutional Siamese approaches for object tracking by augmenting their loss with a binary segmentation task. In this way, our tracker gains a better instance-level understanding towards the object to track by exploiting the rich object mask representations offline. Once trained, SiamMask solely relies on a single bounding box initialisation and operates online, producing class-agnostic object segmentation masks and rotated bounding boxes. Code is publicly available at https://github.com/foolwood/SiamMask.

### A.45. Fitting Siamese Mask with Ellipses for Object Tracking (SiamMsST)

*B. X. Chen, J. Tsotsos*
*baoxchen@cse.yorku.ca, tsotsos@eecs.yorku.ca*

The SiamMsST tracker is an optimized version of SiamMask [96]. SiamMask tracks the target by generating masks (segmentation) on the target. SiamMsST applies an ellipse fitting algorithm [27] to the masks to compute the bounding boxes. By fitting an ellipse to a contour, the rotation of the bounding boxes has a better probability to match the human hand drawing bounding boxes. Then, rotate the mask and apply an up-right bounding rectangle to the rotated mask. We also developed a new version called SiamMask_E [11].

### A.46. SiamRPN++ (SiamRPNpp)

*Q. Wang, B. Li, F. Zhang*
*wangqiang2015@ia.ac.cn, libo@sensetime.com, fangyi.zhang@vipl.ict.ac.cn*

SiamRPN++ utilizes spatial aware sampling strategy to train a Deep Siamese network for visual tracking. SiamRPN++ is composed of a multi-layer aggregation module which assembles the hierarchy of connections to aggregate different levels of representation and a depth-wise correlation layer which allows our network to reduce computation cost and redundant parameters while also leading to better convergence. Code is available at https://github.com/STVIR/pysot.

### A.47. SiamRPNX (SiamRPNX)

*S. Guan, L. Guo, Y. Zhang, X. Sun*
*guanshs@mail2.sysu.edu.cn, guoleida@qq.com, vcheungyi@163.com, winfredsun@tencent.com*

SiamRPNX is based on the idea of SiamRPN++ A.46 but focuses on utilizing historical information to alleviate the problem of longe-term appearance change. Concretely, instead of initializing the template head only once like SiamRPN and SiamPPN++, SiamRPNX maintains a historical window containing K frames ($K = 6$) during tracking and adaptively updates the template with the frames in the window according the confidence scores. Furthermore, it integrates long-term and short-term information by applying correlation searching on the initial template and the previous predicted target respectively and fuses the results to predict the current target. In addition, the number of RPNs in our model is reduced to speed up the inference performance.

### A.48. SPM-Tracker: Series-Parallel Matching for Real-Time Visual Object Tracking (SPM)

*G. Wang, C. Luo, A. He, Z. Xiong, W. Zeng*
*wgting96@gmail.com, chong.luo@microsoft.com, heanfeng@mail.ustc.edu.cn, zwxiong@ustc.edu.cn, wezeng@microsoft.com*

SPM-tracker is a two-stage coarse-to-fine tracker that adopts the SiamRPN as the first stage and the relation network as the second stage. The motivation of the work is the simultaneous requirements on robustness and discrimination power of a visual object tracker. The basic idea of SPM-Tracker is to address the two seemingly contradictory requirements in two separate matching stages. Robustness is strengthened in the coarse matching (CM) stage through generalized training. Seven highest scored proposals produced by the CM stage are passed to the fine matching (FM) stage, which adopts a relation network to enhance the discrimination power. The matching scores and box location refinements of the two stages are fused to generate the final results.

### A.49. Selective Spatial Regularization for Correlation Filter based Tracking (SSRCCOT)

*Q. Guo, R. Han, Z. Chen, W. Feng*
*{tsingqguo, han_ruize, zh_chen, wfeng}@tju.edu.cn*

We propose selective spatial regularization (SSR) for the CF-tracking scheme that selectively uses target or context related filters to track the target by employing three different weight maps for spatial regularization. We formulate the online selection of these weight maps as a Markov Decision

Process (MDP). We equip the SSR to an existing CF tracker, namely CCOT [19], to get the final tracker SSRCCOT.

### A.50. Struck: Structured output tracking with kernels (struck2011)

*Submitted by VOT Committee*

Struck [34] is a framework for adaptive visual object tracking based on structured output prediction. The method uses a kernelized structured output support vector machine (SVM), which is learned online to provide adaptive tracking.

### A.51. Semantic Tracking Network: Tracking the Known and the Unknown by Leveraging Semantic Information (STN)

*A. Tripathi, M. Danelljan, L. Van Gool, R. Timofte*
*{ardhendu-shekhar.tripathi, martin.danelljan, vangool, timofter}@vision.ee.ethz.ch*

Current research in visual tracking is largely focused on the generic case, where no prior knowledge about the target object is assumed. However, many real-world tracking applications stem from specific scenarios where the class or type of object is known. Here, we propose a tracking framework that can exploit this semantic information (even when no semantic information is provided during inference), without sacrificing the generic nature of the tracker. In addition to the target-specific appearance, we model the class of the object through a semantic module (for both, classification of the target into one of the predefined classes and detection of objects of different classes in a scene) that provides complementary class-specific predictions.

### A.52. Target-Aware Deep Tracking (TADT)

*X. Li, C. Ma, B. Wu, Z. He, M.-H. Yang*
*xinlihitsz@gmail.com, chaoma@sjtu.edu.cn, wubaoyuan1987@gmail.com, zhenyuhe@hit.edu.cn, mhyang@ucmerced.edu*

The TADT [61] tracker learns target-aware features for robust visual tracking. The learning is based on the gradients of specifically designed losses, which include a regression loss for generating target-active features and a ranking loss for generating scale-sensitive features. With the generated target-aware features, the tracking process is performed under a VGG based Siamese framework.

### A.53. Temporal confidence learning based correlation filter tracker (TCLCF)

*C.-Y. Tsai*
*chiyi_tsai@gms.tku.edu.tw*

TCLCF is a real-time ensemble correlation filter tracker based on a temporal confidence learning method. In the current implementation, we use three different correlation filters to track the same target cooperatively. The TCLCF tracker is a fast and robust generic object tracker without GPU acceleration; therefore, it can be implemented on embedded platforms with limited computing resources.

### A.54. Tracking and Detection: A Unified Approach (TDE)

*C. Zhang, S. Zhao, S. Li, K. Zhang, T. Xu, Z. Luo, S. Ge*
*{zhangchunhui, zhaoshengwei, lishikun, zhangkangkai}@iie.ac.cn, tianyang_xu@163.com, {luochao, geshiming}@iie.ac.cn*

The TDE tracker unifies tracking and detection technique for adaptive target state estimation, which is based on a discriminative correlation filter method [106]. Moreover, an adaptive spatial feature selection scheme [100] is employed to learn a robust deep tracking model. We also introduce an explicit measure to identify the tracking failure and utilize the best detection result to refine the target state. In this way, the TDE tracker achieves robust and accurate target localization in a unified fashion.

### A.55. Dynamic optimization tracking algorithm based on ATOM combined with static pictures (Trackyou)

*P. Zheng, X. Qiu, J. Wu*
*{1023567918, 584237193, 454666966}@qq.com*

The tracker Trackyou improves ATOM tracker [16] in the following three aspects: (1) Training data: it increases the number of static image pairs to increase diversity of sample categories. (2) Offline training: it adopts the triplet loss and the offline classification for extracting discriminative features during offline training, which sets the network parameters learned from the offline training as the initialization parameters of online classification to keep the performance of the network more robust. (3) Online tracking phase: it dynamically updates the tracking algorithm according to the fuzzy factor defined by ourselves and the feature map of the classification network.

### A.56. UInet( Single Object tracking based on Unet and IouNet ) (UInet)

*P. Zhang, Y. Xu, D. Tao*
*pengfeizhang0520@gmail.com, xyf97@mail.ustc.edu.cn, dacheng.tao@gmail.com*

The UInet tracker is based on an extension of ATOM [16]. A new conv-net segmentation module is introduced to optimize the output to provide a more detailed localization. The module is trained offline on segmentation datasets. Multi attention are implemented on the extracted feature map: Spatial attention enhance its ability to distinguish similar targets while channel attention improves its robustness on complex targets. The tracker uses Resnet [36] as a backbone.

### A.57. Weighted samples based CF tracker (WSCF-ST)

*R. Han, W. Feng, Q. Guo, Z. Chen*
{*han_ruize, wfeng, tsingqguo, zh_chen*}*@tju.edu.cn*

In WSCF a simple yet effective energy function, which can be regarded as assigning the weights to different training samples, is defined to remedy the annoying boundary effects of CF tracking.

## B. VOT-LT2019 submissions

This appendix provides a short summary of trackers considered in the VOT-LT2019 challenge.

### B.1. Assisted Siamese Instance Search Tracking (ASINT)

*D. Gupta, E. Gavves, A. Smeulders*
{*d.k.gupta, e.gavves, ArnoldSmeulders*}*@uva.nl*

This tracker is based on the Siamese tracking framework and is a modified version of the LTSINT tracker from VOTLT2018 challenge. In addition to the local and global search methodologies, ASINT uses an evidence-gathering approach. Under conditions where the object changes its appearance, invoking global search too early can easily cause the tracker to lose the target object. Thus, ASINT invokes global search more cautiously. To compensate for the uncertain similarity range, a short-duration motion model is employed which validates the predictions from Siamese search, and decides whether the global search needs to be invoked or not. In addition, under uncertainty, the choice of the best candidate also depends on its spatial distance from the last prediction obtained from local search.

### B.2. Complementary Local-Global Search for Robust Long-term Tracking (CLGS)

*H. Zhao*, B. Yan*, D. Wang, H. Lu, X. Yang*
*zhaohj1995@gmail.com, yan_bin@mail.dlut.edu.cn,*
{*wdice, lhchuan*}*@dlut.edu.cn,*
*xiaoyun.yang@intellicloud.ai*

In this work, we develop a complementary local-global search (CLGS) framework to conduct robust long-term tracking. CLGS tracker is based on SiamMask [96] tracker, a global detection based on cascade R-CNN [10] and an online verifier based on Real-time MDNet [42]. During online tracking, the SiamMask model locates the target in local region and estimates the size of the target according to the predicted mask. The online verifier is used to judge whether the target is found or lost. Once the target is lost, a global R-CNN detector without class prediction is used to generate region proposals on the whole image. Then, the online verifier will find the target from region proposals again.

### B.3. Synergistic CooSiam Framework based on Comprehensive Template's Feature and Detection (CooSiam)

*R. Zhang, J. Gao, J. Chen*
*ruohan950427@163.com, gaojie_jiangsu@126.com,*
*chenjie818826@163.com*

We proposed a synergistic CooSiam framework based on comprehensive templates and detection. The templates are updated by choosing one of the three following possibilities: (i) the first frame's template, (ii) the average of the first frame template's feature and the latest one useful feature, and (iii) the superposition and average of all the useful templates' features obtained before the current frame. The selected template is the template achieving the highest score. By comparing the tracking results and the detection results obtained by applying YOLO V3 on the current frame, the new template is obtained.

### B.4. Fully Correlational Long-Term Tracker (FuCoLoT)

*Submitted by VOT Committee*

FuCoLoT is a Fully Correlational Long-term Tracker. It exploits the novel DCF constrained filter learning method to design a detector that is able to re-detect the target in the whole image efficiently. Several correlation filters are trained on different time scales that act as the detector components. A mechanism based on the correlation response is used for tracking failure estimation.

### B.5. long-term tracking by diving videos into successive short episodes (LT-DSE)

*K. Dai, Y. Zhang, J. Li, D. Wang, X. Yang, H. Lu*
*dkn2014@mail.dlut.edu.cn,*
{*zhangyunhua@mail., jianhual@, wdice@*}*dlut.edu.cn,*
*xiaoyun.yang@intellicloud.ai, lhchuan@dlut.edu.cn*

The tracker LT-DSE divides each long-term sequence into several short episodes and tracks the target in each episode using short-term tracking techniques. If the target disappears, the image-wide re-detection outputs the possible location and size of the target. The tracker crops the local search region and sends it to the RPN based regression network. Then, the candidate proposals from the regression network will be scored by the online learned verifier. The candidate with the maximum score will be regarded as the target and the tracker conducts short-term tracking which contains two components. One is for target localization based on ATOM algorithm [16]. It uses ResNet18 as the backbone network and adds two convolutional layers above it. The other component is the SiamMask network [96] used for refining the bounding box after locating the centre of the target. For the verifier RT-MDNet network [42] is used as backbone and is pre-trained on ILSVRC VID dataset. The architecture of the region-proposal network is based

on [108]. The network is trained using LaSOT dataset [24] and ILSVRC image detection dataset.

### B.6. mbdet (mbdet)

*J. Chen, J. Gao, R. Zhang*
*chenjie818826@163.com, gaojie_jiangsu@162.com, ruohan950427@163.com*

Based on MBMD, mbdet tracker modifies the positive and negative sample screening mechanism of the classifier, improves the robustness of the classifier, and adds detection mechanism and motion information.

### B.7. Online Deeper and Wider Siamese Networks for Long-Term Visual Tracking (SiamDW-LT)

*H. Du, H. Peng, J. Fu*
{*v-had, houwen.peng, jianf*}*@microsoft.com*

SiamDW-LT is a long-term tracker that equips deeper and wider tracking networks with fast online updates. The basic tracking module is a short-term Siamese tracker, which returns a confidence score to indicate the tracking reliability. When the Siamese tracker is uncertain on its tracking accuracy, an online correction module is triggered to refine the results. When the Siamese tracker is failed, a re-detection module is activated to search the target in the images globally. Moreover, object disappearance or occlusion is also identified by the tracking confidences. Finally, we introduce model ensemble to further improve the tracking accuracy and robustness. Code is available at https://github.com/researchmm/VOT2019.

### B.8. Siamese Fully Convolutional One-Stage Network for Long-Term Tracking (Siamfcos-LT)

*X. Chen, Y. Lian, Y. Li, Y. Chen*
{*xechen, yclian*}*@stu.xidian.edu.cn,*
{*18792687583, 15764395531*}*@163.com*

Siamfcos-LT tracker is based on the tracker Siamfcos A.40. Siamfcos-LT adds yolov3 detection to assist tracking in order to prevent the tracking box from shifting to meaningless background or to prevent the error accumulation in the follow-up tracking process caused by the inaccuracy of the tracking box regression. Thus, the addition of yolov3 detection algorithm helps finding the target back when it dissapears.

### B.9. Optimize SiamRPN with Random Search for Long-Term Tracking (SiamRPNsLT)

*B. X. Chen, J. Tsotsos*
*baoxchen@cse.yorku.ca, tsotsos@eecs.yorku.ca*

The SiamRPNsLT tracker uses the backbone of SiamRPN++ [58] with optimized random search strategies to enhance long-term tracking capability. SiamRPNsLT has two random search strategies: (1) 5-point random locations, and (2) random bounding box size. With 5-point random location, the tracker allows the target moving out of the frame and re-entering in the nearest location. By randomizing the bounding box size, the tracker allows the target to have different sizes once re-entered.

## C. VOT-RGBT2019 submissions

This appendix provides a short summary of trackers considered in the VOT-RGBT2019 challenge.

### C.1. Channel Independent Spatially Regularized Discriminative Correlation Filter Tracker (CISRDCF)

*A. Varfolomieiev*
*a.varfolomieiev@kpi.ua*

The CISRDCF method is based on SRDCF formulation [17]. CISRDCF tracker defers from the original SRDCF in two main points: 1) it calculates the filter channels for each feature channel independently, and 2) the regularization in the filter is performed iteratively using the ADMM approach [29]. To suppress the information outside the object's bounding-box, the rectangular regularization window with slightly blurred edges is applied. The method uses the HOG features augmented with additional channel, which represents the back-projection of object histogram. This channel is equivalent to the per-pixel scores used in histogram-related part of Staple tracker [6]. The TIR version of the tracker extracts HOG features and object histogram from grayscale images and IR-images, and find the final correlation by summing the partial correlation results over channels.

### C.2. Fusion SiamRPN Tracker with Spatial Attention Fusion Strategy (FSRPN)

*H. Li, X.-J Wu, J. Kittler, T. Xu, X. Zhu, Y. Li*
*lihui@stu.jiangnan.edu.cn, xiaojun_wu_jnu@163.com,*
*j.kittler@surrey.ac.uk,*
{*tianyang_xu, xuefeng_zhu95*}*@163.com,*
*6171910026@stu.jiangnan.edu.cn*

In FSRPN tracker, spatial attention-based fusion strategy is applied to Siamese CNN framework. The deep features are extracted by ResNet-50 from RGB and the thermal images are fused to get more accurate and more plentiful information of object. Then, these fused deep features are utilized to track objects by the RPN-based network [58].

### C.3. Gradient of Entropy Sensor based Background Trackable Tracker (GESBTT)

*B. Kim, A. Lukezic, J. Lee, H. Jung, J. Lee, E. Yi, M. Kim*
{*durumy98, leewer354, ytr789, ark986*}*@knu.ac.kr,*
*eunu.yi@hanwha.com, minykim@knu.ac.kr*

The tracker GESBTT is a Gradient of Entropy Sensor based Background Trackable Tracker. The proposed tracker

consists of a global motion-aware method using a gradient of entropy sensor with multiple analysis both RGB and TIR. The method increases the robustness of the tracker when the RGB and TIR cameras are moved. This framework of GES-BTT can be easily integrated into any visual trackers and it excelled in the tracker for the camera moving condition with fast calculation time. The basis of filtering methods are DSST and Staple. For more details, we refer the reader to [33, 6].

### C.4. Joint Modeling Motion and Appearance Cues for Robust RGB-T Tracking (JMMAC)

*P. Zhang, J. Zhao, M. Ni, D. Wang, H. Lu, X. Yang*
*zpy.dut@gmail.com,*
*{zj982853200, ningmeng}@mail.dlut.edu.cn,*
*{wdice, lhchuan}@dlut.edu.cn,*
*xiaoyun.yang@intellicloud.ai*

In this work, we have found that both motion and appearance cues are important for designing a robust RGB-T tracker. The motion cue includes two components: camera motion and object motion. The camera motion is inferred based on the key-point-based image registration technique; and the object motion is estimated based on the camera motion estimation and the Kalman filter method. The appearance cue is captured based on an improved ECO model, where complementary features are selected for the RGB-T tracking task. When the object suffers from heavy or full occlusion, a motion-guided tracking mechanism is used to avoid drifting, which makes the tracker be dynamically switched between the tracking and prediction states.

### C.5. Multi-Adapter Convolutional Networks for RGBT Tracking (MANet)

*A. Lu, C. Li, L. Liu, J. Tang*
*{adlu_ah, lcl1314}@foxmail.com, 1210568677@qq.com,*
*tangjin@ahu.edu.cn*

We propose a novel Multi-Adapter convolutional Network (MANet) to jointly perform modality-shared, modality-specific and instance-aware feature learning in an end-to-end trained deep framework for RGBT tracking. We design three kinds of adapters within our network. In a specific, the generality adapter is to extract shared object representations, the modality adapter aims at encoding modality-specific information to deploy their complementary advantages, and the instance adapter is to model the appearance properties and temporal variations of a certain object.

### C.6. Multi-modal fusion for end-to-end RGB-T tracking (mfDiMP)

*L. Zhang, A. Gonzalez-Garcia, J. van de Weijer*
*{lichao, agonzalez, joost}@cvc.uab.es*

The mfDiMP tracker contains an end-to-end tracking framework for fusing the RGB and TIR modalities in RGB-T tracking [107]. The baseline tracker is DiMP (Discriminative Model Prediction) [8], which employs a carefully designed target prediction network trained end-to-end using a discriminative loss. The mfDiMP tracker fuses modalities at the feature level on both the IoU predictor and the model predictor of DiMP [107].

### C.7. More precise box and accurate object tracking (MPAT)

*L. Zhou*
*lijun13548184189@outlook.com*

For a tracker description, the reader is referred to A.30.

### C.8. Online Deeper and Wider Siamese Networks for RGBT Visual Tracking (SiamDW-T)

*Z. Zhang, H. Peng, J. Fu, B. Li, W. Hu*
*zhipeng.zhang2017@outlook.com, {houwen.peng,*
*jianf}@microsoft.com, {bli, wmhu}@nlpr.ia.ac.cn*

SiamDW-T is based on our previous work [109], and extends it with two fusion strategies for RGBT tracking. First, we get two localizations from the RGB and TIR images. After that, several random bounding boxes are proposed around these two positions. We fuse these proposals into a more extensive collection. Then, the individual RGB and TIR features responded to each localization are extracted. We introduce a cross-attention module to fuse features of different domains. Specifically, RGB and TIR features are fused with channel-wise dot operation. Finally, a simple fully connected layer is appended to classify each fused feature to background or foreground. Code is available at https://github.com/researchmm/VOT2019.

## D. VOT-RGBD2019 submissions

This appendix provides a short summary of trackers considered in the VOT-RGBD2019 challenge.

### D.1. Accurate Tracking by Category-Agnostic Instance Segmentation for RGBD Image (ATCAIS)

*Y. Wang, L. Wang, D. Wang, H. Lu, X. Yang*
*{wym097, wlj}@mail.dlut.edu.cn,*
*{wdice, lhchuan}@dlut.edu.cn,*
*xiaoyun.yang@intellicloud.ai*

The proposed tracker combines both instance segmentation and the depth information for accurate tracking. The tracker ATCAIS is based on the ATOM tracker [16] and the HTC instance segmentation method [12] which is re-trained in a category-agnostic manner. The instance segmentation results are used to detect background distractors and to refine the target bounding boxes to prevent drifting. The depth value is used to detect the target occlusion or disappearance and re-find the target. The submitted tracker did not report

the confidence. A version with the confidence is available at https://github.com/tangjiuqi097/ATCAIS and our own measured F-1 measure for that updated tracker is 0.7016.

### D.2. long-term tracking using depth information by diving videos into successive short episodes (LTDSEd)

*Y. Zhang, K. Dai, L. Wang, J. Qi, H. Lu*
*{zhangyunhua, dkn2014}@mail.dlut.edu.cn,*
*xingkong19890806@gmail.com,*
*{jinqing, lhchuan}@dlut.edu.cn*

The tracker LTDSEd divides each long-term sequence into several short episodes and tracks the target in each episode using short-term tracking techniques. The visibility of the target is judged by the outputs from short-term components. See also the description of LT-DSE from the same authors (B.5).

### D.3. Online Deeper and Wider Siamese Networks for RGBD Visual Tracking (SiamDW-D)

*H. Yu, H. Peng, Z. Wu, Y. Huang, J. Fu, L. Wang*
*{v-hongyy, houwen.peng, Wu.Zhirong}@microsoft.com,*
*yhuang@nlpr.ia.ac.cn, jianf@microsoft.com,*
*wangliang@nlpr.ia.ac.cn*

SiamDW-D is a long-term tracker which mainly addresses the problems of target appearance variations, and frequent disappearance and re-appearance of target objects. It contains three parts, i.e. a main tracker, a re-detection module, and a multi-template matching module. The main tracker is based on [109], and further equips it with an online updating model, similar to [16, 75]. The re-detection module is triggered when the main tracker is not confident on its predictions. However, in some cases, the confidence of main tracker is not reliable for re-detection module triggering, therefore we introduce a multi-template matching module. It matches the unreliable tracking results with history templates, and outputs a more reliable estimation. Moreover, the depth information is also used to estimate the disappearance of target objects. Code is available at https://github.com/researchmm/VOT2019. For more information see the short-term tracker from the same group (A.39).

### D.4. Enhance SiamMask Tracker Using RGBD Images (SiamM_Ds)

*B. X. Chen, J. Tsotsos*
*baoxchen@cse.yorku.ca, tsotsos@eecs.yorku.ca*

The SiamM_Ds tracker is a modified version of SiamMask [96] to track objects in RGB and Depth images. SiamMask produces segmentation on the tracking target. So that, by averaging the depth from the depth image in the same mask could determine the depth of the target. Then, we apply the constraint that the target can not have a very large displacement in two consecutive frames.

## References

[1] B. Babenko, M. H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(8):1619–1632, 2011.

[2] C. Bao, Y. Wu, H. Ling, and H. Ji. Real time robust l1 tracker using accelerated proximal gradient approach. In *CVPR*, 2012.

[3] A. Berg, J. Ahlberg, and M. Felsberg. A Thermal Object Tracking Benchmark. In *12th IEEE International Conference on Advanced Video- and Signal-based Surveillance, Karlsruhe, Germany, August 25-28 2015*. IEEE, 2015.

[4] A. Berg, J. Ahlberg, and M. Felsberg. Generating visible spectrum images from thermal infrared. In *CVPR Workshops*, 2018.

[5] A. Berg, J. Johnander, F. D. de Gevigney, J. Ahlberg, and M. Felsberg. Semi-automatic annotation of objects in visual-thermal video. In *IEEE International Conference on Computer Vision, ICCV Workshops*, 2019.

[6] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr. Staple: Complementary learners for real-time tracking. In *CVPR*, pages 1401–1409, 2016.

[7] L. Bertinetto, J. Valmadre, J. Henriques, P. H. S. Torr, and A. Vedaldi. Fully convolutional siamese networks for object tracking. In *ECCV Workshops*, pages 850–865, 2016.

[8] G. Bhat, M. Danelljan, L. V. Gool, and R. Timofte. Learning discriminative model prediction for tracking. In *IEEE International Conference on Computer Vision, ICCV*, 2019.

[9] G. Bhat, J. Johnander, M. Danelljan, F. S. Khan, and M. Felsberg. Unveiling the power of deep tracking. In *ECCV*, pages 483–498, 2018.

[10] Z. Cai and N. Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6154–6162, 2018.

[11] B. X. Chen and J. K. Tsotsos. Fast visual object tracking with rotated bounding boxes. *arXiv preprint arXiv:1907.03892*, 2019.

[12] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, C. C. Loy, and D. Lin. Hybrid task cascade for instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[13] C. Choi and H. Christensen. RGB-d object tracking: A particle filter approach on GPU. In *IROS*, 2013.

[14] W. Choi, C. Pantofaru, and S. Savarese. A General Framework for Tracking Multiple People from a Moving Camera. *IEEE PAMI*, 2013.

[15] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg. ECO: efficient convolution operators for tracking. In *CVPR*, pages 6638–6646, 2017.

[16] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg. ATOM: Accurate tracking by overlap maximization. In *CVPR*, 2019.

[17] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg. Learning spatially regularized correlation filters for visual tracking. In *Int. Conf. Computer Vision*, pages 4310–4318, 2015.

[18] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg. Discriminative scale space tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(8):1561–1575, 2016.

[19] M. Danelljan, A. Robinson, F. Shahbaz Khan, and M. Felsberg. Beyond Correlation Filters: Learning Continuous Convolution Operators for Visual Tracking. In *ECCV*, 2016.

[20] G. De Ath and R. Everson. Part-based tracking by sampling. *CoRR*, abs/1805.08511, 2018.

[21] G. De Ath and R. Everson. Visual object tracking: The initialisation problem. In *2018 15th Conference on Computer and Robot Vision (CRV)*, pages 142–149, 5 2018.

[22] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: a large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[23] A. Ess, B. Leibe, K. Schindler, and L. van Gool. A Mobile Vision System for Robust Multi-Person Tracking. In *CVPR*, 2008.

[24] H. Fan, L. Lin, F. Yang, P. Chu, G. Deng, H. B. S. Yu, Y. Xu, C. Liao, and H. Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *Comp. Vis. Patt. Recognition*, 2019.

[25] M. Felsberg, A. Berg, G. Häger, J. Ahlberg, M. Kristan, A. Leonardis, J. Matas, G. Fernández, L. Čehovin, and et al. The thermal infrared visual object tracking VOT-TIR2015 challenge results. In *ICCV2015 workshop proceedings, VOT2015 Workshop*, 2015.

[26] M. Felsberg, M. Kristan, J. Matas, A. Leonardis, R. Pflugfelder, G. Häger, A. Berg, A. Eldesokey, J. Ahlberg, L. Čehovin, T. Vojír, A. Lukežič, G. Fernández, and et al. The thermal infrared visual object tracking VOT-TIR2016 challenge results. In *ECCV2016 Workshop Proceedings, VOT2016 Workshop*, volume 9914 of *Lecture Notes in Computer Science*, pages 824–849, 2016.

[27] A. W. Fitzgibbon, R. B. Fisher, et al. *A buyer's guide to conic fitting*. University of Edinburgh, Department of Artificial Intelligence, 1996.

[28] H. K. Galoogahi, A. Fagg, C. Huang, D. Ramanan, and S. Lucey. Need for speed: A benchmark for higher frame rate object tracking. *CoRR*, abs/1703.05884, 2017.

[29] H. K. Galoogahi, A. Fagg, and S. Lucey. Learning background-aware correlation filters for visual tracking. In *ICCV*, pages 1144–1152, 2017.

[30] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim. First-Person Hand Action Benchmark with RGB-D Videos and 3D Hand Pose Annotations. In *CVPR*, 2018.

[31] N. Goyette, P. M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar. Changedetection.net: A new change detection benchmark dataset. In *CVPR Workshops*, pages 1–8. IEEE, 2012.

[32] C. Guo and L. Zhang. A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression. *IEEE Transactions on Image Processing*, 19(1):185–198, 2009.

[33] J. Guo et al. Dynamic displacement measurement of large-scale structures based on the Lucas–Kanade template tracking algorithm. *Mechanical Systems and Signal Processing*, 66:425–436, 2016.

[34] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured output tracking with kernels. In D. N. Metaxas, L. Quan, A. Sanfeliu, and L. J. V. Gool, editors, *Int. Conf. Computer Vision*, pages 263–270. IEEE, 2011.

[35] A. He, C. Luo, X. Tian, and W. Zeng. A twofold siamese network for real-time object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4834–4843, 2018.

[36] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[37] J. Henriques, R. Caseiro, P. Martins, and J. Batista. High-speed tracking with kernelized correlation filters. *PAMI*, 37(3):583–596, 2015.

[38] L. Huang, X. Zhao, and K. Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *arXiv:1810.11981*, 2018.

[39] V. Jack, B. Luca, H. J. ao F., T. Ran, V. Andrea, S. Arnold, T. Philip, and G. Efstratios. Long-term tracking in the wild: A benchmark. *arXiv:1803.09502*, 2018.

[40] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang. Acquisition of localization confidence for accurate object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 784–799, 2018.

[41] J. Johnander, M. Danelljan, E. Brissman, F. S. Khan, and M. Felsberg. A generative appearance model for end-to-end video object segmentation. In *CVPR*, 2019.

[42] I. Jung, J. Son, M. Baek, and B. Han. Real-time mdnet. In *ECCV*, pages 83–98, 2018.

[43] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(7):1409–1422, 2012.

[44] U. Kart, J.-K. Kämäräinen, and J. Matas. How to Make an RGBD Tracker ? In *ECCV Workshops*, 2018.

[45] U. Kart, A. Lukežič, M. Kristan, J.-K. Kämäräinen, and J. Matas. Object Tracking by Reconstruction with View-Specific Discriminative Correlation Filters. In *CVPR*, 2019.

[46] F. S. Khan, J. Weijer, A. D. Bagdanov, and M. Vanrell. Portmanteau vocabularies for multi-cue image representation. In *Advances in Neural Information Processing Systems 24*, 2011.

[47] P. Koschorrek, T. Piccini, P. Öberg, M. Felsberg, L. Nielsen, and R. Mester. A multi-sensor traffic scene dataset with omnidirectional video. In *CVPR Workshops*, 2013.

[48] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojíř, G. Bhat, A. Lukežič, A. Eldesokey, G. Fernández, and et al. The visual object tracking vot2018 challenge results. In *ECCV2018 Workshops, Workshop on visual object tracking challenge*, 2018.

[49] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojíř, G. Häger, A. Lukežič, A. Eldesokey, G. Fernández, and et al. The visual object

tracking vot2017 challenge results. In *ICCV2017 Workshops, Workshop on visual object tracking challenge*, 2017.

[50] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojíř, G. Häger, A. Lukežič, G. Fernández, and et al. The visual object tracking vot2016 challenge results. In *ECCV2016 Workshops, Workshop on visual object tracking challenge*, 2016.

[51] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Čehovin, G. Fernández, T. Vojíř, G. Häger, G. Nebehay, R. Pflugfelder, and et al. The visual object tracking vot2015 challenge results. In *ICCV2015 Workshops, Workshop on visual object tracking challenge*, 2015.

[52] M. Kristan, J. Matas, A. Leonardis, T. Vojíř, R. Pflugfelder, G. Fernández, G. Nebehay, F. Porikli, and L. Čehovin. A novel performance evaluation methodology for single-target trackers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2137–2155, 2016.

[53] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, F. Porikli, L. Čehovin, G. Nebehay, G. Fernández, T. Vojíř, and et al. The visual object tracking vot2013 challenge results. In *ICCV2013 Workshops, Workshop on visual object tracking challenge*, pages 98 –111, 2013.

[54] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, L. Čehovin, G. Nebehay, T. Vojíř, G. Fernández, and et al. The visual object tracking vot2014 challenge results. In *ECCV2014 Workshops, Workshop on visual object tracking challenge*, 2014.

[55] L. Leal-Taixé, A. Milan, I. D. Reid, S. Roth, and K. Schindler. Motchallenge 2015: Towards a benchmark for multi-target tracking. *CoRR*, abs/1504.01942, 2015.

[56] K. Lebeda, R. Bowden, and J. Matas. Long-term tracking through failure cases. In *Vis. Obj. Track. Challenge VOT2013, In conjunction with ICCV2013*, 2013.

[57] A. Li, M. Li, Y. Wu, M.-H. Yang, and S. Yan. Nus-pro: A new visual tracking challenge. *IEEE-PAMI*, 2015.

[58] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan. SiamRPN++: Evolution of Siamese Visual Tracking with Very Deep Networks. *arXiv preprint arXiv:1812.11703*, 2018.

[59] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu. High performance visual tracking with siamese region proposal network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8971–8980, June 2018.

[60] C. Li, X. Liang, Y. Lu, N. Zhao, and J. Tang. RGB-T object tracking: Benchmark and baseline. *Pattern Recognition*, 2019. submitted.

[61] X. Li, C. Ma, B. Wu, Z. He, and M.-H. Yang. Target-aware deep tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[62] P. Liang, E. Blasch, and H. Ling. Encoding color information for visual tracking: Algorithms and benchmark. *IEEE Transactions on Image Processing*, 24(12):5630–5644, 2015.

[63] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[64] Y. Liu, X.-Y. Jing, J. Nie, H. Gao, J. Liu, and G.-P. Jiang. Context-aware 3-D Mean-shift with Occlusion Handling for Robust Object Tracking in RGB-D Videos. *IEEE TMM*, 2018.

[65] A. Lukežič, U. Kart, J. Kämäräinen, J. Matas, and M. Kristan. CDTB: A Color and Depth Visual Object Tracking Dataset and Benchmark. In *ICCV*, 2019.

[66] A. Lukežič, L. Č. Zajc, and M. Kristan. Deformable parts correlation filters for robust visual tracking. *IEEE Transactions on Cybernetics*, PP(99):1–13, 2017.

[67] A. Lukežič, L. Čehovin Zajc, T. Vojíř, J. Matas, and M. Kristan. Now you see me: evaluating performance in long-term visual tracking. *CoRR*, abs/1804.07056, 2018.

[68] A. Lukežič, L. Čehovin Zajc, T. Vojíř, J. Matas, and M. Kristan. Performance evaluation methodology for long-term visual object tracking. *CoRR*, abs/1906.08675, 2019.

[69] A. Lukežič, T. Vojíř, L. Čehovin Zajc, J. Matas, and M. Kristan. Discriminative correlation filter with channel and spatial reliability. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6309–6318, July 2017.

[70] A. Lukežič, L. Čehovin Zajc, T. Vojiř, J. Matas, and M. Kristan. FuCoLoT - A Fully-Correlational Long-Term Tracker. In *ACCV*, 2018.

[71] A. Memarmoghadam and P. Moallem. Size-aware visual object tracking via dynamic fusion of correlation filter-based part regressors. *Signal Processing*, 164:84–98, 2019.

[72] A. Moudgil and V. Gandhi. Long-term visual object tracking benchmark. *arXiv preprint arXiv:1712.01358*, 2017.

[73] M. Mueller, N. Smith, and B. Ghanem. A benchmark and simulator for uav tracking. In *Proc. European Conf. Computer Vision*, pages 445–461, 2016.

[74] 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 *ECCV*, pages 300–317, 2018.

[75] H. Nam and B. Han. Learning multi-domain convolutional neural networks for visual tracking. In *CVPR*, pages 4293–4302, 2016.

[76] F. Pernici and A. del Bimbo. Object tracking by oversampling local features. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(12):2538–2551, 2013.

[77] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss. The feret evaluation methodology for face-recognition algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(10):1090–1104, 2000.

[78] E. Real, J. Shlens, S. Mazzocchi, X. Pan, and V. Vanhoucke. YouTube-BoundingBoxes: a large high-precision human-annotated data set for object detection in video. In *Comp. Vis. Patt. Recognition*, pages 7464–7473, 2017.

[79] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. *arXiv e-prints*, page arXiv:1902.09630, Feb 2019.

[80] S. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for Data: Ground Truth from Computer Games. In *ECCV*, 2016.

[81] D. A. Ross, J. Lim, R. S. Lin, and M. H. Yang. Incremental learning for robust visual tracking. *Int. J. Comput. Vision*, 77(1-3):125–141, 2008.

[82] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual Tracking: an Experimental Survey. *TPAMI*, 2013.

[83] F. Solera, S. Calderara, and R. Cucchiara. Towards the evaluation of reproducible robustness in tracking-by-detection. In *Advanced Video and Signal Based Surveillance*, pages 1 – 6, 2015.

[84] S. Song and J. Xiao. Tracking Revisited Using RGBD Camera: Unified Benchmark and Baselines. In *ICCV*, 2013.

[85] L. Spinello and K. O. Arras. People detection in RGB-D data. In *IROS*, 2011.

[86] R. Tao, E. Gavves, and A. W. M. Smeulders. Tracking for half an hour. *CoRR*, abs/1711.10217, 2017.

[87] Z. Tian, C. Shen, H. Chen, and T. He. Fcos: Fully convolutional one-stage object detection. *arXiv preprint arXiv:1904.01355*, 2019.

[88] L. Čehovin. TraX: The visual Tracking eXchange Protocol and Library. *Neurocomputing*, 2017.

[89] L. Čehovin, M. Kristan, and A. Leonardis. Is my new tracker really better than yours? Technical Report 10, ViCoS Lab, University of Ljubljana, Oct 2013.

[90] L. Čehovin, M. Kristan, and A. Leonardis. Robust visual tracking using an adaptive coupled-layer visual model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(4):941–953, 2013.

[91] L. Čehovin, A. Leonardis, and M. Kristan. Visual object tracking performance measures revisited. *IEEE Transactions on Image Processing*, 25(3), 2015.

[92] L. Čehovin, A. Leonardis, and M. Kristan. Robust visual tracking using template anchors. In *WACV*. IEEE, Mar 2016.

[93] L. Čehovin Zajc, A. Lukežič, A. Leonardis, and M. Kristan. Beyond standard benchmarks: Parameterizing performance evaluation in visual object tracking. *ICCV*, abs/1612.00089, 2017.

[94] T. Vojíř and J. Matas. Pixel-wise object segmentations for the VOT 2016 dataset. Research Report CTU–CMP–2017–01, Center for Machine Perception, K13133 FEE Czech Technical University, Prague, Czech Republic, January 2017.

[95] T. Vojíř, J. Noskova, and J. Matas. Robust scale-adaptive mean-shift for tracking. *Pattern Recognition Letters*, 49:250–258, 2014.

[96] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. Torr. Fast online object tracking and segmentation: A unifying approach. In *CVPR*, pages 1328–1338, 2019.

[97] Y. Wu, J. Lim, and M. H. Yang. Online object tracking: A benchmark. In *Comp. Vis. Patt. Recognition*, 2013.

[98] Y. Wu, J. Lim, and M.-H. Yang. Object tracking benchmark. *PAMI*, 37(9):1834–1848, 2015.

[99] J. Xiao, R. Stolkin, Y. Gao, and A. Leonardis. Robust Fusion of Color and Depth Data for RGB-D Target Tracking Using Adaptive Range-Invariant Depth Models and Spatio-Temporal Consistency Constraints. *IEEE Transactions on Cybernetics*, 48:2485 – 2499, 2018.

[100] T. Xu, Z. Feng, X. Wu, and J. Kittler. Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking. *IEEE Transactions on Image Processing*, pages 1–14, 2019.

[101] T. Yang and A. B. Chan. Learning Dynamic Memory Networks for Object Tracking. In *ECCV*, 2018.

[102] T. Yang and A. B. Chan. Visual Tracking via Dynamic Memory Networks. *TPAMI*, 2019.

[103] T. Yang, X. Pengfei, H. Runbo, C. Hua, and A. B. Chan. ROAM: Recurrently Optimizing Tracking Model. *arXiv*, 2019.

[104] L. Yiming, J. Shen, and M. Pantic. Mobile face tracking: A survey and benchmark. *arXiv:1805.09749v1*, 2018.

[105] D. P. Young and J. M. Ferryman. PETS Metrics: On-line performance evaluation service. In *ICCCN '05 Proceedings of the 14th International Conference on Computer Communications and Networks*, pages 317–324, 2005.

[106] C. Zhang, S. Ge, Y. Hua, and D. Zeng. Robust deep tracking with two-step augmentation discriminative correlation filters. In *IEEE International Conference on Multimedia and Expo*, pages 1774–1779, 2019.

[107] L. Zhang, M. Danelljan, A. Gonzalez-Garcia, J. van de Weijer, and F. S. Khan. Multi-modal fusion for end-to-end rgb-t tracking. In *IEEE International Conference on Computer Vision, ICCV Workshops*, 2019.

[108] Y. Zhang, D. Wang, L. Wang, J. Qi, and H. Lu. Learning Regression and Verification Networks for Long-term Visual Tracking. *CoRR*, abs/1809.04320, 2018.

[109] Z. Zhang and H. Peng. Deeper and wider siamese networks for real-time visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[110] P. Zhu, L. Wen, X. Bian, L. Haibin, and Q. Hu. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*, 2018.