The Eighth Visual Object Tracking VOT2020 Challenge Results

Matej Kristan[1], Aleš Leonardis[2], Jiří Matas[3], Michael Felsberg[4], Roman Pflugfelder[5,6], Joni-Kristian Kämäräinen[7], Martin Danelljan[8], Luka Čehovin Zajc[1], Alan Lukežič[1], Ondrej Drbohlav[3], Linbo He[4], Yushan Zhang[4,9], Song Yan[7], Jinyu Yang[2], Gustavo Fernández[5], Alexander Hauptmann[10], Alireza Memarmoghadam[39], Álvaro García-Martín[36], Andreas Robinson[4], Anton Varfolomieiev[25], Awet Haileslassie Gebrehiwot[36], Bedirhan Uzun[12], Bin Yan[11], Bing Li[18], Chen Qian[29], Chi-Yi Tsai[35], Christian Micheloni[43], Dong Wang[11], Fei Wang[29], Fei Xie[33], Felix Jaremo Lawin[4], Fredrik Gustafsson[44], Gian Luca Foresti[43], Goutam Bhat[8], Guangqi Chen[29], Haibin Ling[34], Haitao Zhang[46], Hakan Cevikalp[12], Haojie Zhao[11], Haoran Bai[32], Hari Chandana Kuchibhotla[17], Hasan Saribas[13], Heng Fan[34], Hossein Ghanei-Yakhdan[45], Houqiang Li[41], Houwen Peng[23], Huchuan Lu[11], Hui Li[19], Javad Khaghani[37], Jesus Bescos[36], Jianhua Li[11], Jianlong Fu[23], Jiaqian Yu[28], Jingtao Xu[28], Josef Kittler[42], Jun Yin[46], Junhyun Lee[21], Kaicheng Yu[16], Kaiwen Liu[18], Kang Yang[24], Kenan Dai[11], Li Cheng[37], Li Zhang[40], Lijun Wang[11], Linyuan Wang[46], Luc Van Gool[8], Luca Bertinetto[14], Matteo Dunnhofer[43], Miao Cheng[46], Mohana Murali Dasari[17], Ning Wang[24], Ning Wang[41], Pengyu Zhang[11], Philip H.S. Torr[40], Qiang Wang[26], Radu Timofte[8], Rama Krishna Sai Gorthi[17], Seokeon Choi[20], Seyed Mojtaba Marvasti-Zadeh[37], Shaochuan Zhao[19], Shohreh Kasaei[31], Shoumeng Qiu[30], Shuhao Chen[11], Thomas B. Schön[44], Tianyang Xu[42], Wei Lu[46], Weiming Hu[18,26], Wengang Zhou[41], Xi QIu[22], Xiao Ke[15], Xiao-Jun Wu[19], Xiaolin Zhang[30], Xiaoyun Yang[27], Xuefeng Zhu[19], Yingjie Jiang[19], Yingming Wang[11], Yiwei Chen[28], Yu Ye[15], Yuezhou Li[15], Yuncon Yao[33], Yunsung Lee[21], Yuzhang Gu[30], Zezhou Wang[11], Zhangyong Tang[19], Zhen-Hua Feng[42], Zhijun Mai[38], Zhipeng Zhang[18], Zhirong Wu[23], and Ziang Ma[46]

[1] University of Ljubljana, Slovenia
[2] University of Birmingham, United Kingdom
[3] Czech Technical University, Czech Republic
[4] Linköping University, Sweden
[5] Austrian Institute of Technology, Austria
[6] TU Wien, Austria
[7] Tampere University, Finland
[8] ETH Zürich, Switzerland
[9] Beijing Institute of Technology, China
[10] Carnegie Mellon University, USA
[11] Dalian University of Technology, China
[12] Eskisehir Osmangazi University, Turkey
[13] Eskisehir Technical University, Turkey
[14] Five AI, United Kingdom
[15] Fuzhou University, China
[16] High School Affiliated to Renmin University of China, China

[17] Indian Institute of Technology, India
[18] Institute of Automation, Chinese Academy of Sciences, China
[19] Jiangnan University, China
[20] KAIST, Korea
[21] Korea University, Korea
[22] Megvii, China
[23] Microsoft Research, USA
[24] Nanjing University of Information Science & Technology, China
[25] National Technical University of Ukraine, Ukraine
[26] NLP, China
[27] Remark Holdings, United Kingdom
[28] Samsung Research China-Beijing (SRC-B), China
[29] Sensetime, Hong Kong
[30] Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, China
[31] Sharif University of Technology, Iran
[32] Sichuan University, China
[33] Southeast University, China
[34] Stony Brook University, USA
[35] Tamkang University, Taiwan
[36] Universidad Autónoma de Madrid, Spain
[37] University of Alberta, Canada
[38] University of Electronic Science and Technology of China, China
[39] University of Isfahan, Iran
[40] University of Oxford, United Kingdom
[41] University of Science and Technology of China, China
[42] University of Surrey, United Kingdom
[43] University of Udine, Italy
[44] Uppsala University, Sweden
[45] Yazd University, Iran
[46] Zhejiang Dahua Technology, China

Abstract. The Visual Object Tracking challenge VOT2020 is the eighth annual tracker benchmarking activity organized by the VOT initiative. Results of 58 trackers are presented; many are state-of-the-art trackers published at major computer vision conferences or in journals in the recent years. The VOT2020 challenge was composed of five sub-challenges focusing on different tracking domains: (i) VOT-ST2020 challenge focused on short-term tracking in RGB, (ii) VOT-RT2020 challenge focused on "real-time" short-term tracking in RGB, (iii) VOT-LT2020 focused on long-term tracking namely coping with target disappearance and reappearance, (iv) VOT-RGBT2020 challenge focused on short-term tracking in RGB and thermal imagery and (v) VOT-RGBD2020 challenge focused on long-term tracking in RGB and depth imagery. Only the VOT-ST2020 datasets were refreshed. A significant novelty is introduction of a new VOT short-term tracking evaluation methodology, and introduction of segmentation ground truth in the VOT-ST2020 challenge – bounding boxes will no longer be used in the VOT-ST challenges. A new VOT Python toolkit that implements all these novelites was introduced. 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[47].

Keywords: Visual object tracking, performance evaluation protocol, state-of-the-art benchmark, RGB, RGBD, depth, RGBT, thermal imagery, short-term trackers, long-term trackers

1 Introduction

Visual object tracking remains a core computer vision problem and a popular research area with many open challenges, which has been promoted over the last decade by several tracking initiatives like PETS [86], CAVIAR[48], i-LIDS [49], ETISEO[50], CDC [20], CVBASE [51], FERET [58], LTDT [52], MOTC [39,66] and Videonet [53]. However, prior to 2013, a consensus on performance evaluation was missing, which made objective comparison of tracking results across papers impossible. In response, the VOT[47] initiative has been formed in 2013. The primary goal of VOT was establishing datasets, evaluation measures and toolkits as well as creating a platform for discussing evaluation-related issues through organization of tracking challenges. This lead to organization of seven challenges, which have taken place in conjunction with ICCV2013 (VOT2013 [36]), ECCV2014 (VOT2014 [37]), ICCV2015 (VOT2015 [35]), ECCV2016 (VOT2016 [34]), ICCV2017 (VOT2017 [33]), ECCV2018 (VOT2018 [32]) and ICCV2019 (VOT2019 [31]).

Initially the VOT considered 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. *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. In 2018, the VOT tracker categories were extended by an additional one: 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.

This paper presents the VOT2020 challenge, organized in conjunction with the ECCV2020 Visual Object Tracking Workshop, and the results obtained. Several novelties are introduced in VOT2020 with respect to VOT2019, which

[47] http://votchallenge.net

[48] http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1

[49] http://www.homeoffice.gov.uk/science-research/hosdb/i-lids

[50] http://www-sop.inria.fr/orion/ETISEO

[51] http://vision.fe.uni-lj.si/cvbase06/

[52] http://www.micc.unifi.it/LTDT2014/

[53] http://videonet.team

consider encoding of the ground truth target positions, performance measures and a complete re-implementation of the VOT toolkit in a widely-used programming language for easier tracker integration. In the following we overview the most closely related works, discuss issues with exisiting performance measures and point out the contributions of VOT2020.

1.1 Short-term tracker evaluation

Over the last eight years, the Visual Object Tracking initiative (VOT) has been gradually developing performance evaluation methodology with an overall guideline to develop interpretable measures that probe various tracking properties. At VOT inception in 2013, a simple evaluation protocol was popularized by OTB [77]. This 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. The overall performance is summarized by area-under-the-curve principle, which has been showed in [70,72] 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 [38].

Alternatively, based on the analysis later published in [72,38], VOT proposed two basic performance measures: accuracy and robustness. The goal was to promote trackers that well approximate the target position, and even more importantly, do not fail often. The first methodology introduced in VOT2013 [36] was based on ranking trackers along each measure and averaging the ranks. Due to a reduced interpretation power and dependency of ranks on the tested trackers, this approach was replaced in VOT2015 [35] by the expected average overlap measure (EAO), which principally combines the individual basic measures.

To provide 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, VOT introduced the so-called state-of-the-art bound (*SotA* bound). Any tracker exceeding this bound is considered state-of-the-art by the VOT standard.

While most of the tracking datasets [77,40,65,44,53,19,93,25,85,18,54] have partially followed the trend in computer vision of increasing the number of sequences, the VOT [36,37,35,38,34,33,32,31] datasets have been constructed with diversity in mind and were kept sufficiently small to allow fast tracker development-and-testing cycles. Several recent datasets [25,18] have adopted elements of the VOT dataset construction principles for rigorous tracker evaluation. In VOT2017 [33] a sequestered dataset was introduced to reduce the influence of tracker over-fitting without requiring to increase the public dataset size. 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.

In 2015, the VOT introduced a new short-term tracking challenge dedicated to tracking in thermal imagery. The VOT short-term performance evaluation methodology was used with the LTIR [2] dataset. The challenge gradually evolved into an RGB+Thermal short-term tracking and constructed a new dataset based on [43]. The targets were re-annotated by rotated bounding boxes using a semi-automatic protocol [3].

The VOT evaluation protocols have promoted development of robust short-term trackers. But with increased robustness of modern trackers, a drawback of the reset-based evaluation protocol has emerged. In the VOT performance evaluation protocol a tracker is initialized in the first frame and whenever the overlap between the reported and the ground truth target location (i.e., bounding box) falls to zero, a failure is detected and the tracker is reset a fixed number of frames later. The robustness is measured as the number of times the tracker is reset and the accuracy is the average overlap between the periods of successful tracking. This setup reflects the tracker performance in a practical application, where the task is to track the target throughout the sequence, either automatically, or by user intervention, i.e., a tracker reset. Furthermore, this approach enables utilization of all sequence frames in the evaluation.

However, a point of tracking failure will affect the point of reset (tracker re-initialization) and initialization points profoundly affect the tracking performance. With recent development of very robust trackers, the initialization points started to play a significant role in the final tracker ranking. In particular, we have noticed that initialization at some frame might result in another failure later on in the sequence, while initializing a few frames later might not. This allows a possibility (although not trivially) for fine-tuning the tracker to fail on more *favorable* frames and by that reducing the failure rate and increase the overall apparent robustness as measured by the reset-based protocol.

Another potential issue of the existing VOT reset-based protocol is the definition of a tracking failure. A failure is detected whenever the overlap between the prediction and ground truth falls to zero. Since resets directly affect the performance, a possible way to reduce the resets is to increase the predicted bounding box size, so to avoid the zero overlap. While we have not observed such *gaming* often, there were a few cases in the last seven challenges where the trackers attempted this and one of the trackers has been disqualified upon identifying the use of the bounding box inflation strategy. But some trackers did resort to reporting a slightly larger bounding box due to the strictness of the failure protocol – the tracker will be reset if the zero overlap is detected in a single frame, even if the tracker would have jumped right back on the target in the next frame. We call this a short-term failure and the current protocol does not distinguish between trackers robust to short-term failures and trackers that fail completely.

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. Early work [30,57] directly adapted existing object-detection measures precision, recall and F-measure based on 0.5

IoU (overlap) threshold and several authors [68,52] proposed a modification of the short-term average overlap measure. Valmadre et al. [28] introduced a measure that directly addresses the evaluation of the re-detection ability and most recently Lukežič et. al. [49] proposed *tracking* precision, *tracking* recall and *tracking* F-measure that do not depend on specifying the IoU threshold. Their primary measure, the tracking F-measure, reduces to a standard short-term measure (average overlap) when computed in a short-term setup, thus closing the gap between short- and log-term tracking measures. The measure is shown to be extremely robust and allows using a very sparse temporal target annotation, thus enabling using very long evaluation sequences at reduced annotation effort. For these reasons, the measures and the evaluation protocol from [49] were selected in 2018 as the main methodology for all VOT sub-challenges dealing with long-term trackers.

Several datasest have been proposed for RGB long-term tracking evaluation, starting with LTDT [52] and followed by [53,52,48,28,49]. The authors of [48] 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 and was selected in VOT2018 for the first VOT long-term tracking challenge and later replaced by the updated dataset from [49].

In 2019 VOT introduced another long-term tracking challenge to promote tracker operating with RGB and depth (RGBD). At the time, only two public datasets were available, namely the PTB [67] and STC [78], with relatively short sequences and limited range of scenes due to acquisition hardware restrictions. Recently, a more elaborate dataset called CDTB [46] was proposed, which contains many long sequences with many target disappearances, captured with a range of RGBD sensors both indoor and outdoor under various lighting conditions. This dataset was used in VOT2019 in the VOT-RGBD challenge.

1.3 The VOT2020 challenge

Since VOT2020 considers short-term as well as long-term trackers in separate challenges, we adopt the definitions from [49] to position the trackers on the short-term/long-term spectrum:

- **Short-term tracker** (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** (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** (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** (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 VOT2020 organizers. The participants were required to use the new Python VOT toolkit that implements the new evaluation protocols and the new ground truth encoding. A toolkit beta testing period opened in early March 2020, and the challenge officially opened on March 20th 2020 with approximately a month available for results submission. Due to Covid-19 crisis, the VOT-RGBT team could not complete all the preparations in time and has decided to postpone the opening of the VOT-RGBT2020 sub-challenge to May 10th. The results submission deadline for all sub-challenges was May 3rd. The VOT2020 challenge thus contained five challenges:

1. **VOT-ST2020 challenge**: This challenge was addressing short-term tracking in RGB images and has been running since VOT2013 with annual updates and modifications. A significant novelty compared to 2019 was that the target position was encoded by a segmentation mask.
2. **VOT-RT2020 challenge**: This challenge addressed the same class of trackers as VOT-ST2020, except that the trackers had to process the sequences in real-time. The challenge was introduced in VOT2017. A significant novelty compared to 2019 was that the target position was encoded by a segmentation mask.
3. **VOT-LT2020 challenge**: This challenge was addressing long-term tracking in RGB images. The challenge was introduced in VOT2018. The target positions were encoded by bounding boxes.
4. **VOT-RGBT2020 challenge**: This challenge was addressing short-term tracking in RGB+thermal imagery. This challenge was introduced in VOT2019 and can be viewed as evolution of the VOT-TIR challenge introduced in VOT2015. The target positions were encoded by bounding boxes.
5. **VOT-RGBD2020 challenge**: This challenge was addressing long-term tracking in RGB+depth (RGBD) imagery. This challenge was introduced in VOT2019. The target positions were encoded by bounding boxes.

The authors participating in the challenge were required to integrate their tracker into the new VOT2020 evaluation kit, which automatically performed a set of standardized experiments. The results were analyzed according to the VOT2020 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. The committee reserved the right to disqualify any tracker that, by their judgement, attempted to cheat the evaluation protocols.

To compete for the winner of VOT2020 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 in VOT2019, while the remaining 9k+ sequences were allowed for learning. The reason was that part of the GOT10k was used for VOT-ST2020 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 VOT2020 were allowed to participate in the challenge, but did not compete for the winner titles. Further details are available from the challenge homepage[54].

VOT2020 goes beyond previous challenges by updating the datasets in VOT-ST, VOT-RT, challenges, as well as introduction of the segmentation ground truth. New performance evaluation protocol and measures were used in the short-term tracking challenges and the new VOT2020 Python toolkit was developed that implements all the novelties and ensures seamless use of challenge-specific modalities and protocols.

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. Section 2.1 introduces the new short-term performance evaluation protocol and measures, while the standard VOT long-term tracking evaluation protocol is overviewed in Section 2.2.

[54] http://www.votchallenge.net/vot2020/participation.html

2.1 The new anchor-based short-term tracking evaluation protocol

The main drawback of the existing VOT short-term performance evaluation protocol are the tracker-dependent resets, which induce a causal correlation between the first reset and the later ones. To avoid this, the notion of reset is replaced by *initialization points* (called *anchors* for short), which are made equal for all trackers in the new protocol. In particular, anchors are placed on each sequence Δ_{anc} frames apart, with the first and last anchor on the first and the last frame, respectively. A tracker is run from *each* anchor forward or backward in the sequences, whichever direction yields the longest sub-sequence. For example, if the anchor is placed before the middle of the sequence, the tracker is run forward, otherwise backward in the sequence. Each anchor is manually checked and potentially moved by a few frames to avoid placing the initialization point on an occluded target. Figure 1 shows example of the anchor placement and the tracking direction.

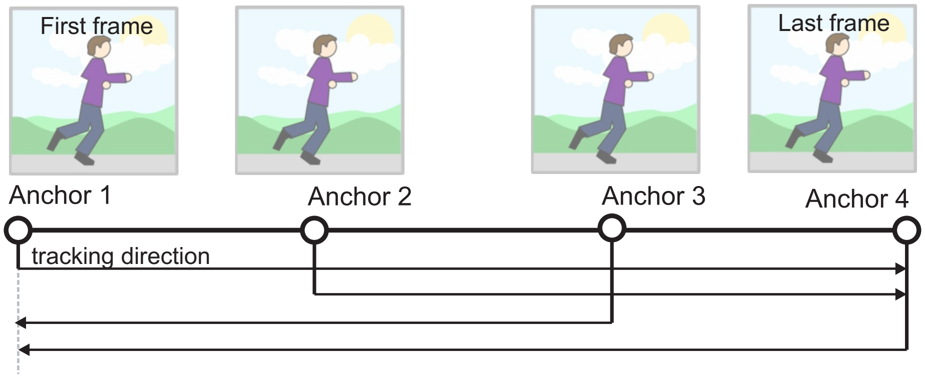

Fig. 1. Anchors are placed 50 frames apart. At each anchor the tracker is initialized and tracks in the direction that yields the longest subsequence.

The distance between the anchors was set to $\Delta_{\text{anc}} = 50$. At approximately 25 frames per second, this amounts to 2 second distances. We have experimentally tested that this value delivers stable results for the measures described in the next section computed on typical-length short-term sequences, while keeping the computational complexity of the evaluation at a moderate level.

Like in previous VOT challenges, we use the accuracy and robustness as the basic measures to probe tracking performance and the overall performance is summarized by the expected average overlap (EAO). In the following we redefine these in the context of the new anchor-based evaluation protocol.

The new accuracy and robustness measures. On a subsequence starting from an anchor a of sequence s, the accuracy $A_{s,a}$ is defined as the average overlap between the target predictions and the ground truth calculated from the frames before the tracker fails on that subsequence, i.e.,

$$A_{s,a} = \frac{1}{N^F_{s,a}} \sum\nolimits_{i=1:N^F_{s,a}} \mathit{\Omega}_{s,a}(i), \tag{1}$$

where $N^F_{s,a}$ is the number of frames before the tracker failed in the subsequence starting at anchor a in the sequence s (see Section 2.1 for the failure definition) and $\mathit{\Omega}_{s,a}(i)$ is the overlap between the prediction and the ground truth at frame i. The new robustness measure $R_{s,a}$ is defined as the *extent* of the sub-sequence before the tracking failure, i.e.,

$$R_{s,a} = N^F_{s,a}/N_{s,a}, \tag{2}$$

where $N_{s,a}$ is the number of frames of the subsequence.

The results from the sub-sequences are averaged in a weighted fashion such that each sub-sequence contributes proportionally to the number frames used in calculation of each measure. In particular, the per-sequence accuracy and robustness are defined as

$$A_s = \frac{1}{\sum_{a=1:N^A_s} N^F_{s,a}} \sum\nolimits_{a=1:N^A_s} A_{s,a} N^F_{s,a}, \tag{3}$$

$$R_s = \frac{1}{\sum_{a=1:N^A_s} N_{s,a}} \sum\nolimits_{a=1:N^A_s} R_{s,a} N_{s,a}, \tag{4}$$

where N^A_s is the number of anchors in the sequence s. The overall accuracy and robustness are calculated by averaging the per-sequence counterparts proportionally to the number of frames used for their calculation, i.e.,

$$A = \frac{1}{\sum_{s=1:N} N^F_s} \sum\nolimits_{s=1:N} A_s N^F_s, \tag{5}$$

$$R = \frac{1}{\sum_{s=1:N} N_s} \sum\nolimits_{s=1:N} R_s N_s, \tag{6}$$

where N is the number of sequences in the dataset, N_s is the number of frames in sequence s and $N^F_s = \sum_{a=1:N^A_s} N^F_{s,a}$ is the number of frames used to calculate the accuracy in that sequence.

The new EAO measure. As in previous VOT challenges, the accuracy and robustness are principally combined into a single performance score called the expected average overlap (EAO). We use the same approach as in the previous VOT challenges, i.e., the expected average overlap curve is calculated and averaged over an interval of typical short-term sequence lengths into the EAO measure.

Note that the computation considers virtual sequences of overlaps generated from the sub-sequence results. In particular, if a tracker failed on a sub-sequence

(s, a), the overlap falls to zero at the failure frame, and the overlaps can be extended to i-th frame by zeros, even if i exceeds the sub-sequence length. But if the tracker did not fail, the overlaps cannot be extrapolated beyond the original sub-sequence length.

The value of the EAO curve $\hat{\Phi}_i$ at sequence length i is thus defined as

$$\hat{\Phi}_i = \frac{1}{|\mathcal{S}(i)|} \sum_{s,a \in \mathcal{S}(i)} \Phi_{s,a}(i), \tag{7}$$

where $\Phi_{s,a}(i)$ is the average overlap calculated between the first and i-th frame of the extended sub-sequence starting at anchor a of sequence s, $\mathcal{S}(i)$ is the set of the extended sub-sequences with length greater or equal to i and $|\mathcal{S}(i)|$ is the number of these sub-sequences.

The EAO measure is then calculated by averaging the EAO curve from N_{lo} to N_{hi}, i.e.,

$$EAO = \frac{1}{N_{\mathrm{hi}} - N_{\mathrm{lo}}} \sum\nolimits_{i=N_{\mathrm{lo}}:N_{\mathrm{hi}}} \hat{\Phi}_i. \tag{8}$$

Similarly to VOT2015 [35], the interval bounds $[N_{\mathrm{lo}}, N_{\mathrm{hi}}]$ were determined from the mean ± one standard deviation of the anchor-generated sub-sequences.

Failure definition. The tracking failure event is also redefined to (i) reduce the potential for the *gaming*, i.e., outputting the entire image as the prediction to prevent failure detection during an uncertain tracking phase, and (ii) allow for recovery from short-term tracking failures. A *tentative* failure is detected when the overlap falls below a non-zero threshold θ_Φ. The non-zero threshold punishes an actual drift from the target as well as speculation by outputting a very large bounding box to prevent failure detection. If a tracker does not recover within the next θ_N frames, i.e., the overlap does increase to over θ_Φ, a failure is detected. Note that in some rare situations a frame might contain the target fully occluded. Since short-term trackers are not required to report target disappearance, these frames are ignored in tracking failure detection.

By using several well-known trackers from different tracker design classes we experimentally determined that the threshold values $\theta_\Phi = 0.1$ and $\theta_N = 10$ reduce the *gaming* potential, allow recoveries from short-term failures, while still penalizing the trackers that fail more often.

Per-attribute analysis. Per-attribute accuracy and robustness are computed by accounting for the fact that the attributes are not equally distributed among the sequences and that the attribute at a frame may affect the tracker performance a few frames later in the sequence. Thus the two per-attribute measures $(A_{\mathrm{atr}}, R_{\mathrm{atr}})$ are defined as weighted per-sequence measures with weights proportional to the amount of attribute in the sequence, i.e.,

$$A_{\mathrm{atr}} = \frac{1}{\sum_{s=1:N} N_s^{\mathrm{atr}}} \sum\nolimits_{s=1:N} N_s^{\mathrm{atr}} A_s, \tag{9}$$

$$R_{\mathrm{atr}} = \frac{1}{\sum_{s=1:N} N_s^{\mathrm{atr}}} \sum\nolimits_{s=1:N} N_s^{\mathrm{atr}} R_s, \tag{10}$$

where N_s^{atr} is the number of frames with attribute "atr" in a sequence s.

2.2 The VOT long-term 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 [49] first used in the VOT2018 is adapted.

Three long-term tracking performance measures proposed in [48] 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 [48] 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 τ_θ, 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

$$F(\tau_\theta) = 2Pr(\tau_\theta)Re(\tau_\theta)/(Pr(\tau_\theta) + Re(\tau_\theta)). \tag{11}$$

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 τ_θ [48]. The final values of Pr, Re and F-measure are obtained by selecting τ_θ 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 (11) 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 VOT2020 challenge.

3.1 VOT-ST2020 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.

The dataset. Results of the VOT2019 showed that the dataset was not saturated [31], and the public dataset has been refreshed by replacing one sequence (see Figure 2.) A single sequence in the sequestered dataset has been replaced as well to calibrate the attribute distribution between the two datasets. Following the protocols from VOT2019, the list of 1000 diverse sequences[55] from the GOT-10k [25] training set was used. The sequence selection and replacement procedure followed that of VOT2019. In addition, object category and motion diversity was ensured by manual review.

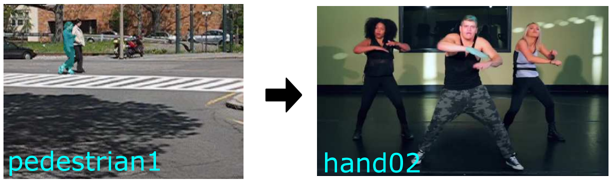

Fig. 2. The `pedestrian1` sequence of the VOT2019 public dataset has been replaced by a more challenging `hand02` sequence for VOT2020.

The bounding boxes are no longer used in the VOT-ST/RT tracking subchallenges. The target position is now encoded by the segmentation masks. Since 2016, VOT has already been using segmentation masks for fitting the rotated bounding box ground truth in the previous years. However, a closer inspection revealed that while these masks were valid for fitting rectangles, their accuracy was insufficient for segmentation ground truth. Thus *the entire dataset* (public and sequestered) was re-annotated. The initial masks were obtained by a semi-automatic method and then all sequences were frame-by-frame manually corrected. Examples of segmentation masks are shown in Figure 3.

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 EAO interval bounds in (8) were estimated to be $[N_{\text{lo}}, N_{\text{hi}}] = [115, 755]$ on the public VOT-ST2020 dataset.

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

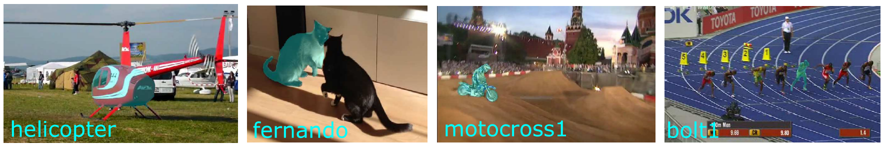

Fig. 3. Images from the VOT-ST2020 sub-challenge with segmentation masks superimposed (in cyan).

Winner identification protocol. The VOT-ST2020 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 VOT2020 committee on the sequestered dataset. The top ranked tracker on the sequestered dataset not submitted by the VOT2020 committee members was the winner of the VOT-ST2020 challenge.

3.2 VOT-RT2020 challenge outline

This challenge addressed *real-time* RGB tracking in a short-term tracking setup. The dataset was the same as in the VOT-ST2020 challenge, but the evaluation protocol was modified to emphasize the real-time component in tracking performance. In particular, the VOT-RT2020 challenge requires predicting bounding boxes faster or equal to the video frame-rate. The toolkit sends images to the tracker via the Trax protocol [71] 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). The same performance evaluation protocol as in VOT-ST2020 is then applied.

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 VOT2020 committee members.

3.3 VOT-LT2020 challenge outline

This challenge addressed RGB tracking in a long-term tracking setup and is a continuation of the VOT-LT2019 challenge. We adopt the definitions from [48], 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-LT2020.

Trackers were evaluated on the LTB50 [49], the same dataset as used in VOT-LT2019. 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 × 720 and 290 × 217.

Each sequence contains on average 10 long-term target disappearances, each lasting on average 52 frames.

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. Please see [49] for more details.

Winner identification protocol. The VOT-LT2020 winner was identified as follows. Trackers were ranked according to the tracking F-score on the LTB50 dataset (no sequestered dataset available). The top ranked tracker on the dataset not submitted by the VOT2020 committee members was the winner of the VOT-LT2020 challenge.

3.4 VOT-RGBT2020 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.

Trackers were evaluated on the VOT-RGBT2019 dataset, derived from [43], but extended with anchor frames for the new re-initialization approach. The VOT-RGBT2019 dataset contains 60 public and 60 sequestered sequences containing partially aligned RGB and thermal images. The longest three sequences in the sequestered dataset are among the simpler ones and were sub-sampled by factor five to avoid a positive bias in the EAO measure. Due to acquisition equipment, the RGB and thermal channels are slightly temporally de-synchronized, which adds to the challenge in RGBT tracking. All frames are annotated with the attributes (i) occlusion, (ii) dynamics change, (iii) motion change, (iv) size change, and (v) camera motion. Due to the slight temporal de-synchronization and the partially very small objects, the consistency and accuracy of segmentation masks was not sufficient for unbiased tracker evaluation. A decision was thus made to use rotated bounding boxes already created for the VOT-RGBT2019 dataset instead. Examples of images from the dataset are shown in Figure 4.

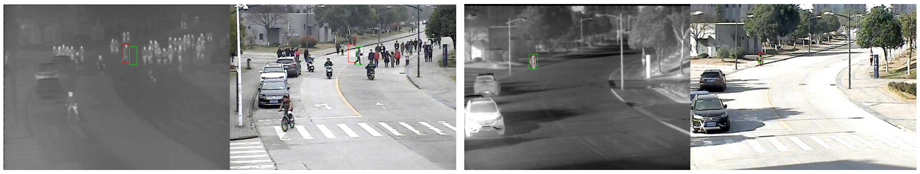

Fig. 4. Example images from the VOT-RGBT2020 dataset. The left two frames illustrate the synchronization issue in the RGBT234 dataset [43] and the right two frames the small object issue.

Winner identification protocol. The VOT-RGBT2020 winner has been identified as follows. Trackers were ranked according to the EAO measure on the public VOT-RGBT2020 dataset. The top five trackers have then been re-run by the VOT2020 committee on the sequestered VOT-RGBT dataset. The top ranked tracker on the sequestered dataset not submitted by the VOT2020 committee members was the winner of the VOT-RGBT2020 challenge.

3.5 VOT-RGBD2020 challenge outline

This challenge addressed long-term trackers using the RGB and depth channels (RGBD). The long-term performance evaluation protocol from Section 2.2 was used. The VOT-RGBD2020 trackers were evaluated on the CDTB dataset described in detail in [46]. The dataset contains 80 sequences acquired with three different sensor configurations: 1) a single Kinect v2 RGBD sensor, 2) a combination of the Time-of-Flight (Basler tof640) and 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. The dataset contains tracking of various household and office objects (Figure 5). The sequences contain target in-depth rotations, occlusions and disappearance that are challenging for only RGB and depth-only trackers. The total number of frames is 101,956 in various resolutions. For more details, see [46].

Winner identification protocol. The VOT-RGBD2020 winner was identified as follows. Trackers were ranked according to the F-score on the public VOT-RGBD2020 dataset (no sequestered dataset available). The reported numbers were computed using the submitted results, but the numbers were verified by re-running the submitted trackers multiple times. The top ranked tracker not submitted by the VOT2020 committee members was the winner of the VOT-RGBD2020 challenge.

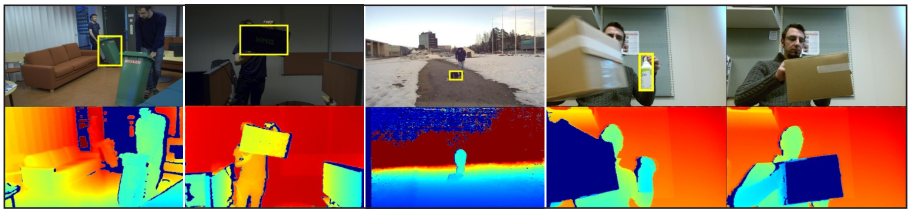

Fig. 5. RGB and depth (D) frames from the VOT-RGBD dataset.

4 The VOT2020 challenge results

This section summarizes the trackers submitted, results analysis and winner identification for each of the five VOT2020 challenges.

4.1 The VOT-ST2020 challenge results

Trackers submitted In all, 28 valid entries were submitted to the VOT-ST2020 challenge. Each submission included the binaries or source code that allowed verification of the results if required. The VOT2020 committee and associates additionally contributed 9 baseline trackers. For these, the default parameters were selected, or, when not available, were set to reasonable values. Thus in total 37 trackers were tested on VOT-ST2020. 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, 17 trackers (46%) were categorized as ST_0, 18 trackers (49%) as ST_1 and 2 as LT_0. 92% applied discriminative and 8% applied generative models. Most trackers (95%) used holistic model, while 5% of the participating trackers used part-based models. Most trackers applied a locally uniform dynamic model[56] or a random walk dynamic model (95%) and only (5%) applied a nearly-constant-velocity dynamic model. 38% of trackers localized the target in a single stage, while the rest applied several stages, typically involving approximate target localization and position refinement. Most of the trackers (86%) use deep features, which shows that the field has moved away from using hand-crafted features, which were still widely used on their own or in combination with the deep features even a few years ago. 54% of these trackers re-trained their backbone on tracking or segmentation/detection datasets.

A particular novelty of the VOT-ST2020 is that target location ground truth is encoded as a segmentation mask. We observe a strong response in the VOT community to this: 57% of trackers reported target position as a segmentation mask, while the rest (43%) reported a bounding box. Among the segmentation trackers, 5 apply a deep grab-cut-like segmentation [79], 3 apply a nearest-neighbor segmentation akin to [50] and 13 apply patch-based Siamese segmentation akin to [74].

The trackers were based on various tracking principles. The two dominant tracking methodologies are discriminative correlation filters[57] (used in 68% of all submissions) and Siamese correlation networks, e.g. [4,41,74], (used in 46% of all submissions). 15 trackers were based only on DCFs (DET50 A.6, TRASTmask A.9, TRASFUSTm A.11, DPMT A.13, TRAT A.19, DiMP A.21, SuperDiMP A.22, LWTL A.23, TCLCF A.25, AFOD A.26, FSC2F A.27, KCF A.33,

[56] 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.

[57] This includes standard FFT-based as well as more recent deep learning based DCFs (e.g., [13,5]).

CSRpp A.31, ATOM A.30, UPDT A.37). 10 trackers were based only on Siamese correlation networks (DCDA A.3, igs A.4, SiamMaskS A.5, VPUSiamM A.7, Ocean A.10, Siammask A.14, SiamMargin A.17, SiamEM A.18, AFAT A.24, SiamFc A.35). 6 trackers applied a combination of DCFs and Siamese networks (A3CTDmask A.2, RPT A.8, AlphaRef A.12, OceanPlus A.15, fastOcean A.16, DESTINE A.28), one tracker combined a DCF with nearest-neighbor deep feature segmentation D3S A.1. One tracker was based on generative adversarial networks InfoVital A.20, one entry was a state-of-the-art video segmentation method STM A.36, one entry was a subspace tracker (IVT A.32), one used multiple-instance learning (MIL A.34) and one entry was a scale-adaptive mean-shift tracker (ASMS A.29).

Results The results are summarized in the AR-raw plots and EAO plots in Figure 6, and in Table 8. The top ten trackers according to the primary EAO measure (Figure 6) are RPT A.8, OceanPlus A.15, AlphaRef A.12, AFOD A.26, LWTL A.23, fastOcean A.16, TRASTmask A.9, DET50 A.6, D3S A.1 and Ocean A.10. All of these trackers apply CNN features for target localization. Nine (RPT, OceanPlus, AlphaRef, AFOD, LWTL, fastOcean, DET50, D3S, TRASTmask) apply a deep DCF akin to [13] (many of these in combination with a Siamese correlation net – RPT, OceanPlus, AlphaRef), while Ocean applies a Siamese correlation network without a DCF. All trackers provide the target location in form of a segmentation mask. Most trackers localize the target in multiple stages, except for AFOD, LWTL and D3S, which produce the target mask in a single stage. Several methods apply deep DCFs such as ATOM/DiMP [13,5] for target localization, bounding box estimation by region proposals [41] or fully-convolutional [69,84], while the final segmentation draws on approaches from [74,50,64]. A ResNet50 backbone pre-trained on general datasets is used in all top 10 trackers.

The top performer on the public dataset is RPT A.8. This is a two-stage tracker. The first stage combines the response of a fully-convolutional region proposal RepPoints [84] with a deep DCF DimP [5] response for initial bounding box estimation. In the second stage, a single-shot segmentation tracker D3S [50] is applied on the estimated target bounding box to provide the final segmentation mask. D3S appears to be modified by replacing the target presence map in geometrically constrained model by the more robust output from the approximate localization stage. This tracker significantly stands out from the rest according to the EAO measure.

The second-best ranked tracker is OceanPlus A.15. This is a multi-stage tracker based on Siamese region proposal nets SiamDW[91] (a top-performer of several VOT2019 sub-challenges) that matches template features in three parallel branches with various filter dilation levels. Fused outputs are used to predict the target bonding box akin to Fcos [69] and DiMP [5] is applied for increased robustness. Attention maps akin to TVOS [89] are computed and a UNet-like architecture [61] is then applied to fuse it with the correlation features

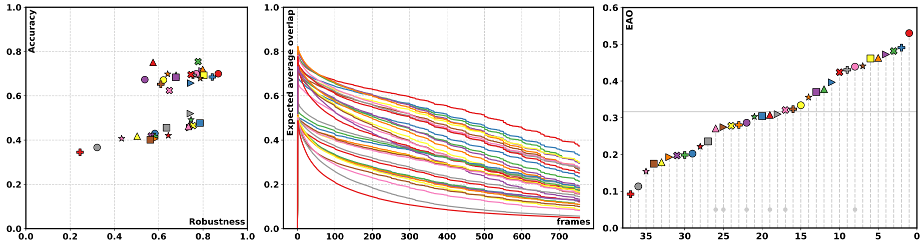

Fig. 6. The VOT-ST2020 AR-raw plots generated by sequence pooling (left) and EAO curves (center) and the VOT-ST2020 expected average overlap graph with trackers ranked from right to left. The right-most tracker is the top-performing according to the VOT-ST2020 expected average overlap values. The dashed 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.

into the final segmentation. The tracker shows a comparable robustness to the top performer.

The third top-performing tracker is AlphaRef A.12. This is a two-stage tracker that applies DiMP [5] to localize the target region. The region is then passed to a refine network akin to the one used in [60] to merge pixel-wise correlation and non-local layer outputs into the final segmentation mask.

The top-three trackers stand out from the rest in different performance measures. RPT and OceanPlus are much more robust than the other trackers, meaning that they remain on the target for longer periods. Combined with their very accurate target segmentation mask estimation, they achieve a top EAO. The third tracker, AlphaRef, also obtains a very high EAO, but not due to robustness – its robustness is actually lower than a lower-ranked fastOcean. The very high EAO can be attributed to the high accuracy. This tracker achieves a remarkable 0.753 localization accuracy, meaning that the segmentation masks are of much higher quality than the competing trackers whenever the target is well localized. From the submitted tracker descriptions, we can speculate that all three trackers with top accuracy (AlphaRef, AFOD and LWTL) apply similar approaches for segmentation. This comes at a cost of a reduced robustness of several percentage points compared to the two top EAO performers.

Since the VOT-ST2020 challenge has shifted toward target localization by segmentation, the VOT committee added a recent state-of-the-art video object segmentation (VOS) method STM A.36 [56] (2019) as a strong VOS baseline. Small modifications were made like rescaling the input to a fixed resolution to allow running on longer sequences with smaller targets than those typical for VOS challenge. Interstingly STM is ranked 19th, outperforming state-of-the-art bounding-box-based trackers such as ATOM [13] and DiMP [5]. In fact, STM achieves a second-best segmentation accuracy among all submissions and runs with decent robustness – for example, it outperforms an improved bounding-box tracker SuperDiMP. These results show a great tracking potential for the video object segmentation methods.

The trackers which have been considered as baselines or state-of-the-art a few years ago (e.g., SiamFc, KCF, IVT, CSRpp, ASMS) are positioned at the lower part of the AR-plots and at the tail of the EAO rank list, and even some of the recent state-of-the-art like ATOM [13] and DiMP [5] are ranked in the lower third of the submissions. This is a strong indicator of the advancements made in the field. Note that 6 of the tested trackers have been published in major computer vision conferences and journals in the last two years (2019/2020). These trackers are indicated in Figure 6, along with their average performance (EAO= 0.3173), which constitutes the VOT2020 state-of-the-art bound. Approximately 46% 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**	**OC**	**SC**	**MC**
Accuracy	0.53③	0.54	0.45①	0.54	0.51②
Robustness	0.70	0.77	0.60①	0.69	0.63②

Table 1. VOT-ST2020 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 per-attribute robustness analysis is shown in Figure 7 for individual trackers. The overall top performers remain at the top of per-attribute ranks as well. None of the trackers consistently outperforms all others on all attributes, but RPT is consistently among the top two trackers. According to the median failure over each attribute (Table 1) the most challenging attributes remain occlusion and motion change as in VOT2019. The drop on these two attributes is consistent for all trackers (Figure 7). Illumination change, motion change and scale change are challenging, but comparatively much better addressed by the submitted trackers.

The VOT-ST2020 challenge winner Top five trackers from the baseline experiment (Table 8) were re-run on the sequestered dataset. Their scores obtained on sequestered dataset are shown in Table 2. The top tracker according to the EAO is RPT A.8 and is thus the VOT-ST2020 challenge winner.

4.2 The VOT-RT2020 challenge results

Trackers submitted The trackers that entered the VOT-ST2020 challenge were also run on the VOT-RT2019 challenge. Thus the statistics of submitted trackers was the same as in VOT-ST2020. For details please see Section 4.1 and Appendix A.

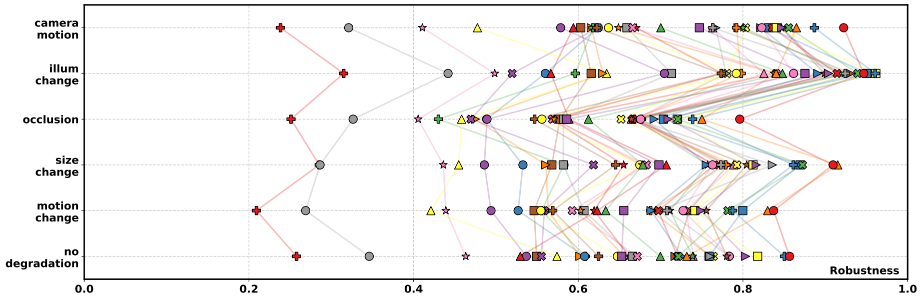

Fig. 7. Robustness with respect to the visual attributes.

	Tracker	EAO	A	R
1.	RPT	0.547①	0.766	0.850
2.	AFOD	0.536②	0.795	0.816
3.	LWTL	0.526③	0.781	0.822
4.	OceanPlus	0.513	0.760	0.818
5.	AlphaRef	0.510	0.823	0.762

Table 2. The top five trackers from Table 8 re-ranked on the VOT-ST2020 sequestered dataset.

Results The EAO scores and AR-raw plots for the real-time experiments are shown in Figure 8 and Table 8. The top ten real-time trackers are AlphaRef A.12, OceanPlus A.15, AFOD A.26, fastOcean A.16, Ocean A.10, D3S A.1, AFAT A.24, SiamMargin A.17, LWTL A.23 and TRASTmask A.9.

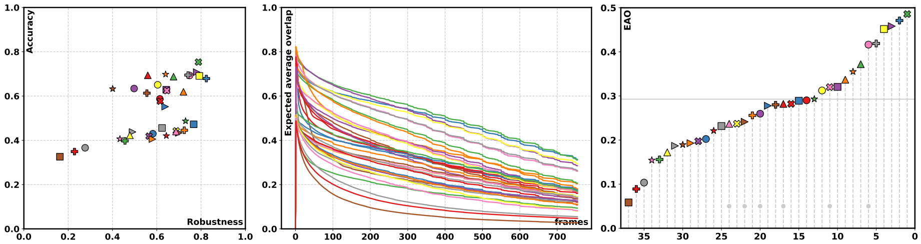

Fig. 8. The VOT-RT2020 AR plot (left), the EAO curves (center) and the EAO plot (right).

The top three trackers, AlphaRef, OceanPlus and AFOD are ranked 3rd, 2nd and 4th on the VOT-ST2020 challenge, respectively. These, in addition to fastOcean stand out from the rest in EAO, owing this to an excellent robustness. AlphaRef has slightly lower robustness than OceanPlus, but a much better accuracy, which results in a higher EAO.

Astonishingly, 8 out of 10 top real-time trackers are among the top ten performers on VOT-ST challenge. This is in stark contrast to the previous years, where most of the top performers from VOT-ST challenge substantially dropped in ranks under the realtime constraint. The two additional trackers among to 10 are SiamMargin and TRASTmask. SiamMargin is the VOT-RT2019 challenge winner based on SiamRPN++[41], while TRASTmask is a teacher-student network that uses DiMP [5] for the teacher for bounding box prediction. Both trackers apply SiamMask [74] for final segmentation.

Seven trackers (AlphaRef, OceanPlus, AFOD, fastOcean, Ocean, D3S and AFAT) outperform the VOT-RT2019 winner SiamMargin, which shows that the real-time performance bar has been substantially pushed forward this year. The tracking speed obviously depends on the hardware used, but overall, we see emergence of deep tracking architectures that no longer sacrifice speed and computational efficiency for performance (or vice versa).

Like in VOT-ST2020 challenge, 6 of the tested trackers have been published in major computer vision conferences and journals in the last two years (2019/2020). These trackers are indicated in Figure 8, along with their average performance (EAO= 0.2932), which constitutes the VOT2020 realtime state-of-the-art bound. Approximately 32% of submitted trackers exceed this bound, which is slightly lower than in the VOT-ST2020 challenge.

The VOT-RT2020 challenge winner According to the EAO results in Table 8, the top performer and the winner of the real-time tracking challenge VOT-RT2020 is AlphaRef (A.12).

4.3 The VOT-LT2020 challenge results

Trackers submitted The VOT-LT2020 challenge received 5 valid entries. The VOT2020 committee contributed additional three top performers from VOT-LT2019 as baselines, thus 8 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 available.

All participating trackers were categorized as LT_1 according to the ST-LT taxonomy from Section 1.3 in that they implemented explicit target re-detection. All methods are based on convolutional neural networks. Several methods are based on region proposal networks akin to [41] for approximate target localization at detection stage (Megtrack B.1, SPLT B.2, LTDSE B.7) and several approaches apply the MDNet classifier [55] for target presence verification (LTMUB B.3, ltMDNet B.5, CLGS B.6). One tracker is based purely on a deep DCF and applies the DCF for localization as well as for the detection module (RLTDiMP B.4) and one tracker applies an ensamble for improved robustness and accuracy (SiamDWLT B.8). Four trackers update their short-term and long-term visual models only when confident (Megtrack, LTMUB, RLTDiMP, LTDSE), while SPLT never updates the visual models, LTMDNet updates the short-term visual model at fixed intervals, but keeps the long-term model fixed, CLGS never updates the short-term model and updates the long-term model at fixed intervals, and SiamDWLT applies PN learning to update both visual models.

Tracker	Pr	Re	F-Score
●LT_DSE	0.715②	0.677③	0.695①
✚LTMU_B	0.701	0.681②	0.691②
✖Megtrack	0.703③	0.671	0.687③
▶CLGS	0.739①	0.619	0.674
▲RLT_DiMP	0.657	0.684①	0.670
■SiamDW_LT	0.678	0.635	0.656
★ltMDNet	0.649	0.514	0.574
●SPLT	0.587	0.544	0.565

Table 3. List of trackers that participated in the VOT-LT2020 challenge along with their performance scores (Pr, Re, F-score) and ST/LT categorization.

Results The overall performance is summarized in Figure 9 and Table 3. The top-three performers are LTDSE B.7, LTMUB B.3 and Megtrack B.1. LTDSE is the winner of the VOT-LT2019 challenge as was included by the VOT2020 committee as a strong baseline. This tracker applies a DCF [13] short-term tracker on top of extended ResNet18 features for initial target localization. The target position is refined by a SiamMask [74] run on the target initial position. The target presence is then verified by RT-MDNet [29]. If the target is deemed absent, an image-wide re-detection using a region proposal network akin to MBMD [90] is applied. The region proposals are verified by the online trained verifier.

LTMUB architecture is composed of a local tracker, verifier, global detector and meta-updater. Similarly to LTSDE, the short-term tracker is a combination of DiMP [5] and SiamMask [74]. Adaptation of the MDNet [55] is used for a verifier, an LSTM-absed meta updater from [11] to decide whether to update and [26] is used for image-wide re-detection. According to the authors, the method can be thought of as a simplified version of LTDSE.

Megtrack architecture applies a short-term tracker composed of ATOM [13] and SiamMask [74] for inter-frame target localization and presence verification. Target re-detection is performed by GlobalTrack [26] within a gradually increasing search region and verified using a combination of online-learned real-time MDNet [29] and offline-learned one-shot matching module. Short-term tracker is re-initialized on the re-detected target.

LTDSE achieves an overall best F-measure and slightly surpasses LTMUB (by 0.6%). LTDSE has a better precision, meaning that its target detections

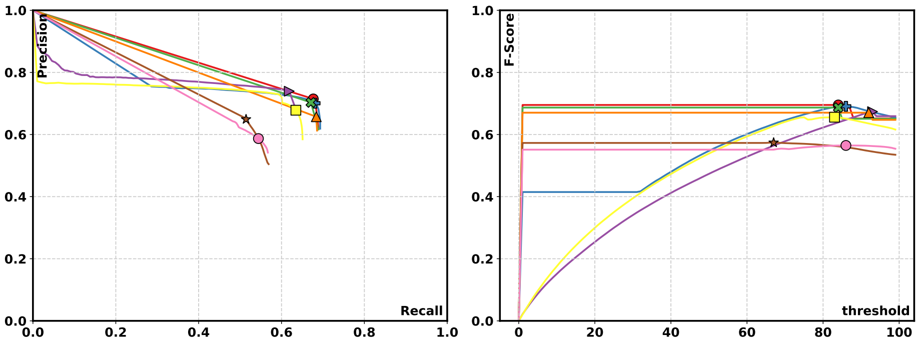

Fig. 9. VOT-LT2020 challenge average tracking precision-recall curves (left), the corresponding F-score curves (right). Tracker labels are sorted according to maximum of the F-score.

more reliably contain the target. On the other hand, LTMUB recovers more target positions, but at a cost of a reduced precision.

Figure 10 shows tracking performance with respect to nine visual attributes from Section 3.3. The most challenging attributes are fast motion, partial and full occlusion and target leaving the field of view (out-of-view attribute).

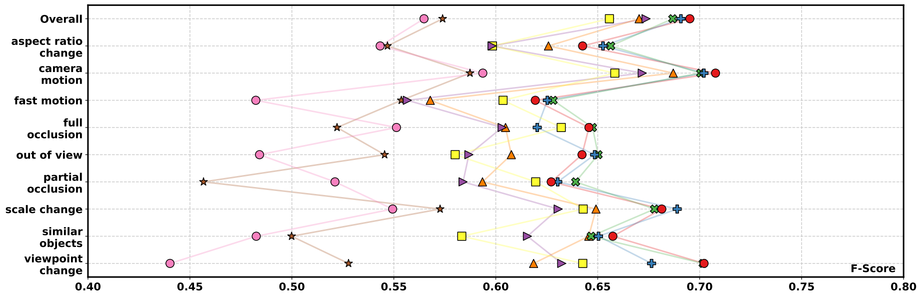

Fig. 10. VOT-LT2020 challenge maximum F-score averaged over overlap thresholds for the visual attributes. The most challenging attributes are partial and full occlusion and out-of-view.

The VOT-LT2020 challenge winner According to the F-score, the top-performing tracker is LTDSE, closely followed by LTMUB. LTDSE was provided by the VOT committee as a baseline tracker and as such does not compete for the winner of the VOT-LT2020. Thus the winner of the VOT-LT2020 challenge is LTMUB B.3.

4.4 The VOT-RGBT2020 challenge results

Trackers submitted In all, 5 entries were submitted to the VOT-RGBT2020 challenge. All submissions included the source code that allowed verification of the results if required. Two additional trackers were contributed by the VOT committee: mfDiMP C.6 [87] and SiamDW-T C.7. Thus in total 7 trackers were compared on VOT-RGBT2020. In what follows we briefly overview the entries and provide the references to original papers in the Appendix C where available.

All five submitted trackers use discriminative models with a holistic representation. 2 trackers (40%) were categorized as ST_1 and 3 trackers (60%) as ST_0. All 5 trackers applied a locally uniform dynamic model.

The trackers were based on various tracking principles: 4 trackers (80%) are single-stage trackers based on discriminative correlation filters (M2C2Frgbt C.1, JMMAC C.2, AMF C.3, and SNDCFT C.4) and 1 tracker (20%) is a multi-stage tracker based on a Siamese network (DFAT C.5). Respectively 1 tracker (20%) makes use of subspace methods (M2C2Frgbt C.1) and RANSAC (JMMAC C.2). Most of the trackers (80%) use deep features, only M2C2Frgbt C.1 uses hand-crafted features. Except for JMMAC C.2 and SNDCFT C.4, all deep-feature-based trackers train their backbones.

Results The results are summarized in the AR-raw plots, EAO curves, and the expected average overlap plots in Figure 11. The values are also reported in Table 4.

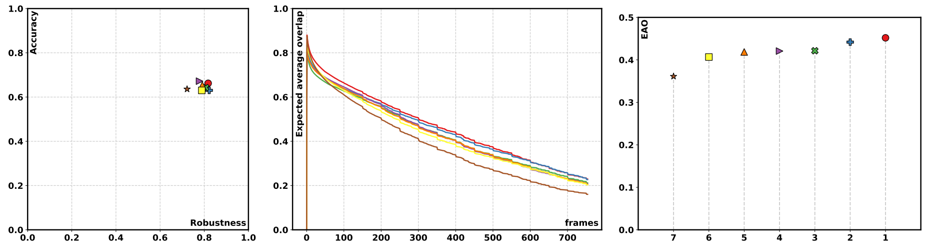

Fig. 11. The VOT-RGBT2020 AR plot (left), the EAO curves (center), and the EAO plot (right). The legend is given in Table 4.

The top performer on the public dataset is JMMAC C.2 with an EAO score of 0.420. This tracker thus repeats its top rank on the public dataset from 2019, even though it does not perform backbone training.

The second-best ranked tracker is AMF C.3 with an EAO score of 0.412. It follows the standard recipe of singe-stage discriminative correlation filter applied to deep features with backbone training.

The third top-performing position is taken by DFAT C.5, the only Siamese-network-based tacker among the submissions, with an EAO score of 0.390.

Tracker	EAO	A	R
●JMMAC	0.420①	0.662②	0.818②
✚AMF	0.412②	0.630	0.822①
✖DFAT	0.390③	0.672①	0.779
▶SiamDW-T	0.389	0.654③	0.791
▲mfDiMP	0.380	0.638	0.793③
□SNDCFT	0.378	0.630	0.789
★M2C2Frgbt	0.332	0.636	0.722

Table 4. The ranking of the five submitted trackers and the two top-ranked trackers from VOT-RGBT2019 on the VOT-RGBT2020 public dataset.

Since this has been the second RGBT-challenge within VOT, we can compare the newly submitted trackers to top-performing trackers from 2019: JMMAC C.2 (also submitted 2020), SiamDW-T C.7, and mfDiMP C.6. In comparison to these three trackers, only AMF C.3 and DFAT C.5 beat previous top-performers. Note that EAO scores from 2019 and 2020 differ due to the different restart policies. Note further that the number of trackers from 2019 is too small to introduce a state-of-the-art bound as for the VOT-ST challenge.

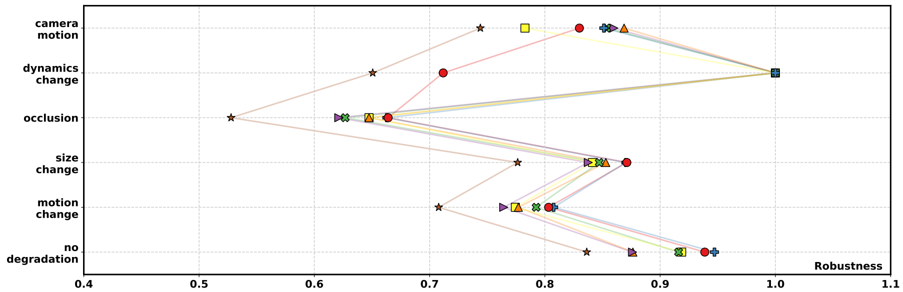

Fig. 12. Failure rate with respect to the visual attributes. The legend is given in Table 4.

However, similar to VOT-ST, we analyzed the number of failures with respect to the visual attributes (replacing illumination change with dyunamics change), see Figure 12. The overall top performers remain at the top of per-attribute ranks as well, with the only exception that JMMAC shows degraded performance for dynamics changes. SiamDW-T and mfDiMP perform comparably weak if no degradation is present. The most challenging attributes in terms of failures are occlusion and motion change. Dynamics change is the least challenging attribute.

The VOT-RGBT2020 challenge winner The top five trackers (besides 2019 top-performers AMF C.3 and DFAT C.5) were re-run on the sequestered dataset and their scores are shown in Table 5. Interestingly, the order of trackers has changed significantly. Looking at the individual scores, it becomes evident that the trackers submitted by the committee, SiamDW-T C.7 and mfDiMP C.6, were the only trackers with better EAO-score on the sequestered dataset than on the public dataset. The top tracker according to the EAO that has not been submitted by the committee is DFAT C.5, which achieved basically the same EAO-score on both datasets, and is thus the VOT-RGBT2020 challenge winner.

Tracker	EAO	A	R
1. SiamDW-T	0.403①	0.664①	0.702③
2. mfDiMP	0.402②	0.623③	0.734①
3. DFAT	0.385③	0.654②	0.674
4. AMF	0.373	0.590	0.705②
5. JMMAC	0.158	0.576	0.287

Table 5. The top five trackers from Table 4 re-ranked on the VOT-RGBT2020 sequestered dataset.

4.5 The VOT-RGBD2020 challenge results

Trackers submitted The VOT-RGBD2020 challenge received 4 valid entries: ATCAIS (D.1), DDiMP (D.2), CLGS_D (D.3) and Siam_LTD (D.4). We also included the best and the third best tracker from the previous year (VOT-RGBD2019): SiamDW_D and LTDSEd. The previous version of ATCAIS was submitted in 2019 as well and obtained the second best F-score (0.676). In addition, to study the performance gap between the best RGB and RGBD trackers the best performing RGB trackers from the VOT-LT2020 and VOT-ST2020 challenges were included: LTMU_B, Megtrack, RLT_DiMP, RPT, OceanPlus and AlphaRef. In total, 12 trackers were considered for the challenge. In the following we briefly overview the entries.

ATCAIS is based on the ATOM tracker [13] and HTC instance segmentation [9]. In ATCAIS the depth channel is used to detect occlusion and disappearance and in target re-detection. DDiMP is an extension of the original DiMP RGB tracker. DDiMP uses better features from ResNet50 and depth information is used to robustify scale changes during tracking. CLGS_D tracker utilizes a set of deep architectures (SiamMask, FlowNetV2, CenterNet and MDNet) and uses the optical flow (FlowNet) and depth maps to filter the region proposals for target re-detection. Similar to the other RGBD trackers Siam_LTD is based on deep architectures, but it is unclear how the depth information is integrated to the processing pipeline.

The SoTA RGB trackers are explained in more details in Section 4.3 and the two RGB-D trackers from the previous year can be found in the VOT2019 report [31].

Tracker	Pr	Re	F-Score	ST/LT	RGB/RGBD
●ATCAIS	0.709②	0.696①	0.702①	LT	RGBD
✚DDiMP	0.703③	0.689②	0.696②	ST	RGBD
✖CLGS_D	0.725①	0.664	0.693③	LT	RGBD
▶SiamDW_D	0.677	0.685③	0.681	LT	RGBD
▲LTDSEd	0.674	0.643	0.658	LT	RGBD
□RLT_DiMP	0.625	0.632	0.629	LT	RGB
★LTMU_B	0.680	0.581	0.626	LT	RGB
○Megtrack	0.694	0.551	0.614	LT	RGB
✚RPT	0.601	0.546	0.572	ST	RGB
✖Siam_LTD	0.626	0.489	0.549	LT	RGBD
▶OceanPlus	0.577	0.502	0.537	ST	RGB
▲AlphaRef	0.491	0.547	0.518	ST	RGB

Table 6. List of trackers that participated in the VOT-RGBD2020 challenge along with their performance scores (Pr, Re, F-score) and categorizations (ST/LT, RGB/RGBD). 2020 submissions are ATCAIS, DDiMP, CLGS_D and Siam_LTD. SiamDW_D and LTDSEd are 2019 submissions (SiamDW_D was the winner). RGB trackers are the three top performers of VOT-ST2020 and VOT-LT2020.

Results The overall performances are summarized in Figure 13 and Table 6. *ATCAIS* obtains the highest F-score in 2020 while it obtained the second best in 2019. The improvement on the same data is from 0.676 (F-score) to 0.702 while the last year winner (SiamDW_D) obtains 0.681. All the results are based on the submitted numbers, but these were verified by running the codes multiple times.

The three best RGBD trackers, ATCAIS, DDiMP and CLGS_D, provide better results than the last year winner, SiamDW_D, but the improvement of the best (ATCAIS) is only 3%. Moreover, the Precision, Recall and F-score values of the three best trackers are within 1.2% (F-score) to 4.5% (Recall) and the numbers are similar to VOT-LT2020 challenge which indicate that the results are saturating and a new dataset is needed to make the RGBD data more challenging.

ATCAIS is the best performer in 8 out of the 14 attribute categories (Figure 14). It is noteworthy that in 2019 competition ATCAIS performance was particularly poor on *full occlusion* and *out-of-frame* categories, on which ATCAIS substantially improved this year. The second best RGBD tracker DDiMP has very similar per category performance to ATCAIS and DDiMP obtains better performance on *deformable*, *full occlusion*. *occlusion*, *out-of-frame*, and *similar objects* categories. On the other hand, the performance of the RLT_DiMP RGB tracker is moderately good across all attribute categories despite of not using the depth channel at all.

The VOT-RGBD2020 challenge winner It should be noted that there are only minor differences among the four best RGBD trackers and the last year

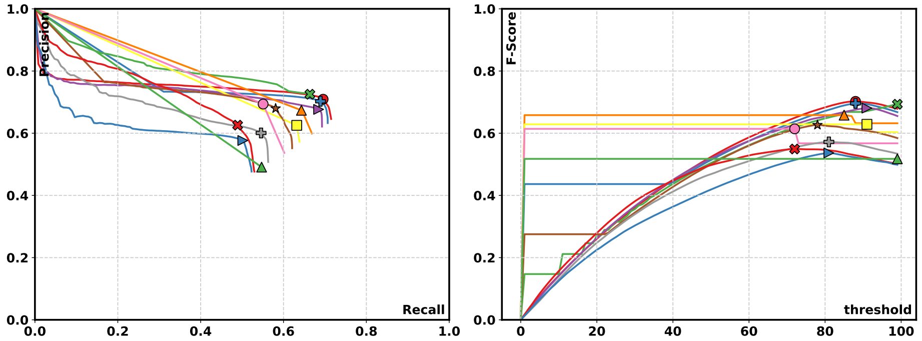

Fig. 13. VOT-RGBD2020 challenge average tracking precision-recall curves (left), the corresponding F-score curves (right). The tracker labels are sorted according to the maximum of the F-score.

winner, SiamDW_D, is among them. They all achieve the maximum F-measure near the same Precision-Recall region (Figure 13) and differences between their Recall, Precision and F-score values are from 1% to 4%. The five best trackers are RGBD trackers which indicates the importance of the depth cue.

The winner is selected based on the best F-score and is ATCAIS (F-score 0.702). For the winning F-score ATCAIS also obtains the best recall (0.696) and the second best precision (0.709). ATCAIS also obtains the best performance on 7 out of 13 assigned attributes for all sequences (inc. "unassigned"). According to the VOT winner rules, the VOT-RGBD2020 challenge winner is therefore ATCAIS (D.1).

5 Conclusion

Results of the VOT2020 challenge were presented. The challenge is composed of the following five challenges focusing on various tracking aspects and domains: (i) the VOT2020 short-term RGB tracking challenge (VOT-ST2020), (ii) the VOT2020 short-term real-time RGB tracking challenge (VOT-RT2020), (iii) the VOT2020 long-term RGB tracking challenge (VOT-LT2020), (iv) the VOT2020 short-term RGB and thermal tracking challenge (VOT-RGBT2020) and (v) the VOT2020 long-term RGB and depth (D) tracking challenge (VOT-RGBD2020).

Several novelties were introduced in the VOT2020 challenge. A new VOT short-term performance evaluation methodology was introduced. The new methodology is an extension of the VOT2019 methodology that avoids tracker-dependent re-starts and addresses short-term failures. Another important novelty is transition from bounding boxes to segmentation masks in the VOT-ST challenge. Both, the VOT-ST public and sequestered dataset were refreshed and the targets were manually segmented in each frame.

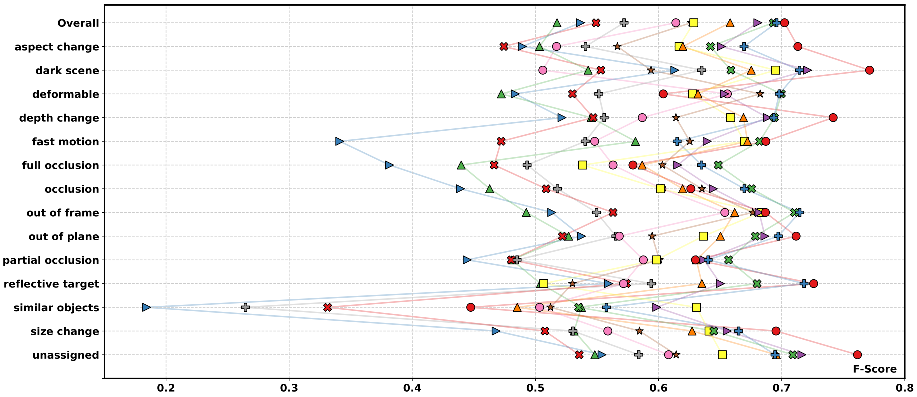

Fig. 14. VOT-RGBD2020 challenge: tracking performance w.r.t. visual attributes.

A major technical novelty is transition to a new Python toolkit that implements the VOT2020 challenges and the new performance evaluation protocols. This toolkit will be extended to support VOT2019 challenges in near future, after which the previous Matlab-based toolkits will be made obsolete and only the VOT Python toolkit will be maintained.

The overall results of the VOT-ST2020 challenges show that the majority of tested trackers apply either deep discriminative correlation filters or Siamese networks. Majority of trackers report a segmentation mask, including the top performers. Interestingly, results show that video-object-segmentation state-of-the-art method STM [56] obtained competitive performance and outperformed several state-of-the-art bounding box trackers. Another observation is that, 8 out of 10 top real-time trackers (VOT-RT2020 challenge) are among the top ten performers on the VOT-ST2020, which shows emergence of deep learning architectures that no longer sacrifice the speed for tracking accuracy (assuming a sufficiently powerful GPU is available). As in VOT2019 short-term challenges, the most difficult attributes remain occlusion and motion change.

The VOT-LT2020 challenge top performers apply short-term localization and long-term re-detection tracker structure. Similarly to VOT2019, the dominant methodologies are deep DCFs [13,5] and Siamese correlation [4], region proposals and online trained CNN classifiers [55].

The participating trackers of the VOT-RGBT2020 challenge did not go significantly beyond the top-performers from VOT-RGBT2019. It was even observed that some of the participating trackers over-fitted to the public dataset such that the two top-ranked trackers from the VOT-RGBT2019 sequestered dataset are still top ranked in 2020. Notably, Siamese network approaches seem to have bypassed DCF-based trackers now, even though with a small margin.

All trackers submitted to the VOT-RGBD2020 challenge are based on the SoTA deep RGB trackers. Depth information is used to improve occlusion de-

	VOT-ST2020			VOT-RT2020			Unsupervised
Tracker	EAO	A	R	EAO	A	R	AO
● RPT	0.530①	0.700	0.869①	0.290	0.587	0.614	0.632①
✚ OceanPlus	0.491②	0.685	0.842②	0.471②	0.679	0.824①	0.575③
✖ AlphaRef	0.482③	0.754①	0.777	0.486①	0.754①	0.788③	0.590②
▶ AFOD	0.472	0.713	0.795	0.458③	0.708②	0.780	0.539
▲ LWTL	0.463	0.719③	0.798	0.337	0.619	0.720	0.570
■ fastOcean	0.461	0.693	0.803③	0.452	0.691	0.792②	0.566
★ DET50	0.441	0.679	0.787	0.189	0.633	0.401	0.524
● D3S	0.439	0.699	0.769	0.416	0.693	0.748	0.508
✚ Ocean	0.430	0.693	0.754	0.419	0.695	0.741	0.533
✖ TRASFUSTm	0.424	0.696	0.745	0.282	0.576	0.616	0.524
▶ DESTINE	0.396	0.657	0.745	0.278	0.552	0.638	0.463
▲ AFAT	0.378	0.693	0.678	0.372	0.687	0.676	0.502
■ TRASTmask	0.370	0.684	0.677	0.321	0.628	0.643	0.494
★ SiamMargin	0.356	0.698	0.640	0.355	0.698③	0.640	0.465
● SiamMask_S	0.334	0.671	0.621	0.312	0.651	0.604	0.449
✚ VPU_SiamM	0.323	0.652	0.609	0.280	0.613	0.555	0.405
✖ siammask	0.321	0.624	0.648	0.320	0.624	0.645	0.405
▷ SiamEM	0.310	0.520	0.743	0.187	0.438	0.491	0.418
▲ STM	0.308	0.751②	0.574	0.282	0.694	0.559	0.445
■ SuperDiMP	0.305	0.477	0.786	0.289	0.472	0.767	0.417
★ DPMT	0.303	0.492	0.745	0.293	0.487	0.730	0.383
● A3CTDmask	0.286	0.673	0.537	0.260	0.634	0.498	0.371
✚ TRAT	0.280	0.464	0.744	0.256	0.445	0.724	0.367
✖ UPDT	0.278	0.465	0.755	0.237	0.443	0.688	0.374
▶ DiMP	0.274	0.457	0.740	0.241	0.434	0.700	0.367
▲ ATOM	0.271	0.462	0.734	0.237	0.440	0.687	0.378
■ DCDA	0.236	0.456	0.635	0.232	0.456	0.624	0.315
★ igs	0.222	0.421	0.643	0.221	0.421	0.643	0.286
● TCLCF	0.202	0.430	0.582	0.202	0.430	0.582	0.216
✚ FSC2F	0.199	0.416	0.581	0.156	0.397	0.456	0.269
✖ asms	0.197	0.419	0.565	0.197	0.419	0.565	0.256
▶ CSR-DCF	0.193	0.406	0.582	0.193	0.405	0.580	0.242
▲ SiamFC	0.179	0.418	0.502	0.172	0.422	0.479	0.229
■ InfoVital	0.175	0.401	0.562	0.058	0.326	0.163	0.239
★ KCF	0.154	0.407	0.432	0.154	0.406	0.434	0.178
● MIL	0.113	0.367	0.322	0.104	0.366	0.276	0.146
✚ IVT	0.092	0.345	0.244	0.089	0.349	0.229	0.096

Table 8. Results for VOT-ST2020 and VOT-RT2020 challenges. Expected average overlap (EAO), accuracy and robustness are shown. For reference, a no-reset average overlap AO [76] is shown under *Unsupervised*.

tection and target re-detection. The five best RGBD trackers (three of them submitted 2020 and two 2019) are better than the best RGB-only tracker with a clear margin. The results indicate that the depth provides complementary information for visual object tracking and therefore more research and new datasets are expected for RGBD tracking.

The top performer on the VOT-ST2020 *public dataset* is RPT A.8. This is a two-stage tracker that integrates a state-of-the-art region proposal network [84] with a state-of-the-art segmentation tracker [50]. On the public set, this tracker obtains a significantly better performance than the second-best tracker. RPT is also the top tracker on the sequestered dataset, on which the performance difference to the second-best is still large, albeit reduced compared to the difference observed on the public set. RPT A.8 is thus a clear winner of the VOT-ST2020 challenge.

The top performer and the winner of the VOT-RT2020 challenge is AlphaRef A.12. This is a two-stage tracker that applies DiMP [5] to localize the target region and refines it with a network akin to [60] to merge pixel-wise correlation and non-local layer outputs into the final segmentation mask. This tracker is also ranked quite high (3rd) on the public VOT-ST2020 challenge.

The top performer of the VOT-LT2020 challenge is LTDSE B.7, wich combines a CNN-based DCF [13] with a Siamese segmentation tracker [74] and an fast version of an online trained CNN classifier [55]. This tracker is also the VOT-LT2019 challenge winner, which was included by the VOT2020 committee as a strong baseline. The top submitted tracker and the winner of the VOT-LT2020 challenge, is LTMUB, which, accroding to the authors, can be considered as a simplified version of the LTDSE.

The top performer on the VOT-RGBT2020 *public dataset* is JMMAC (C.2), an approach that combines DCF-based tracking with RANSAC. The top-ranked participating tracker on the sequestered dataset and the VOT-RGBT2020 challenge winner is DFAT (C.5), the only Siamese-network-based participating tracker.

The top performer and the winner of the VOT-RGBD2020 challenge is ATCAIS (D.1) that improved its rank from the last year second place to this year first place. ATCAIS is based on the ATOM tracker [13] and HTC instance segmentation [9]. The depth value is used to detect the target occlusion or disappearance and re-find the target. For the winning F-score (0.702), ATCAIS obtains the best performance on 8 out of 14 attributes.

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 VOT2020 was the eighth effort toward this, following the very successful VOT2013, VOT2014, VOT2015, VOT2016, VOT2017, VOT2018 and VOT2019.

This VOT edition started a transition to a fully segmented ground truth. We believe this will boost the research in tracking, which will result in a class of robust trackers with per-pixel target localization. In future editions we expect more sub-challenges to follow this direction, depending on man-power, as producing high-quality segmentation ground truth requires substantial efforts.

A new Python toolkit that implements the new evaluation protocols follows the trend of majority of trackers transiting to Python as the main programming language. Our future work will follow these lines of advancements.

Acknowledgements

This work was supported in part by the following research programs and projects: Slovenian research agency research programs P2-0214, Z2-1866, 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 and Linbo He were supported by WASP, VR (ELLIIT and NCNN), and SSF (SymbiCloud). Roman Pflugfelder and Gustavo Fernández were supported by the AIT Strategic Research Programme 2020 Visual Surveillance and Insight. The challenge was sponsored by the Faculty of Computer Science, University of Ljubljana, Slovenia.

A VOT-ST2020 and VOT-RT2020 submissions

This appendix provides a short summary of trackers considered in the VOT-ST2020 and VOT-RT2020 challenges.

A.1 Discriminative Sing-Shot Segmentation Tracker (D3S)

A. Lukezic
alan.lukezic@fri.uni-lj.si

Template-based discriminative trackers are currently the dominant tracking paradigm due to their robustness, but are restricted to bounding box tracking and a limited range of transformation models, which reduces their localization accuracy. We propose a discriminative single-shot segmentation tracker named D3S [50], which narrows the gap between visual object tracking and video object segmentation. A single-shot network applies two target models with complementary geometric properties, one invariant to a broad range of transformations, including non-rigid deformations, the other assuming a rigid object to simultaneously achieve high robustness and online target segmentation.

A.2 Visual Tracking by means of Deep Reinforcement Learning and an Expert Demonstrator (A3CTDmask)

M. Dunnhofer, G. Foresti, C. Micheloni
{*matteo.dunnhofer, gianluca.foresti, christian.micheloni*}*@uniud.it*

A3CTDmask is the combination of the A3CTD tracker [16] with a one-shot segmentation method for target object mask generation. A3CTD is a real-time tracker built on a deep recurrent regression network architecture trained offline using a reinforcement learning based framework. After training, the proposed tracker is capable of producing bounding box estimates through the learned policy or by exploiting the demonstrator. A3CTDmask exploits SiamMask [74] by reinterpreting it as a one-shot segmentation module. The target object mask is generated inside a frame patch obtained through the bounding box estimates given by A3CTD.

A.3 Deep Convolutional Descriptor Aggregation for Visual Tracking (DCDA)

Y. Li, X. Ke
liyuezhou.cm@gmail.com, kex@fzu.edu.cn

This work aims to mine the target representation capability of pre-trained VGG16 model for visual tracking. Based on spatial and semantic priors, a central attention mask is designed for robust-aware feature aggregation, and an edge attention mask is used for accuracy aware feature aggregation. To make full use of the scene context, a regression loss is developed to learn a discriminative feature for complex scenes. DCDA tracker is implemented based on the Siamese network, with a feature fusion and template enhancement strategies.

A.4 IOU guided Siamese networks for visual object tracking (igs)

M. Dasari, R. Gorthi
{ee18d001, rkg}@iittp.ac.in

In the proposed IOU-SiamTrack framework, a new block called 'IOU module' is introduced. This module accepts the above feature domain response maps, convert them into image domain with the help of anchor boxes, as is done in the inference stage in [42,41]. Using the classification response map, top-K 'probable' bounding boxes, having top-K responses are selected. IOU module then calculates the IOU of probable bounding boxes w.r.t. estimated bounding box and produce the one with maximum IOU score as predicted output bounding box. Through training progress, predicted box is more aligned with ground truth, as network is guided to minimise the IOU loss.

A.5 SiamMask_SOLO (SiamMask_S)

Y. Jiang, Z. Feng, T. Xu, X. Song
yj.jiang@stu.jiangnan.edu.cn, {z.feng, tianyang.xu}@surrey.ac.uk,
x.song@jiangnan.edu.cn

The SiamMask_SOLO tracker is based on the SiamMask algorithm. It utilizes a multi-layer aggregation module to make full use of different levels of deep CNN features. Besides, to balance all the three branches, the mask branch is replaced by a SOLO [75] head that uses CoordConv and FCN, which improves the performance of the proposed SiamMask_SOLO tracker in terms of both accuracy and robustness. The original refined module is kept for a further performance boost.

A.6 Diverse Ensemble Tracker (DET50)

N. Wang, W. Zhou, H. Li
wn6149@mail.ustc.edu.cn, {zhwg, lihq}@ustc.edu.cn

In this work, we leverage an ensemble of diverse models to learn manifold representations for robust object tracking. Based on the DiMP method, a shared backbone network (ResNet-50) is applied for feature extraction and multiple head networks for independent predictions. To shrink the representational overlaps among multiple models, both model diversity and response diversity regularization terms are used during training. This ensemble framework is end-to-end trained in a data-driven manner. After box-level prediction, we use SiamMask for mask generation.

A.7 VPU_SiamM: Robust Template Update Strategy for Efficient Object Tracking (VPU_SiamM)

A. Gebrehiwot, J. Bescos, Á. García-Martín
awet.gebrehiwot@estudiante.uam.es, {j.bescos, alvaro.garcia}@uam.es

The VPU_SiamM tracker is an improved version of the SiamMask [74]. The SiamMask tracks without any target update strategy. In order to enable more discriminant features and to enhance robustness, the VPU_SiamM applies a target template update strategy, which leverages both the initial ground truth template and a supplementary updatable template. The initial template provides highly reliable information and increase robustness against model drift and the updatable template integrates the new target information from the predicted target location given by the current frame. During online tracking, VPU_SiamM applies both forward and backward tracking strategies by updating the updatable target template with the predicted target. The tracking decision on the next frame is determined where both templates yield a high response map (score) in the search region. Data augmentation strategy has been implemented during the training process of the refinement branch to become robust in handling motion-blurred and low-resolution datasets during inference.

A.8 RPT: Learning Point Set Representation for Siamese Visual Tracking (RPT)

H. Zhang, L. Wang, Z. Ma, W. Lu, J. Yin, M. Cheng
1067166127@qq.com, {wanglinyuan, kobebean, lwhfh01}@zju.edu.cn, {yin_jun, cheng_miao}@dahuatech.com

RPT tracker is formulated with a two-stage structure. The first stage is composed with two parallel subnets, one for target estimation with RepPoints [84] in an offline-trained embedding space, the other trained online to provide high robustness against distractors [13]. The online classification subnet is set to a lightweight 2-layer convolutional neural network. The target estimation head is constructed with Siamese-based feature extraction and matching. For the second stage, the set of RepPoints with highest confidence (i.e. online classification score) is fed into a modified D3S [50] to obtain the segmentation mask. A segmentation map is obtained by combining enhanced target location channel with target and background similarity channels. The backbone is ResNet50 pretrained on ImageNet, while the target estimation head is trained using pairs of frames from YouTube-Bounding Box [59], COCO [45] and ImageNet VID [63] datasets.

A.9 Tracking Student and Teacher (TRASTmask)

M. Dunnhofer, G. Foresti, C. Micheloni
{matteo.dunnhofer, gianluca.foresti, christian.micheloni}@uniud.it

TRASTmask is the combination of the TRAST tracker [17] with a one-shot segmentation method for target object mask generation. TRAST tracker consists of two components: (i) a fast processing CNN-based tracker, i.e. the Student; and (ii) an off-the-shelf tracker, i.e. the Teacher. The Student is trained offline based on knowledge distillation and reinforcement learning, where multiple tracking teachers are exploited. Tracker TRASTmask uses DiMP [5] as the Teacher. The target object mask is generated inside a frame patch obtained through the bounding box estimates given by TRAST tracker.

A.10 Ocean: Object-aware Anchor-free Tracking (Ocean)

Z. Zhang, H. Peng
zhangzhipeng2017@ia.ac.cn, houwen.peng@microsoft.com

We extend our object-aware anchor-free tracking framework [92] with novel transduction and segmentation networks, enabling it to predict accurate target mask. The transduction network is introduced to infuse the knowledge of the given mask in the first frame. Inspired by recent work TVOS [89], it compares the pixel-wise feature similarities between the template and search features, and then transfers the mask of the template to an attention map based on the similarities. We add the attention map to backbone features to learn target-background aware representations. Finally, a U-net shape segmentation pathway is designed to progressively refine the enhanced backbone features to target mask. The code will be completely released at https://github.com/researchmm/TracKit.

A.11 Tracking by Student FUSing Teachers (TRASFUSTm)

M. Dunnhofer, G. Foresti, C. Micheloni
{*matteo.dunnhofer, gianluca.foresti, christian.micheloni*}*@uniud.it*

The tracker TRASFUSTm is the combination of the TRASFUST tracker [17] with a one-shot segmentation method for target object mask generation. TRASFUSTm tracker consists of two components: (i) a fast processing CNN-based tracker, i.e. the Student; (ii) a pool of off-the-shelf trackers, i.e. Teachers. The Student is trained offline based on knowledge distillation and reinforcement learning, where multiple tracking teachers are exploited. After learning, through the learned evaluation method, the Student is capable to select the prediction of the best Teacher of the pool, thus performing robust fusion. Both trackers DiMP [5] and ECO [12] were chosen as Teachers. The target object mask is generated inside a frame patch obtained through the bounding box estimates given by TRASFUSTm tracker.

A.12 Alpha-Refine (AlphaRef)

B. Yan, D. Wang, H. Lu, X. Yang
yan_bin@mail.dlut.edu.cn, {*wdice, lhchuan*}*@dlut.edu.cn,*
xyang@remarkholdings.com

We propose a simple yet powerful two-stage tracker, which consists of a robust base tracker (super-dimp) and an accurate refinement module named Alpha-Refine [82]. In the first stage, super-dimp robustly locates the target, generating an initial bounding box for the target. Then in the second stage, based on this result, Alpha-Refine crops a small search region to predict a high-quality mask for the tracked target. Alpha-Refine exploits pixel-wise correlation for fine feature aggregation, and uses non-local layer to capture global context information.

Besides, Alpha-Refine also deploys a delicate mask prediction head [60] to generate high-quality masks. The complete code and trained models of Alpha-Refine will be released at github.com/MasterBin-IIAU/AlphaRefine.

A.13 Hierarchical Representations with Discriminative Meta-Filters in Dual Path Network for Tracking (DPMT)

f. xie, n. wang, k. yang, y. yao
220191672@seu.edu.cn, 20181222016@nuist.edu.cn,
yangkang779@163.con, 220191672@seu.edu.cn

We propose a novel dual path network with discriminative meta-filters and hierarchical representations to solve these issues. DPMT tracker consists of two pathways: (i) Geographical Sensitivity Pathway (GASP) and (ii) Geometrically Sensitivity Pathway (GESP). The modules in Geographical Sensitivity Pathway (GASP) are more sensitive to the spatial location of targets and distractors. Subnetworks in Geometrically Sensitivity Pathway (GESP) are designed to refine the bounding box to fit the target. According to this dual path network design, Geographical Sensitivity Pathway (GASP) should be trained to own more discriminative power between foreground and background while Geographical Sensitivity Pathway (GASP) should focus more on the appearance model of the object.

A.14 SiamMask (siammask)

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

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.15 OceanPlus: Online Object-aware Anchor-free Tracking (OceanPlus)

Z. Zhang, H. Peng, Z. Wu, K. Liu, J. Fu, B. Li, W. Hu
zhangzhipeng2017@ia.ac.cn, houwen.peng@microsoft.com,
Wu.Zhirong@microsoft.com, liukaiwen2019@ia.ac.cn, jianf@microsoft.com,
bli@nlpr.ia.ac.cn, wmhu@nlpr.ia.ac.cn

This model is the extension of the Ocean tracker A.10. Inspired by recent online models, we introduce an online branch to accommodate to the changes of object scale and position. Specifically, the online branch inherits the structure and parameters from the first three stages of the Siamese backbone network. The fourth stage keeps the same structure as the original ResNet50, but its initial parameters are obtained through the pre-training strategy proposed in [5]. The segmentation refinement pathway is the same as Ocean. We refer the readers to Ocean tracker A.10 and https://github.com/researchmm/TracKit for more details.

A.16 fastOcean: Fast Object-aware Anchor-free Tracking (fastOcean)

Z. Zhang, H. Peng
zhangzhipeng2017@ia.ac.cn, houwen.peng@microsoft.com

To speed up the inference of our submitted tracker OceanPlus, we use TensorRT[58] to re-implement the model. All structure and model parameters are the same as OceanPlus. Please refer to OceanPlus A.15 and Ocean A.10 for more details.

A.17 Siamese tracker with discriminative feature embedding and mask prediction. (SiamMargin)

G. Chen, F. Wang, C. Qian
{*chenguangqi, wangfei, qianchen*}*@sensetime.com*

SiamMargin is based on the SiamRPN++ algorithm [41]. In the training stage, a discrimination loss is added to the embedding layer. In the training phase the discriminative embedding is offline learned. In the inference stage the template feature of the object in current frame is obtained by ROIAlign from features of the current search region and it is updated via a moving average strategy. The discriminative embedding features are leveraged to accommodate the appearance change with properly online updating. Lastly, the SiamMask [74] model is appended to obtain the pixel-level mask prediction.

A.18 Siamese Tracker with Enhanced Template and Generalized Mask Generator (SiamEM)

Y. Li, Y. Ye, X. Ke
liyuezhou.cm@gmail.com, yyfzu@foxmail.com, kex@fzu.edu.cn

SiamEM is a Siamese tracker with enhanced template and generalized mask generator. SiamEM improves SiamFC++ [81] by obtaining feature results of the template and flip template in the network header while making decisions based on quality scores to predict bounding boxes. The segmentation network presented in [10] is used as a mask generation network.

[58] https://github.com/NVIDIA/TensorRT

A.19 TRacker by using ATtention (TRAT)

H. Saribas, H. Cevikalp, B. Uzun
{*hasansaribas48, hakan.cevikalp, eee.bedirhan*}*@gmail.com*

The tracker 'TRacker by using ATtention' uses a two-stream network which consists of a 2D-CNN and a 3D-CNN, to use both spatial and temporal information in video streams. To obtain temporal (motion) information, 3D-CNN is fed by stacking the previous 4 frames with one stride. To extract spatial information, the 2D-CNN is used. Then, we fuse the two-stream network outputs by using an attention module. We use ATOM [13] tracker and ResNet backbone as a baseline. Code is available at https://github.com/Hasan4825/TRAT.

A.20 InfoGAN based tracker: InfoVITAL (InfoVital)

H. Kuchibhotla, M. Dasari, R. Gorthi
{*ee18m009, ee18d001, rkg*}*@iittp.ac.in*

Architecture of InfoGAN (Generator, Discriminator and a Q-Network) is incorporated in the Tracking-By-Detection Framework using the Mutual Information concept to bind two distributions (latent code) to the target and the background samples. Additional Q Network helps in proper estimation of the assigned distributions and the network is trained offline in an adversarial fashion. During online testing, the additional information from the Q-Network is used to obtain the target location in the subsequent frames. This greatly helps to assess the drift from the exact target location from frame-to-frame and also during occlusion.

A.21 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 tracking architecture, capable of fully exploiting both target and background appearance information for target model prediction. The target model here constitutes the weights of a convolution layer which performs the target-background classification. The weights of this convolution layer are predicted by the target model prediction network, which is derived from a discriminative learning loss by applying an iterative optimization procedure. The model prediction network employs a steepest descent based methodology that computes an optimal step length in each iteration to provide fast convergence. The online learned target model is applied in each frame to perform target-background classification. The final bounding box is then estimated using the overlap maximization approach as in [13]. See [5] for more details about the tracker.

A.22 SuperDiMP (SuperDiMP)

G. Bhat, M. Danelljan, F. Gustafsson, T. B. Schön, L. Van Gool, R. Timofte
{goutam.bhat, martin.danelljan}@vision.ee.ethz.ch, {fredrik.gustafsson, thomas.schon}@it.uu.se, {vangool, timofter}@vision.ee.ethz.ch

SuperDiMP [23] combines the standard DiMP classifier from [5] with the EBM-based bounding-box regressor from [22,14]. Instead of training the bounding box regression network to predict the IoU with an L_2 loss [5], it is trained using the NCE+ approach [23] to minimize the negative-log likelihood. Further, the tracker uses better training and inference settings.

A.23 Learning What to Learn for Video Object Segmentation (LWTL)

G. Bhat, F. Jaremo Lawin, M. Danelljan, A. Robinson, M. Felsberg, L. Van Gool, R. Timofte
goutam.bhat@vision.ee.ethz.ch, felix.jaremo-lawin@liu.se, martin.danelljan@vision.ee.ethz.ch, {andreas.robinson, michael.felsberg}@liu.se, {vangool, timofter}@vision.ee.ethz.ch

LWTL is an end-to-end trainable video object segmentation VOS architecture which captures the current target object information in a compact parametric model. It integrates a differentiable few-shot learner module, which predicts the target model parameters using the first frame annotation. The learner is designed to explicitly optimize an error between target model prediction and a ground truth label, which ensures a powerful model of the target object. Given a new frame, the target model predicts an intermediate representation of the target mask, which is input to the offline trained segmentation decoder to generate the final segmentation mask. LWTL learns the ground-truth labels used by the few-shot learner to train the target model. Furthermore, a network module is trained to predict spatial importance weights for different elements in the few-shot learning loss. All modules in the architecture are trained end-to-end by maximizing segmentation accuracy on annotated VOS videos. See [7] for more details.

A.24 Adaptive Failure-Aware Tracker (AFAT)

T. Xu, S. Zhao, Z. Feng, X. Wu, J. Kittler
tianyang.xu@surrey.ac.uk, zsc960813@163.com, z.feng@surrey.ac.uk, wu_xiaojun@jiangnan.edu.cn, j.kittler@surrey.ac.uk

Adaptive Failure-Aware Tracker [80] is based on Siamese structure. First, multi-RPN module is employed to predict the central location with Resnet-50. Second, a 2-cell LSTM is established to perform quality prediction with an additional motion model. Third, fused mask branch is exploited for segmentation.

A.25 Ensemble correlation filter tracking based on temporal confidence learning (TCLCF)

C. Tsai
chiyi_tsai@gms.tku.edu.tw

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

A.26 AFOD: Adaptive Focused Discriminative Segmentation Tracker (AFOD)

Y. Chen, J. Xu, J. Yu
{*yiwei.chen, jingtao.xu, jiaqian.yu*}*@samsung.com*

The proposed tracker is based on D3S and DiMP [5], employing ResNet-50 as backbone. AFOD calculates the feature similarity to foreground and background of the template as proposed in D3S. For discriminative features, AFOD updates the target model online. AFOD adaptively utilizes different strategies during tracking to update the scale of search region and to adjust the prediction. Moreover, the Lovasz hinge loss metric is used to learn the IoU score in offline training. The segmentation module is trained using both databases YoutubeVOS2019 and DAVIS2016. The offline training process includes two stages: (i) BCE loss is used for optimization and (ii) the Lovasz hinge is applied for further fine tuning. For inference, two ResNet-50 models are used; one for the segmentation and another for the target.

A.27 Fast Saliency-guided Continuous Correlation Filter-based tracker (FSC2F)

A. Memarmoghadam
a.memarmoghadam@yahoo.com

The tracker FSC2F is based on the ECOhc approach [12]. A fast spatio temporal saliency map is added using the PQFT approach [21]. The PQFT model utilizes intensity, colour, and motion features for quaternion representation of the search image context around the previously pose of the tracked object. Therefore, attentional regions in the coarse saliency map can constrain target confidence peaks. Moreover, a faster scale estimation algorithm is utilised by enhancing the fast fDSST method [15] via jointly learning of the sparsely-sampled scale spaces.

A.28 Adaptive Visual Tracking and Instance Segmentation (DESTINE)

S.M. Marvasti-Zadeh, J. Khaghani, L. Cheng, H. Ghanei-Yakhdan, S. Kasaei
mojtaba.marvasti@ualberta.ca, khaghani@ualberta.ca, lcheng5@ualberta.ca, hghaneiy@yazd.ac.ir, kasaei@sharif.edu

DESTINE is a two-stage method consisting of an axis-aligned bounding box estimation and mask prediction, respectively. First, DiMP50 [5] is used as the baseline tracker switching to ATOM [13] when IoU and normalized L1-distance between the results meet predefined thresholds. Then, to segment the estimated bounding box, the segmentation network of FRTM-VOS [60] uses the predicted mask by SiamMask [74] as its scores. Finally, DESTINE selects the best target mask according to the ratio of foreground pixels for two predictions. The codes are publicly released at https://github.com/MMarvasti/DESTINE.

A.29 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 [73] 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.30 ATOM: Accurate Tracking by Overlap Maximization (ATOM)

Submitted by VOT Committee

ATOM separates the tracking problem into two sub-tasks: 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 [13] for more details.

A.31 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 [47].

A.32 Incremental Learning for Robust Visual Tracking (IVT)

Submitted by VOT Committee

The idea of the IVT tracker [62] 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.33 Kernelized Correlation Filter (KCF)

Submitted by VOT Committee

This tracker is a C++ implementation of Kernelized Correlation Filter [24] 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.34 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.35 Robust Siamese Fully Convolutional Tracker (RSiamFC)

Submitted by VOT Committee

RSiamFC tracker is an extended SiamFC tracker [4] with a robust training method which puts a transformation on training sample to generate a pair of samples for feature extraction.

A.36 VOS SOTA method (STM)

Submitted by VOT Committee

Please see the original paper for details [56].

A.37 (UPDT)

Submitted by VOT Committee

Please see the original paper for details [6].

B VOT-LT2020 submissions

This appendix provides a short summary of trackers considered in the VOT-LT2020 challenge.

B.1 Long-Term Visual Tracking with Assistant Global Instance Search (Megtrack)

Z. Mai, H. Bai, K. Yu, X. QIu
marchihjun@gmail.com, 522184271@qq.com, valjean1832@outlook.com, qiuxi@megvii.com

Megtrack tracker applies a 2-stage method that consists of local tracking and multi-level search. The local tracker is based on ATOM [13] algorithm improved by initializing online correlation filters with backbone feature maps and by inserting a bounding box calibration branch in the target estimation module. SiamMask [74] is cascaded to further refining the bounding box after locating the centre of the target. The multi-level search uses RPN-based regression network to generate candidate proposals before applying GlobalTrack [26]. Appearance scores are calculated using both the online-learned RTMDNet [29] and the offline-learned one-shot matching module and linearly combine them to leverage the former's high robustness and the latter's discriminative power. Using a pre-defined threshold, the highest-scored proposal is considered as the current tracker state and used to re-initialize the local tracker for consecutive tracking.

B.2 Skimming-Perusal Long-Term Tracker (SPLT)

B. Yan, H. Zhao, D. Wang, H. Lu, X. Yang
{yan_bin, haojie_zhao}@mail.dlut.edu.cn, {wdice, lhchuan}@dlut.edu.cn, xyang@remarkholdings.com

This is the original SPLT tracker [83] without modification. SPLT consists of a perusal module and a skimming module. The perusal module aims at obtaining precise bounding boxes and determining the target's state in a local search region. The skimming module is designed to quickly filter out most unreliable search windows, speeding up the whole pipeline.

B.3 A Baseline Long-Term Tracker with Meta-Updater (LTMU_B)

K. Dai, D. Wang, J. Li, H. Lu, X. Yang
dkn2014@mail.dlut.edu.cn, {wdice, jianhual}@dlut.edu.cn, lhchuan@dlut.edu.cn, xyang@remarkholdings.com

The tracker LTMU_B is a simplified version of LTMU [11] and LTDSE with comparable performance adding a RPN-based regression network, a sliding-window based re-detection module and a complex mechanism for updating models and target re-localization. The short-term tracker LTMU_B contains two components. One is for target localization and based on DiMP algorithm [5] using ResNet50 as the backbone network. The update of DiMP is controlled by meta-updater which is proposed by LTMU [59]. The second component is the SiamMask network [74] used for refining the bounding box after locating the centre of the target. It also takes the local search region as the input and outputs the tight

[59] https://github.com/Daikenan/LTMU

bounding boxes of candidate proposals. For the verifier, we adopts MDNet network [5] which uses VGGM as the backbone and is pre-trained on ILSVRC VID dataset. The classification score is finally obtained by sending the tracking result's feature to three fully connected layers. GlobalTrack [26] is utilised as the global detector.

B.4 Robust Long-Term Object Tracking via Improved Discriminative Model Prediction (RLTDiMP)

S. Choi, J. Lee, Y. Lee, A. Hauptmann
seokeon@kaist.ac.kr, {ljhyun33, swack9751}@korea.ac.kr, alex@cs.cmu.edu

We propose an improved Discriminative Model Prediction method for robust long-term tracking based on a pre-trained short-term tracker. The baseline tracker is SuperDiMP which combines the bounding-box regressor of PrDiMP [14] with the standard DiMP [5] classifier. To make our model more discriminative and robust, we introduce uncertainty reduction using random erasing, background augmentation for more discriminative feature learning, and random search with spatio-temporal constraints. Code available at https://github.com/bismex/RLT-DIMP.

B.5 Long-term MDNet (ltMDNet)

H. Fan, H. Ling
{hefan, hling}@cs.stonybrook.edu

We designate a long-term tracker by adapting MDNet [55]. In specific, we utilize an instance-aware detector [26] to generate target proposals. Then, these proposals are forwarded to MDNet for classification. Since the detector performs detection on the full image, the final tracker can locate the target in the whole image which can robustly deal with full occlusion and out-of-view. The instance-aware detector is implemented by on Faster R-CNN using ResNet-50. The MDNet is implemented as in the original paper.

B.6 (CLGS)

Submitted by VOT Committee

In this work, we develop a complementary local-global search (CLGS) framework to conduct robust long-term tracking, which is a local robust tracker based on SiamMask [74], a global detection based on cascade R-CNN [8], and an online verifier based on Real-time MDNet [29]. 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. Besides, we design an effective online update strategy to improve the discrimination of the verifier.

B.7 (LT_DSE)

Submitted by VOT Committee

This algorithm divides each long-term sequence into several short episodes and tracks the target in each episode using short-term tracking techniques. Whether the target is visible or not is judged by the outputs from the short-term local tracker and the classification-based verifier updated online. If the target disappears, the image-wide re-detection will be conducted and output the possible location and size of the target. Based on these, the tracker crops the local search region that may include the target 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. If the candidate with the maximum score is above the pre-defined threshold, the tracker will regard it as the target and re-initialize the short-term components. Finally, the tracker conducts short-term tracking until the target disappears again.

B.8 (SiamDW_LT)

Submitted by VOT Committee

SiamDW_LT is a long-term tracker that utilizes deeper and wider backbone networks with fast online model updates. The basic tracking module is a short-term Siamese tracker, which returns confidence scores 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 global re-detection module is activated to search the target in the images. Moreover, object disappearance and occlusion are also detected by the tracking confidence. In addition, we introduce model ensemble to further improve the tracking accuracy and robustness.

C VOT-RGBT2020 submissions

This appendix provides a short summary of trackers considered in the VOT-RGBT2020 challenge.

C.1 Multi-Model Continuous Correlation Filter for rgbt visual object tracking (M2C2Frgbt)

A. Memarmoghadam
a.memarmoghadam@yahoo.com

Inspired by ECO tracker [12], we propose a robust yet efficient tracker namely as M2C2Frgbt that utilizes multiple models of the tracked object and estimates its position every frame by weighted cumulative fusion of their respective regressors in a ridge regression optimization problem [51]. Moreover, to accelerate tracking performance, we propose a faster scale estimation method in which the target scale filter is jointly learned via sparsely sampled scale spaces constructed by just the thermal infrared data. Our scale estimation approach enhances the running speed of fDSST [15] as the baseline algorithm better than 20% while maintaining the tracking performance as well. To suppress unwanted samples mostly belong to the occlusion or other non-object data, we conservatively update every model on-the-fly in a non-uniform sparse manner.

C.2 Jointly Modelling Motion and Appearance Cues for Robust RGB-T Tracking (JMMAC)

P. Zhang, S. Chen, D. Wang, H. Lu , X. Yang
pyzhang@mail.dlut.edu.cn, shuhaochn@mail.dlut.edu.cn, wdice@dlut.edu.cn, lhchuan@dlut.edu.cn, xyang@remarkholdings.com

Our tracker is based on [88], consisting of two components, i.e. multimodal fusion for appearance trackers and camera motion estimation. In multimodal fusion, we develop a late fusion method to infer the fusion weight maps of both RGB and thermal (T) modalities. The fusion weights are determined by using offline-trained global and local Multimodal Fusion Networks (MFNet), and then adopted to linearly combine the response maps of RGB and T modalities obtained from ECOs. In MFNet, the truncated VGG-M networks is used as backbone to extract deep feature. In camera motion estimation, when the drastic camera motion is detected, we compensate movement to correct the search region by key-point-based image registration technique. Finally, we employ YOLOv2 to refine the bounding box. The scale estimation and model updating methods are borrowed from ECO in default.

C.3 Accurate Multimodal Fusion for RGB-T Object Tracking (AMF)

P. Zhang, S. Chen, B. Yan, D. Wang, H. Lu, X. Yang
{*pyzhang, shuhaochn, yan_bin*}*@mail.dlut.edu.cn,* {*wdice, lhchuan*}*@dlut.edu.cn, xyang@remarkholdings.com*

We achieve multimodal fusion for RGB-T tracking by linear combining the response maps obtained from two monomodality base trackers, i.e., DiMP. The fusion weight is obtained by the Multimodal Fusion Network proposed in [88]. To achieve high accuracy, the bounding box obtained from fused DiMP is then refined by a refinement module in visible modality. The refinement module, namely Alpha-Refine, aggregates features via a pixel-level correlation layer and a non-local layer and adaptively selects the most adequate results from three branches, namely bounding box, corner and mask heads, which can predict more accurate bounding boxes. Note that the target scale estimated by IoUNet in DiMP is also applied in visible modality which is followed by Alpha-Refine and the model updating method is borrowed from DiMP in default.

C.4 SqueezeNet Based Discriminative Correlation Filter Tracker (SNDCFT)

A. Varfolomieiev
a.varfolomieiev@kpi.ua

The tracker uses FHOG and convolutional features extracted from both video and infrared modalities. As the convolutional features, the output of the 'fire2/concat' layer of the original SqueezeNet network [27] is used (no additional pre-training for the network is performed). The core of the tracker is the spatially regularized discriminative correlation filter, which is calculated using the ADMM optimizer. The calculation of the DCF filter is performed independently over different feature modalities. The filter is updated in each frame using simple exponential forgetting.

C.5 Decision Fusion Adaptive Tracker (DFAT)

H. Li, Z. Tang, T. Xu, X. Zhu, X. Wu, J. Kittler
hui_li_jnu@163.com, 1030415519@vip.jiangnan.edu.cn, tianyang.xu@surrey.ac.uk, xuefeng_zhu95@163.com, wu_xiaojun@jiangnan.edu.cn, j.kittler@surrey.ac.uk

Decision Fusion Adaptive Tracker is based on Siamese structure. Firstly, the multi-layer deep features are extracted by Resnet-50. Then, multi-RPN module is employed to predict the central location with multi-layer deep features. Finally, an adaptive weight strategy for decision level fusion is utilized to generate the final result. In addition, the template features are updated by a linear template update strategy.

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

Submitted by VOT Committee

The mfDiMP tracker contains an end-to-end tracking framework for fusing the RGB and TIR modalities in RGB-T tracking [87]. The mfDiMP tracker fuses modalities at the feature level on both the IoU predictor and the model predictor of DiMP [87] and won the VOT-RGBT2019 challenge.

C.7 Online Deeper and Wider Siamese Networks for RGBT Visual Tracking (SiamDW-T)

Submitted by VOT Committee

SiamDW-T is based on previous work by Zhang and Peng [91], and extends it with two fusion strategies for RGBT tracking. A simple fully connected layer is appended to classify each fused feature to background or foreground. SiamDW-T achieved the second rank in VOT-RGBT2019 and its code is available at https://github.com/researchmm/VOT2019.

D VOT-RGBD2020 submissions

This appendix provides a short summary of trackers considered in the VOT-RGBD2020 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,wdice,lhchuan}@dlut.edu.cn, xyang@remarkholdings.com

The proposed tracker combines both instance segmentation and the depth information for accurate tracking. ATCAIS is based on the ATOM tracker and the HTC instance segmentation method which is re-trained in a category-agnostic manner. The instance segmentation results are used to detect background distractors and to re-fine the target bounding boxes to prevent drifting. The depth value is used to detect the target occlusion or disappearance and re-finding the target.

D.2 Depth Enhanced DiMP for RGBD Tracking (DDiMP)

S. Qiu, Y. Gu, X. Zhang
{shoumeng, gyz, xlzhang}@mail.sim.ac.cn

DDiMP is based on SuperDiMP which combines the standard DiMP classifier from [5] with the bounding box regressor from [5]. The update strategy of the model during the tracking process is enhanced by using the model's confidence for the current tracking results. Output of IoU-Net is used to determine whether to fine-tune the shape, size, and position of the target. To handle scale variations, the target is searched over five scales $1.025^{\{-2,-1,0,1,2\}}$, and depth information is utilized to prevent scale from changing too quickly. Finally, two trackers with different model update confidence thresholds run in parallel, and the output with higher confidence is selected as the tracking result of the current frame.

D.3 Complementary Local-Global Search for RGBD Visual Tracking (CLGS-D)

H. Zhao, Z. Wang, B. Yan. D. Wang, H. Lu, X. Yang
{haojie_zhao,zzwang,yan_bin,wdice,lhchuan@dlut.edu.cn}@mail.dlut.edu.cn, xyang@remarkholdings.com

CLGS-D tracker is based on SiamMask, FlowNetv2 , CenterNet, Real-time MDNet and a novel box refine module. The SiamMask model is used as the base tracker. MDNet is used to judge whether the target is found or lost. Once the target is lost, CenterNet is used to generate region proposals on the whole image. FlowNetv2 is used to estimate the motion of the target by generating a flow map. Then, the region proposals are filtered with aid of the flow and depth maps. Finally, an online "verifier" will find the target from the remaining region proposals again. A novel module is also used in this work to refine the bounding box.

D.4 Siamese Network for Long-term RGB-D Tracking (Siam_LTD)

X.-F. Zhu, H. Li, S. Zhao, T. Xu, X.-J. Wu
{xuefeng_zhu95,hui_li_jnu,zsc960813,wu_xiaojun}@163.com,
tianyang.xu@surrey.ac.uk

Siam_LTD employes ResNet-50 to extract backbone features and RPN branch to locate the centre. In addition, a re-detection mechanism is introduced.

References

1. Babenko, B., Yang, M.H., Belongie, S.: Robust object tracking with online multiple instance learning. IEEE Trans. Pattern Anal. Mach. Intell. **33**(8), 1619–1632 (2011)
2. Berg, A., Ahlberg, J., Felsberg, M.: 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)
3. Berg, A., Johnander, J., de Gevigney, F.D., Ahlberg, J., Felsberg, M.: Semi-automatic annotation of objects in visual-thermal video. In: IEEE International Conference on Computer Vision, ICCV Workshops (2019)
4. Bertinetto, L., Valmadre, J., Henriques, J., Torr, P.H.S., Vedaldi, A.: Fully convolutional siamese networks for object tracking. In: ECCV Workshops. pp. 850–865 (2016)
5. Bhat, G., Danelljan, M., Gool, L.V., Timofte, R.: Learning discriminative model prediction for tracking. In: IEEE International Conference on Computer Vision, ICCV (2019)
6. Bhat, G., Johnander, J., Danelljan, M., Khan, F.S., Felsberg, M.: Unveiling the power of deep tracking. In: ECCV. pp. 483–498 (2018)
7. Bhat, G., Lawin, F.J., Danelljan, M., Robinson, A., Felsberg, M., Gool, L.V., Timofte, R.: Learning what to learn for video object segmentation. In: European Conference on Computer Vision ECCV (2020)
8. Cai, Z., Vasconcelos, N.: Cascade r-cnn: Delving into high quality object detection. In: CVPR (2018)
9. Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., Loy, C.C., Lin, D.: Hybrid task cascade for instance segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (2019)
10. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 801–818 (2018)
11. Dai, K., Zhang, Y., Wang, D., Li, J., Lu, H., Yang, X.: High-performance long-term tracking with meta-updater. In: CVPR (2020)
12. Danelljan, M., Bhat, G., Khan, F.S., Felsberg, M.: ECO: efficient convolution operators for tracking. In: CVPR. pp. 6638–6646 (2017)
13. Danelljan, M., Bhat, G., Khan, F.S., Felsberg, M.: ATOM: Accurate tracking by overlap maximization. In: CVPR. pp. 4660–4669 (2019)
14. Danelljan, M., Gool, L.V., Timofte, R.: Probabilistic regression for visual tracking. In: CVPR (2020)
15. Danelljan, M., Häger, G., Khan, F.S., Felsberg, M.: Discriminative scale space tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(8), 1561–1575 (2016)

16. Dunnhofer, M., Martinel, N., Luca Foresti, G., Micheloni, C.: Visual tracking by means of deep reinforcement learning and an expert demonstrator. In: The IEEE International Conference on Computer Vision (ICCV) Workshops (Oct 2019)
17. Dunnhofer, M., Martinel, N., Micheloni, C.: A distilled model for tracking and tracker fusion (2020)
18. Fan, H., Lin, L., Yang, F., Chu, P., Deng, G., S. Yu, H.B., Xu, Y., Liao, C., Ling, H.: Lasot: A high-quality benchmark for large-scale single object tracking. In: Comp. Vis. Patt. Recognition (2019)
19. Galoogahi, H.K., Fagg, A., Huang, C., Ramanan, D., Lucey, S.: Need for speed: A benchmark for higher frame rate object tracking. CoRR **abs/1703.05884** (2017), `http://arxiv.org/abs/1703.05884`
20. Goyette, N., Jodoin, P.M., Porikli, F., Konrad, J., Ishwar, P.: Changedetection.net: A new change detection benchmark dataset. In: CVPR Workshops. pp. 1–8. IEEE (2012)
21. Guo, C., Zhang, L.: 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)
22. Gustafsson, F.K., Danelljan, M., Bhat, G., Schön, T.B.: Energy-based models for deep probabilistic regression. In: European Conference on Computer Vision ECCV (2020)
23. Gustafsson, F.K., Danelljan, M., Timofte, R., Schön, T.B.: How to train your energy-based model for regression. CoRR **abs/2005.01698** (2020), `https://arxiv.org/abs/2005.01698`
24. Henriques, J., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. PAMI **37**(3), 583–596 (2015)
25. Huang, L., Zhao, X., Huang, K.: Got-10k: A large high-diversity benchmark for generic object tracking in the wild. arXiv:1810.11981 (2018)
26. Huang, L., Zhao, X., Huang, K.: GlobalTrack: A simple and strong baseline for long-term tracking. In: AAAI (2020)
27. Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K.: Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size. arXiv:1602.07360 (2016)
28. Jack, V., Luca, B., ao F., H.J., Ran, T., Andrea, V., Arnold, S., Philip, T., Efstratios, G.: Long-term tracking in the wild: A benchmark. arXiv:1803.09502 (2018)
29. Jung, I., Son, J., Baek, M., Han, B.: Real-time mdnet. In: ECCV. pp. 83–98 (2018)
30. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **34**(7), 1409–1422 (2012). https://doi.org/10.1109/TPAMI.2011.239
31. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Kamarainen, J.K., Čehovin, L., Drbohlav, O., Lukezic, A., Berg, A., Eldesokey, A., Kapyla, J., Fernández, G., et al.: The seventh visual object tracking vot2019 challenge results. In: ICCV2019 Workshops, Workshop on visual object tracking challenge (2019)
32. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Čehovin, L., Vojíř, T., Bhat, G., Lukežič, A., Eldesokey, A., Fernández, G., et al.: The visual object tracking vot2018 challenge results. In: ECCV2018 Workshops, Workshop on visual object tracking challenge (2018)
33. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Čehovin, L., Vojíř, T., Häger, G., Lukežič, A., Eldesokey, A., Fernández, G., et al.: The visual object tracking vot2017 challenge results. In: ICCV2017 Workshops, Workshop on visual object tracking challenge (2017)

34. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Čehovin, L., Vojíř, T., Häger, G., Lukežič, A., Fernández, G., et al.: The visual object tracking vot2016 challenge results. In: ECCV2016 Workshops, Workshop on visual object tracking challenge (2016)
35. Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Čehovin, L., Fernández, G., Vojíř, T., Häger, G., Nebehay, G., Pflugfelder, R., et al.: The visual object tracking vot2015 challenge results. In: ICCV2015 Workshops, Workshop on visual object tracking challenge (2015)
36. Kristan, M., Pflugfelder, R., Leonardis, A., Matas, J., Porikli, F., Čehovin, L., Nebehay, G., Fernández, G., Vojíř, T., et al.: The visual object tracking vot2013 challenge results. In: ICCV2013 Workshops, Workshop on visual object tracking challenge. pp. 98 –111 (2013)
37. Kristan, M., Pflugfelder, R., Leonardis, A., Matas, J., Čehovin, L., Nebehay, G., Vojíř, T., Fernández, G., et al.: The visual object tracking vot2014 challenge results. In: ECCV2014 Workshops, Workshop on visual object tracking challenge (2014)
38. Kristan, M., Matas, J., Leonardis, A., Vojíř, T., Pflugfelder, R., Fernández, G., Nebehay, G., Porikli, F., Čehovin, L.: A novel performance evaluation methodology for single-target trackers. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**(11), 2137–2155 (2016)
39. Leal-Taixé, L., Milan, A., Reid, I.D., Roth, S., Schindler, K.: Motchallenge 2015: Towards a benchmark for multi-target tracking. CoRR **abs/1504.01942** (2015), `http://arxiv.org/abs/1504.01942`
40. Li, A., Li, M., Wu, Y., Yang, M.H., Yan, S.: Nus-pro: A new visual tracking challenge. IEEE-PAMI (2015)
41. Li, B., Wu, W., Wang, Q., Zhang, F., Xing, J., Yan, J.: Siamrpn++: Evolution of siamese visual tracking with very deep networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4282–4291 (2019)
42. Li, B., Yan, J., Wu, W., Zhu, Z., Hu, X.: High performance visual tracking with siamese region proposal network. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8971–8980 (June 2018)
43. Li, C., Liang, X., Lu, Y., Zhao, N., Tang, J.: RGB-T object tracking: Benchmark and baseline. Pattern Recognition (2019), submitted
44. Liang, P., Blasch, E., Ling, H.: Encoding color information for visual tracking: Algorithms and benchmark. IEEE Transactions on Image Processing **24**(12), 5630–5644 (2015)
45. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European Conference on Computer Vision. pp. 740–755. Springer (2014)
46. Lukežič, A., Kart, U., Kämäräinen, J., Matas, J., Kristan, M.: CDTB: A Color and Depth Visual Object Tracking Dataset and Benchmark. In: ICCV (2019)
47. Lukežič, A., Vojíř, T., Čehovin Zajc, L., Matas, J., Kristan, M.: Discriminative correlation filter with channel and spatial reliability. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6309–6318 (July 2017)
48. Lukežič, A., Čehovin Zajc, L., Vojíř, T., Matas, J., Kristan, M.: Now you see me: evaluating performance in long-term visual tracking. CoRR **abs/1804.07056** (2018), `http://arxiv.org/abs/1804.07056`
49. Lukezic, A., Cehovin Zajc, L., Vojir, T., Matas, J., Kristan, M.: Sperformance evaluation methodology for long-term single object tracking. IEEE Transactions on Cybernetics (2020)
50. Lukezic, A., Matas, J., Kristan, M.: D3S - a discriminative single shot segmentation tracker. In: CVPR (2020)

51. Memarmoghadam, A., Moallem, P.: Size-aware visual object tracking via dynamic fusion of correlation filter-based part regressors. Signal Processing **164**, 84 – 98 (2019). https://doi.org/https://doi.org/10.1016/j.sigpro.2019.05.021, `http://www.sciencedirect.com/science/article/pii/S0165168419301872`
52. Moudgil, A., Gandhi, V.: Long-term visual object tracking benchmark. arXiv preprint arXiv:1712.01358 (2017)
53. Mueller, M., Smith, N., Ghanem, B.: A benchmark and simulator for uav tracking. In: Proc. European Conf. Computer Vision. pp. 445–461 (2016)
54. Muller, M., Bibi, A., Giancola, S., Alsubaihi, S., Ghanem, B.: TrackingNet: A Large-Scale Dataset and Benchmark for Object Tracking in the Wild. In: ECCV. pp. 300–317 (2018)
55. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: CVPR. pp. 4293–4302 (2016)
56. Oh, S.W., Lee, J.Y., Xu, N., Kim, S.J.: Video object segmentation using space-time memory networks. In: ICCV (2019)
57. Pernici, F., del Bimbo, A.: Object tracking by oversampling local features. IEEE Trans. Pattern Anal. Mach. Intell. **36**(12), 2538–2551 (2013). https://doi.org/http://doi.ieeecomputersociety.org/10.1109/TPAMI.2013.250
58. Phillips, P.J., Moon, H., Rizvi, S.A., Rauss, P.J.: The feret evaluation methodology for face-recognition algorithms. IEEE Trans. Pattern Anal. Mach. Intell. **22**(10), 1090–1104 (2000)
59. Real, E., Shlens, J., Mazzocchi, S., Pan, X., Vanhoucke, V.: YouTube-BoundingBoxes: a large high-precision human-annotated data set for object detection in video. In: Comp. Vis. Patt. Recognition. pp. 7464–7473 (2017)
60. Robinson, A., Lawin, F.J., Danelljan, M., Khan, F.S., Felsberg, M.: Learning fast and robust target models for video object segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Computer Vision Foundation (June 2020)
61. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer Assisted Intervention. vol. 9351, pp. 234–241 (2015)
62. Ross, D.A., Lim, J., Lin, R.S., Yang, M.H.: Incremental learning for robust visual tracking. Int. J. Comput. Vision **77**(1-3), 125–141 (2008)
63. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: Imagenet large scale visual recognition challenge. IJCV **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y, `http://dx.doi.org/10.1007/s11263-015-0816-y`
64. Seoung, W.O., Lee, J.Y., Kim, S.J.: Fast video object segmentation by reference-guided mask propagation. In: Comp. Vis. Patt. Recognition. pp. 7376–7385 (2018)
65. Smeulders, A.W.M., Chu, D.M., Cucchiara, R., Calderara, S., Dehghan, A., Shah, M.: Visual Tracking: an Experimental Survey. TPAMI (2013). https://doi.org/10.1109/TPAMI.2013.230
66. Solera, F., Calderara, S., Cucchiara, R.: Towards the evaluation of reproducible robustness in tracking-by-detection. In: Advanced Video and Signal Based Surveillance. pp. 1 – 6 (2015)
67. Song, S., Xiao, J.: Tracking Revisited Using RGBD Camera: Unified Benchmark and Baselines. In: ICCV (2013)
68. Tao, R., Gavves, E., Smeulders, A.W.M.: Tracking for half an hour. CoRR **abs/1711.10217** (2017), `http://arxiv.org/abs/1711.10217`

69. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. arXiv preprint arXiv:1904.01355 (2019)
70. Čehovin, L., Kristan, M., Leonardis, A.: Is my new tracker really better than yours? Tech. Rep. 10, ViCoS Lab, University of Ljubljana (Oct 2013), `http://prints.vicos.si/publications/302`
71. Čehovin, L.: TraX: The visual Tracking eXchange Protocol and Library. Neurocomputing (2017). https://doi.org/http://dx.doi.org/10.1016/j.neucom.2017.02.036
72. Čehovin, L., Leonardis, A., Kristan, M.: Visual object tracking performance measures revisited. IEEE Transactions on Image Processing **25**(3), 1261–1274 (2016)
73. Vojíř, T., Noskova, J., Matas, J.: Robust scale-adaptive mean-shift for tracking. Pattern Recognition Letters **49**, 250–258 (2014)
74. Wang, Q., Zhang, L., Bertinetto, L., Hu, W., Torr, P.H.: Fast online object tracking and segmentation: A unifying approach. In: CVPR. pp. 1328–1338 (2019)
75. Wang, X., Kong, T., Shen, C., Jiang, Y., Li, L.: SOLO: segmenting objects by locations. arXiv preprint arXiv:1912.04488 (2019)
76. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: A benchmark. In: Comp. Vis. Patt. Recognition (2013)
77. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. PAMI **37**(9), 1834–1848 (2015)
78. Xiao, J., Stolkin, R., Gao, Y., Leonardis, A.: 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)
79. Xu, N., Price, B., Yang, J., Huang, T.: Deep grabcut for object selection. In: Proc. British Machine Vision Conference (2017)
80. Xu, T., Feng, Z.H., Wu, X.J., Kittler, J.: Afat: Adaptive failure-aware tracker for robust visual object tracking. arXiv preprint arXiv:2005.13708 (2020)
81. Xu, Y., Wang, Z., Li, Z., Ye, Y., Yu, G.: Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. arXiv preprint arXiv:1911.06188 (2019)
82. Yan, B., Wang, D., Lu, H., Yang, X.: Alpha-refine: Boosting tracking performance by precise bounding box estimation. arXiv preprint arXiv:2007.02024 (2020)
83. Yan, B., Zhao, H., Wang, D., Lu, H., Yang, X.: Skimming-Perusal Tracking: a framework for real-time and robust long-term tracking. In: IEEE International Conference on Computer Vision (ICCV) (2019)
84. Yang, Z., Liu, S., Hu, H., Wang, L., Lin, S.: Reppoints: Point set representation for object detection. In: The IEEE International Conference on Computer Vision (ICCV). pp. 9657–9666 (Oct 2019)
85. Yiming, L., Shen, J., Pantic, M.: Mobile face tracking: A survey and benchmark. arXiv:1805.09749v1 (2018)
86. Young, D.P., Ferryman, J.M.: PETS Metrics: On-line performance evaluation service. In: ICCCN '05 Proceedings of the 14th International Conference on Computer Communications and Networks. pp. 317–324 (2005)
87. Zhang, L., Danelljan, M., Gonzalez-Garcia, A., van de Weijer, J., Khan, F.S.: Multimodal fusion for end-to-end rgb-t tracking. In: IEEE International Conference on Computer Vision, ICCV Workshops (2019)
88. Zhang, P., Zhao, J., Wang, D., Lu, H., Yang, X.: Jointly modeling motion and appearance cues for robust rgb-t tracking. CoRR **abs/2007.02041** (2020)
89. Zhang, Y., Wu, Z., Peng, H., Lin, S.: A transductive approach for video object segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4000–4009 (June 2020)

90. Zhang, Y., Wang, D., Wang, L., Qi, J., Lu, H.: Learning Regression and Verification Networks for Long-term Visual Tracking. CoRR **abs/1809.04320** (2018)
91. Zhang, Z., Peng, H.: Deeper and wider siamese networks for real-time visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4591–4600 (June 2019)
92. Zhang, Z., Peng, H., Fu, J., Li, B., Hu, W.: Ocean: Object-aware anchor-free tracking. arXiv preprint arXiv:2006.10721 (2020)
93. Zhu, P., Wen, L., Bian, X., Haibin, L., Hu, Q.: Vision meets drones: A challenge. arXiv preprint arXiv:1804.07437 (2018)