The First Visual Object Tracking Segmentation VOTS2023 Challenge Results

Matej Kristan[1], Jiří Matas[2], Martin Danelljan[3], Michael Felsberg[4], Hyung Jin Chang[5], Luka Čehovin Zajc[1], Alan Lukežič[1], Ondrej Drbohlav[2], Zhongqun Zhang[5], Khanh-Tung Tran[6], Xuan-Son Vu[6], Johanna Björklund[6], Christoph Mayer[3], Yushan Zhang[4], Lei Ke[3], Jie Zhao[13], Gustavo Fernández[7], Noor Al-Shakarji[38], Dong An[20], Michael Arens[15], Stefan Becker[15], Goutam Bhat[3], Sebastian Bullinger[15], Antoni B. Chan[11], Shijie Chang[13], Hanyuan Chen[14], Xin Chen[13], Yan Chen[19], Zhenyu Chen[13], Yangming Cheng[42], Yutao Cui[29], Chunyuan Deng[16], Jiahua Dong[32], Matteo Dunnhofer[41], Wei Feng[34], Jianlong Fu[27], Jie Gao[19], Ruize Han[34], Zeqi Hao[13], Jun-Yan He[14], Keji He[20], Zhenyu He[18], Xiantao Hu[17], Kaer Huang[25], Yuqing Huang[18], Yi Jiang[9], Ben Kang[13], Jin-Peng Lan[14], Hyungjun Lee[30], Chenyang Li[14], Jiahao Li[42], Ning Li[17], Wangkai Li[39], Xiaodi Li[42], Xin Li[31], Pengyu Liu[13], Yue Liu[23], Huchuan Lu[13], Bin Luo[14], Ping Luo[33], Yinchao Ma[39], Deshui Miao[18], Christian Micheloni[41], Kannappan Palaniappan[38], Hancheol Park[30], Matthieu Paul[3], HouWen Peng[26], Zekun Qian[34], Gani Rahmon[38], Norbert Scherer-Negenborn[15], Pengcheng Shao[23], Wooksu Shin[30], Elham Soltani Kazemi[38], Tianhui Song[29], Rainer Stiefelhagen[24], Rui Sun[39], Chuanming Tang[37], Zhangyong Tang[23], Imad Eddine Toubal[38], Jack Valmadre[35], Joost van de Weijer[12], Luc Van Gool[3], Jash Vira[35], Stèphane Vujasinović[15], Cheng Wan[16], Jia Wan[8], Dong Wang[13], Fei Wang[39], Feifan Wang[34], He Wang[23], Limin Wang[29], Song Wang[40], Yaowei Wang[31], Zhepeng Wang[25], Gangshan Wu[29], Jiannan Wu[33], Qiangqiang Wu[11], Xiaojun Wu[23], Anqi Xiao[20], Jinxia Xie[17], Chenlong Xu[17], Min Xu[10], Tianyang Xu[23], Yuanyou Xu[42], Bin Yan[13], Dawei Yang[39], Ming-Hsuan Yang[36], Tianyu Yang[22], Yi Yang[42], Zongxin Yang[42], Xuanwu Yin[28], Fisher Yu[3], Hongyuan Yu[28], Qianjin Yu[39], Weichen Yu[10], YongSheng Yuan[13], Zehuan Yuan[9], Jianlin Zhang[37], Lu Zhang[13], Tianzhu Zhang[39], Guodongfang Zhao[21], Shaochuan Zhao[23], Yaozong Zheng[17,19], Bineng Zhong[17], Jiawen Zhu[13], Xuefeng Zhu[23], Yueting Zhuang[42], ChengAo Zong[13], and Kunlong Zuo[28]

[1]University of Ljubljana, Slovenia
[2]Czech Technical University, Czech Republic
[3]ETH Zürich, Switzerland
[4]Linköping University, Sweden
[5]University of Birmingham, United Kingdom
[6]Umeå University, Sweden
[7]Austrian Institute of Technology, Austria
[8]Boston College, USA
[9]ByteDance, China
[10]Carnegie Mellon University, USA
[11]City University of Hong Kong, Hong Kong, China
[12]Computer Vision Center, Spain
[13]Dalian University of Technology, China
[14]DAMO Academy, China
[15]Fraunhofer IOSB, Germany
[16]Georgia Institute of Technology, USA
[17]Guangxi Normal University, China
[18]Harbin Institute of Technology (Shenzhen), China

[19]HuaQiao University, China
[20]Institute of Automation, China
[21]Institute of Computing Technology, Chinese Academy of Sciences, China
[22]International Digital Economy Academy, China
[23]Jiangnan University, China
[24]Karlsruhe Institute of Technology (KIT), Germany
[25]Lenovo Research, China
[26]Microsoft Research, China
[27]Microsoft Research Asia, China
[28]Multimedia Department Xiaomi Inc., China
[29]Nanjing University, China
[30]Nota AI, South Korea
[31]Peng Cheng Laboratory, China
[32]Shenyang Institute of Automation, Chinese Academia of Science, China
[33]The University of Hong Kong, Hong Kong, China
[34]Tianjin University, China
[35]University of Adelaide, Australia
[36]University of California at Merced, USA
[37]University of Chinese Academy of Sciences, China
[38]University of Missouri, USA
[39]University of Science and Technology of China, China
[40]University of South Carolina, USA
[41]University of Udine, Italy
[42]Zhejiang University, China

Abstract

The Visual Object Tracking Segmentation VOTS2023 challenge is the eleventh annual tracker benchmarking activity of the VOT initiative. This challenge is the first to merge short-term and long-term as well as single-target and multiple-target tracking with segmentation masks as the only target location specification. A new dataset was created; the ground truth has been withheld to prevent overfitting. New performance measures and evaluation protocols have been created along with a new toolkit and an evaluation server. Results of the presented 47 trackers indicate that modern tracking frameworks are well-suited to deal with convergence of short-term and long-term tracking and that multiple and single target tracking can be considered a single problem. A leaderboard, with participating trackers details, the source code, the datasets, and the evaluation kit are publicly available at the challenge website[1].

1. Introduction

Visual object tracking remains one of fundamental computer vision problems. The significant progress witnessed in the last two decades has been driven by the research efforts of the community, as well as by the emergence of a multitude of initiatives and challenges aimed at advancing the state-of-the-art in this area. A decade ago, the VOT[1] initiative was founded to address the lack of performance evaluation consensus in visual object tracking. Since, VOT challenges were held in conjuction with all subsequent ICCVs and ECCVs, culminating in the Tenth VOT Challenge, organized last year at ECCV2022. In the last 10 years, the initiative has successfully identified the major tracking trends that got reflected in publications at later major computer vision conferences, making the VOT events central in the tracking community.

Considering the significant challenges in tracking, VOT was restricted to single-target tracking and explored short-term and long-term tracking challenges separately. This approach has provided a suitable environment for exploring novel discriminative frame-to-frame localization mechanisms for short-term tracking and target re-detection and constrained adaptation mechanisms for long-term tracking. Particular attention was given to development and revision of performance measures, evaluation protocols and toolkits. To keep raising the bar for ever-improving tracker methodologies, the target location specification has evolved from reporting bounding boxes in the initial challenges [34, 35, 33, 31, 30, 29, 28] to per-pixel segmentation in the latest challenges [27, 32, 26].

In parallel to VOT, a wealth of impactful activities emerged. Most closely related are UAVision[2], VisDrone[3] and Anti-UAV[4] challenges addressing detection and tracking of pre-defined surveillance-related object types. Another strand of related work are segmentation-oriented multi-target tracking challenges. MOTComplex[5] addresses multiple instance tracking with segmentation and considers four challenges: YouTubeVIS (video instance segmentation); VIS (occluded video instance segmentation); DanceTrack (multi-human tracking); UVO (detect and segment all instances of unknown objects that appear in images or videos). TAO-OW[6] addresses open-world instance tracking, while STEP benchmark[7] addresses tracking instances such as humans and cars along with semantic scene segmentation. LaGOT [42] introduced a validation dataset for multiple-object generic tracking. Pioneered by the DAVIS challenge [11], workshops featuring challenges focusing on video object segmentation (VOS) have emerged as well. Most prominent is the recent YouTube-VOS[8] challenge, which includes video instance and video object segmentation.

The aforementioned datasets and initiatives have opened new exciting challenges and greatly contributed to the field. Nevertheless, they are dedicated to tracking entire object instances and thus tightly coupled with instance detectors. The video segmentation workshops primarily focus on challenging video editing tasks, consider relatively large objects undergoing short-term (partial) occlusions and merely momentary disappearance. As such, they do not directly address the needs of the traditional tracking community interested in the general trackers.

The holy grail of tracking are algorithms for tracking 'any' region, not just known instances, and even individual parts of objects, given a single training example in the first frame. This requires development of efficient general object representation, self-supervised adaptation mechanisms to cope with appearance changes, discriminative models that localize and distinguish the target from the neighborhood and finally, efficient object image-wide re-detection mechanisms to cope with long-term target absence.

[1] https://www.votchallenge.net/vots2023/

[2] CV for UAVs, ECCV 2020, https://sites.google.com/site/uavisionvisdrone2020/

[3] VisDrone challenge, ICCV 2021,http://aiskyeye.com/challenge_2021/

[4] Anti-UAV Challenge, CVPR 2023, https://anti-uav.github.io/dataset/

[5] Multiple Object Tracking and Segmentation in Complex Environments, ECCV 2022, https://motcomplex.github.io/

[6] 2nd Workshop on Tracking and Its Many Guises: Tracking Any Object in Open-World, CVPR 2023, https://taodataset.org/workshop/cvpr23/

[7] Segmenting and Tracking Every Point and Pixel: 6th Workshop on Benchmarking Multi-Target Tracking, ICCV 2021 https://motchallenge.net/workshops/bmtt2021/

[8] YouTube-VOS, The 4th Large-scale Video Object Segmentation Challenge, CVPR2022, https://youtube-vos.org/challenge/2022/

In our opinion, the field has matured to a point where the constraints enforced in the prior VOT challenges can be relaxed and general object tracking should be considered in a broader context. Thus we propose a challenge that no longer distinguishes between single- and multi-target tracking nor between short- and long-term tracking. We propose a single challenge that requires tracking one or more targets simultaneously by segmentation over long or short sequences, while the targets may disappear during tracking and reappear later in the video. The targets may be whole instances or only their parts. To distinguish this new evolutionary stage from the legacy VOT challenges, the new series is called Visual Object Tracking and Segmentation (VOTS) challenges.

This paper presents the first VOTS2023 challenge, organized in conjunction with the ICCV2023 Visual Object Tracking and Segmentation Workshop, and the results obtained. In the following, we overview the challenge and participation requirements.

1.1. The VOTS2023 challenge

The evaluation toolkit and the datasets were provided by the VOTS2023 organizers. The challenge opened on May 4th and closed on June 18th. The results, along with the winners were disclosed in early July. The analysis of the results were presented at ICCV2023 VOTS2023 workshop on October 3rd. The *VOTS2023 Benchmark* opened[9] with a continually updated leaderboard to facilitate tracker development in the post-challenge period.

For the VOTS2023 challenge, the participants integrated their tracker into the VOTS2023 evaluation kit, the new version of the VOT toolkit, which implements the most recent evaluation protocols and the new dataset, and automatically performed a standardized experiment. Each participant then registered the trackers on the evaluation server and submitted the tracker outputs produced in the experiment. Note that only the initialization frames were publicly available, while the ground truth of the remaining frames was sequestered on the server side to prevent overfitting. Furthermore, each registered participant was allowed only 10 attempts to run the evaluation.

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. Each submission was accompanied by a short abstract describing the tracker, which was used for the short tracker descriptions in Appendix A, and a questionnaire to categorize their tracker along various design properties.

Participants with sufficiently well performing submissions (i.e., exceeding the VOTS2023 baseline tracker described in Section 3) who contributed with the text for this paper and agreed to make their tracker code publicly available on the VOTS 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. The VOTS committee members could participate in the challenge with their own submissions, but could not compete for the winner title. All co-authors of this paper, including the VOTS2023 committee members and the tracker authors were required to specify divison of work in Appendix A.

Validation and test splits of popular tracking datasets are *not allowed for training* the trackers. These include OTB [56], VOT, ALOV [1], UAV123 [44], NUS-PRO [1], TempleColor [36], AVisT [46], LaSOT-val [17], LaGOT [42] GOT10k-val/test [1], TrackingNet-val/test [1], TOTB [1]. Training split of any dataset is allowed (including LaSOT-train, TrackingNet-train, YouTubeVOS, COCO, etc.). To include transparent objects, the Trans2k[10] dataset is suggested.

Beyond VOT challenges. The VOTS challenge merges short-term and long-term, single-target and multiple-target tracking, which were until now considered as separate tasks, and considers segmentation as the only target location specification. A new larger dataset with the ground truth withheld was created. New performance measures (that address single, multitarget long- and short-term tracking) and evaluation protocols were created along with the new toolkit and the evaluation server, that features a public learboard.

The remainder of this report is structured as follows. Section 2 describes the new performance evaluation protocol and performance measures, Section 2.2 presents the new dataset, results are discussed in Section 3, conclusions are drawn in Section 4. Short descriptions of the tested trackers and divison of work are available in Appendix A.

2. The VOTS performance evaluation protocol

The tracker is initialized in the first frame on all specified targets. For each subsequent frame, the tracker is required to report the locations for all visible targets in that frame. Specifically, a segmentation mask is required for each visible target, ”not present” label is reported for the absent targets. The tracker is then evaluated with the new performance measures presented below.

2.1. VOTS performance measures

The goal of a multi-target tracker is to reliably track each individual target selected in the first frame. Drifting off a target to the background or another target is both considered failed tracking. This allows the definition of per-target performance measures, which are averaged over all targets

[9] https://eu.aihub.ml/competitions/201

[10] https://github.com/trojerz/Trans2k

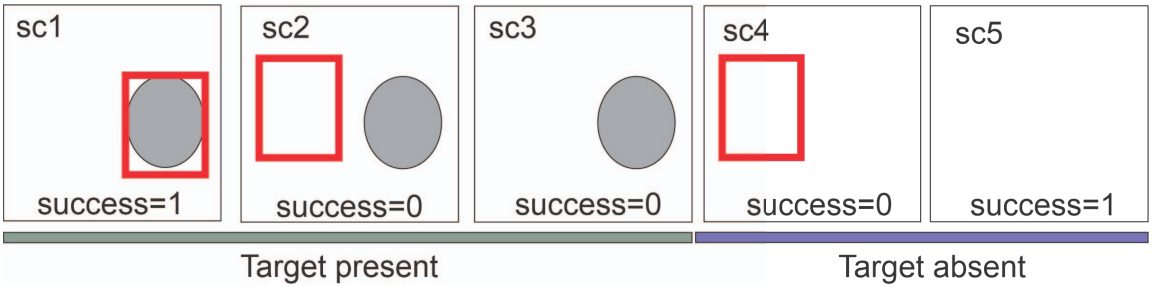


Figure 1. Five scenarios emerge from combinations of target presence and tracker outputs.

to obtain the final score. From the perspective of tracking a single target, five scenarios visualized in Figure 1 are possible. Three scenarios cover cases with the target present: target successfuly localized (sc1), tracker drift (sc2), target incorrectly predicted as absent (sc3). Two scenarios cover the cases with the target absent: target predicted as present (sc4), and target predicted as absent (sc5). In the following we introduce performance measures based on the notion of tracking success that take all these scenarios into account.

Tracking of i-th target on n-th frame of sequence s is considered successful if the predicted target location and the ground truth (i.e., segmentation masks) match sufficiently well. The success is measured by an intersection-over-union (IoU), binarized by some threshold θ (i.e., 1 for values greater than θ, and 0 otherwise). Note that the IoU generalizes well to the case with target absent – if the tracker reports the empty mask in this case (i.e., *target absent flag*), it receives the IoU=1, since the reported mask is in total agreement with the ground truth, otherwise the IoU=0. The overall tracking success for the considered target at threshold θ is thus defined as

$$S(\theta) = \frac{1}{N} \sum_{s=1:N} \frac{1}{T_s N_s} \sum_{i=1:T_s} \sum_{n=1:N_s} [o_{sin} > \theta], \qquad (1)$$

where T_s and N_s are the number of targets and frames[11] in the sequence s, N is the number of sequences and $[o_{sin} > \theta]$ is the operator that binarizes o_{sin} (i.e., the IoU) at a given frame. The performance can be summarized by a tracking quality plot akin to [56] for all thresholds $\theta \in [0, 1)$ as shown in Figure 2. Note that the threshold interval is open, since IoU cannot exceed $\theta = 1.0$, and the definition (1) uses $>$ rather than $\leq$. For visualization purposes, the right-most point is thus evaluated with $[\cdot \equiv \theta]$.

The tracking quality plot has similar interpretation properties as the standard success plot [56], with a difference that the right-most point at $\theta = 1.0$ can be typically higher. The reason is that it accounts for long-term tracking properties in addition to short-term tracking properties. The values IoU=1 can only occur when the prediction completely matches the ground truth (sc1 and sc5 in Figure1). In practice, this is very rare when the target is visible, thus the value is dominated by cases of correctly predicting the target absence (sc5). The practically maximal achievable value will thus be a percentage of the target absent frames in the dataset. This value is indicated in the plot for better interpretation.

[11] Note that the initalization frames are excluded from evaluation, since the tracker does not *predict* the target location at those frames.

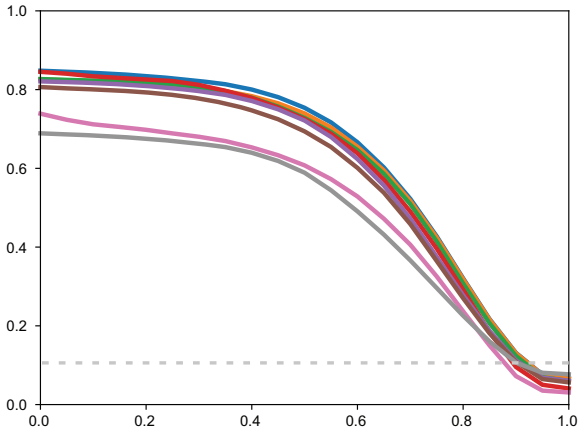


Figure 2. Tracking quality plot with the dashed line indicating percentage of target-absent frames.

The primary VOTS performance measure, called the tracking quality Q summarizes the tracking quality plot by the area under the curve. Following the success plot derivation in [51], it can be shown that the tracking quality is equal to the sequence-normalized average overlap to avoid errors in numerical area-under-the-curve computation, i.e.,

$$Q = \frac{1}{N} \sum_{s=1:N} \frac{1}{T_s N_s} \sum_{i=1:T_s} \sum_{n=1:N_s} o_{sin}. \qquad (2)$$

2.1.1 Secondary performance measures

Additional *secondary performance measures* are proposed for further tracking insights. The first two measures, traditionally used in VOT [27], are localization *accuracy* and *robustness*. The accuracy (Acc) is defined as the sequence-normalized average overlap over successfully

tracked frames, i.e.,

$$Acc = \frac{1}{N}\sum_{s=1:N}\frac{1}{T_s}\sum_{i=1:T_s}\frac{1}{N_{si}}\sum_{n=1:N_{si}} o_{sin}, \quad (3)$$

where N_{si} is the number of successfully tracked frames (i.e., with IoU>0) with the target i visible in sequence s. The tracking robustness (Rob) is defined as the percentage of frames with IoU>0 and target i visible (i.e., a recall),

$$Rob = \frac{1}{N}\sum_{s=1:N}\frac{1}{T_s}\sum_{i=1:T_s}\frac{N_{si}^{\mathrm{sc1}}}{N_s^{\mathrm{sc1+sc2+sc3}}}, \quad (4)$$

where N_{si}^{sc1} is the number of frames with scenario sc1 (Figure 1). Following our prior works [27], the tracker performance on frames with visible target is summarized by the AR plots [26], with the top-right position indicating the better performance.

The next two secondary performance measures answer the question *"Why did tracker fail while the target was visible?"*. The first measure, called *Not-Reported Error* (NRE), gives the percentage of frames where the tracker incorrectly reported the target as absent, i.e.,

$$NRE = \frac{1}{N}\sum_{s=1:N}\frac{1}{T_s}\sum_{i=1:T_s}\frac{N_{si}^{\mathrm{sc3}}}{N_s^{\mathrm{sc1+sc2+sc3}}}, \quad (5)$$

while the second, called *Drift-Rate Error* DRE, gives the percentage of frames where the tracker drifted off the target, and claimed target present, i.e.,

$$DRE = \frac{1}{N}\sum_{s=1:N}\frac{1}{T_s}\sum_{i=1:T_s}\frac{N_{si}^{\mathrm{sc2}}}{N_s^{\mathrm{sc1+sc2+sc3}}}. \quad (6)$$

The final secondary measure answers the question *"How well is the target absence determined?"*. This measure, called *Absence-Detection Quality* ADQ, gives the percentage of frames with target correctly predicted as absent, i.e.,

$$ADQ = \frac{1}{N}\sum_{s=1:N}\frac{1}{T_s}\sum_{i=1:T_s}\frac{N_{si}^{\mathrm{sc5}}}{N_s^{\mathrm{sc4+sc5}}}. \quad (7)$$

Note that in practice, to ensure numerical stability, we consider only those targets, that are absent for at least 10 frames in a sequence.

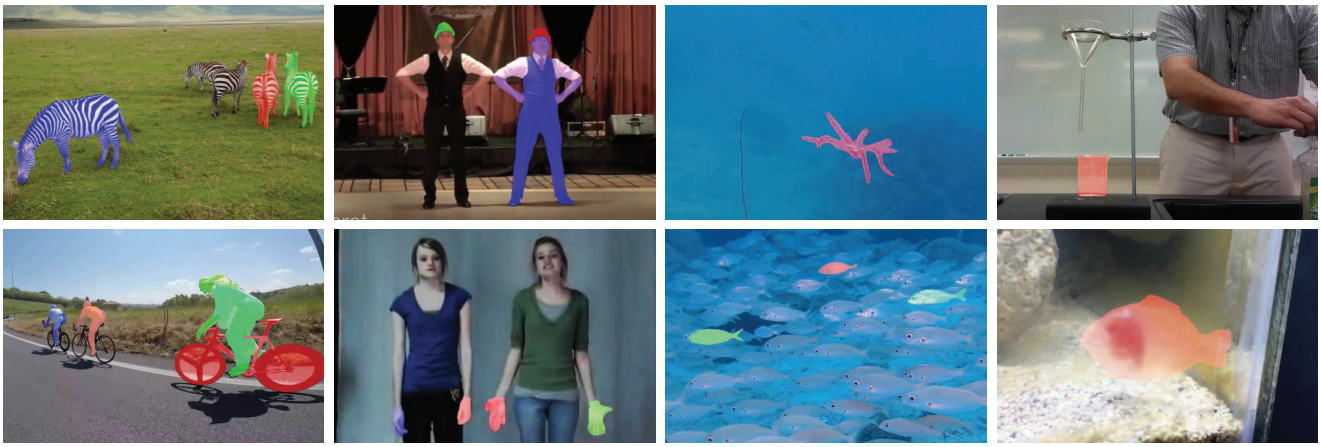
2.2. The VOTS2023 dataset

A new dataset was constructed for evaluation of the new single/multi-target, short/long-term segmentation tracking task considered in the new challenge. This dataset was constructed by including sequences from the following existing datasets: LaGOT [42], VOT-LT2021 [32], VOT-LT2022 [26] UTB180 [2], VOT-ST2022 [26] and TOTB [18]. Note that this does not mean that the same targets were annotated in the final VOTS2023 dataset.

The main selection criterion was to create a dataset containing situations which are in our experience challenging for modern tracking architectures and that cover a wide range of target appearances and object types. We included scenes containing several visually-similar objects, and objects undergoing substantial appearance changes either due to deformation or out-of-plane rotation (e.g., a fish flipping front-to-side). Care was taken to include objects on cluttered backgrounds. Sequences containing objects exiting the field of view and re-entering were considered in addition to sequence with partial occlusions to enable evaluation of long-term tracking properties. We made sure that the sequences covered a diverse set of object types and scenes. For example, in addition to classical air and ground sequences, underwater sequences were considered as well. In addition to opaque objects, several challenging sequences with transparent objects were included to further increase the target diversity.

In most existing tracking benchmarks, the targets are exclusively entire objects. To emphasize the importance of capability to track general appearances, we included also objects, which are a part of other objects (i.e., foot, hat, hand, etc.). In each sequence, potentially several targets was selected in the first frame. The annotation then proceeded in several stages. In the first stage, each of the selected targets was annotated manually by a bounding box (not necessarily in all frames). Then a state-of-the-art bounding box tracker was run to interpolate the missing bounding boxes. All boxes were manually verified and corrected. In the second stage, the bounding boxes were used to guide state-of-the-art segmentation algorithms which gave initial segmentation masks [22, 23]. Finally, the segmentations were manually edited by professional annotators. All annotations were verified by a supervisor and the insufficiently precise annotations were sent back for correction. Figure 3 showcases examples of target diversity and annotation quality.

The final VOTS2023 dataset is composed of 144 sequences, and contains 341 targets in total. The average length of a sequence is ≈2000 frames (min = 63, max = 10700, median = 1810). Number of targets in a sequence ranges from 1 to 8 (median = 2, mean = 2.37). Of the 144 sequences, 93 contain a target which at least once leaves the field of view and then returns. Of the 341 targets, this happens with 168 of them. In cases when the target leaves and returns to the field of view, the median number of absences is 3, with maximum being 23. The median absence length in terms of frame number is 18. For reference, Figure 4 shows the first frames for all 144 sequences.



(a) (b) (c) (d)

Figure 3. VOTS2023 dataset content - examples. (a) Sequences contain multiple targets. (b) Targets may be parts of objects (as opposed to entire objects), e.g. a hat (hat-1 sequence) or hands (hand-9 sequence.) (c) Many sequences are shot underwater, with possibly highly articulated objects (seastar) or multiple distractors (whitefish). (d) Some sequences contain transparent objects (beaker, transparent_fish).

3. Results

A total of 77 trackers was submitted to the evaluation server, including the baselines contributed by the VOTS committee. After removing the duplicate, near-duplicate and incomplete submissions, 47 valid entries remained in the VOTS2023 challenge: DMAOT (A.9), HQTrack (A.42), M-VOSTracker (A.20), Dynamic_DEAOT (A.8), seqtrack (A.38), DMNet (A.12), aot (A.25), MCMOT (A.27), rts_rts50_002 (A.34), VAPT (A.46), MiOTS-ST (A.22), DropTrackSamb (A.11), vttrack (A.47), mmtrack (A.3), MTCTrack (A.29), MixItUp-3 (A.15), MixItUp-2 (A.14), MixFormer (A.23), MixItUp (A.2), PriMem (A.31), UNINEXT_Huge (A.45), SAM-MixFormer, CoCoLoT, MixFormerSAMHDeAOT, T-S-AM, AOTsup, vil_net2, stark_st50_ar (A.40), MixFormerV2 (A.24), UniTD (A.43), alpha_refine_tomp101_seg_000 (A.5), MiOTS (A.21), SAM_Tracker (A.35), alpha_refine_super_dimp_seg_000 (A.4), UNINEXT_R50 (A.44), READMem_MiVOS (A.32), d3sv2 (A.10), LOVD (A.18), starkmulti (A.17), starkplusplus (A.39), Mstark (A.26), MixSAMB (A.36), SRZLT_HSE_IPPM_ClipSegmentAnything (A.37), pytest800_convnext (A.30), ReptileFPN (A.33), TCLCF (A.13), TrackerPRO (A.28).

Each submission included the link to the source code to allow verification of the results if required. The source codes are publicly accessible. In the following we summarize the statistics of the submissions and refer the reader to the Appendix A for the trackers short descriptions.

Of the participating trackers, 13 (28%) were categorized as ST_0, 16 (34%) were categorized as ST_1, 6 (13%) were categorized as LT_0, and 12 (26%) were categorized as LT_1. Most trackers (42; 89%) applied a uniform dynamic model, while (5; 11%) applied a nearly-constant velocity model. The dominant tracking methodology was transformers. In fact, 40 (85%) of the submissions utilized transformers, while 7 (15%) applied deep or classical discriminative filters (in some cases in combination with transformers). Most of the trackers localized the targets in multiple stages (27; 57%), while 20 (43%) performed a single-stage localization. Over a third of the submissions utilized the general object segmentation network SAM [25] (17; 36%), nearly a quarter applied object-specific network AlphaRef [61] for target segmentation or for refining the segmentation (11; 23%), while one quarter directly segmented the target (12; 26%). 14 (30%) trackers applied a fixed template updating mechanism, 19 (40%) updated the template only when confident, 7 (15%) always updated the template and 7 (15%) never updated the template. Majority of the submissions (45; 96%) applied the same network for frame-to-frame localization and target re-detection, while (2; 4%) applied separate methods.

The results are summarized in the tracking quality plots (Figure 2), AR plots (Figure 5) and Table 2. The top 10 trackers according to the primary tracking quality score (Q) are: DMAOT A.9, HQTrack A.42, M-VOSTracker A.20, Dynamic_DEAOT A.8, seqtrack A.38, DMNet A.12, aot A.25, MCMOT A.27, rts_rts50_002 A.34 and VAPT A.46. Of these, 8 are categorized as ST_1 or LT_0, 9 are based on transformers, while rts_rts50_002 is based on deep DCFs, most (7) apply single-stage localization, 6 update their templates at fixed intervals (only 4 when confident), and all apply the same network for frame-to-frame localization and target re-detection.

The top three trackers were all ST_1 single-stage transformer trackers with fixed template updating and a common architecture for frame-to-frame target localization

Figure 4. The first frames of the 144 sequences in the VOTS2023 dataset indicating high diversity in targets, scenes and annotated targets.

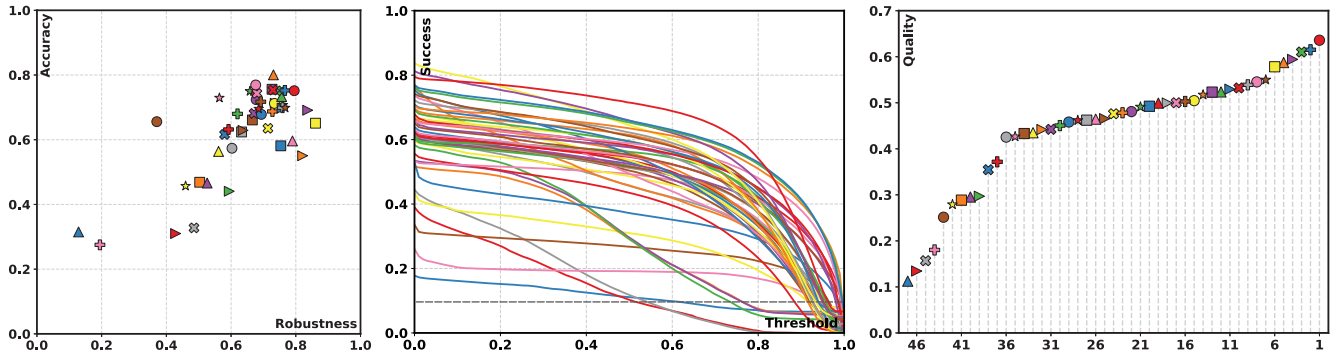


Figure 5. The VOTS2023 accuracy/robustness AR-plots (left), tracking quality Q-plots (center) and all trackers ranked according to Q score (right).

and target re-detection. In particular, the top performer DMAOT (A.9) is built upon the VOT2022 [26] winner AOT [63]. This tracker separates the target-wise long-term memories and updates them only when sufficiently certain, applies a hierarchical gated propagation module (GPM) [64] for better visual embedding propagation and a nearly-constant velocity motion model. The tracker applies a SwinTransformer pretrained on ImageNet [12] and is trained on COCO [37], YouTube-VOS [57], Davis [11], MOSE [13], GOT10k [21], LASOT [17], VIPSeg [43] and OVIS [49]. The next three best-performing trackers HQTRACK A.42, M-VOSTracker A.20 and Dynamic_DEAOT A.8 are similarly designed as DMAOT, i.e., extensions of DeAOT [64].

DMAOT obtains the highest tracking quality (Q=0.6360), which is a 3% improvement over the second-best entry. The AR plot indicates that DMAOT strikes a good balance between accurate target segmentation (Acc=0.751) and very good robustness (Rob=0.795) – the latter indicates that this tracker successfully tracked nearly 80% of an average test sequence length. The tracker drifted off the target in only 7% of cases (DRE=0.07), and falsely predicted the target as absent in 14% of cases (NRE=0.14). This also means that when the target was present, approximately 66% of failures were due to falsely reporting the target absent, and 33% were cases when the tracker drifted off the target while reporting it present. Overall, the target absence was correctly predicted in 73% of cases (ADQ=0.73).

The best robustness was achieved by DMNet (Rob=0.86), which well exceeds the robustness of DMAOT (Rob=0.795). This might be due to application of optimal transport in local correspondence optimization in DMNet, which could be responsible for robust segmentation. However, we note that DMNet also more liberally reports target as present than DMAOT (0.56 ADQ vs 0.73 ADQ). The NRE, which for DMNet is half that of DMAOT, suggests this might be the case, but only under assumption that at least half of the DMAOT false target absent predictions occur when the tracker is actually on the target. The best segmentation accuracy was obtained by Seqtrack, which is a bounding box tracker with SAM [25] segmentation. This might imply very good exploitation and bounding-box-based initialization of SAM.

Considering the above discussed results in terms of the performance scores, additional insights can be made by including the analysis of the Q-plots shapes (Figure 5). Note that the height of the Q-plot at low thresholds indicates the tracker robustness, while the amount by which the graph's "bump" extends to the right (i.e., higher thresholds) indicates the tracker's accuracy. Looking at the trackers that form clusters of Q-plot shapes, it appears that a common property of the trackers that obtain high Q-values at low thresholds is the use of transformer feature extraction backbones, which might lead to high robustness. Similar analysis for medium-to-high thresholds on Q-plot indicates that a common property of many trackers achieving good accuracy is the (careful) use of SAM for segmenting the targets, either initialized by the predicted masks or by predicted bounding boxes.

The VOTS2023 committee provided a baseline tracker for validating the general quality of the submissions. The tracker was created as set of independent STARK [60] trackers which predict the target position by bounding boxes. The tracker called starkmulti A.17 achieved Q=0.297, which is approximately 47% of the Q-score achieved by the top performer. Approximately 80% of submissions outperform the baseline tracker. In addition, the VOTS2023 committee provided a strong state-of-the-art baseline created from the VOT-STs2022 [26] winning tracker AOT [63], which was already designed for multi-object segmentation. 6 trackers (13% of submissions) outperform it, indicating very strong top VOTS2023 submissions.

The VOTS2023 challenge winner. The top tracker according to the tracking quality score Q is DMAOT (A.9) and thus the VOTS2023 challenge winner. A brief analysis indicates that while DMAOT strikes a good balance between the accuracy and robustness, there might be opportunity to further improve performance by reducing the *target-absence threshold* in case substantial percentage of the incorrect target absent predictions occur when the tracker is correctly on the target. The additionally recovered target locations might not be as accurate, which could open an opportunity for improving the performance in those cases by following Seqtrack's strategy with bounding-box conditioned SAM.

4. Conclusion

The first VOTS2023 challenge and results were presented. The challenge merges short-term and long-term, single-target and multiple-target tracking with segmentation as the only target location specification. A new challenging dataset, with the ground truth withheld, was created. New performance measures and evaluation protocols were created along with the new toolkit and an evaluation server, that will hold a public leaderboard.

The paper presents results of 47 trackers. We observe a major increase in the application of transformers. While in VOT2022 [26], 47% of the submissions were from this class, the ratio has increased to 85% in VOTS2023. Interestingly, while the tracking task in VOTS2023 includes both short-term and long-term tracking, only a third of the trackers was categorized as LT. Nearly all trackers apply the same methodological framework for target frame-to-frame localization and long-term re-detection, suggesting that the two tracking classes are indeed methodologically converging. We also observe an increase of single-stage trackers that primarily localize the target by segmentation (close to 43% of submissions). The winner of the VOTS2023 challenge is DMAOT (A.9), which builds upon the VOT2022 [26] winner AOT [63]. After the challenge closure, the *VOTS2023 Benchmark* was opened[12] to facilitate continual evaluation of new general object trackers, primarily segmentation-based.

As bounding-box trackers still dominate the publications at major computer vision conferences and journals, we point out an opportunity for a popular paradigm shift. The workshop results presentation of the last-year's VOT2022 [26] challenge revealed that the best segmentation-based tracker outperformed all bounding-box trackers on the bounding-box tracking task, indicating that bounding-box trackers are rivaled by segmentation trackers not only in accuracy, but also in robustness. This was particularly interesting, since bounding box trackers have been traditionally thought of as more robust than the segmentation trackers, as they estimate a smaller number of output parameters (i.e., bounding box vs per-pixel mask). The VOTS2023 results further support the VOT2022 observations and challenge the traditional belief.

Stronger evidence could be established by the tracking community investing efforts in evaluating their bounding box trackers also on the VOTS2023 challenge in addition to other standard datasets. A bounding box tracker can easily be converted into a segmentation tracker by applying AlphaRef- or SAM-like post-processing on the predicted bounding box. The focus of this experiment should be placed on the robustness measure, which will reveal how well these trackers handle the challenging conditions present in VOTS2023 dataset, and how well they rival the best segmentation trackers. We believe such an effort has a potential to drive development of modern general object trackers towards substantial advancements of the field.

The primary objective of VOT for over a decade has been to establish a platform for discussion of tracking performance evaluation and supporting the tracking community by challenging datasets and toolkits. The VOTS2023 challenge has pushed towards convergence of the tracking tasks, which was a risk, since such trackers are not widely-explored in the community. The remarkable response from the tracking community, which delivered highly competitive trackers in a short time span, encourages us to continue with the efforts in future VOTS editions, and with a hope to witness exciting developments leading to substantial tracking improvements.

Statement of the co-authors contributions

The following abbreviations are used for the VOTS2023 organizers: Matej Kristan (MK), Alan Lukežič (AL), Gustavo Fernandez (GF), Michael Felsberg (MF), Khanh-Tung Tran (TT), Martin Danelljan (MD), Xuan-Son Vu (SV), Johanna Björklund (JB), Jie Zhao (JZ), Yushan Zhang (YZ), Christoph Mayer (CM), Lei Ke (LK), Ondrej Drbohlav (OD), Jiri Matas (JM), Hyung Jin Chang (HJC), Zhongqun Zhang (ZZ), Luka Čehovin Zajc (LCZ).

Dataset construction: MK, AL, MD, CM, LK, OD, JM; Dataset annotation and annotation supervision: OD; Results interpretation: MK, AL; Toolkit development: LCZ, AL; Toolkit team supervision: LCZ; Performance measures design: MK, AL, MF, MD, JM; Paper drafting: MK, GF; Paper proofing: MK, GF, MF, AL; Coordination of public review: GF; Camera ready submission: GF; Evaluation server implementation: TT, SV; Evaluation team supervision: MF, LCZ; VOTS teams coordination:MK; Evaluation server test: JZ, YZ; Evaluation of baselines: CM; Tutorial development: AL, ZZ, LCZ; Tutorial team supervision: AL, HJC, LCZ

The authors of sufficiently well performing trackers con-

[12] https://eu.aihub.ml/competitions/201

	Quality	AR		Auxiliary measures		
Tracker	Q ↑	A ↑	R ↑	NRE ↓	DRE ↓	ADQ ↑
DMAOT	0.636①	0.751	0.795	0.139	0.066	0.731
HQTrack	0.615②	0.752	0.766	0.155	0.079	0.694
M-VOSTracker	0.610③	0.751	0.757	0.159	0.084	0.706
Dynamic_DEAOT	0.594	0.691	0.837②	0.069	0.095	0.568
seqtrack	0.587	0.801①	0.731	0.099	0.171	0.441
DMNet	0.578	0.651	0.861①	0.068	0.071	0.560
aot	0.550	0.698	0.767	0.096	0.137	0.470
MCMOT	0.545	0.769②	0.676	0.156	0.168	0.497
rts_rts50_002	0.539	0.699	0.759	0.068	0.173	0.346
VAPT	0.532	0.753	0.730	0.025	0.245	0.112
MiOTS-ST	0.529	0.698	0.758	0.032	0.211	0.205
DropTrackSamb	0.524	0.734	0.756	0.013	0.231	0.000
vttrack	0.523	0.756③	0.727	0.013	0.261	0.000
mmtrack	0.517	0.707	0.760	0.013	0.227	0.009
MTCTrack	0.505	0.712	0.734	0.013	0.253	0.004
MixItUp-3	0.503	0.717	0.692	0.038	0.271	0.147
MixItUp-2	0.500	0.743	0.680	0.022	0.298	0.046
MixFormer	0.499	0.713	0.736	0.013	0.250	0.013
MixItUp	0.499	0.753	0.675	0.013	0.312	0.000
PriMem	0.493	0.581	0.754	0.158	0.088	0.689
UNINEXT_Huge	0.491	0.750	0.658	0.075	0.266	0.226
SAM-MixFormer	0.481	0.724	0.678	0.022	0.300	0.046
CoCoLoT	0.478	0.687	0.728	0.017	0.255	0.015
MixFormerSAMHDeAOT	0.476	0.635	0.714	0.093	0.193	0.389
T-S-AM	0.465	0.629	0.637	0.166	0.197	0.538
AOTsup	0.464	0.596	0.790	0.027	0.184	0.179
vil_net2	0.462	0.624	0.632	0.178	0.189	0.557
stark_st50_ar	0.462	0.695	0.685	0.016	0.299	0.010
MixFormerV2	0.458	0.678	0.693	0.026	0.281	0.050
UniTD	0.451	0.680	0.620	0.197	0.183	0.473
alpha_refine_tomp101_seg_000	0.442	0.681	0.671	0.022	0.307	0.040
MiOTS	0.442	0.550	0.821③	0.134	0.152	0.442
SAM_Tracker	0.435	0.564	0.561	0.388	0.050	0.868
alpha_refine_super_dimp_seg_000	0.433	0.661	0.666	0.029	0.305	0.056
UNINEXT_R50	0.426	0.729	0.564	0.184	0.252	0.376
READMem_MiVOS	0.425	0.574	0.603	0.332	0.065	0.667
d3sv2	0.372	0.632	0.592	0.042	0.365	0.084
LOVD	0.355	0.616	0.580	0.013	0.407	0.000
starkmulti	0.297	0.441	0.594	0.220	0.186	0.486
starkplusplus	0.295	0.466	0.526	0.337	0.137	0.681
Mstark	0.289	0.469	0.503	0.352	0.145	0.709
MixSAMB	0.279	0.457	0.459	0.013	0.528	0.000
SRZLT_HSE_IPPM_ClipSegmentAnything	0.251	0.656	0.370	0.013	0.617	0.000
pytest800_convnext	0.180	0.275	0.195	0.602	0.203	0.870
ReptileFPN	0.157	0.327	0.485	0.013	0.502	0.000
TCLCF	0.134	0.310	0.429	0.013	0.559	0.000
TrackerPRO	0.112	0.315	0.129	0.669	0.202	0.729

Table 2. Numerical results for VOTS2023 challenge. Tracking quality (Q), accuracy (Acc), robustness (Rob), not-reported error (NRE), drift-rate error (DRE) and absence-detection quality (ADQ).

tributed in the public paper review and tracker descriptions editing. Their contributions to individual trackers are specified in the Appendix A.

Acknowledgements

This work was supported in part by the following research programs and projects: Slovenian research agency research program P2-0214 and project J2-2506, the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. Hyung Jin Chang was supported by the Institute of Information and communications Technology Planning and evaluation (IITP) grant funded by the Korea government (MSIT) (2021-0-00537). Gustavo Fernández was supported by the AIT Strategic Research Programme 2023. Zhenyu He was supported by the National Natural Science Foundation of China (No. 62172126), and the Shenzhen Research Council (No. JCYJ20210324120202006). Xin Li was supported by the National Natural Science Foundation of China (No. 62002241). The challenge was sponsored by the Faculty of Computer Science, University of Ljubljana, Slovenia and the School of Computer Science at the University of Birmingham, UK.

A. Submitted tracker details

This appendix summarizes the VOTS2023 challenge trackers and authors' contributions.

A.1. A Hybrid Approach for Multi-Object Tracking Using DeAOT, Stark, and SAM (SAM-MixFormer)

Authors: E. Soltani Kazemi (esdft@umsystem.edu), I. Toubal, G. Rahmon, K. Palaniappan
Contributions: Conceptualization, ESK, IET, GR, KP; Implementation: ESK, IET, GR; Supervision, KP

This approach proposes a novel hybrid method for multi-object tracking, integrating three leading models: Decoupling Features in Hierarchical Propagation (DeAOT) [64] for video object segmentation, Learning Spatio-Temporal Transformer for Visual Tracking (Stark) [60], and Segment Anything Model (SAM) [25] for image segmentation. The method adapts to the quantity of objects to be tracked. When there are more than five objects, the DeAOT model is used for real-time tracking. For fewer objects, the Stark model tracks objects as bounding boxes, which are then used as prompts to the SAM model to generate high-quality masks. This approach effectively leverages each model's strengths, resulting in a versatile and high-performing solution for multi-object tracking across various scenarios.

A.2. A hybrid method of Mixformer, Stark and Sam for object tracking (MixItUp)

Authors: I. Eddine Toubal (itoubal@mail.missouri.edu), E. Soltani kazemi, G. Rahmon, Kannappan Palaniappan
Contributions: Conceptualization, ESK, IET, GR, KP; Implementation: ESK, IET, GR; Supervision, KP

The tracker employed in this study is a hybrid method that utilizes various algorithms based on the number of objects of interest in the video sequences. When dealing with a small number of objects, the MixFormer tracker [9] is employed, as it excels in accurately estimating bounding boxes. However, in scenarios where a large number of objects are present, the ensemble tracker switches to the Stark tracker [60]. Additionally, the Segment Anything Model (SAM) [25] is utilized to generate masks using the predicted bounding boxes.

A.3. A Simple yet Powerful video-stream Tracker (mmtrack)

Authors: Y. Zheng (20014083057@stu.hqu.edu.cn), B. Zhong, J. Gao, X. Hu, N. Li, C. Xu, J. Xie
Contributions: Conceptualisation, YZ, JG; Implementation, YZ, JG; Validation, YZ, JG, XH, NL, LC, JX; Project leader, BZ

We propose a simple yet powerful video-stream tracker. We adopt ViT [14] as our visual encoder, which 1) models the spatio-temporal trajectory information of the target object by an auto-regressive approach, and 2) propagates rich temporal information about the target by a long short term video-stream manner. Finally, AlphaRefine [61] is used as a segmentation network to predict the target mask.

A.4. alpha_refine_super_dimp_seg_000 (alpha_refine_super_dimp_seg_000)

Authors: G. Bhat (goutam.bhat@vision.ee.ethz.ch), M. Danelljan, C. Mayer, L. Van Gool
Contributions: Conceptualization, GB, MD; Design: GB, MD; Implementation: GB, MD; Validation and Integration, CM; Paper writing: GB; Supervision, LVG

This tracker consists of Super DiMP [3] and uses AlphaRefine [61] to generate segmentation mask using the predicted bounding boxes.

A.5. alpha_refine_tomp101_seg_000 (alpha_refine_tomp101_seg_000)

Authors: C. Mayer (chmayer@vision.ee.ethz.ch), M. Danelljan, G. Bhat, M. Paul, F. Yu, L. Van Gool
Contributions: Idea, CM, MD; Conceptualization, GB, MD, FY; Design, CM, MD; Implementation, CM; Paper writing, CM, MD, GB, MP; Validation and Integration, CM, MP; Supervision: LVG

This tracker consists of ToMP [40] and it uses AlphaRefine [61] to generate segmentation mask using the predicted

bounding boxes. Please see the original paper [40] for more details.

A.6. AOTsup (AOTsup)

Authors: C. Wan (cwan38@gatech.edu), H. Yu, W. Yu
Contributions: Conceptualisation, HY; Implementation, CW; Validation, WY; Project leader, CW

Our multi-target, long-short time tracker, AOTsup, automatically optimizes the tracking model used based on the mask size. It intelligently employs two different models, the MixformerV2 [10] and AOT [63], each activated depending on the specific size ratio. For larger ratios, the MixformerV2 model is utilized due to its superior accuracy. On the other hand, for smaller ratios, AOTsup opts for the AOT model which is renowned for its excellent recall capacity. By doing so, AOTsup capitalizes on the unique strengths of both models, ensuring a combination of accuracy and robustness in our short-term tracking and segmentation capabilities.

A.7. Combining Complementary Trackers in Long-Term Visual Tracking (CoCoLoT)

Authors: M. Dunnhofer (matteo.dunnhofer@uniud.it), C. Micheloni
Contributions: Conceptualisation MD, CM; Implementation MD; Validation MD; Project leader and supervision CM

The single-object CoCoLoT tracker [15, 16] generalizes mlpLT [32]. It implements a strategy that combines the complementary behaviors of Stark [60] and KeepTrack [41] trackers. The combination of these trackers is managed by a decision strategy based on an online learned target verifier akin to MDNet [45]. At every frame, the trackers are run in parallel to predict their target localizations. Based on the evaluation of the target localization, the decision strategy selects the output for the current frame and to correct the tracker that performed worse. Additional strategies such as the computation of adaptive search areas and the avoidance of wrong target size estimations, have been implemented to the baseline trackers in order to make their localizations more consistent. After the bounding-box given by CoCoLoT, AlphaRefine [61] is run to obtain the segmentation mask of each target.

A.8. Decouple Association objects with Dynamic Memory (Dynamic_DEAOT)

Authors: D. Miao (22B951002@stu.hit.edu.cn), X. Li, Y. Huang, Z. He, Y. Wang, M. Yang
Contributions: Conceptualisation and initial idea, DM, XL, M-HY; Implementation and update, DM, XL; Validation, YH; Improvement of the Memory module, ZH, YW; Discussion and idea improvement, M-HY; Project leader, XL, ZH

Dynamic-DEAOT is constructed based on a video object segmentation (VOS) framework borrowed from DEAOT [64] which provides accurate mask predictions and achieves global search. To better handle long-term sequences, we develop a dynamic memory bank to leverage the modeling of long-term and short-term target appearances. In addition, we apply a SOT method (MixFormer) [9] with local search to handle tiny objects by providing a coarse position of the target and then use the segmentation part to generate finer mask predictions. We train our approach on the VOS datasets including YouTube VOS, COCO, and DAVIS using the AdamW optimization method.

A.9. Decoupled Memory AOT (DMAOT)

Authors: Y. Cheng (chengyangming@zju.edu.cn), Z. Yang, Y. Xu, X. Li, J. Li, Y. Yang, Y. Zhuang
Contributions: Conceptualisation, YC, ZY, YX; Implementation, YC; Validation, YC; Information gathering, XL, JL; Project leader, YY, YZ

We propose an adjusted version of DeAOT [64] & AOT [63] called DMAOT that stores object-wise long-term memories instead of frame-wise long-term memories used by AOT. With this object-wise long-term memory, DMAOT ensures that the masks of all objects to be tracked are stored in the memory with a high degree of similarity to the current mask. DMAOT then uses these memories to predict the current object mask, achieving better results.

A.10. Discriminative Single-Shot Segmentation Tracker v2 (d3sv2)

Authors: A. Lukezic (alan.lukezic@fri.uni-lj.si), J. Matas, M. Kristan
Contributions: Conceptualization: AL, MK, JM; Implementation: AL

D3Sv2 [39] is an extended version of the D3S [38]. The original method is extended in the following aspects: (i) a better backbone, (ii) channel attention mechanism in the upscaling modules in GIM, (iii) trainable MLP-based similarity computation in GIM, which replaces the 'handcrafted' top-K average operation and (iv) the new scale estimation module used for robust target size estimation.

A.11. DropTrackSamb: DropTrack with DropMAE pre-training and SAM-base model for mask prediction (DropTrackSamb)

Authors: Q. Wu (qiangqwu2-c@my.cityu.edu.hk), T. Yang, J. Wan, A. Chan
Contributions: Conceptualisation: TY; Implementation QW; validation: JW; Project leader, AC

DropTrackSamb consists of two main modules including a ViT-based DropTrack motion module and a SAM-base segmentation module. The DropTrack employs a pre-

trained DropMAE [55] initialization and uses the standard fine-tuning in OSTrack for downstream tracking representation learning. There is no additional online updating or memory used in DropTrack, and no tracking failure detection is applied since our tracker is a short-term tracker. After obtaining the bounding box predicted by DropTrack, we use it as the box prompt input to the SAM-base model for mask prediction.

A.12. dynamic matching network (DMNet)

Authors: Y. Ma (imyc@mail.ustc.edu.cn), W. Li, D. Yang, R. Sun, Q. Yu, F. Wang, T. Zhang
Contributions: Conceptualization, Method investigation, YM, RS, QY; Implementation and optimization, YM, WL, RS; Test and analysis, DY, FW; Project leader, TZ

We propose a dynamic matching network (DMNet) for pixel-level and part-level matching, which includes a dynamic pixel-aware correspondence module (Pixel-CM) and a dynamic part-aware alignment module (Part-AM). These two modules are trained in an adversarial way, where Pixel-CM generate more accurate mask approaching the ground truth to fool Part-AM. Moreover, Pixel-CM optimizes the correspondences within the local window to reduce false matches and Part-AM divides objects into diverse parts and discriminates detailed local differences between the predicted mask and the ground truth. Finally, we apply test-time augmentations and model ensemble [6, 64] to further improve accuracy.

A.13. Ensemble correlation filter tracking based on temporal confidence learning (TCLCF)

Authors: C. Tsai (chiyi_tsai@gms.tku.edu.tw)
Contributions: Conceptualization, Implementation and Validation C-YT

TCLCF is a real-time ensemble correlation filter tracker based on temporal confidence learning. In the current implementation, we use two different correlation filters to cooperatively track the same target. TCLCF tracker is a high-speed and robust generic object tracker that does not require GPU acceleration. Therefore, it can be implemented on embedded platforms with limited computing resources.

A.14. Ensemble Different Trackers to Make a Robust Single and Multi-Object Tracking (MixItUp-2)

Authors: G. Rahmon (gani.rahmon@mail.missouri.edu), I. Eddine Toubal, E. Soltani Kazemi, N. Al-Shakarji, K. Palaniappan
Contributions: Conceptualization, GR, ESK, IET, KP; Implementation, engineering work, experiments, results and visualization, GR, IET, ESK; Supervision, KP

MixItUp-2 is an ensemble algorithm that adapts to different scenarios based on the number of objects in the video sequences. For scenarios with a number of objects less than 5 in the video sequence, it employs the MixFormer tracker [9]. MixFormer predicts the bounding boxes of the objects, providing accurate position estimations. It is coupled with the Segment Anything Model (SAM) [25], which generates segmentation masks based on the predicted bounding boxes, ensuring precise object identification. In situations with more than or equal to 5 objects present, the ensemble tracker switches to the DeAOT tracker [64]. It utilizes hierarchical feature propagation and attention mechanisms to handle complex scenarios with occlusions and cluttered backgrounds. This enables the tracker to accurately track and distinguish multiple objects in the video sequences. By utilizing the ensemble method and incorporating MixFormer and DeAOT, the tracker ensures robust and accurate object tracking across various scenarios.

A.15. Ensemble Different Trackers to Make a Robust Single and Multi-Object Tracking (MixItUp-3)

Authors: G. Rahmon (gani.rahmon@mail.missouri.edu), I. Eddine Toubal, E. Soltani Kazemi, N. Al-Shakarji, K. Palaniappan
Contributions: Conceptualization, GR, ESK, IET, KP; Implementation, engineering work, experiments, results and visualization, GR, IET, ESK; Supervision, KP

MixItUp-3 is similar to MixItUp-2 (A.14). The difference between both trackers is in the number of objects used to use either the MixFormer tracker or the DeAOT tracker. In case of MixItUp-3 the number of objects is set up as 4.

A.16. Ensemble Different Trackers to Make a Robust Single and Multi-Object Tracking (MixFormerSAMHDeAOT)

Authors: G. Rahmon (gani.rahmon@mail.missouri.edu), I. Eddine Toubal, E. Soltani Kazemi, K. Palaniappan
Contributions: Conceptualization, GR, ESK, IET, KP; Implementation, engineering work, experiments, results and visualization, GR, IET, ESK; Supervision, KP

MixFormerSAMHDeAOT is similar to MixItUp-2 (A.14). The difference between both trackers is in the number of objects used to use either the MixFormer tracker or the DeAOT tracker. In case of MixFormerSAMHDeAOT the number of objects is set up as 2.

A.17. Learning Spatio-Temporal Transformer for Visual Tracking (starkmulti)

Authors: A. Lukezic (alan.lukezic@fri.uni-lj.si)
Contributions: Implementation: AL

Stark [60] is an end-to-end tracking approach based on the transformer methodology, which directly predicts one accurate bounding box as the tracking result. The templates and the search region are concatenated into a sin-

gle feature representation and processed using several self-attention operations to get the final feature representation on which the bounding box prediction is performed. Besides, Stark does not use any hyperparameters-sensitive post-processing, leading to stable performances.

A.18. Linking Open-Vocabulary Detections (LOVD)

Authors: J. Vira (jash.vira@student.adelaide.edu.au), J. Valmadre
Contributions: Conceptualisation, J. Va., J. Vi.; Implementation, J. Vi., J. Va.; Validation, J. Vi., J. Va.; Project leader, J. Va.

LOVD performs tracking-by-detection using a pre-trained open-vocabulary detection model, Grounding DINO [67]. The same prompt of about 80 words is used for all sequences. The visual similarity of two detections is measured using the KL divergence of their likelihoods over text tokens. Detections are filtered using their similarity to the correct detections in the first frame. Contiguous tracklets are constructed by matching detections to tracklets from the previous frame, and unmatched detections initialise new tracklets. Tracklets are associated to past tracks considering temporal overlap and similarity of motion and appearance. Masks are obtained for each box using the pre-trained Segment Anything Model [25].

A.19. long_vil_net2 (vil_net2)

Authors: F. Wang (wff@tju.edu.cn), Z. Qian, R. Han, S. Wang
Contributions: Conceptualisation, FW, ZQ, RH; Implementation, FW, ZQ; Validation, FW, ZQ, RH; Project leader, SW

The primary algorithms utilized include the SAM (Segment Anything Models) [25] for automatic/interactive keyframe segmentation and the DeAOT [64] for efficient multi-object tracking and propagation.

A.20. Memory-based video object segmentation tracker (M-VOSTracker)

Authors: J. Zhu (jiawen@mail.dlut.edu.cn), Z. Chen, Z. Hao, S. Chang, L. Zhang, D. Wang, H. Lu, B. Luo, J. He, J. Lan, H. Chen, C. Li
Contributions: Conceptualisation, JZ; Methodology, JZ, ZC; Validation, ZH, SC; Project leader, LZ, DW, HL; Advice and funding, BL, JH, JL, HC, CL

M-VOSTracker mainly consists of an object segmenter and a mask refiner. The object segmenter is a modified version of DeAOT [64], we extern the gated propagation module to 1/8 scale and employ a more powerful InternImage [53] as our backbone. The segmenter uses multi-object segmentation datasets for training for a better understanding of the relationship between multiple objects and it can handle multiple objects at the same time during one single inference. To further improve the accuracy of tracking masks, we utilize a pre-trained SAM model which is trained on large-scale segmented data to refine our tracking results.

A.21. MiOTS (MiOTS (formerly MiOTS_rushmi))

Authors: H. Yu (yuhongyuan@xiaomi.com), C. Wan, W. Yu, D. An, K. He, A. Xiao, J. Dong, C. Deng, M. Xu, X. Yin, K. Zuo
Contributions: Conceptualisation, MX, XY KZ; Implementation, HY, CW, WY, DA; Validation, CD, KH, AX, JD; Project leader, HY

MiOTS is based on a single-object multi-target tracking segmentation model. For each tracking target, MiOTS initializes a tracker. The MiOTS framework consists of two models: MixformerV2 [10] and AOT [63]. MixformerV2 is an extension of the original model with an input size of 384 and we use the model parameters of SAM [25] as pre-training and retrain the model to obtain a larger MixformerV2 model. The second model AOT is based on the R50 backbone network. We directly use the model parameters provided by the official website for this model. During the tracking process, both MixformerV2 and AOT models run simultaneously. MiOTS then calculates the Intersection over Union (IoU) of the tracking results from both models. If the IoU is less than 0.1, we directly use the results from AOT, as its recall performance is superior. If the IoU is greater than 0.5, we use the results from MixformerV2, as its accuracy is better. Finally, if the IoU falls within the range of 0.1 to 0.5, we use the intersection of the results from both models as the final output.

A.22. MiOTS-ST (MiOTS-ST)

Authors: C. Wan (cwan38@gatech.edu), H. Yu, W. Yu, D. An, K. He, A. Xiao, C. Deng, J. Dong, M. Xu, X. Yin, K. Zuo
Contributions: Conceptualisation, MX, XY KZ; Implementation, HY, CW, WY, DA; Validation, CD, KH, AX, JD; Project leader, HY

MiOTS-ST is based on MiOTS (A.21). In MiOTS-ST, the MixformerV2 and AOT models are independently employed based on size ratio. For ratios exceeding 100, MixformerV2 is used due to its accuracy, while for ratios under 100, AOT is selected for its exceptional recall. This approach effectively leverages the strengths of both models, promoting accuracy and resilience in our short-term tracking and segmentation model.

A.23. MixConvMAE-L (MixFormer)

Authors: T. Song (songtianhui799@gmail.com), Y. Cui, G. Wu, L. Wang
Contributions: Conceptualisation, GW, LW; Implementation TS, YC; Validation TS; Project leader, LW

MixFormer-ConvMAE-Large is an End-to-End Tracking with Iterative Mixed Attention (MixConvMAE-L). MixConvMAE-L consists of two stages which perform MixFormer-based tracking and Alpha-Refine-based segmentation respectively. Our core design is to utilize the flexibility of attention operations, and propose a Mixed Attention Module (MAM) for simultaneous feature extraction and target information integration. MixFormer-ConvMAE-Large is constructed based on pretrained ConvMAE-Large.

A.24. MixFormerV2-Base (MixFormerV2)

Authors: T. Song (songtianhui799@gmail.com), Y. Cui, G. Wu, L. Wang
Contributions: Conceptualisation, GW, LW; Implementation TS, YC; Validation TS; Project leader, LW

MixFormerV2 is a well unified fully transformer tracking model, without any dense convolutional operation and complex score prediction module. We propose four key prediction tokens to capture the correlation between target template and search area. Based on them, we can easily predict the tracking box and estimate its confidence score through simple MLP heads. With our distillation design, MixFormerV2 can achieve excellent tradeoff between performance and inference latency. Besides, we place an Alpha Refine model on top for target segmentation.

A.25. MS-AOT: Associating Objects with Multi-scale Transformers for Video Object Segmentation (aot)

Authors: A. Lukezic (alan.lukezic@fri.uni-lj.si)
Contributions: Implementation: AL

The MS-AOT tracker is built based on AOT [63, 62, 65], a transformer-based video object segmentation method, by applying transformers in multiple feature scales. MS-AOT tracks and segments most of the objects end-to-end without using bounding-box information. For tiny objects, we use MixFormer [9], a bounding-box-based tracker, to coarsely locate the objects before applying MS-AOT to predict segmentation results. The backbone of MS-AOT is ResNet-50, and the backbone of MixFormer is CvT [54].

A.26. Mstark (Mstark)

Authors: J. Reddy (jayatejareddypochimireddy@gmail.com), J. Pochimireddy
Contributions: Conceptualisation, JTR; Implementation, JR, JTR; Validation: JTR

Tracker Mstark is based on the Stark model [60] incorporating two key changes. Firstly, an object presence flag is added to the Stark model. This flag serves as an indicator that determines whether an object is present in the scene or not. Secondly, adjustments to the search region of the model were made expanding the search region by a factor of 3.5. By enlarging the search region, the model has a wider field of view, increasing the likelihood of correctly detecting objects that closely resemble the target object. This modification aims to improve the model's ability to distinguish between similar objects and reduce miss-detections. The object presence flag helps eliminate false positives, while the expanded search region reduces mis-detections of similar objects.

A.27. Multiple Context-based Multi-Object Tracker (MCMOT)

Authors: W. Shin (wooksu.shin@nota.ai), H. Lee, H. Park
Contributions: Conceptualisation, WS, HL, HP; Implementation, WS, HL; Paper writing, HP, WS, HL; Project leader, WS

MCMOT utilizes MixFormer [9] for target location detection and Segment Anything Model (SAM) [25] for object masking. This also involves predicting the position of each target independently at each time step. However, in cases where target templates share similar visual appearances, this independent prediction may result in different templates indicating the same object. To address this, MCMOT incorporates contextual information from the previous time step's predictions. Specifically, the pixels corresponding to the locations of other templates predicted in the previous time step are set to 0 in the input search area for the current template. Additionally, MCMOT combines two online templates, namely the long and short-term templates, to provide a more comprehensive context. By doing so, the model can simultaneously benefit from both templates: preserving appearance features for disappeared objects and handling rapid changes in appearances.

A.28. Multiple Object Tracker by Particle Repropagation and Sparse Optical Flow (TrackerPRO)

Authors: D. Lee (ehdgls@ust.ac.kr), J. Yoo
Contributions: Conceptualisation, D-HL, J-HY; Implementation D-HL; Validation D-HL; Project leader, J-HY

TrackerPRO is based on the iterative particle repropagation method [8], which employs particles and HSV color histograms to improve tracking accuracy. In comparison to the previous algorithm, the particle distribution was changed from Gaussian to circular uniform distribution for initializing the particles on a circle with uniform density at all angles. After that, a calculated optical flow was used to adjust the direction of particles. To track objects of various sizes, contraction and expansion areas were generated around the positions of the particles, and a region with a color distribution more similar to the object was selected between these areas. The tracked object was determined based on the degree of similarity distribution of the particles across the varying regions.

A.29. Multiple Target Cues for Tracking (MTCTrack)

Authors: J. Gao (gaoitjie@163.com), Y. Zheng, B. Zhong, Y. Chen
Contributions: Conceptualisation, JG, YZ, YC; Implementation, JG, YZ; Validation, JG, YZ, YC; Project leader, BZ

MTCTrack exploits the more comprehensive information of the target by multiple target cues. MTCTrack is built on top of OSTrack [66] and utilizes long-term contextual information to propagate the appearance state of the target, explicitly modeling the apparent information of the target. Furthermore, Alpha-Refine [61] is employed to produce a mask prediction as the output.

A.30. OmniTracker_pyetest800_convnext (pytest800_convnext)

Authors: J. Wang (wangjk21@m.fudan.edu.cn), D. Chen, Z. Wu, C. Luo, X. Dai, L. Yuan, Y. Jiang
Contributions: Conceptualisation, JW, DC, ZW, CL, XD, LY, Y-GJ; Implementation, JW

Depending on whether the initial states of target objects are specified by provided annotations in the first frame or the categories. Combining the advantages of the best practices developed in both communities, instance tracking (*e.g.*, SOT and VOS) and category tracking (*e.g.*, MOT, MOTS, and VIS), we propose a novel tracking-with-detection paradigm, where tracking supplements appearance priors for detection and detection provides tracking with candidate bounding boxes for association. Equipped with such a design, a unified tracking model, OmniTracker, is further presented to resolve all the tracking tasks with a fully shared network architecture, model weights, and inference pipeline.

A.31. PriMem: A Memory-based Tracker with prior knowledge (PriMem)

Authors: G. Zhao (zhaoguodongfang21s@ict.ac.cn), K. Huang, Z. Wang
Contributions: Conceptualisation, KH, GZ; Implementation GZ; Validation GZ, KH; Project leader, WZ

This PriMem tracker is built on top of XMem [6], a memory-based single-object tracker in video object segmentation. Compared to the original model, we add the prior knowledge of instances to improve the object tracking in complex scenarios. Furthermore, we adopt a SOTA segmentation model with the incorporation of implicit expression and the original feature vector to assist the generation of segmentation masks.

A.32. READMem-MiVOS (READMem_MiVOS)

Authors: S. Vujasinović (stephane.vujasinovic@iosb.fraunhofer.de), S. Bullinger, S. Becker, N. Scherer-Negenborn, M. Arens, R. Stiefelhagen
Contributions: Conceptualisation, SV, SeB; Implementation, SV; Validation, SV, SeB, StB, RS, Project leader, NSN, MA, RS

READMem_MiVOS is based on READMem (Robust Embedding Association for a Diverse Memory) [52], a modular framework for semi-automatic video object segmentation (sVOS) methods designed to handle unconstrained videos. READMem integrates the embedding of a new frame into the memory only if it increases the diversity of the memory content. Furthermore, it uses a robust association of the embeddings stored in the memory with query embeddings during the update process. The tracker consists of two encoders [20] each for the memory and the query frame, a space-time memory read block [7], a decoder [47] and an external memory which stores previously observed frames as reference. The memory encoder takes an image and the object mask jointly to extract memory-key and -value embeddings, while the query encoder exclusively processes the query image to obtain query-key and -value embeddings. Cross-attention (performed by the space-time memory read block) between the query-key and memory-keys determines relevant information of memory-values, utilized by the decoder to segment the current frame.

A.33. Reptile Meta-Tracking (ReptileFPN)

Authors: C. Tsai (chiyi_tsai@gms.tku.edu.tw), S. Jhang
Contributions: Conceptualisation, C-YT, S-JJ; Implementation C-YT, S-JJ; Validation C-YT

ReptileFPN is a tracker based on FPN model and a meta-learning technique called Reptile. We trained a deep learning network offline by repeatedly sampling different tasks. The resulting network can quickly adapt to any domain without the need to train multi-domain branches like MDNet. The original architecture of Reptile Meta-Tracker used a VGG-like backbone. Here we modified it using FPN to further improve the feature extraction ability. During online initialization, the ReptileFPN tracker only requires a few training examples from the first frame and a few steps of optimization.

A.34. Robust Visual Tracking by Segmentation (rts_rts50_002)

Authors: M. Paul (paulma@vision.ee.ethz.ch), M. Danelljan, C. Mayer, L. Van Gool
Contributions: Conceptualisation, MP, MD; Implementation, MP; Validation MP, CM; Project Leader MD, LVG

RTS [48] is a unified tracking architecture capable of predicting accurate segmentation masks. To design a *segmentation-centric* approach, we take inspiration from the VOS method LWL [4]. However, to achieve robust and accurate segmentation on tracking datasets, we propose several new components. In particular, we propose an instance

localization branch that is trained to predict a target appearance model, which allows the detection of occlusions and to identify the correct target even in cluttered scenes. The output of the instance localization branch is further used to condition the high dimensional mask encoding. This enables the segmentation decoder to focus on the localized target, leading to a more robust mask prediction. Since our proposed method contains a segmentation memory and an instance memory that need to be updated with previous tracking results, we design a memory management module. This module first assesses the prediction quality, decides whether the sample should enter into the memory and triggers the tracking model if it should be updated. See [48] for more details.

A.35. Segment Anything Model based AOT tracker (SAM_Tracker)

Authors: Y. Liu (651382513@qq.com)
Contributions: Conceptualisation, ZyT, YL; Implementation, YL

SAM_Tracker uses segment anything model to generate reference segmentation information for AOT tracker [63], also combined with grounding DINO [67] to generate text prompts for tracker.

A.36. Segment Anything Model based MixFormer Tracker (MixSAMB)

Authors: J. H. Lee (wnsgk986@kitech.re.kr), W. S. Shin, J-H. Lee, S. H. Lee, M. Woo, S. K. Kim, J. Lee, M. Y. Kim, J. P. Yun, H-I. Won, B. H. Kim
Contributions: Conceptualisation, MYK, BHK; Model development, JHL, WSS, JPY; Data development and training, J-HL, SHL, MW, SKK, H-IW; Toolkit and tracker evaluation, JHL, JL, BHK; Project leader, MYK, BHK

The proposed tracker model uses the Segment Anything Model (SAM) [25], which has generalization performance, as a back-end to predict image segmentation information for a target. To predict target area segmentation of the tracking target, we use the predicted result of the bounding box tracker as a prompt for SAM. To generate accurate prompt the MixFormer [9] model is used which has shown an excellent performance for bounding inference. The tracker uses the pre-trained MixFormer-vit-base model, and the SAM uses the pre-trained ViT-base model [14].

A.37. SegmentAnything + Open CLIP (HSE University + IPPM RAS) (SRZLT_HSE_IPPM_ClipSegmentAnything)

Authors: R. Solovyev (roman.solovyev.zf@gmail.com), V. Zunin, D. Lyutkin, A. Romanov, D. Telpukhov
Contributions: Concept and coding of first version, RS; Implementation, DL; Testing, VZ; Validation of results, DT; Project leader, AR

Our method depends on two models. The first model is Segment Anything by Facebook [25] based on VIT-H backbone. This model searches for the regions of interest and it returns a set of masks. The second model is the Open CLIP model[13] which find vectors for each region of interest as well as for all objects to be found. After that, cosine similarities between each proposed mask and each object are found. The mask with the maximum value of metric is chosen for the object. The tracker searches for all the objects at once and it is a zero-shot tracker (i.e. it was not trained on any tracking dataset). The developed tracker is freely available at Github[14].

A.38. SeqTrack: Sequence to Sequence Learning for Visual Object Tracking (seqtrack)

Authors: C. Zong (chengaozong@mail.dlut.edu.cn), X. Chen, P. Liu, B. Kang, Y. Yuan, D. Wang, H. Peng, H. Lu
Contributions: Conceptualisation CZ, XC; Implementation CZ, PL, BK, YY; Funding and Guide DW, HP; Project leader HL

We utilize Seqtrack [5] as our primary tracker, which is based on a straightforward encoder-decoder transformer architecture. The object's bounding box is represented as a sequence of discrete tokens, and the encoder extracts visual features while the decoder autoregressively generates the sequence of bounding box tokens using the extracted features. To address the challenge of distractors, we also employ an auxiliary tracker called KeepTrack [41]. Additionally, we incorporate a basic motion module (trained on LaSOT dataset) to predict the target bounding box When there is an abnormal jump in the results of the tracker. We use the SAM [25] model to predict the mask.

A.39. starkplusplus (starkplusplus)

Authors: J. Pochimireddy (jassu0821p@gmail.com), M. Dasari, A. Kumar, R. K. Gorthi
Contributions: Conceptualisation, RKG, JR; Implementation, JR, MM; Validation, AK; Project Guide, RKG

The modified model is based on the stark [60] single object tracking model and incorporates a yolo [50] detection module to enhance its capabilities. The primary goal of the modification is to address the situation where the tracker loses track of the object being monitored. To achieve this, a yolo detection module is integrated into the model architecture. When the tracker fails to locate the object, the yolo detection module is triggered, allowing the model to perform a new detection to locate and reacquire the object. Additionally, a flag is introduced to provide information about the presence or absence of the object in the current frame. By combining the strengths of the stark model, the yolo

[13] https://github.com/mlfoundations/open_clip
[14] https://github.com/ZFTurbo/VOTS2023-Challenge-Tracker

detection module, and the inclusion of the flag, this modified model offers improved tracking performance by autonomously re-detecting the object whenever it is lost and providing real-time information about its presence.

A.40. STARK-ST50 with Alpha-Refine (stark_st50_ar)

Authors: B. Yan (yan_bin@mail.dlut.edu.cn), H. Peng, J. Fu, D. Wang, H. Lu
Contributions: Coding and implementation, BY; Supervisor, HL

Stark_st50_ar combines Transformer-based STARK [60] with Alpha-Refine [61].

A.41. Track_and_Segment_Anything_Model (T-S-AM)

Authors: F. Wang (wff@tju.edu.cn), Z. Qian, R. Han, W. Feng
Contributions: Conceptualisation, FW, ZQ, RH; Implementation, FW, ZQ; Validation, FW, ZQ, RH; Project leader, WF

We select an effective tracker DeAOT [64] as the baseline tracker for efficient multiple target tracking and propagation. We further apply a large model SAM (Segment Anything Models) [25] for automatic key-frame segmentation.

A.42. Tracking Anything in High Quality (HQ-Track)

Authors: J. Zhu (jiawen@mail.dlut.edu.cn), Z. Chen, Z. Hao, S. Chang, L. Zhang, D. Wang, H. Lu, B. Luo, J. He, J. Lan, H. Chen, C. Li
Contributions: Conceptualisation, JZ; Methodology, JZ, ZC; Validation, ZH, SC; Advice and funding, BL, JH, JL, HC, CL; Project leader, LZ, DW, HL

HQTrack mainly consists of a video multi-object segmenter and a mask refiner. The segmenter is an improved version of DeAOT [64], we cascade a 1/8 scale gated propagation module for perceiving small objects in complex scenarios. Besides, Intern-T is employed as our feature extractor to enhance object discrimination capabilities. Our object segmenter uses multi-object segmentation datasets for training for a better understanding of the relationship between multiple objects. It can handle multiple objects at the same time during one single inference. To further improve the quality of tracking masks, we utilize a pre-trained HQ-SAM model [24] to refine our tracking results. HQ-SAM designs a learnable high-quality output token, which is injected into SAM's mask decoder and is responsible for predicting the high-quality mask. We calculate the outer enclosing boxes of the predicted results of our segmenter as box prompts and feed them into HQ-SAM together with the original image to get the refined results, the final tracking results are selected from the segmenter and refiner. For more implementation details, we refer readers to our technical report [68].

A.43. Unified Object Tracking via Target-aware Disappear Detection (UniTD)

Authors: Z. Tang (7211905025@stu.jiangnan.edu.cn), Y. Liu, P. Shao, H. Wang, S. Zhao, X. Zhu, T. xu, X. Wu
Contributions: Conceptualisation, ZT, YL, XZ, TX; Implementation, ZT, YL, PS, HW; Validation, SZ, XZ; Project leader, XW, TX

We use the baseline Unicorn [58] to solve the SOT and MOT tasks at the same time. To address the new VOTS task, we mainly follow the SOT paradigm, but extending the original single ground truth prompt into multiple ones. To address the problem of disappearance of objects, we further design a target-aware disappear detection method. Redetection is activated when the score is below the threshold, and here the threshold is target-aware. We use a pre-trained to model to deal with all the videos, thus a fixed threshold might not be suitable for all kinds of targets. So the threshold is related to the scores computed in the first frame in this method. Specifically, we endow different thresholds to different objects.

A.44. UNINEXT with ResNet-50 backbone (UNINEXT_R50)

Authors: B. Yan (yan_bin@mail.dlut.edu.cn), Y. Jiang, J. Wu, P. Luo, Z. Yuan, D. Wang, H. Lu
Contributions: Coding and implementation, BY; Supervisor, HL

UNINEXT is a powerful unified model for 10 instance perception tasks. It reformulates 10 instance perception tasks into a prompt-guided object discovery and retrieval fashion.

A.45. UNINEXT with ViT-Huge backbone (UNINEXT_Huge)

Authors: B. Yan (yan_bin@mail.dlut.edu.cn), Y. Jiang, J. Wu, P. Luo, Z. Yuan, D. Wang, H. Lu
Contributions: Coding and implementation, BY; Supervisor, HL

UNINEXT is a unified model for 10 instance perception tasks. UNINEXT_Huge [59] takes ViT-Huge as the backbone. Other settings are aligned with UNINEXT_R50 (A.44).

A.46. ViT-adaptive Dense-Fusion Transformer Tracker (VAPT)

Authors: C. Tang (tangchuanming96@gmail.com), J. van de Weijer, J. Zhang
Contributions: Conceptualisation, CT, JW; Implementation CT; Validation CT; Adviser JZ; Project leader, JW

VAPT is a two-stage video tracking-to-segmentation architecture. The tracker is based on a ViT encoder with an adaptive network, a 4-layer Dense-Fusion Decoder (DFD) and two DCF target prediction heads. The adaptive network is built on 12-layer zero-centred attention blocks which integrate the feature context of each near layer into the same feature space. DFD is built with a target query tensor, four zero-centered attention layers and a project MLP layer. DCF target heads extend convolution layers inspired by ToMP [40] head. The segmentation network is following HQ-SAM [24] to generate high-quality masks of targets based on the predicted bounding box. During inference, we propose a strategy named CycleTrack to correct for errors caused by distractors by verifying temporal cycle consistency. This is based on the insight that the tracked target should track the previous-frame target when tracking backwards in time as a posteriori condition. To improve the long-term adaptive tracking ability, we extend the basic memory update strategy in ToMP, into a staggered template update method. In addition, search-region jitter is another inference strategy in VAPT. It will be applied when the target is lost to re-find it in a large-scale region.

A.47. vision transformer tracking (vttrack)

Authors: P. Liu (liupengyu@mail.dlut.edu.cn), X. Chen, C. Zong, B. Kang, Y. Yuan, D. Wang, H. Peng, H. Lu
Contributions: Conceptualisation PL, XC; implementation PL, CZ, BK, YY; Funding and Guide DW, HP; Project leader HL

We fine-tuned the weights generated using the MAE [19] method on the tracking dataset. We used the VIT-large model. First, both the template and search regions were patch embedded, then concatenated together for feature extraction and fusion through transformer block structure. Finally, the fused features are output to the classification and regression heads to complete the generation of bounding boxes. We apply a Hanning window on the output of the classification head to utilize the motion information of the object. After that, we retrieve the output of the regression head at the position with the highest confidence and output the bounding box. We used Segment Anything Segment Anything Model (SAM) [25] as the model for outputting masks. When the confidence value outputted by the tracker is very low, it is considered that the target is no longer in the image, and an empty mask is outputted.

References

[1] The VOT 2016 evaluation kit. `http://www.votchallenge.net`.

[2] Basit Alawode, Yuhang Guo, Mehnaz Ummar, Naoufel Werghi, Jorge Dias, and Sajid Javed Ajmal Mian. Utb180: A high-quality benchmark for underwater tracking. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 3326–3342, 2022.

[3] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6182–6191, 2019.

[4] Goutam Bhat, Felix Jaremo Lawin, Martin Danelljan, Andreas Robinson, Michael Felsberg, Luc Van Gool, and Radu Timofte. Learning what to learn for video object segmentation. In *European Conference on Computer Vision ECCV*, 2020.

[5] Xin Chen, Houwen Peng, Dong Wang, Huchuan Lu, and Han Hu. Seqtrack: Sequence to sequence learning for visual object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14572–14581, June 2023.

[6] Ho Kei Cheng and Alexander G Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*, pages 640–658. Springer, 2022.

[7] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion. *arXiv preprint arXiv:2103.07941*, 2021.

[8] Jin-Woo Choi, Daesung Moon, and Jang-Hee Yoo. Robust multi-person tracking for real-time intelligent video surveillance. *ETRI Journal*, 37(3):551–561, 2015.

[9] Yutao Cui, Cheng Jiang, Limin Wang, and Gangshan Wu. Mixformer: End-to-end tracking with iterative mixed attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13608–13618, 2022.

[10] Yutao Cui, Tianhui Song, Gangshan Wu, and Limin Wang. Mixformerv2: Efficient fully transformer tracking. *arXiv preprint arXiv:2305.15896*, 2023.

[11] James W. Davis and Hui Gao. Gender recognition from walking movements using adaptive three-mode pca. In *Comp. Vis. Patt. Recognition*, 2003.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[13] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, Philip HS Torr, and Song Bai. Mose: A new dataset for video object segmentation in complex scenes. *arXiv preprint arXiv:2302.01872*, 2023.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, and Sylvain et al. Gelly. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[15] Matteo Dunnhofer and Christian Micheloni. Cocolot: Combining complementary trackers in long-term visual tracking. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 5132–5139, 2022.

[16] Matteo Dunnhofer, Kristian Simonato, and Christian Micheloni. Combining complementary trackers for enhanced long-term visual object tracking. *Image and Vision Computing*, 122, 2022.

[17] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. LaSOT: A High-Quality Benchmark for Large-Scale Single Object Tracking. In *IEEE Conf. Comp. Vis. and Patt. Rec. (CVPR)*, 2019.

[18] Heng Fan, Halady Akhilesha Miththanthaya, Harshit, Siranjiv Ramana Rajan, Xiaoqiong Liu, Zhilin Zou, Yuewei Lin, and Haibin Ling. Transparent object tracking benchmark. In *Int. Conf. Computer Vision*, pages 10734–10743, 2021.

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[21] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *arXiv preprint arXiv:1810.11981*, 2018.

[22] Lei Ke, Martin Danelljan, Xia Li, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Mask transfiner for high-quality instance segmentation. In *CVPR*, 2022.

[23] Lei Ke, Henghui Ding, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Video mask transfiner for high-quality video instance segmentation. In *ECCV*, 2022.

[24] Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Segment anything in high quality. *arXiv preprint arXiv:2306.01567*, 2023.

[25] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023.

[26] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kämäräinen, Hyung Jin Chang, Martin Danelljan, Luka Čehovin, Alan Lukežič, Ondrej Drbohlav, Johanna Björklund, Yushan Zhang, Zhongqun Zhang, Song Yan, Wenyan Yang, Dingding Cai, Christoph Mayer, Gustavo Fernandez, and et. al. The tenth visual object tracking vot2022 challenge results. In *ECCV 2022 Workshops. ECCV 2022. Lecture Notes in Computer Science*, 2022.

[27] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kamärainen, Luka Čehovin, Martin Danelljan, Alan Lukežič, Ondrej Drbohlav, Linbo He, Yushan Zhang, Song Yan, Jinyu Yang, Gustavo Fernández, and et al. The eighth visual object tracking vot2020 challenge results. In *ECCV2020 Workshops, Workshop on visual object tracking challenge*, 2020.

[28] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kamärainen, Luka Čehovin, Ondrej Drbohlav, Alan Lukežič, Amanda Berg, Abdelrahman Eldesokey, Jani Käpylä, Gustavo Fernández, and et al. The seventh visual object tracking vot2019 challenge results. In *ICCV2019 Workshops, Workshop on visual object tracking challenge*, 2019.

[29] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Luka Čehovin, Tomáš Voj íř, Goutam Bhat, Alan Lukežič, Abdelrahman Eldesokey, Gustavo Fernández, and et al. The visual object tracking vot2018 challenge results. In *ECCV2018 Workshops, Workshop on visual object tracking challenge*, 2018.

[30] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Luka Čehovin, Tomáš Vojíř, Gustav Häger, Alan Lukežič, Abdelrahman Eldesokey, Gustavo Fernández, and et al. The visual object tracking vot2017 challenge results. In *ICCV2017 Workshops, Workshop on visual object tracking challenge*, 2017.

[31] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Luka Čehovin, Tomáš Vojíř, Gustav Häger, Alan Lukežič, Gustavo Fernández, and et al. The visual object tracking vot2016 challenge results. In *ECCV2016 Workshops, Workshop on visual object tracking challenge* , 2016.

[32] Matej Kristan, Jiří Matas, Aleš Leonardis, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kämäräinen, Hyung Jin Chang, Martin Danelljan, Luka Čehovin, Alan Lukežič, Ondrej Drbohlav, Jani Käpylä, Gustav Hager, Song Yan, Jinyu Yang, Zhongqun Zhang, Gustavo Fernandez, and et. al. The ninth visual object tracking vot2021 challenge results. In *Proceedings of the IEEE/CVF International Conference on Computer Vision ICCV2021 Workshops, Workshop on visual object tracking challenge*, pages 2711–2738, 2021.

[33] Matej Kristan, Jiří Matas, Aleš Leonardis, Michael Felsberg, Luka Čehovin, Gustavo Fernández, Tomáš Vojíř, Gustav Häger, Georg Nebehay, Roman Pflugfelder, and et al. The visual object tracking vot2015 challenge results. In *ICCV2015 Workshops, Workshop on visual object tracking challenge* , 2015.

[34] Matej Kristan, Roman Pflugfelder, Aleš Leonardis, Ji ří Matas, Fatih Porikli, Luka Čehovin, Georg Nebehay, Gustavo Fernández, Tomáš Vojíř, and et al. The visual object tracking vot2013 challenge results. In *ICCV2013 Workshops, Workshop on visual object tracking challenge*, pages 98 – 111, 2013.

[35] Matej Kristan, Roman Pflugfelder, Aleš Leonardis, Ji ří Matas, Luka Čehovin, Georg Nebehay, Tomáš Vojíř, Gustavo Fernández, and et al. The visual object tracking vot2014 challenge results. In *ECCV2014 Workshops, Workshop on visual object tracking challenge*, 2014.

[36] Pengpeng Liang, Erik Blasch, and Haibin Ling. Encoding color information for visual tracking: Algorithms and benchmark. *IEEE Transactions on Image Processing* , 24(12):5630–5644, 2015.

[37] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014.

[38] Alan Lukežič, Jiří Matas, and Matej Kristan. D3S - A discriminative single shot segmentation tracker. In *Proceedings of the IEEE/CVF CVPR*, pages 7131–7140. IEEE, 2020.

[39] Alan Lukežič, Jiří Matas, and Matej Kristan. A discriminative single-shot segmentation network for visual object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9742–9755, 2022.

[40] Christoph Mayer, Martin Danelljan, Goutam Bhat, Matthieu Paul, Danda Pani Paudel, Fisher Yu, and Luc Van Gool. Transforming model prediction for tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8731–8740, June 2022.

[41] Christoph Mayer, Martin Danelljan, Danda Pani Paudel, and Luc Van Gool. Learning target candidate association to keep track of what not to track. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13444–13454, 2021.

[42] Christoph Mayer, Martin Danelljan, Ming-Hsuan Yang, Vittorio Ferrari, Luc Van Gool, and Alina Kuznetsova. Beyond SOT: it's time to track multiple generic objects at once. *CoRR*, abs/2212.11920, 2022.

[43] Jiaxu Miao, Xiaohan Wang, Yu Wu, Wei Li, Xu Zhang, Yunchao Wei, and Yi Yang. Large-scale video panoptic segmentation in the wild: A benchmark. In *CVPR*, 2022.

[44] Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *Proc. European Conf. Computer Vision*, pages 445–461, 2016.

[45] Hyeonseob Nam and Bohyung Han. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4293–4302, 2016.

[46] Mubashir Noman, Wafa Al Ghallabi, Daniya Kareem, Christoph Mayer, Akshay Dudhane, Martin Danelljan, Hisham Cholakkal, Salman Khan, Luc Van Gool, and Fahad Shahbaz Khan. Avist: A benchmark for visual object tracking in adverse visibility. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, page 817. BMVA Press, 2022.

[47] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9226–9235, 2019.

[48] Matthieu Paul, Martin Danelljan, Christoph Mayer, and Luc Van Gool. Robust visual tracking by segmentation. In *European Conference on Computer Vision ECCV*, 2022.

[49] Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip Torr, and Song Bai. Occluded video instance segmentation: A benchmark. *International Journal of Computer Vision*, 2022.

[50] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[51] Luka Čehovin, Aleš Leonardis, and Matej Kristan. Visual object tracking performance measures revisited. *IEEE Transactions on Image Processing*, 25(3), 2015.

[52] Stéphane Vujasinovic, Sebastian Bullinger, Stefan Becker, Norbert Scherer-Negenborn, Michael Arens, and Rainer Stiefelhagen. Readmem: Robust embedding association for a diverse memory in unconstrained video object segmentation. *arXiv preprint arXiv:2305.12823*, 2023.

[53] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. *CVPR*, 2023.

[54] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *ICCV*, pages 22–31, 2021.

[55] Qiangqiang Wu, Tianyu Yang, Ziquan Liu, Baoyuan Wu, Ying Shan, and Antoni B Chan. Dropmae: Masked autoencoders with spatial-attention dropout for tracking tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14561–14571, 2023.

[56] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Object tracking benchmark. *PAMI*, 37(9):1834–1848, 2015.

[57] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018.

[58] Bin Yan, Yi Jiang, Peize Sun, Dong Wang, Zehuan Yuan, Ping Luo, and Huchuan Lu. Towards grand unification of object tracking. In *European Conference on Computer Vision*, pages 733–751. Springer, 2022.

[59] Bin Yan, Yi Jiang, Jiannan Wu, Dong Wang, Ping Luo, Zehuan Yuan, and Huchuan Lu. Universal instance perception as object discovery and retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15325–15336, 2023.

[60] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning spatio-temporal transformer for visual tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10448–10457, 2021.

[61] Bin Yan, Xinyu Zhang, Dong Wang, Huchuan Lu, and Xiaoyun Yang. Alpha-refine: Boosting tracking performance by precise bounding box estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5289–5298, 2021.

[62] Zongxin Yang, Jiaxu Miao, Xiaohan Wang, Yunchao Wei, and Yi Yang. Associating objects with scalable transformers for video object segmentation. *arXiv preprint arXiv:2203.11442*, 2022.

[63] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

[64] Zongxin Yang and Yi Yang. Decoupling features in hierarchical propagation for video object segmentation. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:36324–36336, 2022.

[65] Zongxin Yang, Jian Zhang, Wenhao Wang, Wenhua Han, Yue Yu, Yingying Li, Jian Wang, Yunchao Wei, Yifan Sun, and Yi Yang. Towards multi-object association from foreground-background integration. In *CVPR Workshops*, volume 2, 2021.

[66] Botao Ye, Hong Chang, Bingpeng Ma, and Shiguang Shan. Joint feature learning and relation modeling for tracking: A one-stream framework. *arXiv preprint arXiv:2203.11991*, 2022.

[67] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, and Heung-Yeung Shum. DINO: DETR with improved denoising anchor boxes for end-to-end object detection. https://arxiv.org/abs/2203.03605, 2022.

[68] Jiawen Zhu, Zhenyu Chen, Zeqi Hao, Shijie Chang, Lu Zhang, Dong Wang, Huchuan Lu, Bin Luo, Jun-Yan He, Jin-Peng Lan, Hanyuan Chen, and Chenyang Li. Tracking anything in high quality. *arXiv preprint arXiv:2307.13974*, 2023.