Tracking Small and Fast Moving Objects: A Benchmark[⋆]

Zhewen Zhang[†], Fuliang Wu[†], Yuming Qiu, Jingdong Liang, and Shuiwang Li[*]

Guilin University of Technology, Guilin 541006, China
zwzhang0101@163.com, wufuliang@glut.edu.cn, qiuyuming0706@163.com,
liangjingdong@glut.edu.cn, lishuiwang0721@163.com

Abstract. With more and more large-scale datasets available for training, visual tracking has made great progress in recent years. However, current research in the field mainly focuses on tracking generic objects. In this paper, we present TSFMO, a benchmark for **T**racking **S**mall and **F**ast **M**oving **O**bjects. This benchmark aims to encourage research in developing novel and accurate methods for this challenging task particularly. TSFMO consists of 250 sequences with about 50k frames in total. Each frame in these sequences is carefully and manually annotated with a bounding box. To the best of our knowledge, TSFMO is the first benchmark dedicated to tracking small and fast moving objects, especially connected to sports. To understand how existing methods perform and to provide comparison for future research on TSFMO, we extensively evaluate 20 state-of-the-art trackers on the benchmark. The evaluation results exhibit that more effort are required to improve tracking small and fast moving objects. Moreover, to encourage future research, we proposed a novel tracker S-KeepTrack which surpasses all 20 evaluated approaches. By releasing TSFMO, we expect to facilitate future researches and applications of tracking small and fast moving objects. The TSFMO and evaluation results as well as S-KeepTrack are available at https://github.com/CodeOfGithub/S-KeepTrack.

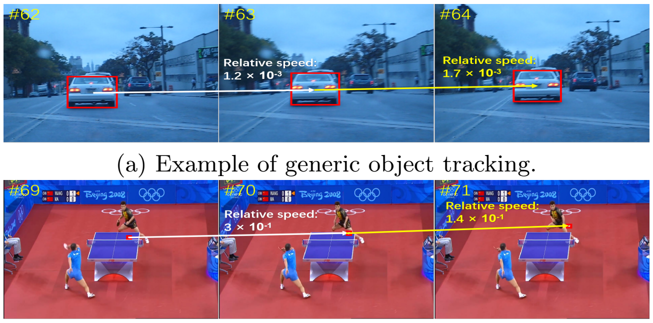
Keywords: Visual tracking · Small and fast moving objets · Benchmark.

1 Introduction

Object tracking is one of the most fundamental problems in computer vision with a variety of applications, including video surveillance, robotics, human-machine interaction, motion analysis and so forth [40,41,43]. Great progress has been witnessed in object tracking thanks to the successful application of deep learning to the field in recent years [29]. Despite considerable progress in the field, current researches mainly focus on tracking generic objects, while very little attention is paid to tracking small and fast moving objects. Nevertheless, small and fast moving objects are common to see in the real world. Many of them are close connection to sports. Tracking of them is crucial to meet the practical

[⋆] [†]These authors contributed equally. [*] Corresponding author.

and accuracy requirements of motion analysis for sports [10,36], and to provide a fairer measure of performance than that provided by human judges, and to spare referees, umpires or judges from immense pressure in making accurate split-second decisions [31,58]. It is also the key to develop automatic sports video recording and recommendation systems [28,66]. However, tracking small and fast moving objects are more challenging. There are several reasons account for this.



(a) Example of generic object tracking.

(b) Example of small and fast moving object tracking.

Fig. 1. Generic object tracking (a) and small and fast moving object tracking (b). In comparison, tracking of small and fast moving objects is more challenging as the objects are visually much smaller and the relative speeds are much higher.

Table 1. Comparison of average target size (in $pixel^2$) and average relative target speed (in $1/pixel$) of object tracking benchmarks.

	OTB[64]	Got-10K[26]	LasoT[18]	TrackingNet[54]	UAV123[53]	**TSFMO (Ours)**
Avg. target size ($\times 10^3$)	6.73	228.97	56.17	45.65	2.48	0.51
Avg. relative speed ($\times 10^{-1}$)	2.02	2.64	30.21	1.97	7.74	58.28

First, small objects in sports are usually balls or objects of regular geometric shapes. Discriminative information provided by their shapes are fairly limited. Without sufficient discriminative cue of the target, existing tracking algorithms are prone to failure. Second, these objects are covered with plain or regular patterns, making the identifying of them in complex scenes very difficult, as they may be treated as parts of the background. Third, they are frequently moving at a high speed, which may cause severe motion blur in images, and they may change moving direction abruptly when at collision or being hit.

In addition to the above technical difficulties, another important reason that tracking small and fast moving objects is hardly touched is the lack of public available benchmarks, which are undoubtedly crucial to attack the problem and to advance the field as without which researchers are unable to effectively design and evaluate novel algorithms for improvement. Despite that there exist plenty of benchmarks for generic object tracking [64,45,35,21,60,54,26,18], there is no benchmark dedicated to tracking small and fast moving objects, especially small objects in sports. Although many of existing benchmarks consist of small and fast moving objects, the numbers of both sequences and object classes are very limited. Tracking algorithms developed to target these benchmarks are generic but not effective in dealing challenges posed by small and fast moving objects.

To facilitate research on tracking small and fast moving objects, in this paper we present a dedicated dataset to serve as the testbed for fair evaluation.

1.1 Contribution

In this work, we make the first attempt to explore tracking small and fast moving objects by introducing the TSFMO benchmark for Tracking Small and Fast Moving Objects. TSFMO is made up of a diverse selection of 26 classes of sports with each containing multiple sequences. TSFMO consists of a total of 250 sequences and about 50k frames. Each sequence is manually annotated with axis-aligned bounding boxes with different attributes for performance evaluation and analysis. As far as we know, TSFMO is the first benchmark dedicated to the task of tracking small and fast moving objects, especially in sports. Fig. 1 illustrates the differences between generic object tracking and small and fast moving object tracking. Compared with generic object tracking in which the target size is relatively larger and the relative speed (the displacement of the target in two neighbouring frames divided by the square root of the average area of the neighbouring bounding boxes) is relatively lower, tracking of small and fast moving objects is more challenging as the objects are visually very small and the relative speeds are much higher. A quantitative comparison of average target size and average relative target speed between four public generic object tracking benchmarks and five TSFMO is shown in Table 1.

In addition, in order to understand the performance of existing tracking algorithms and provide comparisons for future research on TSFMO, we extensively evaluated 20 state-of-the-art tracking algorithms on TSFMO. Meanwhile we conducted an in-depth analysis of the evaluation results and observed several surprising findings. First, the tracking performances of existing trackers are much lower in TSFMO than in most public benchmarks for generic object tracking, which suggests that most existing tracking methods may overlook very important factors so that they are not effective in tracking small and fast moving objects. Second, not all latest trackers whose performances rank high on OTB [64], Got-10K [26], LasoT [18], and TrackingNet [54] are highly-ranked on TSFMO, which suggests that the generalization ability of some existing trackers is questionable. So, as as an unexplored or less studied problem, improving the generalization ability of tracking algorithms is valuable and interesting. These above observations imply the need to develop tracking algorithms devoted to tracking small and fast moving objects, which may also stimulate more generalizable tracking algorithms in the future.

Last but not the least, we introduce a baseline tracker in order to facilitate the development of tracking algorithms on TSFMO. The proposed tracker is based on KeepTrack [50] given that it shows the best performance among state-of-the-art trackers to be evaluated here. In view of that small objects may not have representation in deeper layers' features because of the larger receptive field in deeper layers and that combining low and high level features to boost performance has been extensively studied in tiny object detection [25,32,22], we modify the architecture of KeepTrack [50] so that low-level features are exploited to improve tracking performance. This results in the proposed tracker, which is called S-KeepTrack. The proposed S-KeepTrack outperforms all 20 state-of-the-art trackers on TSFMO. In summary, we have made the following contributions:

- We propose TSFMO, which is, as far as we know, the first benchmark dedicated to track small and fast moving objects, especially in sports.
- We evaluate 20 state-of-the-art tracking algorithms with in-depth analysis to assess their performance and provide comparisons on TSFMO.
- We develop a baseline tracker S-KeepTrack base on the KeepTrack [50] to encourage further research on TSFMO.

2 Related Works

2.1 Visual Tracking Algorithms

Visual tracking has been studied for decades with a huge literature, the comprehensive review of which is out of scope of this paper. In this section, we review two popular trends including discriminative correlation filter (DCF)-based tracking and deep learning (DL)-based tracking in the field and refer readers to [47,51] for comprehensive surveys.

Roughly stated, DCF-based trackers treat visual tracking as an online regression problem. Thanks to the Parseval theorem and the Fast Fourier Transform (FFT), DCF-based tracker can be effectively evaluated in the frequency domain and demonstrate impressive CPU speeds [42]. They started with the minimum output sum of squared error (MOSSE) filter [5]. After that great advance has been witnessed in DCF-based trackers [42]. For instance, an additional scale filter is exploited in [44,13] to deal with target scale variations. The trackers in [12,39] leverage regularization techniques to improve robustness. The approach in [40] generalize the DCF to achieve translation equivariance. The methods in [49,15] utilize deep features instead of handcrafted ones in correlation filter tracking and achieve significant improvements.

The great success of deep learning in other vision tasks motivated the mushrooming development of DL-based trackers in visual tracking in recent years. As one of the pioneering works, SiamFC [2] considered visual tracking as a general similarity-learning problem and took advantage of the Siamese network [9] to measure the similarity between target and search image. Since then, many DL-based trackers base on Siamese architectures have been proposed [38,24,65] and the tracking performances have been significantly improved. Along another line, visual tracking is divided into two sub-tasks, i.e., localization and scale estimation, which are solved, respectively, by an online classifier and an offline intersection-over-union (IoU) network [11,3,14,50].

2.2 Visual Tracking Benchmarks

As standards by which the performances of tracking methods are measured or judged, tracking benchmarks, undoubtedly, are crucial for the development of visual tracking. Existing benchmarks can be roughly divided into two types: generic benchmarks and specific benchmarks [19].

Generic Benchmarks. A generic tracking benchmark is usually designed for tracking objects in general scenes. OTB-2013 [64] is the first generic benchmark with 50 sequences and later extended to OTB-2015 with 100 sequences. TC-128 [45] consists of 128 colorful sequences, and is used to study the impact of color information on tracking performance. VOT [35] organizes a series of tracking competitions with up to 60 sequences. NfS [21] concerns about videos of high frame rate. NUS-PRO [37] collects 365 videos and primarily addresses tracking rigid objects. TracKlinic [20] includes 2,390 videos to evaluate tracking algorithms under various challenges. Recently, many large-scale benchmarks have been proposed to provide training data for developing DL-based trackers. OxUvA [60] provides 366 videos aiming for long-term tracking in the wild. TrackingNet [54] collects a large-scale dataset consisting of more than 30K sequences for deep tracking. GOT-10k [26] provides 10K sequences with rich motion trajectories. LaSOT offers 1,400 long-term videos in [18] and later introduces additional 150 sequences and a new evaluation protocol for unseen objects in [17].

Specific Benchmarks. In addition to generic visual tracking benchmarks, there exist specific benchmarks for particular goals. UAV123 [53] consists of 123 sequences captured by unmanned aerial vehicle (UAV) for low altitude UAV target tracking. VOT-TIR [33] is from VOT and focuses on object tacking in RGB-T sequences, aiming at taking advantage of RGB and thermal infrared images simultaneously. CDTB [48] and PTB [57] are designed to assess tracking performance on RGB-D videos, D indicating depth images. TOTB [19] collects 225 videos from 15 transparent object categories and focus on transparent objects.

Despite of the availability of the above benchmarks, they mainly focus on tracking objects of common sizes and relatively slow speeds. Tracking of small and fast moving objects, especially in sports, has received very little attention. The most important reason, we think, is the lack of public available benchmarks, which motivates our proposal of TSFMO.

2.3 Dealing with Small and Fast Moving Objects in Vision

Small objects here refer to objects with smaller physical sizes in the real world and occupying areas less than and equal to 32 × 32 pixels [59], while fast moving objects refer to the ones that may move over a distance exceeding its size within the exposure time [55]. Small and fast moving objects are common to see in the real-world, and a significant amount of researches have been devoted to deal with them. For example, the methods of [30,52] studied the problem of small object detection utilizing hand-engineered features and shallow classifiers in aerial images. The approach in [27] combined detection and tracking and integrated them into an adaptive particle filter to handle small object localization, but it was evaluated on mere two testing videos for case study. To the best of our knowledge, Chen et al. [7] are perhaps the first to introduce a small object detection (SOD) dataset, an evaluation metric, and provide a baseline score in order to explore small object detection. The work of [1] presented an algorithm for detecting and tracking small dim targets in Infrared (IR) image sequences base on the frequency and spatial domain information. The work of [55] presented a method for detecting and tracking fast moving objects and provided a new dataset consisting of 16 sports videos for evaluation. In [46], an aggregation signature was proposed for small object tracking and 112 sequences were collected for evaluation. The method in [73] implemented a segmentation network that performs near real-time detection and tracking of fast moving objects and introduced a synthetic physically plausible fast moving object sequence generator for training purpose. The method of [68] investigated the problem of small and fast moving object detection and tracking in sports video sequences, using only motion as a cue for detection and multiple filter banks for tracking.

Our work is related to [27,1,55,46,73,68] but different in: (1) TSFMO focuses on tracking small and fast moving objects, while other works concentrate on either tracking small objects [27,1,46], or tracking fast moving objects [55,73]. (2) Although both focus on tracking small and fast moving objects, TSFMO provides a diverse benchmark of hundreds challenging sequences for evaluation, while in [68] only a small number of sequences is provided, which are captured with a stationary camera under indoor conditions where the background remains stationary. Such data is way far from real applications.

3 Tracking Small and Fast Moving Objects

3.1 Video Collection

We select 26 small and fast-moving object categories to construct TSFMO, including baminton, baseball, basketball, beach volleyball, bowling, boxing, curling, discus, football, gateball, golf, hammer, handball, ice hockey, indoor football, kick volleyball, pingpong, polo, ruby, shot, shuttlecock ball, snooker, squash, tennis, volleyball and water polo . Fig. 3 demonstrates some sample sequences from these categories.

After determining the object categories, we search for raw sequences of each class from the Internet, as it is the source of many tracking benchmarks (e.g., LaSOT [17], GOT-10k [26], TrackingNet [54], etc). Initially, we collected at least 10 raw videos for each class and gathered more than 280 sequences in total. We then carefully inspect each sequence for its availability for tracking and drop the undesirable sequences. Afterwards, we verify the content of each raw sequence and remove the irrelevant parts to obtain a video clip that is suitable for tracking. We intentionally limit the number of frames in each video to 900 frames, which is enough for accessing the tracker's performance on tracking small and fast-moving objects, meanwhile manageable for annotation. Eventually, TSFMO is made up of 250 sequences from 26 object classes with about 50K frames. Table 2 summarizes TSFMO, and Fig. 2 shows the average sequence length for each object category in TSFMO.

3.2 Annotation

For sequence annotation, we follow the principle used in [14]: given an initial object, for each frame in the sequence, the annotator draws/edits an axis-aligned bounding box as the tightest bound box to fit any visible part of the object if the object appears; otherwise, either an absence label or out of view (OV) or full occlusion (FOC) is assigned to the frame.

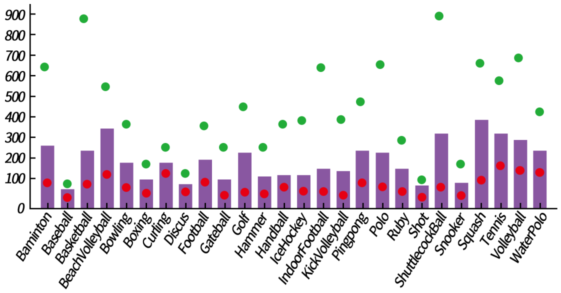


Fig. 2. Average video length for each object class in TSFMO. The red and green dots indicate the minimum and maximum frame numbers of each category.

Table 2. Summary of statistics of the proposed TSFMO.

Number of videos	250	Min frames	16	Frame rate	≤30fps	Max frames	887
Total frames	49k	Avg frames	196	Object categories	26	Avg duration	7.4s

Adhering to the above principle, we finish the annotation in three steps, i.e., manual annotation, visual inspection, and box refinement. In the first step, each video was labelled by an expert, i.e., a student working on tracking. As annotation errors or inconsistencies is hardly avoidable in the first stage, a visual inspection is performed to verify the annotations in the second stage, which is conducted by a validation team. If the validation team do not agree on the annotation unanimously, it will be sent back to the original annotator for refinement in the third step. This three-step strategy ensures high-quality annotation for objects in TSFMO. See Fig. 3 for some examples of box annotations for TSFMO.

3.3 Attributes

In view of that in-depth analysis of tracking methods is crucial to grasp their strengths and limitations, we select twelve attributes that widely exist in video tasks and annotate each sequence with these attributes, including (1) Illumination Variation (IV), (2) Deformation (DEF), (3) Motion Blur (MB), (4) Rotation (ROT), (5) Background Clutter (BC), (6) Scale Variation (SV), which is assigned when the ratio of bounding box is outside the range [0.5, 2], (7) Out-of-view (OV), (8) Low Resolution (LR), which is assigned when the target area is smaller than 900 pixels, (9) Aspect Ratio Change (ARC), which is assigned when the ratio of the bounding box aspect ratio is outside the range [0.5, 2],

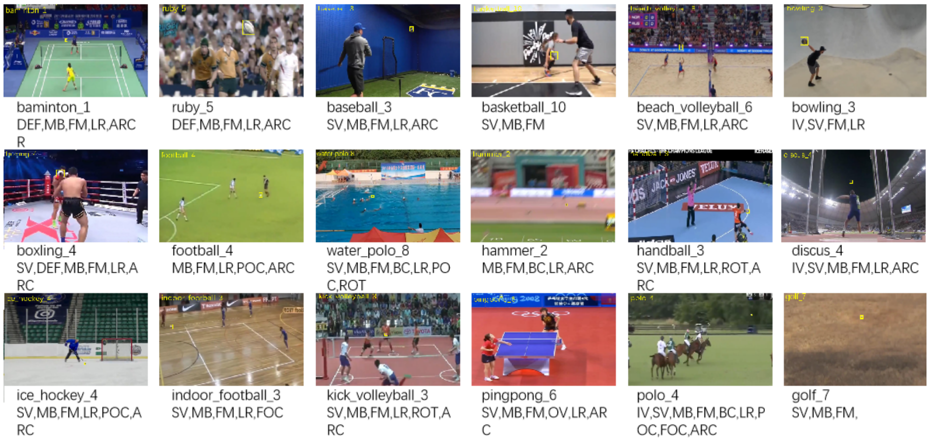


Fig. 3. Example sequences of small and fast moving object tracking in our TSFMO. Each sequence is annotated with axis-aligned bounding boxes and attribues.

Table 3. Distribution of twelve attributes on the TSFMO. The diagonal (shown in **bold**) corresponds to the distribution over the entire benchmark, and each row or column presents the joint distribution for the attribute subset.

	IV	SV	DEF	MB	FM	OV	BC	LR	POC	ROT	FOC	ARC
IV	**31**	29	0	26	31	0	14	30	7	12	1	23
SV	29	**193**	33	170	191	17	49	188	53	96	8	147
DEF	0	33	**45**	39	44	12	5	38	7	26	0	41
MB	26	170	39	**221**	132	18	58	203	56	102	9	158
FM	31	191	44	132	**248**	18	58	231	56	102	9	75
OV	0	17	12	18	18	**21**	0	20	4	17	0	17
BC	14	49	5	58	58	0	**63**	59	18	21	1	40
LR	30	188	38	203	231	20	59	**235**	62	108	8	170
POC	7	53	7	56	56	4	18	62	**71**	31	1	46
ROT	12	96	26	102	102	17	21	108	31	**111**	2	80
FOC	1	8	0	9	9	0	1	8	1	2	**9**	7
ARC	23	147	41	158	75	17	40	170	46	80	7	**176**

(10) Partial Occlusion (POC), (11) Full Occlusion (FOC), and (12) Fast Motion (FM), which is assigned when the target center moves by at least 50% of its size in last frame. Table 3 shows the distribution of these attributes on TSFMO. As can be seen, the most common challenge in TSFMO is Fast Motion. In addition, the Motion Blur and Low Resolution also present frequently in TSFMO.

4 A new baseline : S-KeepTrack

We found that among the state-of-the-art trackers to be evaluated here KeepTrack shows the best performance, despite that it is still far from satisfactory. To facilitate the development of tracking algorithms for tracking small and fast moving objects, we present a new baseline tracker based on the KeepTrack. The proposed tracker, dubbed S-KeepTrack, combines low-level and high-level features to improve tracking performance, considering that small objects may not have representation in deeper layers' features because of their larger receptive field. In contrast to one target candidate association network of KeepTrack, S-KeepTrack has two parallel target candidate association networks that separately

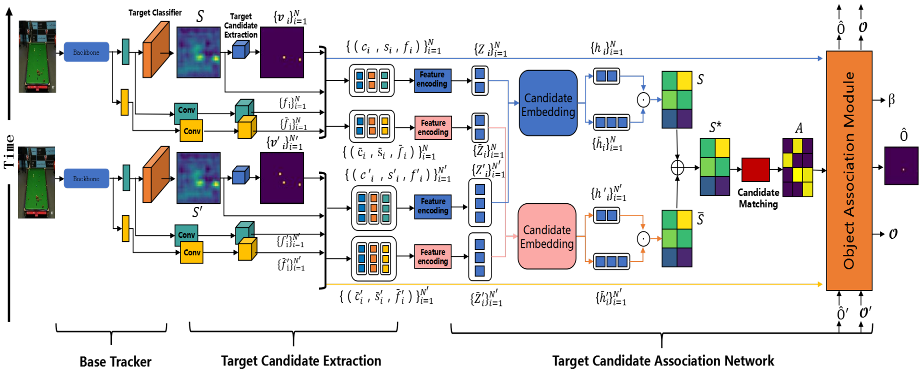


Fig. 4. Overview of the tracking pipline of the proposed S-KeepTrack. Note that the parallel architectures plotted in the same color share parameters.

process the feature encodings of lower and higher level features, respectively. And the result of candidate association is a weighted combination of the result of two parallel networks. Hopefully, this combination of low-level and high-level features will build a stronger tracker than KeepTrack for tracking small and fast moving objects.

Like KeepTrack, S-KeepTrack also consists of three components: i) a base tracker that predicts the target score map s for the current frame and extracts the target candidates V and V' ($V = \{v_i\}_{i=1}^N$ and $V' = \{v'_i\}_{i=1}^{N'}$ denote candidate set of the current and the previous frame, respectively) by finding locations in s with high target score, ii) a target candidate extraction module that extracts for each candidatea a set of features (i.e., target classifier score s_i, location c_i in the image, and two appearance cues $\bar{f}_i$ and f_i from lower and higher level, respectively, where i indexes the ith candidate), and iii) a target candidate association network that estimates the candidate assignment probabilities between two consecutive frames. The difference between S-KeepTrack and KeepTrack lies in the parallel network achitechtures designed for the low level features from the backbone. An overview of our tracking pipeline is shown in Fig. 4.

Base tracker: The base tracker is inherited from KeepTrack with the difference: the backbone of our S-KeepTrack outputs both low and high level features for target candidate extraction instead of only high level one as in KeepTrack.

Target candidate extraction: This module aims to build a feature representation for each target candidate. In KeepTrack, the position c_i and the target classifier score s_i are taken as two discriminative cues of the target candidate v_i, based on two observations: i) the motion of the same object from frame to frame is typically small and thus the same object has similar locations in two neighbouring frames, ii) only small changes in appearance for each object [50]. In addition, it also processes the backbone features with a single learnable convolution layer to add a more discriminative appearance-based feature f_i. Finally, each feature tuple (c_i, s_i, f_i) representing the target candidate v_i is fed into a feature encoding module to get the code [50]

$$z_i = f_i + \psi(s_i, c_i), \tag{1}$$

where ψ denotes a Multi-Layer Perceptron (MLP), mapping s_i and c_i to the same dimensional space as f_i. z_i then passes through a candidate embeding network before candidate matching is conducted. In order to build a better feature representaion for each target candidate, we additionally process a low-level backbone feature with an extra learnable convolution layer in our S-KeepTrack, resulting in an extra discriminative appearance-based feature $\bar{f}_i$. In view of that the feature encoding module and the candidate embedding network have been carefully tailored to the tuple (c_i, s_i, f_i), we avoid fusing f_i and $\bar{f}_i$ to get an entangled feature, as it may not fit well with the original network and will demand modifying the feature encoding module and the candidate embedding network. Instead, from the perspective of ensemble method, we build a new tuple $(c_i, s_i, \bar{f}_i)$ to accompany (c_i, s_i, f_i). And $(c_i, s_i, \bar{f}_i)$ is fed into a new feature encoding module to get another code

$$\bar{z}_i = \bar{f}_i + \psi'(s_i, c_i), \tag{2}$$

where ψ' denotes a MLP that map s_i and c_i to the same dimensional space as $\bar{f}_i$. The feature encoding modules are followed by two parallel candidate embedding networks, which will be detailed in the following.

Candidate embedding network: On an abstract level, candidate association bares similarities with the task of sparse feature matching [56,50], for which KeepTrack adopted the SuperGlue [56] architecture that establishes state-of-the-art sparse feature matching performance to do candidate embedding and matching. With this method, the feature encodings $\{z_i\}_{i=1}^N$ and $\{z'_i\}_{i=1}^N$ of two neighbouring frames translate to nodes of a single complete graph with two types of directed edges: 1) self edges within the same frame and 2) cross edges connecting only nodes between the frames. In SuperGlue [56], a Graph Neural Network (GNN) is utilized to send messages in an alternating fashion across self or cross edges to produce a new feature representation for each node after every layer, in which self and cross attention are used to compute the messages for self and cross edges [56,50]. After the last message passing layer a linear projection layer extracts the final feature representation h_i for each candidate v_i [50]. In our S-KeepTrack, a new candidate embedding network is adopted to deal with the feature encoding related to the low-level features, resulting in an extra feature representation $\bar{h}_i$ for each candidate v_i.

Candiate matching: In KeepTrack, the candidate embeddings h'_i and h_j (corresponding to two candidates $v'_i \in V'$ and $v_j \in V$, respectively.) are used to compute the similarity between v'_i and v_j by the scalar product: $S_{i,j} = \langle h'_i, h_j \rangle$. Given a match may not exist for every candidate, KeepTrack makes use of the dustbin concept [16,56] to actively match candidates that miss their counterparts to the so-called dustbin, ending up with an augmented assignment matrix A with an additional row and column representing dustbins. Note that a dustbin is a virtual candidate without any feature representation, to which a candiate corresponds only if its similarity scores to all other candidates are sufficiently low. To obtain the assignment matrix A between V and V' given the similarity matrix $S = \{S_{i,j}\}$, KeepTrack follow Sarlin et al. [56] and designed a learnable module to predict A. However, we have two parallel similarity matrice S and

$\bar{S}$ in S-KeepTrack because of the parallel feature representations. Therefore, we aggregate the two similarity matrice S and $\bar{S}$ to produce a fused one S^* by the following weighted sum,

$$S^* = \omega S + (1-\omega)\bar{S}, \tag{3}$$

where $\omega \in [0,1]$ is the weight coefficient to balance the contributions of S and $\bar{S}$. S^* is then fed into the candidate matching module to predict the assignment matrix A as in KeepTrack.

Object association: The object association module uses the estimated assignments to determine the object correspondences during online tracking, which follows that of KeepTrack. The idea is to keep track of every object present in each scene over time using a database $\mathcal{O}$ with each entry being an object visible in the current frame. When online tracking, the estimated assignment matrix A is used to determine which objects disappeared, newly appeared, or stayed visible and can be associated unambiguoursly, and to help reason the target object $\hat{o}$. Last but not least, the target detection confidence β is computed to manage the memory and control the sample wieght for updating the target classifier online. This finishes the description of our S-KeepTrack. It is worth noting that the losses and training pipeline of S-KeepTrack is the same as KeepTrack. Please refer to [50] for details.

5 Evaluation

5.1 Evaluation Metrics

We use one-pass evaluation (OPE) and measure each tracker using precision, and success rate as in [18,54]. The precision measures the distance between the centers of the estimated target bounding box and the groundtruth box in pixels. Success rate is based on the intersection over union (IoU) of the estimated target bounding box and the groundtruth box, specifically, it measures the percentage of estimated target bounding boxes with IoU larger than 0.5 [64,18,54]. The precision at 20 pixels (PRC) and the area under curve (AUC) of success plot is usually used for ranking.

5.2 Trackers for Comparison

We evaluate 20 state-of-the-art trackers to understand their performance on TSFMO, including KeepTrack [50], AutoMatch [69], TransT [8], SAOT [72], SiamGAT [23], LightTrack [67], TrDiMP [61], TrSiam [61], KYS [4], HIFT [6], SuperDiMP [34], PrDiMP50 [14], PrDiMP18 [14], Ocean [70], SiamMask [62], SiamRPN++ [38], ATOM [11], DiMP50 [3], DiMP18 [3], and SiamDW [71].

5.3 Evaluation results

Overall performance. 20 state-of-the-art trackers and our S-KeepTrack are extensively evaluated on TSFMO. Note that existing trackers are used without

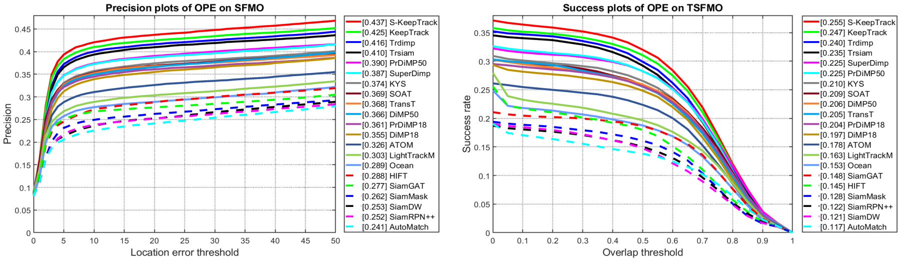


Fig. 5. Overall performance on TSFMO. Precision and success rate for one-pass evaluation (OPE) [63] are used for evaluation.

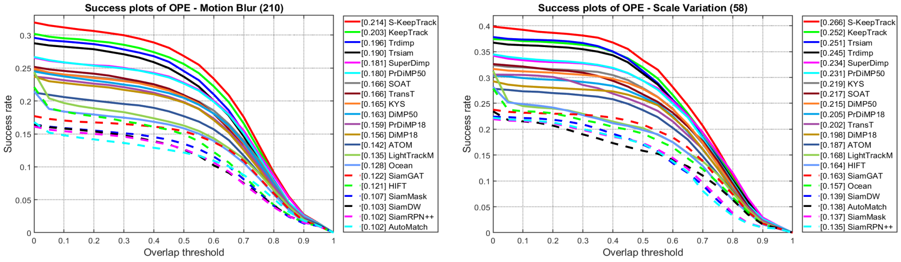


Fig. 6. Attribute-based comparison on motion blur and scale variation.

any modification. The evaluation results are reported in precision and success plot, as shown in Fig. 5. As can be seen, our S-KeepTrack achieved the best results with a PRC of 0.437, AUC of 0.255. KeepTrack got the second best PRC of 0.425, and likewise KeepTrack got the second best AUC of 0.247. In comparison with the second best tracker KeepTrack, S-KeepTrack achiveves improvements of 1.2% and 0.8% in terms of PRC and AUC, respectively, which evidences the effectiveness and advantage of our method of combining low level and high level features for small and fast moving object tracking.

Attribute-based performance. We conduct performance evaluation under twelve attributes to further analyze and understand the performances of different trackers. Our S-KeepTrack achieves the best PRC and AUC on most attributes. Due to space limitation, we demonstrate in Fig. 6 the success plots for the two most frequent challenges, including motion blur and scale variation. We observe that S-KeepTrack performs the best on both attributes. Specifically, S-KeepTrack achieves a AUC of 0.214 on motion blur, surpassing the second best tracker KeepTrack with AUC of 0.203 by 1.1%; on scale variation, S-KeepTrack' AUC is 0.266, outperforming the second best tracker KeepTrack with AUC of 0.252 by 1.4%. This also supports the importance of combining low level and high level features for small and fast moving object tracking.

Qualitative evaluation. In Fig. 7, we show some qualitative tracking results of our method in comparison with eight top trackers, including KeepTrack [50], TrSiam [61], TrDiMP [61], SAOT [72], KYS [4], PrDiMP50 [14], DiMP50 [3], DiMP18 [3]. As can be seen, only our S-KeepTrack succeeds to maintain robustness in these four examples that subject to challenges including rotation,

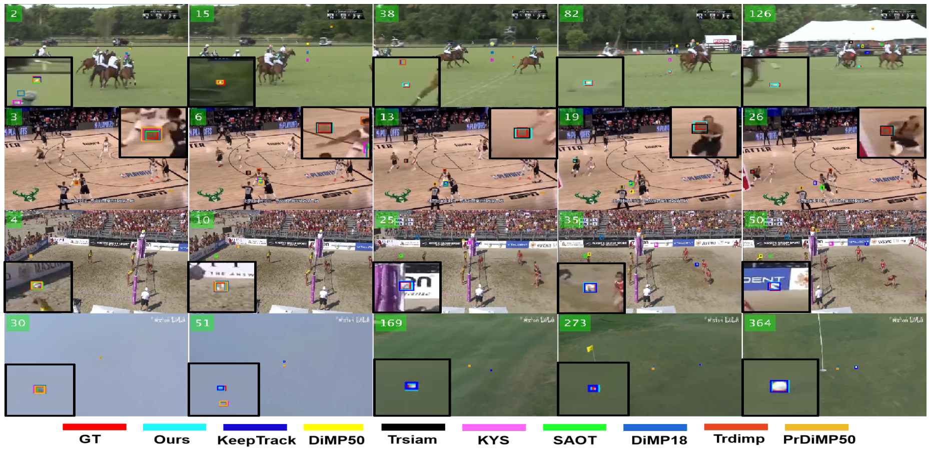


Fig. 7. Qualitative evaluation on 4 sequences from TSFMO, i.e., polo_4, basketball_14, beach_volleyball_8, and golf_4 from top to bottom. The results of different methods have been shown with different colors, and 'GT' denotes the groundtruth.

background clutter, illumination, partial occlusion, motion blur, low resolution, and scale variation. Specifically, all other trackers fail to keep track of the target in polo_4, only S-KeepTrack and Trsiam succeed to track the target in basketball_14, and only S-KeepTrack and KeepTrack succeed to track the target in golf_4. We own the advantage of S-KeepTrack over other trackers, especially over KeepTrack, to the proposed method of combining low level and high level features for representation.

Table 4. Illustration of the impact of different backbones on precision (PRC) and AUC on TSFMO. The better one is marked in bold.

	DiMP	PrDiMP	KeepTrack	S-KeepTrack (ours)
ResNet18	(0.355, 0.197)	(0.361, 0.204)	(0.346, 0.185)	(0.353, 0.190)
ResNet50	(**0.366**, **0.206**)	(**0.390**, **0.225**)	(**0.425**, **0.247**)	(**0.437**, **0.255**)

5.4 Ablation Study

Impact of backbone. To study the impact of the backbone on performance of tracking small and fast moving object. We evaluate several state-of-the-art trackers with ResNet-18 and ResNet-50 as backbone separately, including DiMP [3], PrDiMP [14], KeepTrack [50], and our S-KeepTrack. Note that KeepTrack was implemented with only ResNet-50 as backbone originally. We adapt it and S-KeepTrack to support ResNet-18 as backbone for this ablation study. Table 4 shows the PRC and AUC of these trackers on TSFMO in the form of (PRC, AUC). As can be seen, in each tracker the PRC and AUC are higher with ResNet-50 than with ResNet-18. Specifically, the (PRC, AUC) of DiMP, PrDiMP, KeepTrack, and S-KeepTrack increases by (1.1%, 0.9%), (2.9%, 2.1%), (7.9%, 6.2%), (8.4%, 6.5%) when the backbone is replaced from ResNet-18 to ResNet-50. This suggests that, although low-level features is helpful for tracking small and fast moving object as demonstrated by our method, deeper backbones

Table 5. Illustration of the impact of the importance coefficient of the low-level and high-level features on precision (PRC) and AUC on TSFMO. Red, blue and green indicate the first, second and third place.

ω	1.0	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0.0
PRC	0.424	0.417	0.415	0.424	0.421	0.427	0.428	0.428	**0.437**	**0.433**	**0.432**
AUC	0.247	0.241	0.240	0.246	0.243	0.247	**0.248**	**0.248**	**0.255**	**0.251**	**0.251**

are crucial to learn representation that extract abstract and essential information for this tracking task.

Impact of the importance coefficient of the low-level and high-level features. To study the impact of the weight coefficient that balances the contributions of the similarity matrice S and $\bar{S}$ estimated with low-level and high-level features, respectively, as formulated in Eq. (3), we evaluate the proposed S-KeepTrack trained with different setting of the weight coefficient ω on TSFMO. The tried ω ranges from 0.0 to 1.0 with step size 0.1. Note that $\omega = 0.0$ and $\omega = 1.0$ mean using exclusively low-level and high-level features, respectively, where the latter is just the KeepTrack. Therefore, the larger the ω the more contributions of high-level features and the less of low-level one. The PRC and AUC of S-KeepTrack on TSFMO with respect to ω are shown on Table 5. As shown, the highest PRC and AUC occur when ω is less than or equal to 0.4, suggesting, in a sense, that the more contributions of low-level features the higher the tracking performance. Specifically, when ω=0.2, S-KeepTrack achieves the best PRC and AUC, i.e., 0.437 and 0.255, which is used as the default setting of ω for S-KeepTrack. When ω=1.0 S-KeepTrack reduces to KeepTrack, having PRC and AUC of 0.424 and 0.247, respectively. S-KeepTrack surpasses KeepTrack on PRC and AUC by 1.3% and 0.8%, respectively, owing to introduced low-level features. Remarkably, we can observe that using low-level features is more effective than high-level features, as when ω=0.0, i.e., using low-level features exclusively, S-KeepTrack achieves PRC and AUC of 0.432 and 0.251, surpassing KeepTrack with gaps of 0.8% and 0.4% on PRC and AUC, respectively.

6 Conclusion

In this work, we explore a new tracking task, i.e., tracking small and fast moving objects. In particular, we propose the TSFMO, which is the first benchmark for small and fast moving object tracking, to our best knowledge. In addition, in order to understand the performance of existing trackers and to provide baseline for future comparison, we extensively evaluate 20 state-of-the-art tracking algorithms with in-depth analysis. Moreover, we propose a novel tracker, named S-KeepTrack, by combining low-level and high-level features to obtain a stronger tracker, which outperforms existing state-of-the-art tracking algorithms by a clear margin. Our experiments suggest that there is a big room for us to improve the performance of tracking small and fast moving objects. We believe that, the benchmark, evaluation and the baseline tracker will inspire and facilitate more future research and application on tracking small and fast moving objects.

References

1. Ahmadi, K., Salari, E.: Small dim object tracking using frequency and spatial domain information. Pattern Recognit. **58**, 227–234 (2016)
2. Bertinetto, L., Valmadre, J., et al.: Fully-convolutional siamese networks for object tracking. In: ECCV. pp. 850–865. Springer (2016)
3. Bhat, G., Danelljan, M., et al.: Learning discriminative model prediction for tracking. In: ICCV. pp. 6182–6191 (2019)
4. Bhat, G., Danelljan, M., et al.: Know your surroundings: Exploiting scene information for object tracking. In: ECCV (2020)
5. Bolme, D.S., Beveridge, J.R., et al.: Visual object tracking using adaptive correlation filters. CVPR pp. 2544–2550 (2010)
6. Cao, Z., Fu, C., et al.: Hift: Hierarchical feature transformer for aerial tracking. ICCV pp. 15437–15446 (2021)
7. Chen, C., Liu, M.Y., et al.: R-cnn for small object detection. In: ACCV (2016)
8. Chen, X., Yan, B., et al.: Transformer tracking. CVPR pp. 8122–8131 (2021)
9. Chicco, D.: Siamese neural networks: An overview. Artificial Neural Networks pp. 73–94 (2021)
10. Colyer, S.L., Evans, M., et al.: A review of the evolution of vision-based motion analysis and the integration of advanced computer vision methods towards developing a markerless system. Sports Medicine - Open **4** (2018)
11. Danelljan, M., Bhat, G., et al.: Atom: Accurate tracking by overlap maximization. CVPR pp. 4655–4664 (2019)
12. Danelljan, M., Hager, G., et al.: Learning spatially regularized correlation filters for visual tracking. In: ICCV. pp. 4310–4318 (2015)
13. Danelljan, M., Hager, G., et al.: Discriminative scale space tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(8), 1561–1575 (2017)
14. Danelljan, M., LucVanGool, Timofte, R.: Probabilistic regression for visual tracking. In: CVPR. pp. 7183–7192 (2020)
15. Danelljan, M., Robinson, A., et al.: Beyond correlation filters: Learning continuous convolution operators for visual tracking. In: ECCV (2016)
16. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. CVPRW pp. 337–33712 (2018)
17. Fan, H., Bai, H., et al.: Lasot: A high-quality large-scale single object tracking benchmark. International Journal of Computer Vision **129**, 439–461 (2021)
18. Fan, H., Lin, L., et al.: Lasot: A high-quality benchmark for large-scale single object tracking. CVPR pp. 5369–5378 (2019)
19. Fan, H., Miththanthaya, H.A., et al.: Transparent object tracking benchmark. arXiv (2020)
20. Fan, H., Yang, F., et al.: Tracklinic: Diagnosis of challenge factors in visual tracking. WACV pp. 969–978 (2021)
21. Galoogahi, H.K., Fagg, A., et al.: Need for speed: A benchmark for higher frame rate object tracking. ICCV pp. 1134–1143 (2017)
22. Gong, Y., Yu, X., et al.: Effective fusion factor in fpn for tiny object detection. WACV pp. 1159–1167 (2021)
23. Guo, D., Shao, Y., et al.: Graph attention tracking. CVPR pp. 9538–9547 (2021)
24. Guo, D., Wang, J., et al.: Siamese fully convolutional classification and regression for visual tracking. In: CVPR. pp. 6269–6277 (2020)
25. Hong, M., Li, S., et al.: Sspnet: Scale selection pyramid network for tiny person detection from uav images. IEEE Geoscience and Remote Sensing Letters **19**, 1–5 (2022)

26. Huang, L., Zhao, X., Huang, K.: Got-10k: A large high-diversity benchmark for generic object tracking in the wild. IEEE Transactions on Pattern Analysis and Machine Intelligence **43**, 1562–1577 (2021)
27. Huang, Y., Llach, J., Zhang, C.: A method of small object detection and tracking based on particle filters. ICPR pp. 1–4 (2008)
28. Jiang, J., Zhang, X.: Research on moving object tracking technology of sports video based on deep learning algorithm. ICISCAE (2021)
29. Jiao, L., Wang, D., et al.: Deep learning in visual tracking: A review. IEEE Transactions on Neural Networks and Learning Systems **PP** (2021)
30. Kembhavi, A., Harwood, D., Davis, L.S.: Vehicle detection using partial least squares. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**, 1250–1265 (2011)
31. Kerr, R.: Technologies for judging, umpiring and refereeing. In: Sport and technology, pp. 114–134. Manchester University Press (2016)
32. Kong, T., Sun, F., et al.: Foveabox: Beyound anchor-based object detection. IEEE Transactions on Image Processing **29**, 7389–7398 (2020)
33. Kristan, M., Leonardis, A., et al.: The visual object tracking vot2017 challenge results. ICCVW pp. 1949–1972 (2017)
34. Kristan, M., Leonardis, A., et al.: The eighth visual object tracking vot2020 challenge results. In: ECCV Workshops (2020)
35. Kristan, M., Matas, J., et al.: A novel performance evaluation methodology for single-target trackers. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**, 2137–2155 (2016)
36. Lapinski, M., Brum Medeiros, C., et al.: A wide-range, wireless wearable inertial motion sensing system for capturing fast athletic biomechanics in overhead pitching. Sensors **19**(17), 3637 (2019)
37. Li, A., Lin, M., et al.: Nus-pro: A new visual tracking challenge. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**, 335–349 (2016)
38. Li, B., Wu, W., et al.: Siamrpn++: Evolution of siamese visual tracking with very deep networks. In: CVPR. pp. 4282–4291 (2019)
39. Li, F., Tian, C., et al.: Learning spatial-temporal regularized correlation filters for visual tracking. In: CVPR. pp. 4904–4913 (2018)
40. Li, S., Jiang, Q., et al.: Asymmetric discriminative correlation filters for visual tracking. Frontiers Inf. Technol. Electron. Eng. **21**, 1467–1484 (2020)
41. Li, S., Liu, Y., et al.: Learning residue-aware correlation filters and refining scale estimates with the grabcut for real-time uav tracking. 3DV pp. 1238–1248 (2021)
42. Li, S., Liu, Y., et al.: Learning residue-aware correlation filters and refining scale estimates with the grabcut for real-time uav tracking. In: 3DV. pp. 1238–1248 (2021)
43. Li, S., Liu, Y., et al.: Learning residue-aware correlation filters and refining scale for real-time uav tracking. Pattern Recognition p. 108614 (2022)
44. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. In: ECCV. pp. 254–265 (2014)
45. Liang, P., Blasch, E., Ling, H.: Encoding color information for visual tracking: Algorithms and benchmark. IEEE Transactions on Image Processing **24**, 5630–5644 (2015)
46. Liu, C., Ding, W., et al.: Aggregation signature for small object tracking. IEEE Transactions on Image Processing **29**, 1738–1747 (2020)
47. Lu, H., Wang, D.D.: Online visual tracking. In: Springer Singapore (2019)
48. Lukei, A., Kart, U., et al.: Cdtb: A color and depth visual object tracking dataset and benchmark. ICCV pp. 10012–10021 (2019)

49. Ma, C., Huang, J.B., et al.: Hierarchical convolutional features for visual tracking. ICCV pp. 3074–3082 (2015)
50. Mayer, C., Danelljan, M., et al.: Learning target candidate association to keep track of what not to track. ICCV pp. 13424–13434 (2021)
51. Mazzeo, P.L., Ramakrishnan, S., Spagnolo, P.: Visual object tracking with deep neural networks (2019)
52. Morariu, V.I., Ahmed, E., et al.: Composite discriminant factor analysis. WCACV pp. 564–571 (2014)
53. Mueller, M., Smith, N.G., Ghanem, B.: A benchmark and simulator for uav tracking. In: ECCV (2016)
54. Müller, M., Bibi, A., et al.: Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In: ECCV (2018)
55. Rozumnyi, D., Matas, J., et al.: The world of fast moving objects. CVPR pp. 4838–4846 (2017)
56. Sarlin, P.E., DeTone, D., et al.: Superglue: Learning feature matching with graph neural networks. CVPR pp. 4937–4946 (2020)
57. Song, S., Xiao, J.: Tracking revisited using rgbd camera: Unified benchmark and baselines. ICCV pp. 233–240 (2013)
58. Tamir, I., Bar-eli, M.: The moral gatekeeper: Soccer and technology, the case of video assistant referee (var). Frontiers in Psychology **11** (2020)
59. Tong, K., Wu, Y., Zhou, F.: Recent advances in small object detection based on deep learning: A review. Image Vis. Comput. **97**, 103910 (2020)
60. Valmadre, J., Bertinetto, L., et al.: Long-term tracking in the wild: A benchmark. ArXiv **abs/1803.09502** (2018)
61. Wang, N., gang Zhou, W., et al.: Transformer meets tracker: Exploiting temporal context for robust visual tracking. CVPR pp. 1571–1580 (2021)
62. Wang, Q., Zhang, L., et al.: Fast online object tracking and segmentation: A unifying approach. CVPR pp. 1328–1338 (2019)
63. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: A benchmark. In: CVPR (2013)
64. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. IEEE Transactions on Pattern Analysis and Machine Intelligence **37**, 1834–1848 (2015)
65. Xu, Y., Wang, Z., et al.: Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In: AAAI. vol. 34, pp. 12549–12556 (2020)
66. Xue, Y., Song, Y., et al.: Automatic video annotation system for archival sports video. WACVW pp. 23–28 (2017)
67. Yan, B., Peng, H., et al.: Lighttrack: Finding lightweight neural networks for object tracking via one-shot architecture search. CVPR pp. 15175–15184 (2021)
68. Zaveri, M.A., Merchant, S.N., Desai, U.B.: Small and fast moving object detection and tracking in sports video sequences. ICME **3**, 1539–1542 Vol.3 (2004)
69. Zhang, Z., Liu, Y., et al.: Learn to match: Automatic matching network design for visual tracking. ICCV pp. 13319–13328 (2021)
70. Zhang, Z., Peng, H.: Ocean: Object-aware anchor-free tracking. ArXiv **abs/2006.10721** (2020)
71. Zhang, Z., Peng, H., Wang, Q.: Deeper and wider siamese networks for real-time visual tracking. CVPR pp. 4586–4595 (2019)
72. Zhou, Z., Pei, W., et al.: Saliency-associated object tracking. ICCV pp. 9846–9855 (2021)
73. Zita, A., roubek, F.: Tracking fast moving objects by segmentation network. ICPR pp. 10312–10319 (2021)