Beyond SOT: Tracking Multiple Generic Objects at Once

Christoph Mayer[1]* Martin Danelljan[1] Ming-Hsuan Yang[2] Vittorio Ferrari[2]
Luc Van Gool[1] Alina Kuznetsova[2]
[1]ETH Zürich [2]Google Research
{christoph.mayer, martin.danelljan, vangool}@vision.ee.ethz.ch {minghsuan, vittoferrari, akuznetsa}@google.com

Abstract

Generic Object Tracking (GOT) is the problem of tracking target objects, specified by bounding boxes in the first frame of a video. While the task has received much attention in the last decades, researchers have almost exclusively focused on the single object setting. However multi-object GOT poses its own challenges and is more attractive in real-world applications. We attribute the lack of research interest into this problem to the absence of suitable benchmarks. In this work, we introduce a new large-scale GOT benchmark, LaGOT, containing multiple annotated target objects per sequence. Our benchmark allows users to tackle key remaining challenges in GOT, aiming to increase robustness and reduce computation through joint tracking of multiple objects simultaneously. In addition, we propose a transformer-based GOT tracker baseline capable of joint processing of multiple objects through shared computation. Our approach achieves a $4\times$ *faster run-time in case of* 10 *concurrent objects compared to tracking each object independently and outperforms existing single object trackers on our new benchmark. In addition, our approach achieves highly competitive results on single-object GOT datasets, setting a new state of the art on TrackingNet with a success rate AUC of 84.4%. Our benchmark, code, results and trained models are available at* https://github.com/visionml/pytracking.

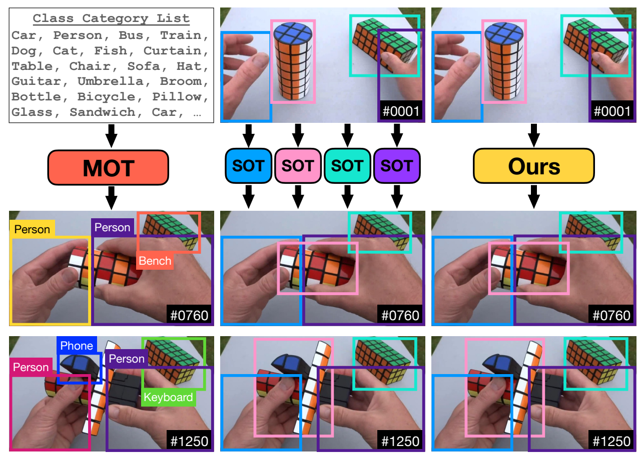


Figure 1. Multiple Object trackers (MOT) track all the objects corresponding to classes in a *predefined* category list, while all other objects are ignored. Single Object Tracking (SOT) methods focus on tracking only a single user-specified object per video. Thus, when encountered with multiple objects, such methods must resort to independent tracking of each object. This leads to a directly linear increase in computation. Our tracker can track multiple *generic* objects jointly that are defined via user-specified bounding boxes, leading to the opportunity of computational savings and to exploit inter-object information for improved robustness. The box colors correspond to track IDs.

1. Introduction

Visual object tracking is a fundamental problem in computer vision. Over the years the research effort has been directed mainly to two different task definitions: Generic Object Tracking (GOT) [2,4,10,21,27,29,53] and Multiple Object Tracking (MOT) [5, 13, 19, 48, 61–63]. MOT aims at detecting and tracking all objects from a predefined class category list (see Fig. 1), whereas all other objects are ignored. In contrast, GOT focuses on the scenario where a priori information about the object's appearance is unknown. Thus, the target model of the object's appearance must be learned at test time from a single user-specified bounding box in the initial frame, see Fig. 1.

While GOT has a long history of active research, GOT methods and benchmarks focused so far on tracking a single object per video such that the term *Single Object Tracking (SOT)* was introduced. However, the task of GOT is not limited to tracking a single object. In fact, the ability to track multiple generic objects is desired in many real-world applications, such as surveillance, video understanding, semi-automatic video annotation, robotics, and industrial quality control. A method that jointly tracks multiple objects can achieve substantial reduction in computational cost through shared elements, compared to running a separate instance of

*Work done while interning at Google Research.

a SOT method for each object. Moreover, processing multiple targets at the same time has the potential of increasing the robustness of the tracker by joint reasoning.

To facilitate the work on tracking multiple generic objects, we introduce the new multi-object GOT benchmark LaGOT. It provides up to 10 user-specified generic objects in the initial frame visible through the large part of the video. The target objects in one video may correspond to completely different and previously unseen classes. Our benchmark features challenging characteristics such as fast moving objects, frequent occlusions, presence of distractors, camera motion, and camouflage. In total LaGOT contains 528k annotated objects of 102 different classes and an average track length of 71 seconds.

Tracking multiple target objects in the same video poses key challenges and research questions that are typically overlooked by SOT methods. A multi-object GOT method needs to jointly track multiple objects using the first-frame annotations. This could allow the tracker to exploit annotations of potential distractors to improve the robustness of each target model. Furthermore, a joint localization step opens the opportunity for global reasoning across all tracks to reduce the risk of confusing similar objects. Finally, operating on multiple local search area [7, 38, 57] is no longer feasible for a multi-object GOT method because it is inefficient and complicates re-detecting of lost objects.

We tackle these challenges by introducing a new multi object GOT tracker. In order to track all desired target objects at once it operates globally by processing the full frame producing a shared feature representation for all targets. Furthermore, we propose a new generic multiple object encoding that allows us to encode multiple targets within the same training sample. We achieve this by learning a fixed size pool of different object embeddings, each representing a different target. Thus, we query the proposed model predictor with these object embeddings to produce all target models. In addition, we employ a Feature Pyramidal Network (FPN) to increase the overall tracking accuracy while operating on full-frame inputs.

Contributions. **(i)** We propose a novel large-scale multi-object GOT evaluation benchmark, LaGOT. It provides multiple annotated objects per frame with an average of 2.9 tracks per sequence. We further evaluate several baselines on LaGOT, including two MOT and six SOT methods. We assess their quality by using GOT and MOT metrics.
(ii) We develop a new baseline, TaMOs, a GOT tracker that tracks multiple generic objects at the same time efficiently. To achieve this, we propose a new multi-object encoding, introduce an FPN and apply the tracker globally on the entire video frame. TaMOs demonstrates near constant run-time when increasing the number of targets and operates at an over $4\times$ faster run-time compared to the SOT baselines when tracking 10 objects.
(iii) We analyze TaMOs by assessing the impact of its different components using multiple benchmarks. Furthermore, TaMOs outperforms all baselines on LaGOT, while achieving excellent results on popular SOT benchmarks.

2. Related Work

Object Tracking Benchmarks. Generic object tracking is a well explored topic and many datasets exist. There are specialized datasets and challenges that focus on short-term [17, 23, 25, 27, 41, 53] or long-term tracking [14, 15, 25, 40, 50]. However, all of these benchmarks and datasets share the same setup of only providing a single user-specified bounding box such that only one target is tracked in each video sequence. Recently, GMOT-40 [1] focused on Generic Multi Object Tracking (GMOT), where a single bounding box is provided in the first video frame and all objects that correspond to the same class as the annotated object should be tracked. In contrast to GMOT, we focus on the setting where multiple user-specified targets are given, potentially from different classes.

MOT aims at tracking multiple objects defined by a list of classes and mainly focuses on a single class [13, 24, 48] (usually pedestrians) or on autonomous driving settings, where only a handful of classes are considered [5, 19, 61]. TAO [12] contains objects of a long-tailed class distributions, but provides only sparse annotations due to the costly annotation process. Another related task is open world tracking [34] that aims at detecting and tracking all objects in a video. However, compared to GOT there is no mechanism to guarantee that a specific object is actually detected and tracked. In the Video Object Segmentation (VOS) domain, DAVIS [45] and YouTubeVOS [54] provide multi-object annotations. However, their videos are extremely short (2.9 and 4.5 seconds on average), and are therefore not suitable for tracking. Moreover, the VOS domain provides less challenges for trackers, instead focuses on large objects and a short-term nature, where the predominant challenge is the prediction of accurate fine-grained masks.

Global Generic Object Tracking. Global trackers operate on the whole video frame, rather than in a restricted search area near the object location in the previous frame. This is not only beneficial when tracking multiple objects in the same scene but also facilitates re-detecting lost objects. GlobalTrack [22] and Siam R-CNN [52] track the target by using global RPNs that retrieve target-specific proposals. Recent method for open vocabulary tracking [31] tracks objects of specified classes in MOT fashion by operating on generic RPN proposals. Methods such as MetaUpdater [8] and SPLT [58] operate on local search areas but use a re-detector to re-localize the target if it disappeared from the search area. In contrast, our tracker TaMOs always operates on the entire frame and generates target specific correlation

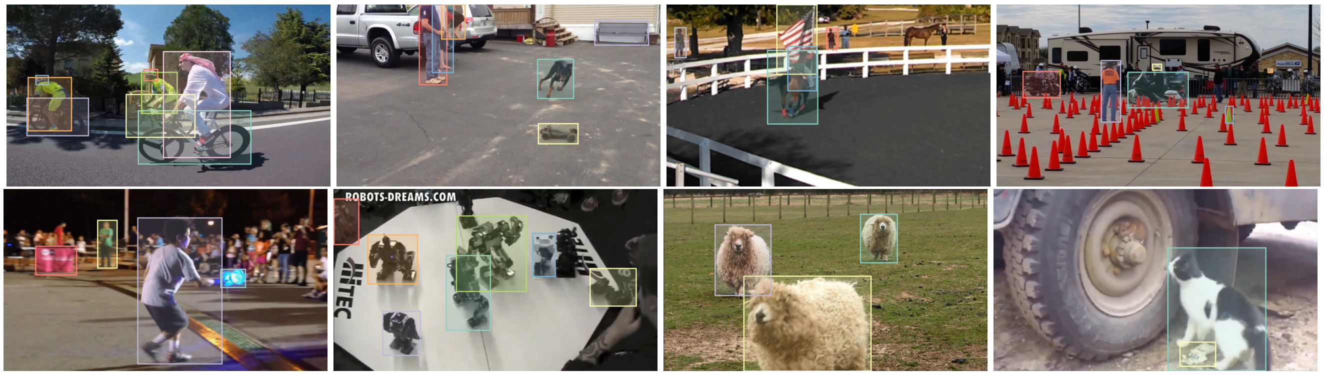


Figure 2. Examples of the annotated objects in the video sequences of our LaGOT dataset. The objects are annotated at 10 FPS. Notice the diversity of the annotated media as well as the complexity of the scenes.

filters instead of target-specific proposals.

Unified Object Tracking. Unified methods aim at tracking both: objects defined by class names or objects defined by a user-specified bounding box. UTT [37] allows to track all pedestrians and one generic object in each video. UTT uses a Transformer to match test frame features with reference features of the detected objects in the initial or previous frame. Unicorn [56] allows to either perform the SOT or the MOT task with the same model and weights solely by varying the input data type. In contrast, our method tracks multiple generic objects at the same time instead of one generic object or multiple objects of known classes.

Transformers for Generic Object Tracking. Tracking has seen a tremendous progress in recent years with the advent of Transformers [51]. Most such trackers share the idea of fusing the search area and the template image features by using a Transformer [6, 7, 38, 57, 59, 60]. MixFormer [7] and OSTrack [59] employ a Transformer to jointly extract and fuse the template and search area features. TransT [6], STARK [57], SwinTrack [32] and ToMP [38] use a backbone to extract features and employ cross attention to fuse the feature representations. However, none of these trackers can easily be extended to jointly track multiple objects, which is addressed in this work.

3. LaGOT Benchmark

In this section we first introduce the multi-object GOT task and discuss its differences to other object tracking tasks. Then, we introduce our new benchmark LaGOT.

3.1. Multi-object GOT Task

Multi-object GOT is the task of tracking multiple generic target objects in a video sequence. The target objects are defined by user-specified bounding boxes in the initial frame of the video. Thus, the target objects are generic in the sense that their class category is unknown and there might be no object of the same category in the training data, see Fig. 1.

Multi-object GOT vs. SOT. SOT requires to track only a single target object defined by the user [15, 27, 53], whereas multi-object GOT focuses on tracking multiple user-specified generic target objects in the same video.

Multi-object GOT vs. MOT. **(i)** The MOT task requires to track all objects of known classes, whereas for multi-object GOT target objects in each video are defined by user-specified boxes. Consequently, multi-object GOT is a one-shot problem where the target objects are unknown at training time and are only available during inference. In contrast, traditionally MOT methods track all objects corresponding to the categories defined at training time. **(ii)** For the multi-object GOT task an object-id switch is equivalent to a complete failure since the user-specified object is no longer recoverable [36, 53]. Conversely, for MOT methods object-id switches are considered less problematic and are penalized less drastically by the MOT metrics [35].

Multi-object GOT vs. GMOT. GMOT focuses on tracking multiple objects of a single generic object class in each video. The class is defined by a single user-specified bounding box in the initial video frame [1, 16]. Thus, in contrast to multi-object GOT, a GMOT method is unable to track multiple objects of different categories in the same video.

3.2. LaGOT

Benchmark Construction. LaSOT [15] contains diverse and relatively long videos (2430 frames or 81 seconds on average) with challenging tracking scenarios including fast moving objects, camera motion, various object sizes, frequent object occlusions, scale changes, motion blur, camouflage and objects that go out of view or change their appearance. LaSOT provides annotations for a single object in each video but typically multiple objects are present throughout the full sequence and are fairly difficult to track, which is desirable for long-term tracking scenarios. Thus, instead of collecting new videos, we used the popular LaSOT evaluation set and add new annotations for multiple objects in each sequence.

Another large-scale video dataset we considered is

Table 1. Comparison of LaGOT with existing benchmarks that focus on related tasks to multi-object GOT.

Dataset	Task	Object Definition	Num Classes per Video	Tracking Metrics	Num Classes	Num Videos	Avg Video Length (s)	Avg Track Length (num anno.)	Avg Tracks per Video	Num Annotations	Annotation Frequency
TAO val [12]	MOT	class list	≥ 1	Track-mAP	302	988	33.5	21	5.55	115k	1 FPS
GMOT-40 [1]	GMOT	1 box	1	MOTA/IDF1	30	40	10	133	50.65	486k	24-30 FPS
LaSOT val [15]	SOT	1 box	1	Success AUC	70	280	81	2430	1	680k	30 FPS
LaGOT	GOT	n boxes	≥ 1	F1-Score	102	294	75.3	707	2.89	528k	10 FPS

TAO [12] and GMOT-40 [1]. However, compared to LaSOT, TAO contains shorter videos with an average of 33 seconds and its outdoor and road sequences mainly focus on pedestrians and vehicles (60% of all objects in TAO). While the indoor sequences contain rarer object categories, they are often static and are only visible for a short time. Furthermore, TAO contains only sparse annotations (1 FPS). For all these reasons, we used LaSOT instead of TAO to build our benchmark. GMOT-40 [1] contains dense annotations, but videos often contain many objects of a single class. Furthermore, GMOT-40 consists of only 40 short sequences (avg 240 frames or 8 seconds) rendering only 10 different object classes, see Tab. 1. Thus, GMOT-40 is unsuitable to serve as a multi-object GOT benchmark.

Annotation Protocol. First, we inspect all 280 sequences in LaSOT and identify in each video challenging target objects that play an active role and meet the previously specified criteria. Next, we entrust professional annotators to annotate the selected objects in all sequences on every third frame, leading to an annotation frequency of 10 FPS. They use an interactive annotation tool which incorporates an object tracker to speed up the annotation process [28]. A group of researchers verifies the newly obtained annotations and sends low-quality annotations back for correction until all annotations meet our high quality standards. Finally, we post-process the annotations to construct the final tracks. First, we remove all tracks shorter than 4 seconds. Second, we define the starting frame by manually selecting the earliest frame where as many annotated objects as possible are clearly visible. Third, it is not always possible to unambiguously associate all object identities over time due to occlusions and out-of-view events — hence, we either remove ambiguous annotations or cut these videos into multiple sub-sequences, where the object association is clear. We follow this protocol to guarantee a high annotation quality, see Fig. 2 for annotated example frames.

Statistics. Our benchmark LaGOT has 294 videos with 850 tracks leading to over 528k annotated objects. Thus, we almost triple the number of tracks compared to the original LaSOT validation set (and the corresponding evaluation time from 378 to 1006 min). Furthermore, we add 31 additional generic object classes, *e.g*. propeller, tires or fabric bag. We compare the proposed benchmark with the most closely related benchmarks in Tab. 1 (and with many more Tab. 2 in suppl. material). Overall our benchmark contains $10\times$ more class categories than GMOT-40. The average track length of LaGOT is 2121 frames (707 annotated frames), which is $3\times$ longer than in TAO, and almost $10\times$ longer than in GMOT-40.

Annotation Frequency. According to Valmadre *et al*. [50] it is more effective to spend a fixed annotation budget on many videos with sparse box annotations than on fewer videos with dense labels. Thus, we annotate every third frame to reduce the overall annotation cost. To analyze the difference between 10 and 30 FPS annotations, we evaluate five recent trackers on the tracks borrowed from LaSOT, where 30 FPS annotations are available. The mean relative error of the success rate AUC is only 0.237%. This shows that 10 FPS is sufficient on large-scale datasets such as LaSOT and LaGOT, leading to only minor score deviations.

4. Method

In this section we present our tracker TaMOs, which employs a Transformer to jointly model and track a set of arbitrary objects defined in the initial frame of a video. We start from ToMP [38], a recent Transformer-based generic single object tracker that operates on local search area cropped from the full frame, as almost all SOT trackers. ToMP employs a transformer to predict a correlation filter (target model) from the target appearance in the initial frame conditioned on the new frame; the predicted target models is later used to localize the target in the subsequent frames. In Sec. 4.1 we introduce the proposed Transformer-based multi-object tracking architecture and in Sec. 4.2 we discuss the used training protocol.

4.1. Generic Multi-Object Tracker - Overview

An overview of the proposed generic multi-object tracker TaMOs is presented in Fig. 3. First, unlike original ToMP, our tracker operates on the full train and test images instead of crops. The target object encoder uses a pool of learnable object embeddings to encode the location and extent of each target object within a single shared feature map (Sec. 4.1.1). The randomly sampled object embedding then represents a particular target in the entire video sequence: we use the object embedding to condition the model predictor to produce the target model that localizes the target object in the test frame (Sec. 4.1.2). Since operating on the entire video frame increases the computational cost of the Transformer operations, we are limited to a certain fea-

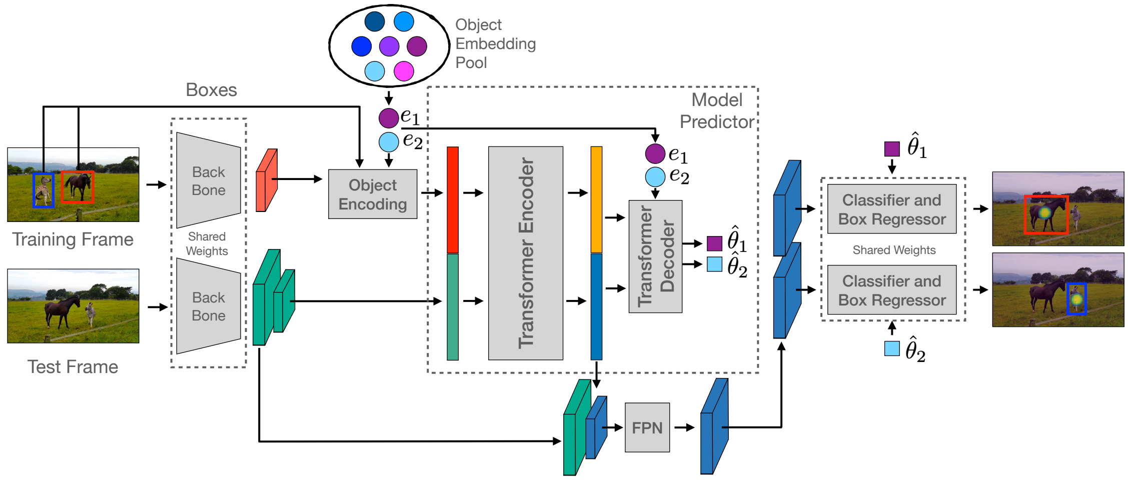


Figure 3. Overview of our tracker TaMOs for joint tracking of multiple targets. First, we extract features from training and test frames. All objects in the training frame are encoded jointly with a multi-object encoding and passed to the model predictor together with the training frame features. The model predictor produces target models $\hat{\theta}_i$ together with enhanced test features. We apply an FPN on the enhanced output features to generate higher resolution test features. Finally, we predict the bounding box of each target by applying the target model $\hat{\theta}_i$ for each target.

ture resolution. To track small objects we propose an FPN-based feature fusion of the test frame features produced by the Transformer with the higher resolution backbone features. We adopt the correlation filter based target localization and bounding box regression mechanism of ToMP but apply both on the higher resolution FPN features instead of the output features of the Transformer (Sec. 4.1.3).

4.1.1 Generic Multiple Object Encoding

To track several target objects efficiently, we propose a novel object encoding to embed multiple objects in a shared feature map without requiring multiple templates.

In particular, we extend the single object encoding formulation of ToMP to be applicable for multiple objects. The idea is to replace the foreground embedding with multiple object embeddings, each representing a different target object. Thus, we create a pool $E \in \mathbb{R}^{m \times c}$ of m object embeddings $e_i \in \mathbb{R}^{1 \times c}$. Then, we sample for each target object a random object embedding from the pool E without replacement. Next, we combine the object embeddings with the Gaussian score map $y_i \in \mathbb{R}^{h \times w \times 1}$ that represents the center location of the target object i and the LTRB [49, 55] bounding box encoding $b_i^{\text{ltrb}} \in \mathbb{R}^{h \times w \times 4}$. The final encoding is thus:

$$f_{\text{train}}^{\text{enc}} = f_{\text{train}} + \sum_{i=0}^{n} e_i \cdot y_i + \sum_{i=0}^{n} e_i \cdot \phi\left(b_i^{\text{ltrb}}\right), \qquad (1)$$

where $f_{\text{train}} \in \mathbb{R}^{h \times w \times c}$ are visual features extracted from the full training frame, ϕ is a Multi-Layer Perceptron (MLP) and $n \leq m$ is the number of tracked objects. Note, that in contrast to the object encoding in ToMP, we not only use the object embedding to encode the Gaussian score map but also the bounding box representation. The object embeddings e_i are learned during training such that the model is able to disentangle the shared feature representation and can identify each object in the training and test features. Note, that the products in Eq. (1) employ multiplications with broadcasting across every dimension whereas the latter uses channel-wise multiplication with broadcasting across the spatial dimensions.

4.1.2 Joint Model Prediction

Now that the target object locations and extents are embedded in the training features, we require a model predictor to produce a target model for each encoded object. The target models are then used to localize the targets in the test frame and to regress their bounding boxes. In order to easily associate the different targets over time, we require a model predictor that can be conditioned on the targets encoded through object embeddings e_i. Furthermore, the model needs to be able to produce all target models jointly to increase the efficiency.

We extend the single target model predictor of ToMP by keeping the Transformer encoder unchanged but by modifying the Transformer decoder. In particular, we query the Transformer decoder with multiple object embeddings e_i at the same time instead of a single foreground embedding,

$$[\hat{\theta}_1, \ldots \hat{\theta}_n] = T_{\text{dec}}([h_{\text{train}}, h_{\text{test}}], [e_1, \ldots e_n])\,. \qquad (2)$$

Here, $\hat{\theta}_i \in \mathbb{R}^c$ is the target model, n is the number of target objects encoded in the training frame and h_{train}, h_{test} are the refined output features of the Transformer encoder for the train and test frame.

4.1.3 Target Localization and Box Regression

We use the generated target models to localize the targets and to regress their bounding boxes. We produce a correlation filter for target classification and adopt the bounding box regression branch of ToMP [38]. But instead of applying the target classifier and box regressor on the low-resolution test features h_{test} of the Transformer encoder, we use high resolution features generated with an FPN $\psi(\cdot)$ and obtain the high-resolution multi-channel score map:

$$\hat{y}_i^{\text{high}} = w_i^{\text{cls}}(\hat{\theta}_i) * \psi(h_{\text{test}}, f_{\text{test}}^{\text{high}}), \quad 0 \leq i < n, \tag{3}$$

where $f_{\text{test}}^{\text{high}} \in \mathbb{R}^{2h \times 2w \times c}$ are the high-resolution test features extracted at an earlier stage of the backbone, $w_i^{\text{cls}}(\hat{\theta}_i)$ refers to the discriminative correlation filter for the target object i obtained from the predicted target model $\hat{\theta}_i$. Similarly we obtain the high-resolution multi-channel bounding box regression maps $\hat{b}_i^{\text{high}}$.

4.2. Training

During training we employ a classification and a bounding box regression loss. We compute both losses for the predictions obtained by processing each FPN feature map (low-res and high-res) as well as the output test features h_{test} of the Transformer encoder. The classification loss is

$$L_{\text{cls}} = \sum_{i=0}^{n} L_{\text{focal}}(\hat{y}_i, y) + \sum_{j=n}^{m} L_{\text{focal}}(\hat{y}_j, 0), \tag{4}$$

Here we assume that the first n object embeddings e_i were used to encode the n objects marked in the training frame whereas the remaining $m - n$ object embeddings were not used to encode any objects. Thus, we require that the resulting score maps $\hat{y}_j$ that correspond to an unused object embedding e_j produce low scores everywhere (second sum in Eq. (4)). This step tightly couples the object encoding and decoding. Omitting this term not only decreases the overall performance but slows down the training progress.

In contrast to classification, we enforce the generalized IoU-Loss [46] for bounding box regression only for the predictions that actually correspond to an encoded object and ignore those corresponding to unused object embeddings.

Training Details. We randomly sample an image pair consisting of one training and one test frame from a training video. The frames are re-scaled and padded to a resolution of 384×576. We train our tracker on the training splits of LaSOT [15], GOT10k [23], TrackingNet [41], MS-COCO [33], ImageNet-Vid [47], TAO [12], and YoutubeVOS [54]. Note, that we remove all videos from the TAO training set that overlap with the evaluation set of LaSOT. We randomly sample for each epoch 40k image pairs with equal probability from all datasets. In order to leverage SOT datasets and training all object embeddings e_i equally, we assign random object ids to all objects in the sampled training pair. Note, that both SOT and MOT datasets are crucial to train the proposed tracker. Without MOT datasets the tracker is unable to learn multiple target models at the same time and avoiding SOT datasets leads to inferior tracking quality. We train the tracker for 300 epochs on 4 Nvidia A100 GPUs. Our method is implemented using PyTracking [9] (see suppl. material for further details).

5. Experiments

To illustrate the challenges of our proposed GOT benchmark, we evaluate several recent trackers along with our proposed tracker TaMOs on LaGOT (Sec. 5.1). In addition, we compare TaMOs to recent trackers on several SOT benchmarks (Sec. 5.2) and present an ablation study (Sec. 5.3), evaluating the impact of different components of our tracker.

5.1. State-of-the-Art Evaluation on LaGOT

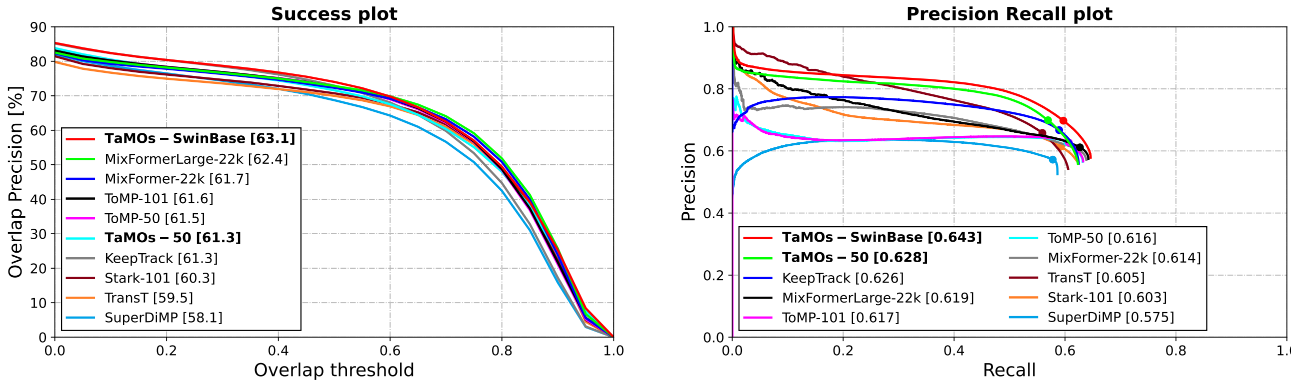
We evaluate our tracker with a ResNet-50 and a Swin-Base backbone as well as six single object trackers (SuperDiMP [9], KeepTrack [39], TransT [6], STARK [57], ToMP [38], and MixFormer [7]) and two multi object trackers (QDTrack [43] and OVTrack [31]) on LaGOT.

Metrics. We measure the performance of a tracker in the One Pass Evaluation (OPE) setting. The standard GOT Success rate Area Under the Curve (AUC) metric [14, 15, 17, 40–42, 53] does not account for false positive predictions when a target gets occluded or is out of view. While this is not a big issue in standard SOT datasets, where the target object is present in the vast majority of frames, it becomes vital in long-term tracking. In LaGOT objects are more frequently invisible due to occlusions or moving out-of-view. To capture this aspect, we employ the VOTLT [26,36] metric that penalizes false positives. It computes the IoU-weighted precision-recall curve and ranks the trackers according to their F1-score.

5.1.1 Comparison to SOT Methods

SOT trackers are limited to track only a single target at once. Thus, multiple instances of the same tracker need to be run in parallel to track multiple objects in the same sequence leading to a linearly increasing run-time, see Fig. 1.

Results. Fig. 4a shows the success rate of all trackers on LaGOT. We observe that SOT trackers perform well on LaGOT. However, our multi-object tracker TaMOs achieves the best AUC, even outperforming the state-of-the-art SOT tracker MixFormerLarge-22k [7]. We further observe that TaMOs is as robust as KeepTrack [39] ($T < 0.4$), where the gap to the remaining trackers is particularly prominent. This demonstrates the potential of a global multiple object GOT method. Fig. 4b shows the *tracking* Precision-Recall curve



(a) Success Plot (b) Precision-Recall Plot

Figure 4. Success plot, showing OP_T, on LaGOT (AUC is reported in the legend). Tracking Precision-Recall curve on LaGOT – VOTLT is reported in the legend (the highest F1-score).

on LaGOT. Both versions of TaMOs outperform all other SOT trackers. The highly robust object presence scores predicted by our tracker lead to a superior precision at all recall rates > 0.2. Moreover, our approach achieves the best maximal recall and outperforms all previous methods in VOTLT by 1.7 points. This demonstrates that joint tracking of multiple objects and global search benefit the object localization and identification capabilities of the tracker. For further insights we show MOT metrics on LaGOT in Tab. 3. Our tracker achieves the best results for every MOT metric and outperforms MixFormerLarge-22k by 5.9 points in MOTA.

Run-Time Analysis. We evaluate the run-time on a single A100 GPU. Tab. 2 reports a run-time analysis of our tracker TaMOs compared to ToMP, with both employing a ResNet-50 backbone. While TaMOs is slower than ToMP for a single object, due to the higher resolution required for full-frame tracking, our approach already reaches an advantage for 2 concurrent objects. As ToMP needs to run a separate independent tracker for each new object, our approach achieves a $4\times$ speedup for 10 concurrent objects. Furthermore, the analysis demonstrates that TaMOs achieves almost a constant run-time even when increasing the number of targets. TaMOs-SwinBase achieves 13.1 FPS for a single object and 9.3 FPS when jointly tracking 10 objects.

5.1.2 Comparison to MOT Methods

MOT methods are designed to track multiple objects in a video sequence and are thus used as baselines for LaGOT. However, in MOT the targets are defined via a list of classes whereas in multi-object GOT targets are defined by user-specified bounding boxes in the initial frame. Hence, to be able to track generic objects MOT methods need to be trained on large vocabulary datasets — then we can greedily match the detected tracks with the bounding boxes on the initial frame to track user-specified objects. Alternatively, the recent open-vocabulary MOT method OVTrack [31] allows to track objects of any class. We employ QDTrack [43]

Table 2. Run-time analysis (in FPS) between our baseline model ToMP and our tracker TaMOs.

	1 Object	2 Objects	5 Objects	10 Objects
ToMP-50	34.7	17.4	7.0	3.4
TaMOs-50	19.2	17.9	16.3	13.9

Table 3. Comparison of GOT and MOT metrics on LaGOT.

		F1-Score	Success	HOTA	MOTA	IDF1	OWTA
GOT	**TaMOs-SwinBase**	**0.643**	**63.1**	**62.1**	**58.2**	**74.7**	**68.9**
	TaMOs-50	**0.628**	61.3	60.0	**52.9**	72.0	67.1
SOT	MixFormerLarge-22k	0.619	**62.4**	**61.5**	52.3	**74.3**	**69.0**
	ToMP-101	0.617	61.6	60.1	51.9	73.8	67.5
	STARK-101	0.603	60.3	59.4	49.0	72.5	67.0
	TransT	0.605	59.5	57.7	46.6	70.7	65.6
	KeepTrack	0.626	61.3	59.1	51.3	73.8	66.2
	SuperDiMP	0.575	58.1	56.1	43.2	69.7	63.8
MOT	QDTrack	0.187	19.2	22.2	-115.8	16.3	36.3
	OVTrack	0.128	13.4	24.4	13.9	23.5	25.9

and open-vocabulary OVTrack [31] as MOT baselines. QDTrack is trained on LVIS [20] and TAO. We provide OVTrack in each video with the class name of the target.

Results. QDTrack and OVTrack achieve a VOTLT F1-Score of 0.187 and 0.128 respectively, performing inferior to all other trackers. Neither of the MOT trackers is robust enough and both fail to track rare or unknown generic objects. To further explore the limitations of MOT methods in our setting, we evaluate 'Oracle' versions, where we select the track ID that maximizes the scores on LaGOT. Even with such oracle information, the performance of QDTrack and OVtrack is by far inferior to any evaluated SOT baseline (VOTLT 33.1 and 23.0 respectively). In addition we evaluate both trackers using all its predicted tracks with MOT metrics, see Tab. 3. QDTrack tracks multiple background objects that are not annotated in LaGOT leading to many False Positives (FPs), and OVTrack tracks unannotated objects as well since the videos are not annotated exhaustively on class levels. Thus, traditional MOT tracking metrics such as MOTA, HOTA and IDF1 are unsuitable to evaluate MOT trackers on LaGOT. Instead, we concentrate on the OWTA metric [34] that focuses on Detection Recall and Association Accuracy and thus ignores FPs. QDTrack achieves 36.3 and OVTrack 25.9, which are still the lowest OWTA scores compared to SOT and GOT trackers.

5.2. State-of-the-Art Comparison on SOT Datasets

While TaMOs is built to track multiple objects in a video it can as well track only a single generic object. Thus, we evaluate TaMOs on popular large-scale SOT benchmarks. We deploy the very same tracker in these settings, without altering its weights or any hyper-parameters.

LaSOT [15]. This large-scale dataset consists of 280 test sequences with 2500 frames on average. Tab. 4 shows a comparison to recent SOT trackers. While primarily designed to cope with multiple objects, our tracker achieves the highest precision and the third highest success rate AUC. Note, that neither MixFormer, SwinTrack nor OSTrack operate on the entire video frame, but rely on a local search area to produce such high tracking accuracy.

TrackingNet [41]. This dataset consists of 511 test sequences and predictions are evaluated on a server. Tab. 4 shows that our tracker with SwinBase sets the new state of the art on TrackingNet in terms of success rate and precision AUC. Similarly, our tracker with ResNet-50 achieves the best results among all trackers using that backbone.

The results on both benchmarks show the great potential of applying trackers *globally* without motion priors, such as search area selection [3,7,59] or spatial windowing [29,30].

5.3. Ablation Study

The ablation experiments shown in Tabs. 5 and 6 are performed before the final annotation verification step such that the results compared to the numbers above slightly differ.

Generic Multiple Object Encoding. Tab. 5 shows the effect of the Gaussian score map encoding, the LTRB bounding box encoding and the total number of object embeddings m stored in the pool E. The first two rows in Tab. 5 show that the LTRB encoding is more important than the Gaussian encoding (as removing LTRB decreases all results more significantly). Another key factor is the number of different object embeddings, that sets an upper limit on the number of objects that can be tracked. LaGOT requires at least 10 embeddings and our tracker achieves the best results when using a pool size of 10. Increasing the number of embeddings decreases the overall tracking performance.

Architecture. Tab. 6 shows that using SwinBase increases the tracking performance on LaSOT and LaGOT. Similarly, adding an FPN improves the results.

Inference. During inference we update the memory by adding a second dynamic training frame similar to ToMP [38]. Since the ground truth bounding boxes are not available, we use the predicted boxes as annotations. We replace the dynamic training frame (update the memory) if the maximal value in each target score map is above the threshold of $\tau = 0.85$. The results in Tab. 6 show that adding a second training frame improves the results on both datasets.

Table 4. State-of-the-art comparison on SOT datasets.

Method	Venue	LaSOT [15] Prec	LaSOT N-Prec	LaSOT Succ	TrackingNet [41] Prec	TrackingNet N-Prec	TrackingNet Succ
TaMOs-SwinBase	WACV'24	**77.8**	79.3	70.2	**84.2**	**88.7**	**84.4**
TaMOs-50	WACV'24	75.0	77.2	67.9	82.0	87.2	82.7
SwinTrack [32]	NIPS'22	76.5	—	**71.3**	82.0	—	**84.0**
Unicorn [56]	ECCV'22	74.1	76.6	68.5	82.2	86.4	83.0
AiATrack [18]	ECCV'22	73.8	79.4	69.0	80.4	87.8	82.7
OSTrack [59]	ECCV'22	**77.6**	**81.1**	**71.1**	**83.2**	88.5	**83.9**
RTS [44]	ECCV'22	73.7	76.2	69.7	79.4	86.0	81.6
MixFormer [7]	CVPR'22	76.3	**79.9**	70.1	83.1	**88.9**	83.9
ToMP [38]	CVPR'22	73.5	79.2	68.5	78.9	86.4	81.5
UTT [37]	CVPR'22	67.2	—	64.6	77.0	—	79.7
KeepTrack [39]	ICCV'21	70.2	77.2	67.1	73.8	83.5	78.1
STARK [57]	ICCV'21	72.2	77.0	67.1	—	86.9	82.0
TransT [6]	CVPR'21	69.0	73.8	64.9	80.3	86.7	81.4
SuperDiMP [11]	CVPR'20	65.3	72.2	63.1	73.3	83.5	78.1

Table 5. Analysis of different object encoding settings. All tested configurations are not employing the FPN.

Gaussian Encoding	LTRB Encoding	Object Embedding Pool size m	LaSOT AUC	LaGOT AUC	LaGOT F1
✓	✗	10	58.3	54.0	0.552
✗	✓	10	66.3	60.2	0.620
✓	✓	10	**67.2**	**61.6**	**0.633**
✓	✓	15	65.7	60.0	0.617
✓	✓	20	65.7	58.9	0.603
✓	✓	50	63.1	57.4	0.587

Table 6. Architecture and memory update analysis.

Backbone	FPN	Memory Update	LaSOT AUC	LaGOT AUC	LaGOT F1
Resnet-50	✗	✓	67.2	60.4	0.621
Resnet-50	✓	✗	66.0	60.2	0.620
Resnet-50	✓	✓	**67.9**	**61.6**	**0.633**
SwinBase	✗	✓	69.5	62.4	0.643
SwinBase	✓	✗	67.9	62.1	0.636
SwinBase	✓	✓	**70.2**	**63.5**	**0.649**

6. Conclusion

We propose a novel multiple object GOT tracking benchmark, LaGOT, that allows to evaluate GOT methods that can jointly track multiple targets in the same sequence. We demonstrate that the proposed task and benchmark are challenging for existing SOT and MOT trackers. We further propose a Transformer-based tracker capable of processing multiple targets at the same time, with a novel generic multi object encoding and an FPN in order to achieve full frame tracking. Our method outperforms recent trackers on the LaGOT benchmark, while operating 4× faster than the SOT baseline when tracking 10 objects. Lastly, our approach also achieves excellent results on popular SOT benchmarks.

Funding: This work was done at Google Research.
Acknowledgement: The authors thank Paul Voigtlaender for his support, fruitful discussions and valuable feedback.

References

[1] Hexin Bai, Wensheng Cheng, Peng Chu, Juehuan Liu, Kai Zhang, and Haibin Ling. Gmot-40: A benchmark for generic multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6719–6728, June 2021. 2, 3, 4

[2] Luca Bertinetto, Jack Valmadre, João F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *Proceedings of the European Conference on Computer Vision Workshops (ECCVW)*, October 2016. 1

[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 (ICCV)*, October 2019. 8

[4] David S. Bolme, J. Ross Beveridge, Bruce A. Draper, and Yui Man Lui. Visual object tracking using adaptive correlation filters. In *CVPR*, 2010. 1

[5] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2

[6] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021. 3, 6, 8

[7] 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 (CVPR)*, pages 13608–13618, June 2022. 2, 3, 6, 8

[8] Kenan Dai, Yunhua Zhang, Dong Wang, Jianhua Li, Huchuan Lu, and Xiaoyun Yang. High-performance long-term tracking with meta-updater. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[9] Martin Danelljan, Goutam Bhat, and Christoph Mayer. PyTracking: Visual tracking library based on PyTorch. https://github.com/visionml/pytracking, 2019. Accessed: 1/07/2022. 6

[10] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. ECO: efficient convolution operators for tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2017. 1

[11] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic regression for visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 8

[12] Achal Dave, Tarasha Khurana, Pavel Tokmakov, Cordelia Schmid, and Deva Ramanan. Tao: A large-scale benchmark for tracking any object. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 436–454. Springer International Publishing, 2020. 2, 4, 6

[13] Patrick Dendorfer, Aljosa Osep, Anton Milan, Konrad Schindler, Daniel Cremers, Ian Reid, Stefan Roth, and Laura Leal-Taixé. Motchallenge: A benchmark for single-camera multiple target tracking. *International Journal of Computer Vision (IJCV)*, 129(4):1–37, 2020. 1, 2

[14] Heng Fan, Hexin Bai, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Mingzhen Huang, Juehuan Liu, Yong Xu, et al. Lasot: A high-quality large-scale single object tracking benchmark. *International Journal of Computer Vision (IJCV)*, 129(2):439–461, 2021. 2, 6

[15] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2, 3, 4, 6, 8

[16] J. Ferryman and A. Shahrokni. Pets2009: Dataset and challenge. In *2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, pages 1–6, 2009. 3

[17] Hamed Kiani Galoogahi, Ashton Fagg, Chen Huang, Deva Ramanan, and Simon Lucey. Need for speed: A benchmark for higher frame rate object tracking. In *ICCV*, 2017. 2, 6

[18] Shenyuan Gao, Chunluan Zhou, Chao Ma, Xinggang Wang, and Junsong Yuan. Aiatrack: Attention in attention for transformer visual tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 146–164, 2022. 8

[19] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 1, 2

[20] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 7

[21] João F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 37(3):583–596, 2015. 1

[22] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Globaltrack: A simple and strong baseline for long-term tracking. In *Proceedings of the Conference on Artificial Intelligence (AAAI)*, February 2020. 2

[23] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 43(5):1562–1577, 2021. 2, 6

[24] Mingzhen Huang, Xiaoxing Li, Jun Hu, Honghong Peng, and Siwei Lyu. Tracking multiple deformable objects in egocentric videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, Canada, 2023. 2

[25] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kämäräinen, Martin Danelljan, Luka Čehovin Zajc, 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 *Proceedings of the European Conference on Computer Vision Workshops (ECCVW)*, August 2020. 2

[26] Matej Kristan, Ales Leonardis, Jiri Matas, Michael Felsberg, Roman Pfugfelder, Luka Cehovin Zajc, Tomas Vojir, Goutam Bhat, Alan Lukezic, Abdelrahman Eldesokey, Gustavo Fernandez, and et al. The sixth visual object tracking vot2018 challenge results. In *Proceedings of the European Conference on Computer Vision Workshops (ECCVW)*, September 2018. 6

[27] Matej Kristan, Jiri Matas, Aleš Leonardis, Tomas Vojir, Roman Pflugfelder, Gustavo Fernandez, Georg Nebehay, Fatih Porikli, and Luka Čehovin. A novel performance evaluation methodology for single-target trackers. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(11):2137–2155, 2016. 1, 2, 3

[28] Alina Kuznetsova, Aakrati Talati, Yiwen Luo, Keith Simmons, and Vittorio Ferrari. Efficient video annotation with visual interpolation and frame selection guidance. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2021, Waikoloa, HI, USA, January 3-8, 2021*, 2021. 4

[29] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 8

[30] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 8

[31] Siyuan Li, Tobias Fischer, Lei Ke, Henghui Ding, Martin Danelljan, and Fisher Yu. Ovtrack: Open-vocabulary multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5567–5577, 2023. 2, 6, 7

[32] Liting Lin, Heng Fan, Zhipeng Zhang, Yong Xu, and Haibin Ling. Swintrack: A simple and strong baseline for transformer tracking. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. 3, 8

[33] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014. 6

[34] Yang Liu, Idil Esen Zulfikar, Jonathon Luiten, Achal Dave, Deva Ramanan, Bastian Leibe, Aljoša Ošep, and Laura Leal-Taixé. Opening up open world tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19045–19055, June 2022. 2, 8

[35] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip H. S. Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. HOTA: A higher order metric for evaluating multi-object tracking. *International Journal of Computer Vision (IJCV)*, 129(2):548–578, 2021. 3

[36] Alan Lukežič, Luka Čehovin Zajc, Tomáš Vojíř, Jiří Matas, and Matej Kristan. Now you see me: evaluating performance in long-term visual tracking, 2018. 3, 6

[37] Fan Ma, Mike Zheng Shou, Linchao Zhu, Haoqi Fan, Yilei Xu, Yi Yang, and Zhicheng Yan. Unified transformer tracker for object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8781–8790, June 2022. 3, 8

[38] 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. 2, 3, 4, 6, 8

[39] 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 (ICCV)*, pages 13444–13454, October 2021. 6, 8

[40] Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, October 2016. 2, 6

[41] Matthias Müller, Adel Bibi, Silvio Giancola, Salman Al-Subaihi, and Bernard Ghanem. Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 2, 6, 8

[42] Mubashir Noman, Wafa H 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 BMVC*, 2022. 6

[43] Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, and Fisher Yu. Quasi-dense similarity learning for multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 164–173, June 2021. 6, 7

[44] Matthieu Paul, Martin Danelljan, Christoph Mayer, and Luc Van Gool. Robust visual tracking by segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 571–588, 2022. 8

[45] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[46] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 6

[47] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 6

[48] Siyu Tang, Mykhaylo Andriluka, Bjoern Andres, and Bernt Schiele. Multiple people tracking by lifted multicut and person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1, 2

[49] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 5

[50] Jack Valmadre, Luca Bertinetto, João F. Henriques, Ran Tao, Andrea Vedaldi, Arnold W.M. Smeulders, Philip H.S. Torr, and Efstratios Gavves. Long-term tracking in the wild: a benchmark. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2, 4

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 3

[52] Paul Voigtlaender, Jonathon Luiten, Philip H.S. Torr, and Bastian Leibe. Siam R-CNN: Visual tracking by re-detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[53] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 37(9):1834–1848, 2015. 1, 2, 3, 6

[54] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark, 2018. 2, 6

[55] Yinda Xu, Zeyu Wang, Zuoxin Li, Ye Yuan, and Gang Yu. Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In *Proceedings of the Conference on Artificial Intelligence (AAAI)*, February 2020. 5

[56] Bin Yan, Yi Jiang, Peize Sun, Dong Wang, Zehuan Yuan, Ping Luo, and Huchuan Lu. Towards grand unification of object tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 733–751. Springer International Publishing, 2022. 3, 8

[57] 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 (ICCV)*, pages 10448–10457, October 2021. 2, 3, 6, 8

[58] Bin Yan, Haojie Zhao, Dong Wang, Huchuan Lu, and Xiaoyun Yang. 'skimming-perusal' tracking: A framework for real-time and robust long-term tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 2

[59] Botao Ye, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Joint feature learning and relation modeling for tracking: A one-stream framework. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 341–357. Springer Nature Switzerland, 2022. 3, 8

[60] Bin Yu, Ming Tang, Linyu Zheng, Guibo Zhu, Jinqiao Wang, Hao Feng, Xuetao Feng, and Hanqing Lu. High-performance discriminative tracking with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9856–9865, October 2021. 3

[61] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2

[62] Qian Yu, Gérard Medioni, and Isaac Cohen. Multiple target tracking using spatio-temporal markov chain monte carlo data association. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2007. 1

[63] Li Zhang, Yuan Li, and Ramakant Nevatia. Global data association for multi-object tracking using network flows. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2008. 1