International Journal of Computer Vision
https://doi.org/10.1007/s11263-023-01908-5

SOTVerse: A User-Defined Task Space of Single Object Tracking

Shiyu Hu[1,2] · **Xin Zhao**[1,2] · **Kaiqi Huang**[1,2,3]

Received: 8 January 2023 / Accepted: 12 September 2023

Abstract
Single object tracking (SOT) research falls into a cycle—trackers perform well on most benchmarks but quickly fail in challenging scenarios, causing researchers to doubt the insufficient data content and take more effort to construct larger datasets with more challenging situations. However, inefficient data utilization and limited evaluation methods more seriously hinder SOT research. The former causes existing datasets can not be exploited comprehensively, while the latter neglects challenging factors in the evaluation process. In this article, we systematize the representative benchmarks and form a **s**ingle **o**bject **t**racking meta**verse** (**SOTVerse**)—a user-defined SOT task space to break through the bottleneck. We first propose a **3E Paradigm** to describe tasks by three components (i.e., environment, evaluation, and executor). Then, we summarize task characteristics, clarify the organization standards, and construct SOTVerse with 12.56 million frames. Specifically, SOTVerse automatically labels challenging factors per frame, allowing users to generate user-defined spaces efficiently via construction rules. Besides, SOTVerse provides two mechanisms with new indicators and successfully evaluates trackers under various subtasks. Consequently, SOTVerse first provides a strategy to improve resource utilization in the computer vision area, making research more standardized. The SOTVerse, toolkit, evaluation server, and results are available at http://metaverse.aitestunion.com.

Keywords Single object tracking · Experimental environment · Evaluation system · Performance analysis

1 Introduction

As the fundamental computer vision task, single object tracking (SOT (Kristan et al., 2013; Wu et al., 2015; Fan et al., 2021; Huang et al., 2021; Hu et al., 2023), i.e., locates a user-specified moving target in a video) aims to model the powerful human dynamic vision ability (JW, 1962; Biederman, 1987; Lee & Seung, 1999; McLeod et al., 2003; Land & McLeod, 2000; Beals et al., 1971; Burg, 1966; Kohl et al., 1991), and has been widely used in daily application scenarios like self-driving cars (Kim et al., 2019; Kong & Fu, 2022; Dendorfer et al., 2021), intelligent monitoring (Yoon et al., 2019; Cook, 2012; Chu et al., 2017), augmented reality (Zhang & Vela, 2015; Abu Alhaija et al., 2018; Gauglitz et al., 2011) and robot navigation (Dupeyroux et al., 2019; Held et al., 2016; Ramakrishnan et al., 2021). When we look back at the evolution of SOT task, we can find that the task definition has drifted three times – from short-term tracking (Wu et al., 2013, 2015) to long-term tracking (Fan et al., 2021; Kristan et al., 2019), and then to spatiotemporal change tracking (Hu et al., 2023). Obviously, the expansion of task definition prompts SOT to gradually model the human tracking vision ability and evolve towards general vision intelligence.

During the humanoid process of task definition, researchers have also contributed to constructing comprehensive benchmarks, aiming to provide high-quality *datasets* and scientific *evaluation* methods for algorithms. The first systematic SOT benchmark OTB (Wu et al., 2013) was successfully

Communicated by Matej Kristan.

✉ Shiyu Hu
hushiyu2019@ia.ac.cn

✉ Xin Zhao
xzhao@nlpr.ia.ac.cn

Kaiqi Huang
kqhuang@nlpr.ia.ac.cn

[1] School of Artificial Intelligence, University of Chinese Academy of Sciences, No. 19(A) Yuquan Road, Beijing 100049, China

[2] Institute of Automation, Chinese Academy of Sciences, 95 Zhongguancun East Road, Beijing 100190, China

[3] Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, 320 Yue Yang Road, Shanghai 200031, China

Published online: 18 October 2023

released in 2013. In the following decade, more and more benchmarks have been successfully constructed with larger dataset scales, and richer video content (Fan et al., 2021; Huang et al., 2021; Hu et al., 2023). At the same time, several researchers (Čehovin et al., 2016; Kristan et al., 2016; Lukežič et al., 2020) also design various evaluation mechanisms to accomplish performance analysis via different perspectives. These benchmarks provide profitable environments and standardized evaluation processes, greatly facilitating the development of data-driven trackers.

Through the above analyses, the ideal research route should be an upward spiral: a more human-like task definition promotes a more complicated benchmark construction, and ultimately guides to more intelligent algorithms. However, some bad cases show that current research falls into a cycle – state-of-the-art (SOTA) algorithms perform well on most benchmarks but quickly fail when facing challenging factors in real application scenarios, causing researchers to doubt the insufficiency of benchmarks; thus, researchers usually spend a lot of effort constructing a larger dataset to solve this problem (e.g., the scale of SOT benchmarks in the past decade has been expended nearly 250 times). But many actual examples, like the frequent self-driving accidents, indicate that *only expanding the dataset scale cannot break this bottleneck.*

To find the core reasons for this phenomenon, we first analyze existing issues separately from the perspective of datasets and evaluation:

- **For the dataset aspect, existing data has not been exploited effectively.** SOT task has evolved different characteristics during the development process, which is the primary reason yielding benchmarks to follow miscellaneous data collection rules. This phenomenon leads to inconsistencies in the construction process, causing experimental environments to become isolated. Existing datasets can only be compared in superficial features like dataset scale but are difficult to contrast in vital components such as content difficulty (e.g., challenging factors are always selected to represent the difficulty, while various benchmarks annotate challenging attributes by different metrics, and many classical benchmarks only provide sequence-level annotations rather than frame-level). Besides, although the well-known VOT competition (Kristan et al., 2016) has designed a sequence sampling algorithm to automatically select representative sequences from a data pool, the efficiency of this strategy in massive data space is not high enough. Thus, when researchers aim to investigate tasks in more complex scenarios, most of them usually reconstruct a larger dataset rather than extracting relevant data from existing datasets.
- **For the evaluation aspect, limitations of evaluation methods lead to neglect of challenging factors.** Multiple researchers always overlook the shortcomings of evaluation methods. In fact, existing benchmarks mainly run trackers on sequences, get frame-by-frame scores, and finally calculate the average value to represent the overall performance. However, SOT is a sequential decision task and is seriously affected by challenging factors (e.g., frames with fast motion or tiny objects), while regular tracking sequences in benchmarks are usually composed of many *simple frames* and scant *challenging frames*. Thus, bad performance is ignored after averaging due to the low proportion of challenging frames.

The above problems hinder related research, increase hardship for resource integration and utilization, and ultimately create bottlenecks in research. In this work, we systematize the representative benchmarks and construct a comprehensive **s**ingle **o**bject **t**racking meta**verse** named **SOTVerse** to solve the issues, as shown in Fig. 1. Like DeepMind (Team et al., 2021) defines reinforcement learning tasks as world, game, and co-players, we propose a **3E Paradigm** to describe computer vision tasks by three components (i.e., *environment*, *evaluation*, and *executor*). Among them, datasets provide the *environment* to portray task characteristics, *evaluation* methods measure performance from multiple aspects, and *executors* can estimate their visual tracking abilities via SOTVerse.

Specifically, to integrate different environments into SOTVerse, we first correspond task characteristics with data collection rules, clarify the organization standards, and construct them as the *normal space*. In particular, we also provide various challenging attribute labels for each frame, allowing users to extract related sub-sequences from SOTVerse and efficiently generate *challenging spaces* with their research purpose. Besides, to overcome the limitations of traditional evaluation methods, SOTVerse provides three novel indicators to focus on tracking robustness under challenging factors.

Obviously, SOTVerse is a customizable and extensible space. We summarize the contributions as follows:

- **A paradigm to describe computer vision tasks.** Computer vision tasks can be characterized by *environment*, *evaluation*, and *executor*. Figure 1 illustrates the **3E paradigm** by analyzing SOT in detail: we synthesize the *environment* and *evaluation* to form **SOTVerse** – a user-defined single object tracking task space, and conduct experiments in this space to judge *executors*' tracking ability. Definitely, this paradigm can be expanded to describe different visual tasks and help users improve their research efficiency.

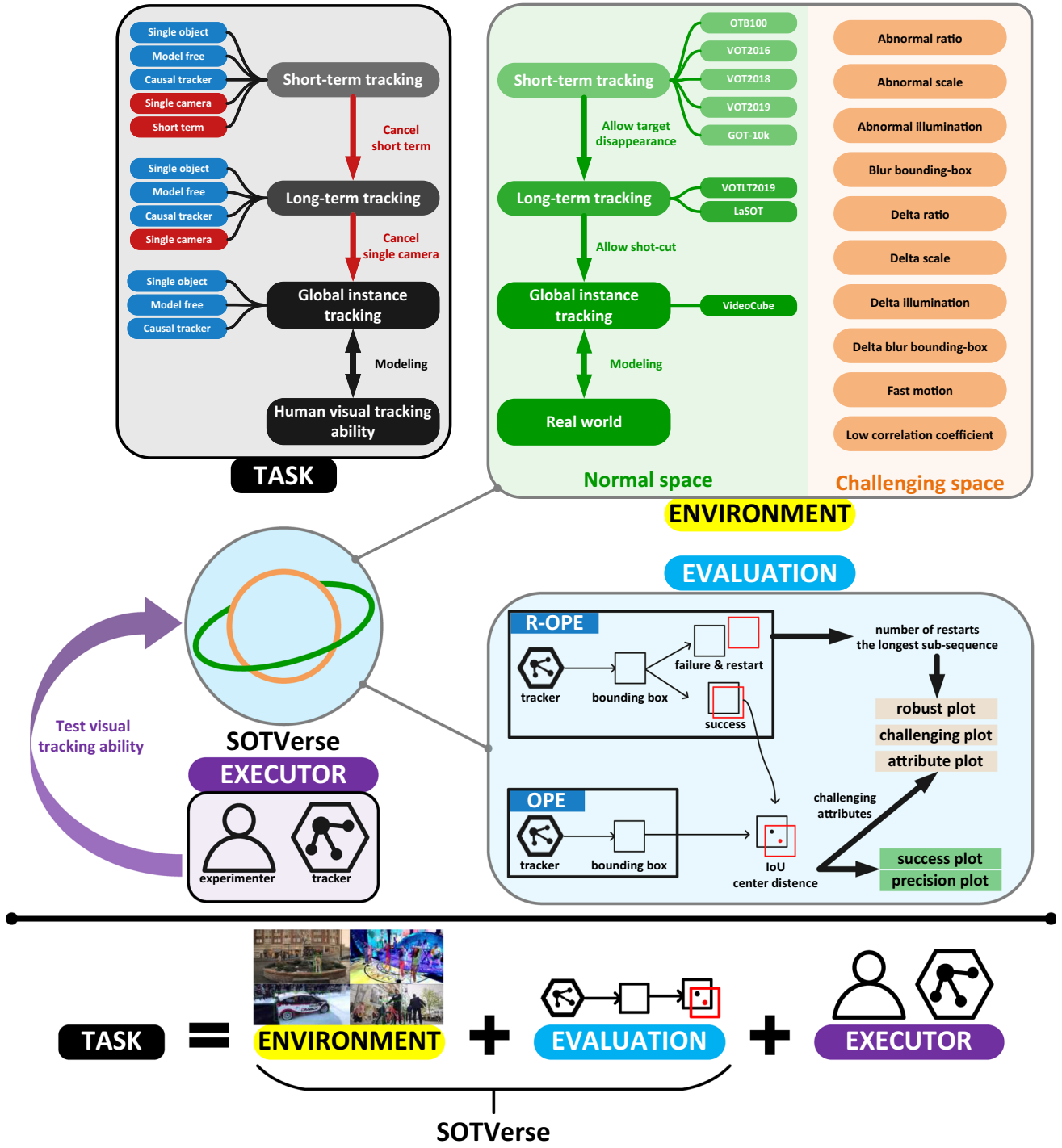


Fig. 1 The **3E Paradigm** to describe the SOT task. A computer vision task can be characterized by three elements (environment, evaluation, and executor). (**TASK**) For the SOT task, constraints contained in the definition are gradually eliminated during development. (**ENVIRONMENT**) Environment portrays task characteristics. We first select eight representative datasets to form the SOTVerse and then label multiple challenging attributes for each frame. Users can quickly extract related sub-sequences for different tasks, such as selecting abnormal ratio sub-sequences to create a single deformable object tracking space. (**EVALUATION**) SOTVerse provides diverse evaluation mechanisms and evaluation indicators to measure performance. (**EXECUTOR**) Both algorithms and human experimenters can test their visual tracking capabilities via SOTVerse

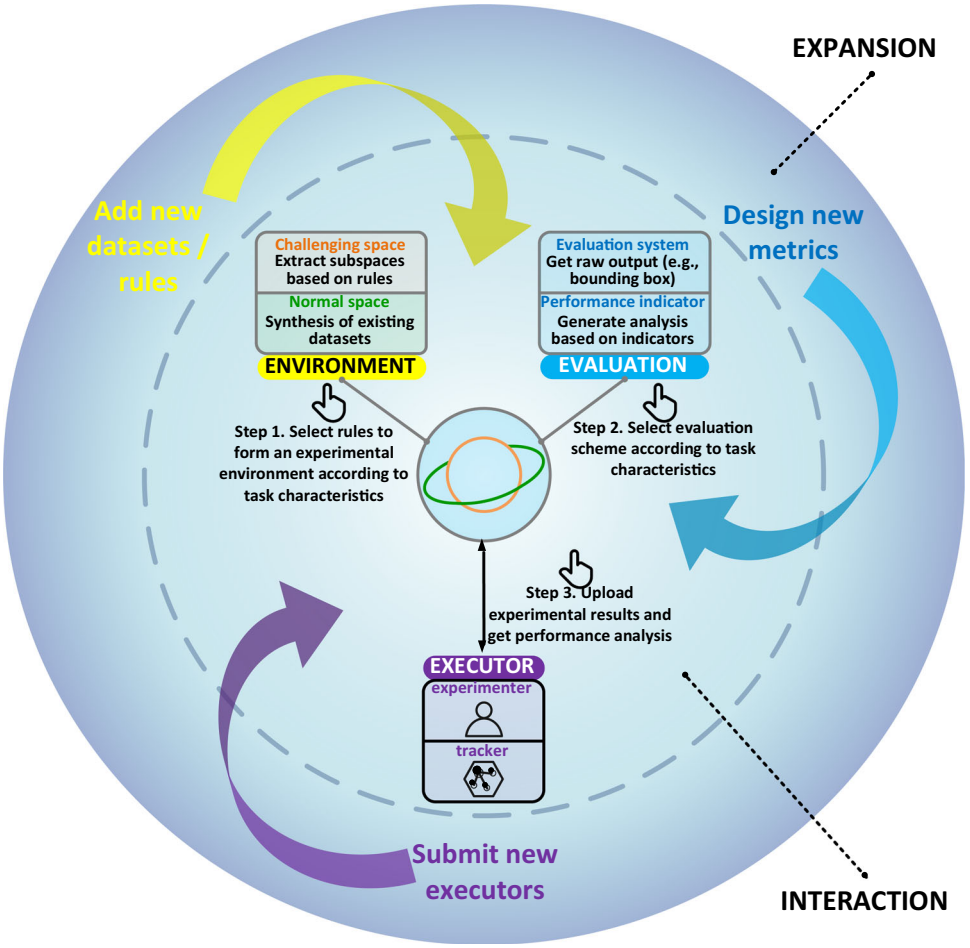


Fig. 2 The user-defined process of SOTVerse, can be divided into interaction and expansion. (**INTERACTION**) Users need three steps to finish the operation: first, select data extraction rules according to task characteristics to generate an experimental environment. Then, determine the appropriate evaluation system and performance indicators. Finally, upload the experimental results and obtain the corresponding performance analysis. (**EXPANSION**) Users can expand the SOTVerse by adding new datasets or extraction rules, designing new metrics, and submitting new executors

- **A comprehensive and user-defined environment.** Through precise analyses of the task definition, we organize existing benchmarks to form the *environment* of SOTVerse. It includes 12.56 million frames and frame-level challenging attribute labels to model the real world. Notably, the thresholds for determining challenging factors are selected by their distribution on the whole environment. Besides, an environment generation method can efficiently help researchers form their own task space. Therefore, unlike traditional benchmarks' isolated and static design, SOTVerse is a comprehensive and dynamic experimental environment.
- **A thoroughgoing evaluation scheme.** We first point out the limitations of existing systems and indicators through detailed analysis; then design a new evaluation scheme, which includes two mechanisms and new metrics to satisfy various tasks.
- **Various experimental executors and detailed analysis.** We conduct extensive experiments in the SOTVerse and perform performance analysis on various executors. Experimental results show that challenging factors severely hamper tracking performance – the proposed challenging plot reveals that high scores are mainly obtained in normal frames, while the success rate of most trackers is less than 0.5 under challenging situations. Finally, we point out the necessity of the re-initialization mechanism for evaluation in long sequences. These results indicate the shortcomings of existing work and verify the effectiveness of the evaluation scheme in SOTVerse.

We provide a comprehensive online platform at http://metaverse.aitestunion.com to help users operate SOTVerse. The user-defined process illustrated by Fig. 2 can be divided into *interaction* and *expansion*. With our platform, users can select the environment generation method according to task characteristics and directly download the generated experimental environment. Besides, an open-sourced toolkit is available to accomplish the evaluation process. Finally, users can upload the experimental results and obtain the corresponding performance analysis. In addition, we accommodate users to expand SOTVerse. For example, users can provide new datasets or develop new environment generation methods to enrich the experimental environment. They

can also formulate new evaluation mechanisms and quickly verify the effectiveness of various subtasks.

Evidently, SOTVerse allows users to customize tasks according to their own research purposes. It not only makes research more targeted, but also can significantly improve research efficiency. Furthermore, the 3E Paradigm successfully performed in the SOT area provides an excellent example, which can be referenced by various visual or other domain tasks in the future.

The rest is organized as follows. Section 2 provides a review of the SOT task. Section 3 introduces the design principles of SOTVerse. Section 4 describes the experimental results and detailed analysis. Finally, the conclusions and discussions of future work are summarized in Sect. 5.

2 Related Work

2.1 Task

Understanding a task includes (1) task definition analysis and (2) task description paradigm. The former is an external description to distinguish a task from others through strict boundaries. The latter is an internal description representing a task through environment and execution standards.

2.1.1 Task Definition Analysis

SOT is usually defined as only providing the initial position of an arbitrary object and continuously locating it in a video sequence (Wu et al., 2013, 2015). Since 2013, researchers have proposed several influential benchmarks (Wu et al., 2015; Muller et al., 2018; Fan et al., 2021; Huang et al., 2021; Hu et al., 2023) – the organized datasets and unified metrics promote the SOT research. However, limited by the research level, early definition adds additional constraints to simplify the task. The influential VOT competition limits this task to five keywords: *single-target*, *model-free*, *causal trackers*, *single-camera*, and *short-term* (Kristan et al., 2016). The first three keywords (*single-target*, *model-free*, *causal trackers*) correspond to the original definition and are the criteria for distinguishing SOT from other visual tasks (e.g., multi-object tracking (Ciaparrone et al., 2019; Geuther et al., 2019) and visual instance detection (Wang et al., 2018; Esteva et al., 2021; Real et al., 2017; Russakovsky et al., 2015)). In comparison, the latter two keywords (*single-camera* and *short-term*) are constraints added to simplify research in the early stage.

The development of SOT is continuously removing hidden constraints and closer to the essential definition, as shown in Fig. 3. Since 2018, some researchers have withdrawn *short-term* and proposed long-term tracking (Valmadre et al., 2018; Fan et al., 2021; Kristan et al., 2019). In 2022, researchers further remove the *single-camera* constraint and propose the global instance tracking (GIT) (Hu et al., 2023), which is supposed to search an arbitrary user-specified instance in a video without any assumptions about camera or motion consistency. Clearly, GIT realizes the basic definition of SOT by gradually removing the constraints.

2.1.2 Task Description Paradigm

The description paradigm analyzes a task from multiple dimensions, establishes specific operating rules, and provides experimental environments for executors. In other words, the paradigm transforms a monotonous task definition into several operational elements concretely. In 2021, DeepMind (Team et al., 2021) provides a task description paradigm for the reinforcement learning task, consisting of a game with a world and co-players. The world is composed of various static and dynamic elements, which can quickly combine an adapted environment according to the task characteristics. A series of goals consist of the game, which aims to guide players to maximize total reward. Players are agents who perform tasks in the world according to the game rules.

Although the computer vision area does not propose a specific task description paradigm like reinforcement learning, different researchers have tried to characterize the task from three aspects: *environment*, *evaluation*, and *executor*. Correspondingly, relevant datasets provide the execution *environment* of the task; *evaluation* methods are similar to the game rules, which measure the performance via different metrics; *executors* are the task entertainer, including related algorithms and human experimenters.

The following parts introduce the experimental environment (datasets) and evaluation methods of SOT in detail.

2.2 Environment

High-quality datasets play a vital role in SOT development. Early datasets represented by VIVID (Collins et al., 2005), CAVIAR (Fisher, 2004), and PETS (Ferryman & Shahrokni, 2009) mainly focus on surveillance scenarios, which aim to track humans or cars but lack canonical build standards. Since 2013, well-organized benchmarks represented by OTB (Wu et al., 2013, 2015) are mainly designed for short-term tracking tasks (Kristan et al., 2017; Liang et al., 2015; Mueller et al., 2016; Li et al., 2015; Kiani Galoogahi et al., 2017), which assumes no complete occlusion or target out-of-view happened in this video, as shown in Fig. 4a, b. The average duration of short-term datasets is always less than one minute, and the following benchmarks mainly innovate in video content. (e.g., TC-128 (Liang et al., 2015) evaluates color-enhanced trackers on color sequences; NUS-PRO (Li et al., 2015) focuses on tracking pedestrian and rigid objects; UAV123 (Mueller et al., 2016) assesses unmanned aerial

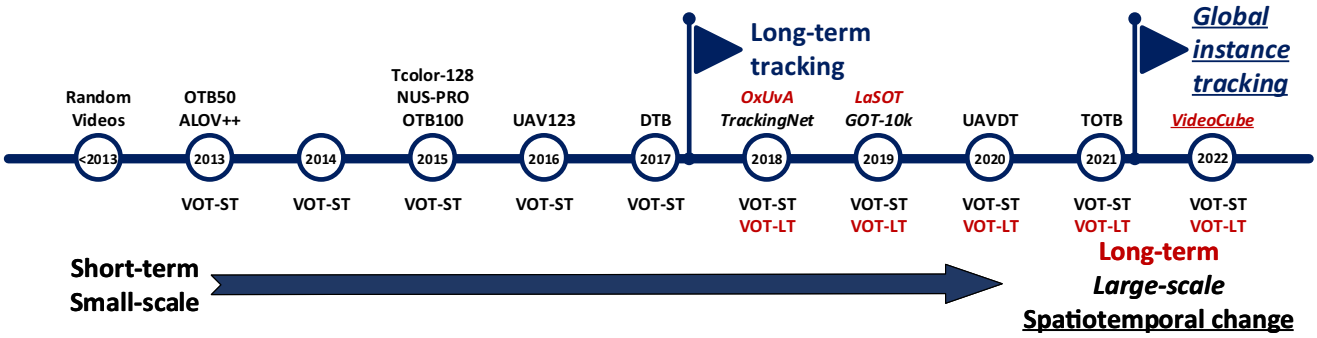


Fig. 3 The development trend of SOT benchmarks. The red font represents long-term tracking datasets, the *italic* represents large-scale datasets, and the underscore represents the spatiotemporal variations. Clearly, SOT benchmarks are developing toward larger-scale, longer-term, and more challenging tracking

Fig. 4 Examples of normal space E_n and challenging space E_c. Sequences in E_n are selected from existing datasets, while sequences in E_c are obtained based on space construction rules

vehicle tracking performance; PTB-TIR (Liu et al., 2019) and VOT-TIR (Kristan et al., 2016) are thermal tracking datasets; GOT-10k (Huang et al., 2021) includes 563 object classes based on the WordNet (Miller, 1995)).

Recently, several new benchmarks represented by LaSOT (Fan et al., 2021) have proposed long-term tracking to satisfy the demands of real scenarios (Moudgil & Gandhi, 2018; Valmadre et al., 2018). However, it is hard to separate the short-term and long-term in the time dimension. Although short-term videos are usually shorter than one minute, only adopting *one minute* as the task boundary is biased. Therefore, the VOT competition proposes a new criterion – a task that allows the target to disappear completely can be regarded as long-term tracking (Lukežič et al., 2020). In contrast to the two criteria, allowing the object to disappear for a short period is more suitable as the decisive factor for long-term tracking. By removing the constraint hidden in the definition of short-term tracking that the target should be present in the tracking process, the experimental environment can include more long-term videos to achieve the expansion from a short to a long term. As shown in Fig. 4c, a target may disappear utterly due to being out of view or be fully occluded, which is excluded in the short-term tracking environment.

Nonetheless, the implicit continuous motion assumption restricts long-term tracking environments to single-camera and single-scene, which is still far from the application scenarios of SOT. Thus, the global instance tracking environment named VideoCube is proposed (Hu et al., 2023). It includes videos with shot-cut and scene-switching to model the real world comprehensively (Fig. 4d).

Existing works build environments from different perspectives with various rules, but no one has tried to unify

the environments. When researchers try to analyze problems from new perspectives, they have to build corresponding datasets from scratch, significantly reducing research efficiency. This status inspired us to summarize and uniform existing environments to construct SOTVerse, and help researchers generate experimental environments effectively.

2.3 Evaluation

2.3.1 Evaluation System

Initialize a tracker in the first frame and continuously record the tracking results – this evaluation system is a one-pass evaluation (OPE). To utilize the failure information and analysis the breakdown reasons, OTB (Wu et al., 2015) benchmark offers a re-initialization mechanism (OPER). The re-initialization mechanism also plays a vital role in VOT competitions (Kristan et al., 2016, 2018, 2019). Trackers will be re-initialized after the assessment system detects tracking failure.

Recently, the VOT2020 challenge (Kristan et al., 2020) proposes a new anchor-based short-term tracking evaluation protocol for performance measurement. They use anchors (i.e., initialization points) to replace the reset mechanism, and require trackers to run from each anchor forward or backward, whichever direction generates the longest subsequences. Specially, the intervals between anchors are constant (e.g., 50 frames), and artificial examination is adopted to ensure each anchor contains complete target information.

Most long-term tracking benchmarks (Fan et al., 2021; Valmadre et al., 2018) select OPE mechanism as an evaluation system. However, the VOTLT competition (Lukeźič et al., 2020), which regards *target disappearance* as the manifestation of long-term tracking, hopes trackers can re-locate the target. Thus, they propose four taxonomies in experiments for accurate performance analysis, include short-term tracker (ST_0), short-term tracker with conservative updating (ST_1), pseudo long-term tracker (LT_0), and re-detecting long-term tracker (LT_1).

2.3.2 Performance Indicator

Most evaluation indicators can be summarized from precision, successful rate, and robustness. Most indicators select the positional relationship between predicted result p_t and ground-truth g_t in the t-th frame to accomplish calculation:

- *Precision* proposed by OTB (Wu et al., 2015) measures the euclidean distance between center points of p_t and g_t in pixels. Calculating the proportion of frames whose distance is less than a threshold and drawing the statistical results based on different thresholds into a curve generates the *precision plot*. Typically, 20 pixels are selected as a threshold to rank trackers. To eliminate the influence of object scale, TrackingNet (Muller et al., 2018) adopts ground-truth scale (width and height) to normalize the center distance. VideoCube (Hu et al., 2023) provides a normalized precision metric to eliminate the effect of target size.
- The overlap of p_t and g_t is calculated by $\Omega(p_t, g_t) = \frac{p_t \bigcap g_t}{p_t \bigcup g_t}$. Frames with $\Omega(p_t, g_t)$ greater than a threshold are defined as successful tracking, and the *successful rate (SR)* measures the percentage of successfully tracked frames. Drawing the results based on various thresholds is the *success plot*. The OTB (Wu et al., 2013) benchmark introduces the area-under-the-curve (AUC) score as a comprehensive measure of tracker performance. Additionally, the VOT competition (Čehovin et al., 2016) establishes and demonstrates that the average overlap (AO) is equivalent to the AUC score of the success plot. Subsequent benchmarks (Huang et al., 2021; Hu et al., 2023; Fan et al., 2021) are mainly based on this indicator to rank the algorithms.
- *Robustness* evaluates the stability of tracking. VOT competition (Kristan et al., 2016) initially applies the number of re-initialization M to calculate robustness, then converts it as $R_s = e^{-SM}$ to interpret the reliability. The VOT2020 challenge (Kristan et al., 2020) further improves it by cooperating with the anchor-based evaluation protocol (please refer to Sect. 2.3.1), and defines the robustness as the extent of the sub-sequence before the tracking failure. This multi-start evaluation is also adopted by TREK-150 (Dunnhofer et al., 2023), a benchmark for visual object tracking in first person vision, and the authors design a new evaluation metrics named generalized success robustness (GSR) for evaluation. For the GIT task (Hu et al., 2023), researchers consider the degree of frame variation and the number of failures to measure the robustness.

Besides, several long-term tracking benchmarks (Valmadre et al., 2018; Lukeźič et al., 2020) require trackers to output disappearance-judgment for calculating the *tracking accuracy*, *recall*, and *F-measure*.

The above introduction illustrates that existing evaluation systems and performance indicators are fragmented. More importantly, the impact of challenging factors has long been identified (Godec et al., 2013; Han et al., 2008; Nejhum et al., 2008; Collins, 2003; Kwon & Lee, 2009), but ignored by existing mechanisms, which mainly focus on the all-around performance of complete sequences. Therefore, when constructing the SOTVerse, we first clarify the calculation formula of performance indicators, then conduct experiments on various tasks to explore the applicable scope of different evaluation methods.

3 The Construction of SOTVerse Space

3.1 3E Paradigm

As shown in Fig. 1, a computer vision task can be described by the combination of environment, evaluation, and executor. Table 1 lists the related concepts in the SOT task. We assume that S denotes a subtask (e.g., short-term tracking task), E is the corresponding experimental environment organized by several videos (e.g., short-term dataset), M_s represents the set of evaluation systems (e.g., OPE mechanism), M_p represents the set of performance indicators (e.g., precision), T symbolizes the set of task executors (e.g., trackers and human subjects). Particularly, $\times$ represents the Cartesian product. Under the 3E Paradigm, the subtask can be represented as:

$$S = E \times M_s \times M_p \times T \tag{1}$$

On the one hand, a complete SOT task space $\mathbb{S}$ can be obtained by integrating the various subtasks. On the other hand, the set of environments, evaluation methods, and task executors can be separately symbolized as $\mathbb{E}$, $\mathbb{M}$, and $\mathbb{T}$, characterizing $\mathbb{S}$ as:

$$\mathbb{S} = \left\{S_{n_1}, S_{n_2}, \ldots, S_{c_1}, S_{c_1}, \ldots\right\} = \mathbb{E} \times \mathbb{M} \times \mathbb{T} \tag{2}$$

According to 3E Paradigm, we build a user-defined task space named SOTVerse, which integrates the existing SOT datasets into a large environmental space $\mathbb{E}$, and provides multiple indicators to combine a comprehensive evaluation space $\mathbb{M}$. With the help of SOTVerse, users can quickly extract relevant data to form the task environment and select appropriate evaluation methods for performance measurement. The following parts introduce the experimental environment $\mathbb{E}$ and evaluation methods $\mathbb{M}$ in detail.

3.2 Environment

Figure 5 illustrates the combination process of SOTVerse, which can be split into three steps.

3.2.1 Step One: Dataset Selection

First, representative datasets e_i are chosen to form normal space E_n according to the relationship between subtasks (short-term tracking S_{n_1}, long-term tracking S_{n_2}, global instance tracking S_{n_3}). Selected benchmarks can cover all subtasks and reflect the characteristics of SOT. Table 2 illustrates that the normal space includes 12.56 million frames to simulate real application scenarios fully.

Besides, we divide the normal space datasets into two categories: (1) Small-scale datasets (i.e., OTB2015, VOT2016, VOT2018, VOT2019, and VOTLT2019) are usually utilized for testing; therefore, we take all the sequences in these datasets as test sets. (2) Large-scale datasets (i.e., LaSOT, GOT-10K, and VideoCube) have divided train/val/test sets while releasing; thus, we retain the division as their original settings.

3.2.2 Step Two: Attribute Selection

We unify the attribute calculation and determine the threshold of abnormal attributes based on its distribution. All attributes are calculated from original files (sequences and ground-truth) without additional manual annotations. We split attributes into two categories: (1) *static attributes* only relate to the current frame, while (2) *dynamic attributes* record changes between consecutive frames.

For the t-th frame F_t in the sequence L, we use four values (x_t, y_t, w_t, h_t) (i.e., the coordinate information of the upper left corner and the shape of the bounding-box) to represent the target bounding-box. The calculation rules are as follows:

- The target *ratio* is defined as $r_t = \frac{h_t}{w_t}$. Original target scale can be calculated via $s_t = \sqrt{w_t h_t}$, to further weaken the impact of image resolution, we calculate *relative scale* by $s_t^{'} = \frac{s_t}{\sqrt{W_t H_t}}$ (W_t and H_t represent the image resolution of F_t).
- Video recorded in special light conditions (e.g., dim light or blinding light) can be transferred to standard illumination by multiplying a correction matrix C_t (Finlayson & Trezzi, 2004). *Illumination* can be quantified by the Euclidean distance between C_t and $\mathbf{1}^{1\times 3}$.
- We use Laplacian transform (Pech-Pacheco et al., 2000) to calculate the *blur bounding-box* degree. We first convert the RGB bounding-box into gray-scale G_t, then convolve G_t with a Laplacian kernel, and calculate the variance as sharpness.
- Dynamic attributes are generated from the variation of static attributes. Correspondingly, we define the variations in two sequential frames as *delta ratio*, *delta relative scale*, *delta illumination*, and *delta blur bounding-box*.
- Besides, we use *fast motion* to quantify the target center distance between consecutive frames by $d_t = \frac{\|c_t - c_{t-1}\|_2}{\sqrt{max(s_t, s_{t-1})}}$.
- *Correlation coefficient* measures the similarity between progressive frames. We select the Pearson product-moment correlation coefficient $\rho_t = \frac{\mathrm{cov}(F_t, F_{t-1})}{\sigma_{F_t}\sigma_{F_{t-1}}}$, in which the numerator calculates the covariance of F_t and F_{t-1}, and the denominator is the product of the standard deviation. The correlation coefficient reflects the changes between consecutive frames and has been normalized in [0, 1]. Based on its definition and calculation process, the corrcoef comprehensively reflects the dynamic variation degree between continuous frames (Fig. 18). According

Table 1 The symbol of task description paradigm in SOT

Symbol		Implication	Definition in SOT
$\mathbb{S}$	Task	The definition of a task	Locating a user-specified target in a video
S	Subtask	Subtask formed by adding constraints to the original task definition	Normal subtask S_n (e.g., short-term tracking, long-term tracking, global instance tracking) and challenging subtask S_c (i.e., locating the target in the challenging situation)
$\mathbb{E}$	Environment	An execution space of a task, usually organized by datasets	Combination of representative SOT datasets
E	Subspace	Execution space of subtask	Normal sub-space E_n and challenging sub-space E_c
$\mathbb{M}$	Evaluation	Methods to evaluate the abilities of executors	Evaluation system M_s and performance indicator M_p
$\mathbb{T}$	Executor	Task executor	Tracker T_t or human subject T_h

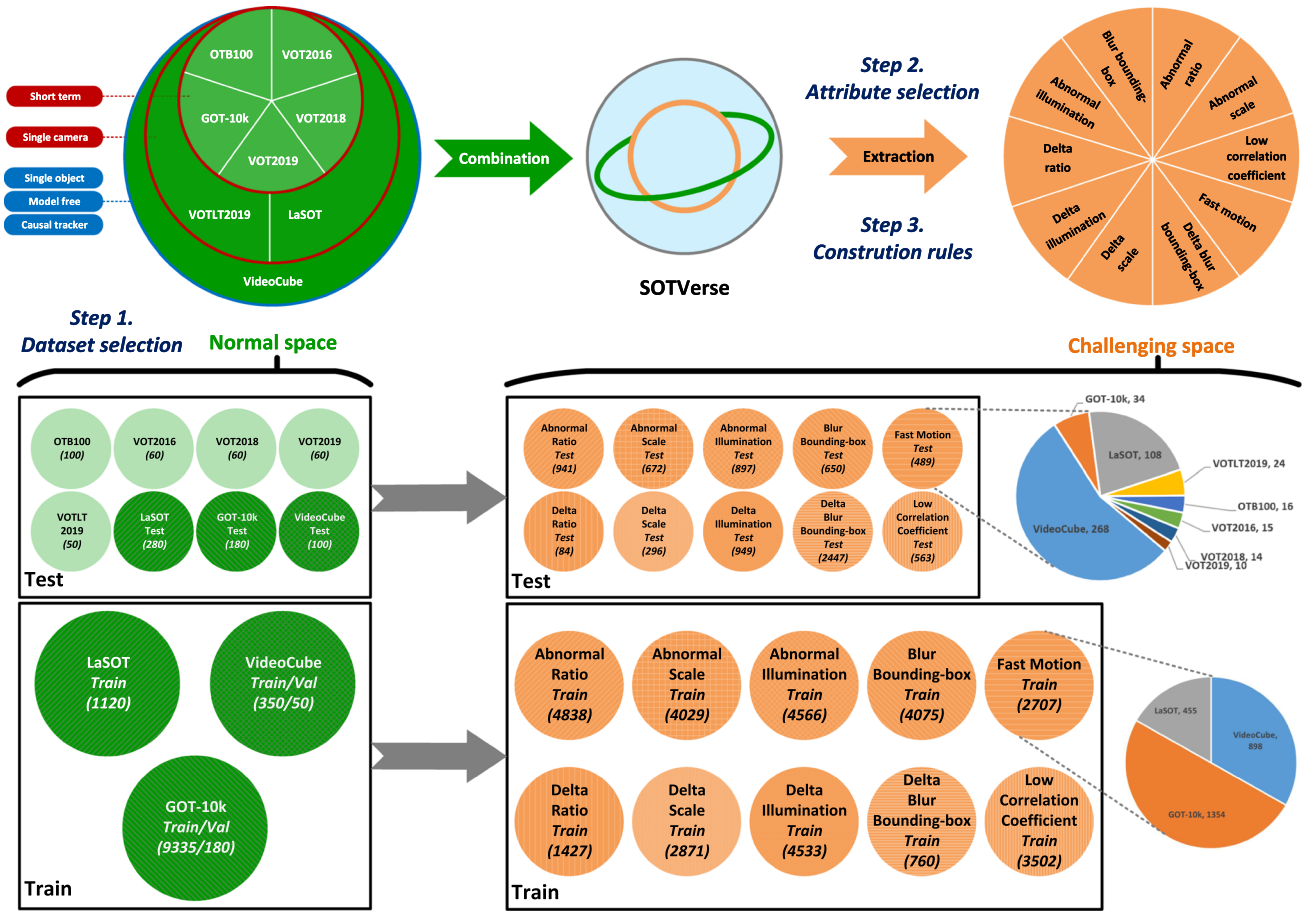


Fig. 5 The combination process of SOTVerse. First, representative datasets e_i are chosen to form the normal space E_n according to the relationship between SOT subtasks (short-term tracking S_{n_1}, long-term tracking S_{n_2}, global instance tracking S_{n_3}). Second, we summarize the challenging factors into ten attributes and automatically label these attributes per frame. Finally, we design space construction rules, which help users quickly extract eligible sub-sequences from SOTVerse to form a challenging space E_c based on research goals. It is worth noting that, due to some repeated sequences in VOT2016, VOT2018, and VOT2019, we have removed duplicates from the three datasets when constructing the challenge space, ensuring that the constructed subspace does not contain any repeated sequences

Table 2 The representative benchmarks selected to form the normal space of SOTVerse

Environment			Videos (train/val/*test*)	Min frame	Mean frame	Max frame	Total frame	Subtask		Target absent	Shotcut
E_{n_1}	e_1	OTB2015 (Wu et al., 2015)	100	71	590	3,872	59K	S_{n_1}	Short-term tracking	✗	✗
	e_2	VOT2016 (Kristan et al., 2016)	60	41	357	1,500	21K			✗	✗
	e_3	VOT2018 (Kristan et al., 2018)	60	41	356	1,500	21K			✗	✗
	e_4	VOT2019 (Kristan et al., 2019)	60	41	332	1,500	20K			✗	✗
	e_5	GOT-10k (Huang et al., 2021)	9,695 (9,335/180/*180*)	29	149	1,418	1.45M			✗	✗
E_{n_2}	e_6	VOTLT2019 (Kristan et al., 2019)	50	1,389	4,305	29,700	215K	S_{n_2}	Long-term tracking	✓	✗
	e_7	LaSOT (Fan et al., 2021)	1,400 (1,120/*280*)	1,000	2,502	11,397	3.5M			✓	✗
E_{n_3}	e_8	VideoCube (Hu et al., 2023)	500(350/50/*100*)	4,008	14,920	29,834	7.46M	S_{n_3}	Global instance tracking	✓	✓

OTB (Wu et al., 2015), VOT series (Kristan et al., 2016, 2018, 2019) and GOT-10k (Huang et al., 2021) represent the short-term tracking benchmarks; VOTLT2019 (Kristan et al., 2019) and LaSOT (Fan et al., 2021) represent the long-term tracking benchmarks; VideoCube (Hu et al., 2023) represents the global instance tracking benchmark

Fig. 6 The attribute distribution and example on SOTVerse. We use a box-plot to illustrate the attribute distribution, the distribution boundaries (i.e., the leftmost level of the 25th percentile or the rightmost level of the 75th percentile) of attribute values over the eight datasets are regarded as abnormal criteria. Notably, the distribution of the OTB (Wu et al., 2015) in the abnormal illumination is significantly different from other datasets, since the partial sequences of OTB are grayscale, resulting in the calculation result being 0. Therefore, we remove the OTB before confirming the lower bound of the abnormal illumination. Similarly, we remove the VideoCube (Hu et al., 2023), which is obviously different from other datasets, to determine the boundary in the blur bounding-box

to its distribution on SOTVerse, $\rho_t \leq 0.75$ can be considered a significant variation between constant frames.

We note that determining the threshold of abnormal attributes in existing benchmarks is subjective. For example, TrackingNet (Muller et al., 2018) and LaSOT (Fan et al., 2021) regard the area smaller than 1,000 pixels as low resolution (i.e., tiny object), while GOT-10k (Huang et al., 2021) considers the target smaller than half of the frames is tiny. Thus, we first ensure the above calculation formulas are applicable to all situations (e.g., we eliminate the influence of image resolution variation). The frame whose attribute value lies in the abnormal interval is defined as a *challenging frame*; otherwise, it is a *normal frame*. To avoid the influence of subjective factors, we select abnormal thresholds via attributes' distribution in 12.56 million frames, as shown in Fig. 6 and Table 3. Consequently, our method excludes human interference and suits all benchmarks.

3.2.3 Step Three: Space Construction Rules

Space construction rules are proposed based on intensive attribute annotation. We hope users can extract relevant data from the normal space, and quickly form challenging spaces according to their research goals. If more than half of the frames in a sequence are challenging frames of attribute a_i, the sequence will be regarded as a *challenging sequence*. The *challenging sub-space* c_i is consisted of challenging sequences of a_i. Figure 7 and Algorithm 1 shows the process of the space construction method, including data screening and deduplication:

- **Data screening** aims to find all challenging sub-sequences in an original sequence. (1) Firstly, we identify all appropriate start points in the original series. It is impractical to initialize trackers in frames with small or blurry targets, and manual selection of start points is time-consuming. Consequently, we utilize the attribute labels of *relative scale* and *blur bounding-box degree* to filter out low-

Algorithm 1 Framework of space construction.

Input: L: original sequence; $|\cdot|$: the cardinality; S: the list of start points in L, sorted by frame number in ascending order; C: the list of challenging frames in L, sorted by frame number in ascending order

Output: L_l: the set of challenging sub-sequences in L

/* Step 1: data screening */
screening set $L_s = \varnothing$
for $i \leftarrow 0$ **to** $|S| - 1$ **do**
  start flag $\alpha \leftarrow S[i]$
  end flag $\beta \leftarrow |L|$
  while $\beta > \alpha$ **do**
    sub-sequence $l \leftarrow L[\alpha : \beta]$
    if $\frac{|l \cap C|}{|l|} \geq 0.5$ **then**
      $L_s \leftarrow L_s \cup l$
    else
      $\beta \leftarrow \beta - 1$
/* Step 2: data deduplication */
$L_s \leftarrow \text{DescendingSort}(L_s)$
extraction set $L_l = \varnothing$
for $i \leftarrow 0$ **to** $|L_s| - 1$ **do**
  sub-sequence $l_s \leftarrow L_s[i]$
  for $j \leftarrow 0$ **to** $|L_c| - 1$ **do**
    sub-sequence $l_c \leftarrow L_c[j]$
    if $\frac{|l_s \cap l_c|}{|l_s|} \geq 0.5$ **then**
      if $(j == |L_c| - 1) \wedge (|L_s| \geq 100)$ **then**
        $L_l \leftarrow L_l \cup l_s$
return L_l

quality frames (those with a relative scale less than the median value or blur degree greater than the median value). Additionally, frames that are within a proximity of less than 10 frames to the subsequent absence of the target are excluded, leaving only the remaining frames as potential start points. (2) Typically, the start flag is sequentially read from the list of start points. Afterwards, the end flag is moved in reverse order through the sequence, and this process continues until the sequence interval meets the condition (i.e., more than half of the frames in the sequence are considered challenging frames). Once the end flag is identified (considering its position as the endpoint of this combination), we save this start-end combination, choose the next start point, and repeat the aforementioned procedure. The data screening rules guarantee that all eligible sub-sequences are not overlooked.

- **Data deduplication** aims to remove the unqualified sub-sequences. Based on their length, we arrange all sub-sequences in descending order and keep the longest series as the first baseline (i.e., in this process, a baseline denotes a sub-sequence that has been selected in the challenging space). Other sub-sequences will be compared with all baselines and calculate the overlapping ratio (i.e., overlapping ratio indicates the proportion of exactly the same frame in two sequences). A series that has a high overlapping ratio or is less than 100 frames will be discarded; otherwise, it will be regarded as a new baseline. We keep all baselines as the extraction result of the original sequence. Eligible sub-sequences combine into the challenging sub-space c_i of attribute a_i. All sub-spaces c_i comprise the challenging space E_c (Table 3).

To better illustrate the performance of the proposed space construction rules,we present Fig. 8 as an example. The VOT official provides manual annotations for datasets on a frame-level basis, which can serve as a human baseline for evaluating the reliability of our method. Here, we choose the *iceskater1* sequence from VOT2016 (Kristan et al., 2016) as an example and compare the *size change* annotations provided by VOT with the *delta scale* annotations generated by SOTVerse.

The proposed automated label generation procedure calculates the variation in the object's bounding box for each frame, and then determines whether there is a *size change* based on a threshold of 0.01. Utilizing this approach, we apply space construction rules to identify sub-sequences that meet the predefined criteria within the given sequence. It is discovered that the range of frames from #530 to #633 exhibit concentrated challenging factors (with at least half of the frames manifesting this challenging factor) and meet the minimum length requirement of 100 frames. In comparison to the manually labeled annotations (where annotators identified *size changes* between frames #563 and #637), the annotations generated by SOTVerse encompass 96% of the manually determined range. Furthermore, several specific cases (Fig. 8a–c) have provided evidence supporting the reliability of SOTVerse.

Figure 5 illustrates the train and test sets for challenging space. All sub-sequences in the train set are selected from train and validation sets of large-scale datasets (i.e., GOT-10k, LaSOT, and VideoCube). All sub-sequences in the test set are chosen from the test sets of large-scale datasets and whole small-scale datasets (i.e., OTB2015, VOT2016, VOT2018, VOT2019, and VOTLT2019). Please refer to "Appendix A" of the appendices for detailed information.

3.3 Evaluation

3.3.1 Evaluation System

SOTVerse provides two evaluation systems – the traditional OPE (Wu et al., 2013, 2015) system and the mechanism with re-initialization (R-OPE) (Hu et al., 2023), as shown in Fig. 9. Unlike OTB (Wu et al., 2015) or VOT competition (Kristan et

Table 3 The threshold for judging the abnormal value and the corresponding challenging space E_c

Environment			Sub-sequences (train/*test*)	**Max** frame	Mean frame
E_{c_1}	$c_1 = \{t : \gamma_t \leq 0.28 \wedge \gamma_t \geq 2.38, t \in E_n\}$	Abnormal ratio	5,779 (4,838/*941*)	29,834	898
	$c_2 = \{t : \varsigma_t \leq 0.02 \wedge \varsigma_t \geq 0.39, t \in E_n\}$	Abnormal scale	4,701 (4,029/*672*)	29,834	1,533
	$c_3 = \{t : \iota_t \leq 0.01 \wedge \iota_t \geq 0.13, t \in E_n\}$	Abnormal illumination	5,463 (4,566/*897*)	28,828	882
	$c_4 = \{t : \beta_t \leq 95, t \in E_n\}$	Blur bounding-box	4,725 (4,075/*650*)	29,834	1,571
E_{c_2}	$c_5 = \{t : \Delta\gamma_t \geq 0.2, t \in E_n\}$	Delta ratio	1,511 (1,427/*84*)	7,162	217
	$c_6 = \{t : \Delta\varsigma_t \geq 0.01, t \in E_n\}$	Delta scale	3,167 (2,871/*296*)	7,638	203
	$c_7 = \{t : \Delta\iota_t \geq 0.0012, t \in E_n\}$	Delta illumination	5,482 (4,533/*949*)	26,268	327
	$c_8 = \{t : \Delta\beta_t \geq 250, t \in E_n\}$	Delta blur bounding-box	1,007 (760/*247*)	28,800	530
	$c_9 = \{t : \varepsilon_t \geq 0.16, t \in E_n\}$	Fast motion	3,196 (2,707/*489*)	22,923	463
	$c_{10} = \{t : \rho_t \leq 0.75, t \in E_n\}$	Low correlation coefficient	4,065 (3,502/*563*)	22,923	320

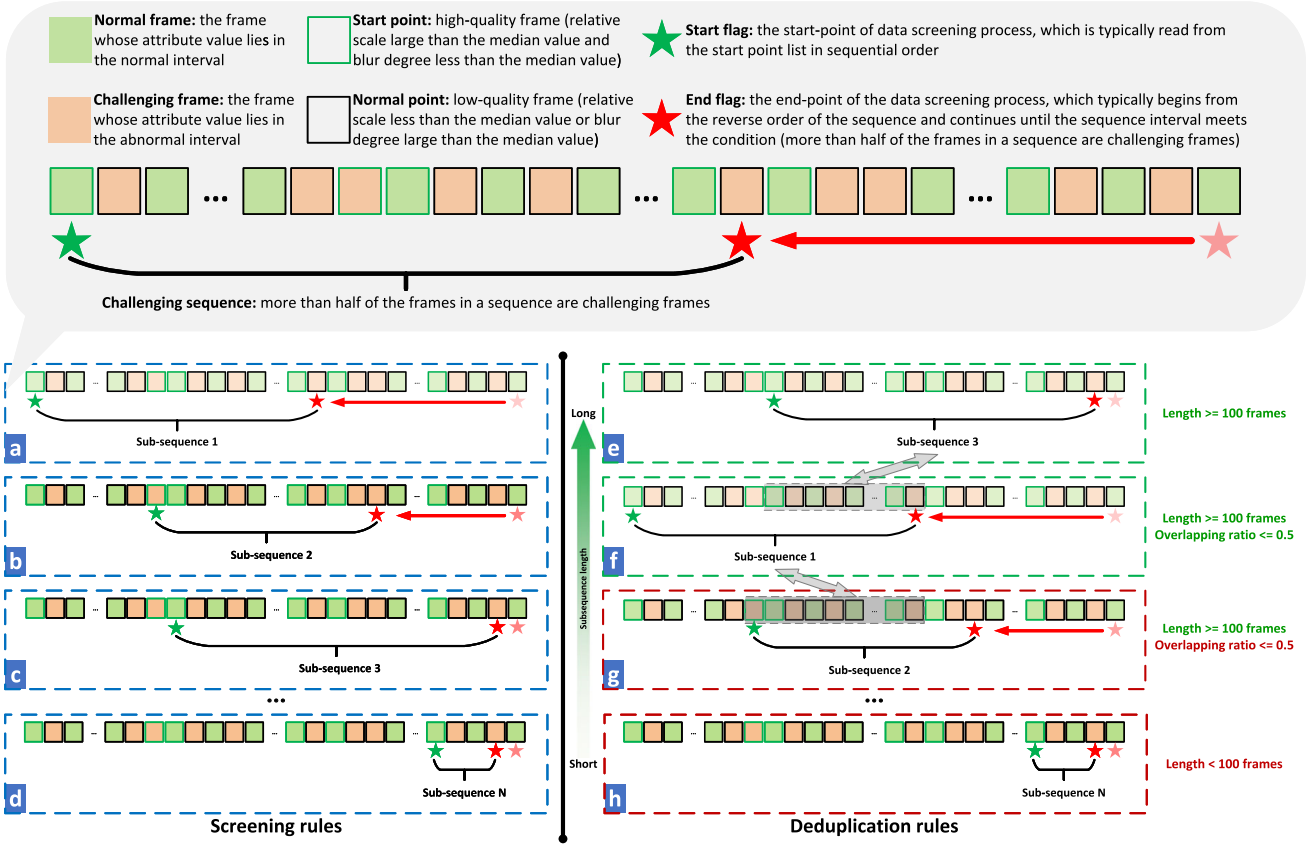


Fig. 7 Schematic diagram of space construction. **a–d** Display screening rules, aiming to find all eligible sub-sequences. **e–h** Display the deduplication rules, aiming to remove the unqualified sub-sequences

al., 2016, 2018, 2019) that only supports reset in the short-term tracking task, SOTVerse allows re-initialization in all subtasks (i.e., short-term tracking, long-term tracking, and global instance tracking) to maximize sequence utilization. Specifically, the tracking failure is decided by the overlap of p_t and g_t ($\Omega(p_t, g_t) = \frac{p_t \bigcap g_t}{p_t \bigcup g_t} < 0.5$ will be regarded as failure). Therefore, the occurrence of a re-initialization requires two conditions: (1) The algorithm fails consecutively for 10 target-present frames (frames where the target is not visible are not counted). (2) The algorithm will be re-initialized at the next start point (the start point must include a high-quality visible target).

3.3.2 Performance Indicator

Suppose the evaluation environment E is composed of $|E|$ sequences, where $|\cdot|$ is the cardinality. For the t-th frame F_t in a sequence L, suppose that p_t is the position predicted by a tracker T, and g_t is the ground-truth. Specifically, a frame without the target is regarded as an empty set (i.e., $g_t = \phi$) and excluded by the evaluation process. Traditional *precision score* and *success score* of frame F_t are calculated by:

$$\begin{aligned} d_t &= \left\| c_p - c_g \right\|_2 \\ s_t &= \Omega(p_t, g_t) = \frac{p_t \bigcap g_t}{p_t \bigcup g_t} \end{aligned} \tag{3}$$

where d_t is the distance between center points c_p and c_g, $\Omega(\cdot)$ is the intersection over union.

Recently, *normalized precision score* ((Hu et al., 2023)) is proposed to exclude the influence of target size and frame resolution. Trackers with a predicted center outside the ground-truth will add a penalty item ${d_t}^p$ (i.e., the shortest distance between center point c_p and the ground-truth edge). For trackers whose center point falls into the ground-truth, the center distance ${d_t}^{'}$ equals the original precision d_t (i.e., ${d_t}^p = 0$).

$$\begin{aligned} N(d_t) &= \frac{{d_t}^{'}}{\max(\{{d_i}^{'} \mid i \in F_t\})} \\ {d_t}^{'} &= d_t + {d_t}^p \end{aligned} \tag{4}$$

Obviously, the *precision* $\mathcal{P}(E)$, *normalized precision* $\mathcal{N}(E)$, and *success* $\mathcal{S}(E)$ of environment E can be defined

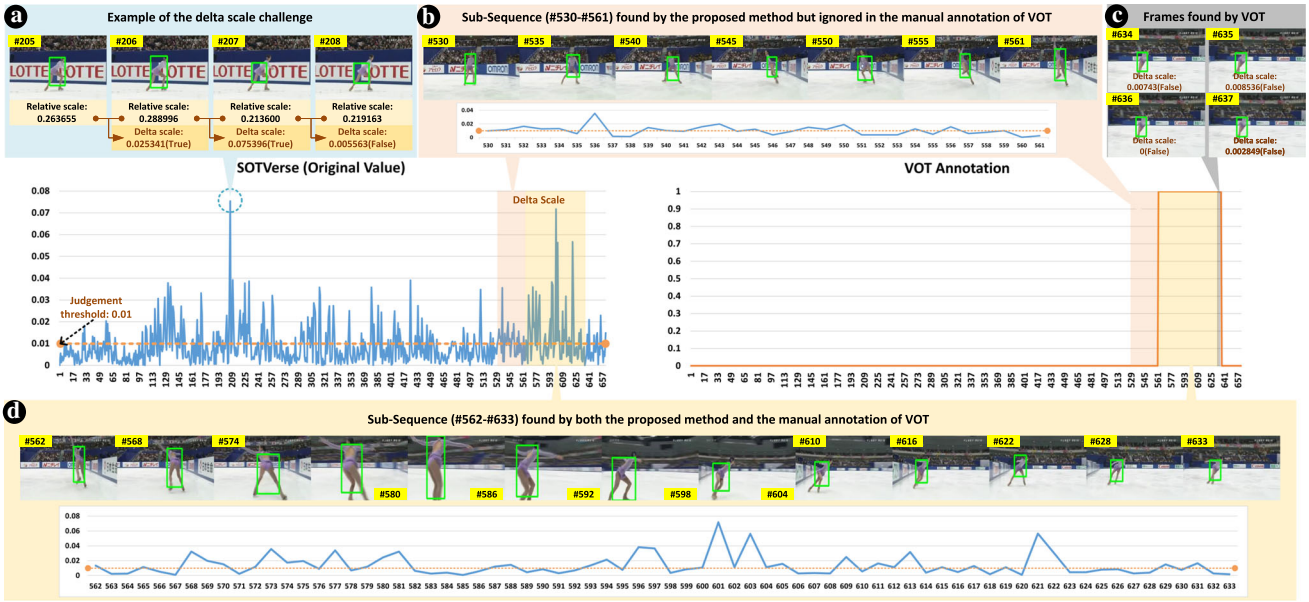


Fig. 8 Comparison of the SOTVerse challenging space construction and the manual annotation. We take the *iceskater1* sequence of VOT2016 (Kristan et al., 2016) as an example, which provides manual annotations for *size change*. These manual annotations can be compared with the *delta scale* annotations generated by SOTVerse. Compared with the manual annotations provided by VOT official, the proposed challenging space construction rules can identify challenging frames that are scattered (**a**, #207) and continuous patterns that are ignored by human annotators (**b**, #530–#561). Additionally, human annotators face challenges in accurately determining the endpoint of the challenging sub-sequence interval, which may result in the inclusion of some redundant frames (**c**, #634–#637). For #562–#633 (**d**), both human annotators and SOTVerse have effectively detected the challenge

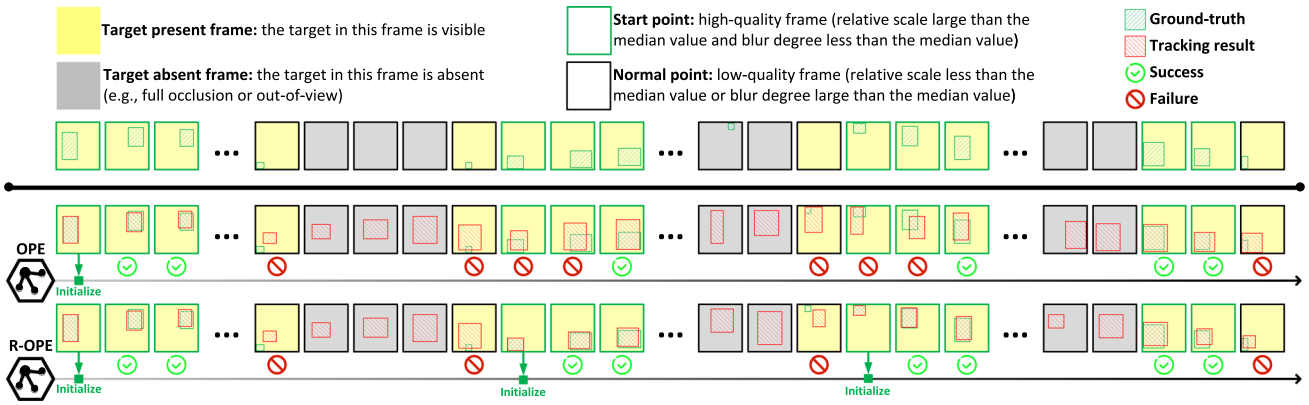


Fig. 9 The execution process of two evaluation mechanisms. The OPE (one-pass evaluation) mechanism aims to utilize the target's position in the first frame to initialize the tracker, and requires the tracker to output the predicted result in each subsequent frame. The R-OPE (one-pass evaluation with restart) mechanism will detect the tracking process, and a tracker that fails for 10 consecutive frames will trigger the re-initialization mechanism and be reset at the next start point (only frames with high-quality target information can be regarded as a start point)

as:

$$\begin{aligned}\mathcal{P}(E) &= \frac{1}{|E|}\sum_{l=1}^{|E|}\frac{1}{|L|}|\{t : d_t \le \theta_d\}| \\ \mathcal{N}(E) &= \frac{1}{|E|}\sum_{l=1}^{|E|}\frac{1}{|L|}|\left\{t : N(d_t) \le \theta_d^{'}\right\}| \\ \mathcal{S}(E) &= \frac{1}{|E|}\sum_{l=1}^{|E|}\frac{1}{|L|}|\{t : s_t \ge \theta_s\}|\end{aligned} \tag{5}$$

Calculating the proportion of frames whose distance $d_t \le \theta_d$ and drawing the statistical results based on different θ_d into a curve generates the *precision plot*. Typically, existing benchmarks always select $\theta_d = 20$ to rank trackers. Similarly, drawing statistical results based on different $\theta_d^{'} \in [0, 1]$ into a curve generates the *normalized precision plot*. However, directly selecting a $\theta_d^{'}$ to rank executors may introduce human factors. Thus, the proportion of frames whose

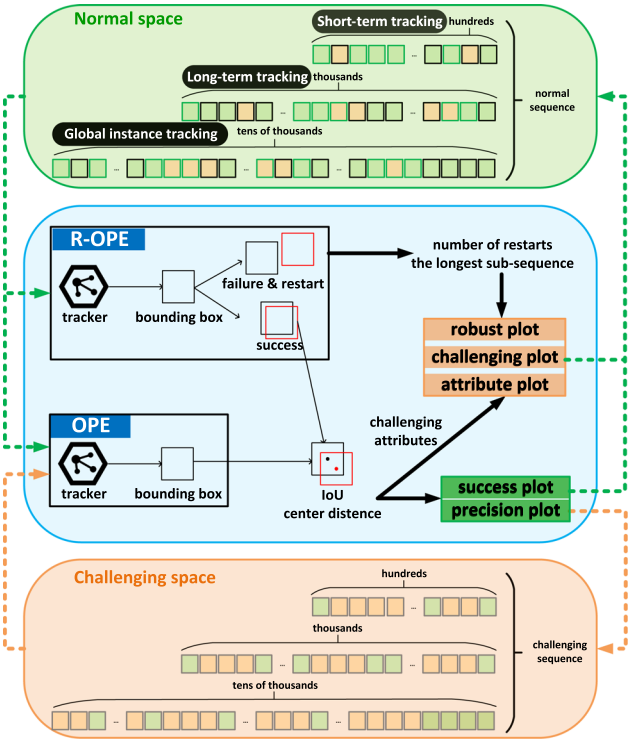


Fig. 10 Evaluation process of normal space E_n and challenging space E_c. The green dotted line and orange dotted lines respectively indicate the evaluation process of E_n and E_c

predicted center c_p successfully falls in the ground-truth rectangle g_t are selected to rank trackers. Frames with overlap $s_t \geq \theta_s$ are defined as successful tracking. Draw the results based on various overlap threshold θ_s into a curve is the *success plot*, where the mAO (mean average overlap) is wildly used to rank trackers.

Evidently, traditional precision plot, normalized precision plot, and success plot average scores on complete series to generate the final result. Challenging space that already contains enough *challenging frames* can directly use these indicators. But for normal sequences composed of most *normal frames* and a few *challenging frames*, the above metrics may ignore the influence of challenging factors due to the average calculation. Thus, SOTVerse provides three novel indicators to concentrate on the impact of challenges:

- **Challenging plot.** For the t-th frame F_t in L, suppose that ρ_t is the correlation coefficient between F_t and F_{t-1}. A frame with $s_t \geq 0.5$ is defined as *success frame*, and vice versa is *fail frame*. The *challenging score* is defined as:

$$\mathcal{C}(E) = \frac{1}{|E|} \sum_{l=1}^{|E|} \frac{|\{t : s_t \geq 0.5\}|}{|\{t : \rho_t \leq \theta_\rho\}|} \tag{6}$$

Calculating the proportion of success frames on the challenging part (i.e., $\rho_t \leq \theta_\rho$) and drawing the statistical results based on different θ_ρ into a curve generates the *challenging plot*. SOTVerse selects $\theta_\rho = 0.75$ to rank trackers.

- **Attribute plot.** *Attribute plot* $\mathcal{A}(\cdot)$ aims to find the attribute that affects tracking most. For each tracker, SOTVerse generates the $\mathcal{A}(\cdot)$ via three steps. (1) SOTVerse first finds all fail frames (i.e., $s_t < 0.5$), then checks their attribute labels, and finally calculates the proportion of a specific challenging factor on the fail frames, and generates the attribute plot based on fail frames $\mathcal{A}_f(\cdot)$. (2) SOTVerse then calculates the proportion of each attribute based on the success frames (i.e., $s_t \geq 0.5$) of each algorithm, and generates the attribute plot based on success frames $\mathcal{A}_s(\cdot)$. (3) Finally, the attribute plot $\mathcal{A}(\cdot)$ is the difference of $\mathcal{A}_f(\cdot)$ and $\mathcal{A}_s(\cdot)$ (i.e., $\mathcal{A}(\cdot) = \mathcal{A}_f(\cdot) - \mathcal{A}_s(\cdot)$). Unlike other indicators to rank algorithms, the attribute plot intuitively reveals the most likely reasons causing failures for each tracker. Please refer to "Appendix C" for an example of attribute plot.
- **Robust plot.** The *robust plot* $\mathcal{R}(\cdot)$ aims to exhibit the performance of trackers in the R-OPE mechanism. SOTVerse counts the number of restarts for each video, divides the entire video into several segments based on the restart point, and returns the longest sub-sequence that the algorithm successfully runs. Taking the number of restarts and the mean value of the longest sub-sequence as abscissa and ordinate can generate a robust plot. Trackers closer to the upper left corner have better performance (indicating successful tracking in longer sequences with rare re-initializations). Considering that the final number of restarts and the longest sub-sequence will be affected by the characteristics of a specific dataset itself, this indicator is more of the tracking robustness through qualitative analysis rather than conducting detailed numerical comparisons.

4 Experiments

4.1 Implementation Details

We select 23 represent algorithms as task executor T_t and conduct experiments based on 3E Paradigm (Table 4). The 23 trackers can be divided into 4 categories based on their model architectures. (1) Correlation filter (CF) based trackers: KCF (Henriques et al., 2014) and ECO (Danelljan et al., 2017). (2) Siamese neural network (SNN) based trackers: SiamFC (Bertinetto et al., 2016), SiamRPN (Li et al., 2018), DaSiamRPN (Zhu et al., 2018), SiamRPN++ (Li et al., 2019), SPLT (Yan et al., 2019), SiamDW (Zhang & Peng, 2019), SiamCAR (Guo et al., 2020), SiamFC++ (Xu et al., 2020),

Table 4 The implementation of experiments, organized by 3E Paradigm. More information about E_n and E_c can reference Tables 2 and 3

Task $\mathbb{S}$	Environment $\mathbb{E}$	Evaluation $\mathbb{M}$	Executor $\mathbb{T}$
		System M_s and indicator M_p	
S_{n_1}: short-term tracking	$E_{n_1} = (e_1, e_2, e_3, e_4, e_5)$	$M_s = (\text{OPE}, \text{R} - \text{OPE})$ $M_p = (\mathcal{P}(\cdot), \mathcal{N}(\cdot), \mathcal{S}(\cdot),$ $\mathcal{C}(\cdot), \mathcal{A}(\cdot), \mathcal{R}(\cdot))$	T_t = (GRM (Gao et al., 2023), Unicorn (Yan et al., 2022), OSTrack (Ye et al., 2022), MixFormer (Cui et al., 2022), KYS (Bhat et al., 2020), KeepTrack (Mayer et al., 2021), Ocean (Zhang et al., 2020), SiamRCNN (Voigtlaender et al., 2020), SuperDiMP (Danelljan et al., 2020), PrDiMP (Danelljan et al., 2020), SiamCAR (Guo et al., 2020), SiamFC++ (Xu et al., 2020), SiamDW (Zhang & Peng, 2019), GlobalTrack (Huang et al., 2020), DiMP (Bhat et al., 2019), SPLT (Yan et al., 2019), SiamRPN++ (Li et al., 2019), ATOM (Danelljan et al., 2019), DaSiamRPN (Zhu et al., 2018), SiamRPN (Li et al., 2018), ECO (Danelljan et al., 2017), SiamFC (Bertinetto et al., 2016), KCF (Henriques et al., 2014))
S_{n_2}: long-term tracking	$E_{n_2} = (e_6, e_7)$		
S_{n_3}: global instance tracking	$E_{n_3} = e_8$		
S_{c_1}: tracking under abnormal ratio	$E_c = c_1$	$M_s = \text{OPE}$ $M_p = (\mathcal{P}(\cdot), \mathcal{N}(\cdot), \mathcal{S}(\cdot))$	
S_{c_2}: tracking under abnormal scale	$E_c = c_2$		
S_{c_3}: tracking under abnormal illumination	$E_c = c_3$		
S_{c_4}: tracking under blur bounding-box	$E_c = c_4$		
S_{c_5}: tracking under delta ratio	$E_c = c_5$		
S_{c_6}: tracking under delta scale	$E_c = c_6$		
S_{c_7}: tracking under delta illumination	$E_c = c_7$		
S_{c_8}: tracking under delta bounding-box	$E_c = c_8$		
S_{c_9}: tracking under fast motion	$E_c = c_9$		
$S_{c_{10}}$: tracking under low correlation coefficient	$E_c = c_{10}$		

Details about M_s and M_p are listed in Sect. 3.3

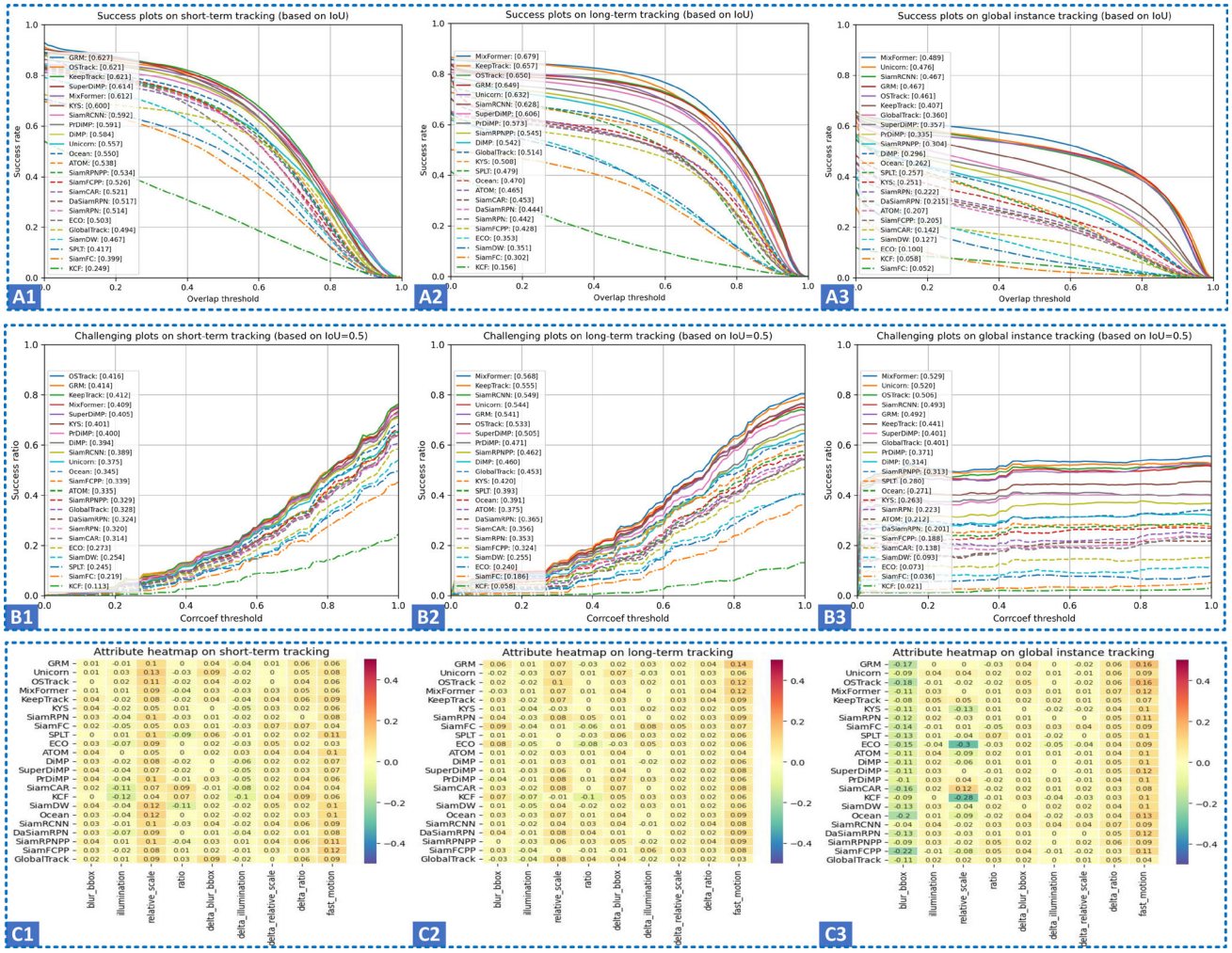


Fig. 11 Experiments in normal space with OPE mechanisms. Three columns represent the results in the short-term tracking task (left), long-term tracking task (middle), and global instance tracking task (right). Each task is evaluated by success plots (**A1–A3**), challenging plots (**B1–B3**) and attribute plots (**C1–C3**)

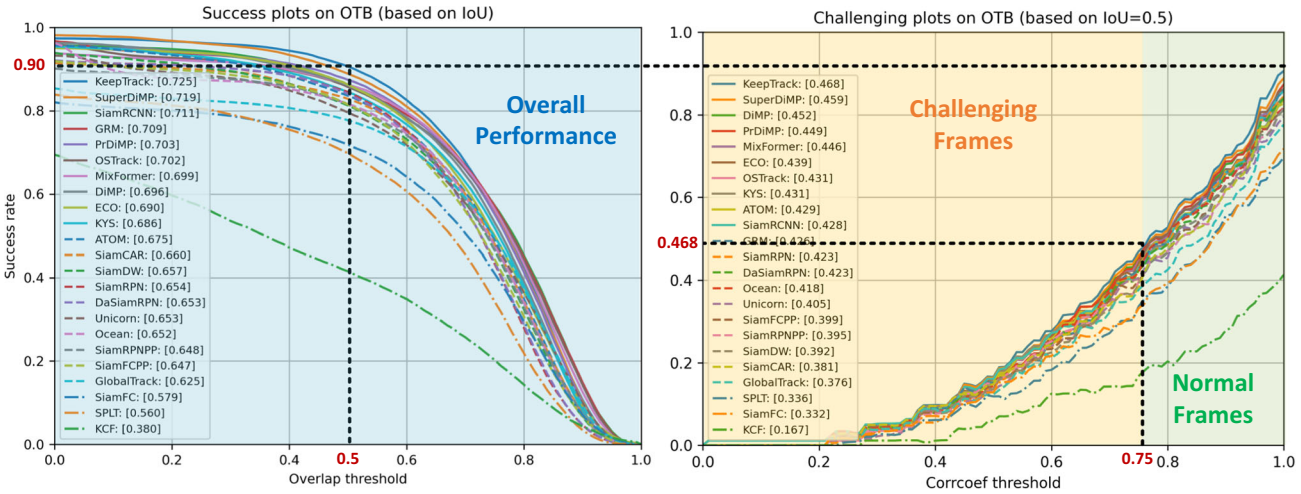


Fig. 12 Comparison of performance under success plot (LEFT) and challenging plot (RIGHT), taking OTB (Wu et al., 2015) as an example

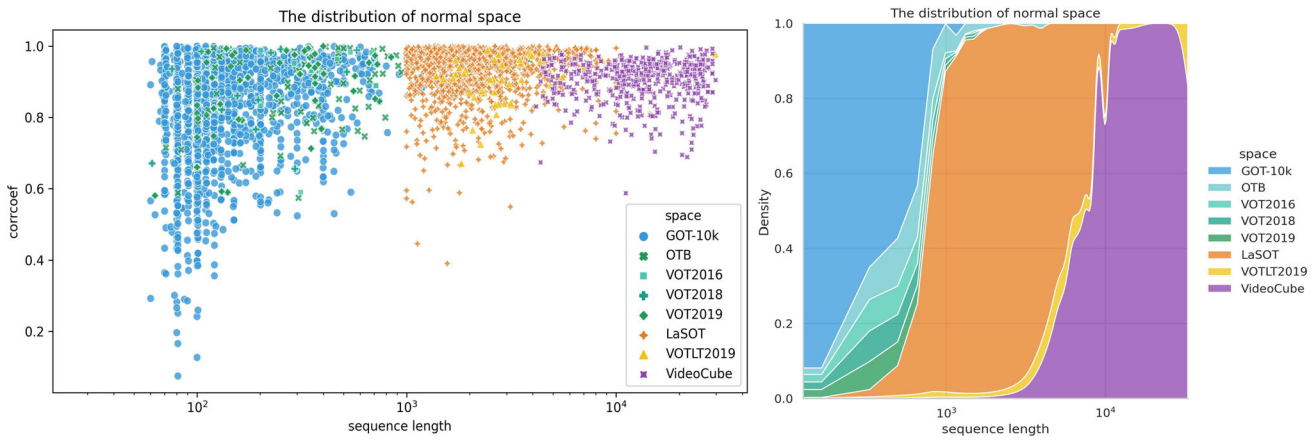


Fig. 13 The composition of normal space. (Left) The distribution of corrcoef values and sequence lengths, each point representing a sub-sequence. (Right) The distribution of sequence lengths

Ocean (Zhang et al., 2020), and SiamRCNN (Voigtlaender et al., 2020). (3) Trackers that combine CF and SNN: ATOM (Danelljan et al., 2019), DiMP (Bhat et al., 2019), PrDiMP (Danelljan et al., 2020), SuperDiMP (Danelljan et al., 2020), and KeepTrack (Mayer et al., 2021). (4) Transformer-based trackers, like MixFormer (Cui et al., 2022), OSTrack (Ye et al., 2022), and GRM (Gao et al., 2023). (5) Trackers with custom networks, like GlobalTrack (Huang et al., 2020) with zero cumulative error, KYS (Bhat et al., 2020) with scene information, and Unicorn (Yan et al., 2022) that accomplishes the unification of learning paradigm for different tracking tasks.

Specifically, by comparing with precision plots and normalized precision plots, the success plots utilize both information about the position and object size for evaluation. Thus, we only retain the success plots in this section to show the results more efficiently. Please refer to "Appendix D" of the appendices for detailed information and comprehensive experimental results for the 23 representative trackers.

4.2 Experiments in Normal Space

Figures 11, 12, 13 and 14 illustrate the experimental results in normal space. We first conduct experiments on the meta-datasets e_i for each subtask S_{n_i}, then average results as the final performance for the current subtask. We add a figure number at the bottom left for each subplot to better illustrate the experimental results.

4.2.1 Experiments in OPE Mechanism

Figure 11 shows the performance of trackers under the OPE mechanism.

Influence of Task Constraints on Tracking Performance

From the task perspective, the widely used success plots (Fig. 11A1–A3) show a downward trend in trackers' performance from short-term tracking to global instance tracking. This phenomenon indicates that with the relaxation of task constraints, more challenging factors are occurred and require higher tracking ability. Especially compared with the first two tasks, performance drops the most on global instance tracking, which indicates that as the SOT task that is closest to the actual application scenario, global instance tracking is still a considerable difficulty to existing methods.

Limitations of Existing Evaluation Metrics.

Before conducting analyses based on new metrics, we first illustrate the limitations of existing metrics through an experiment in Fig. 12. Existing benchmarks only evaluate complete sequences but ignore the challenging frames. Taking the OTB (Wu et al., 2015) as an example, the traditional success plot (left) for KeepTrack (Mayer et al., 2021) indicates it successfully tracks 90% frames, while the challenging plot (right) proposed in this paper shows that the success rate of challenging frames is only 46.8%. Obviously, the existing evaluation system ignores the influence of challenging factors.

Tracking Evaluation via Challenging Plots.

The challenging plots (Fig. 11B1–B3) demonstrate that the algorithm's success rate on challenging frames is basically lower than 50%.

Observe three points in challenging plots: the inflection point, the point with $\theta_\rho = 0.75$, and the endpoint on the right ($\theta_\rho = 1$). Attention that the variation trends of challenging plots are not totally similar, meaning the decisive factors for influencing algorithm performance of various tasks are different:

- **Challenging factors mainly influence short-term tracking task.** In short-term tracking (Fig. 11B1), the success rate of the majority of algorithms shows improvement with higher correlation coefficients, suggesting that challenging factors play a significant role in determining the success rate.

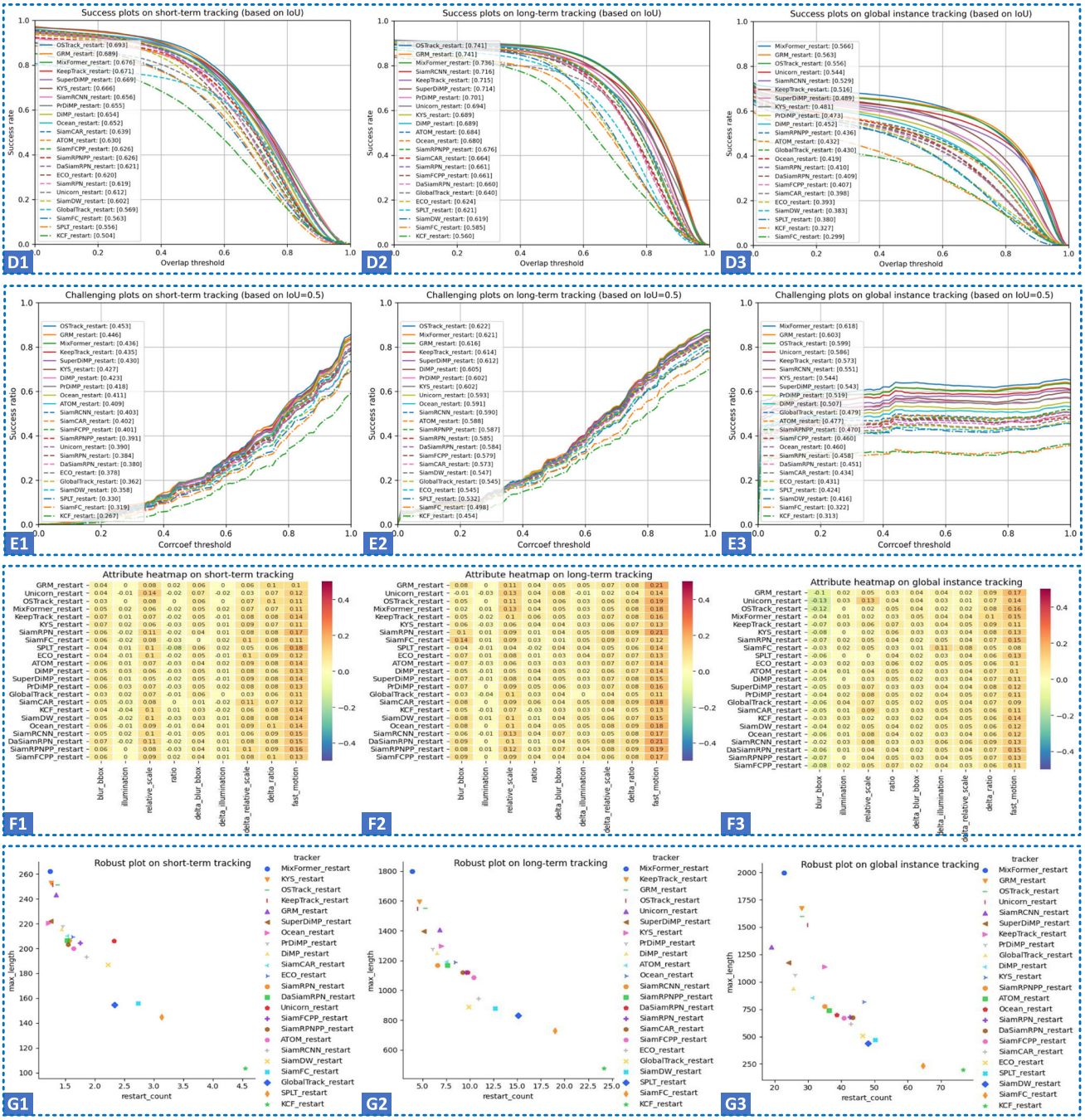


Fig. 14 Experiments in normal space with R-OPE mechanisms. Three columns represent the results in the short-term tracking task (left), long-term tracking task (middle), and global instance tracking task (right). Each task is evaluated by success plots (**D1–D3**), challenging plots (**E1–E3**), attribute plots (**F1–F3**), and robust plots (**G1–G3**)

- **Challenging factors and sequence length mainly influence long-term tracking task.** In long-term tracking (Fig. 11B2), the performance improves with the addition of the correlation coefficient in challenging frame areas. However, this improvement slows down when the sequence contains more normal frames, indicating that sequence length becomes the primary factor influencing the success rate.

- **Shot-switching mainly influences global instance tracking task.** In the global instance tracking task (Fig. 11B3) that involves shot switching, the curve reaches an inflection point at very low correlation coefficients. Beyond this inflection point, the curve exhibits a gentle slope. As a result, the crucial aspect influencing algorithm performance in the GIT task is not challenging factors, but

rather the ability to re-locate the target position after each shot switching.

Based on the definition of attribute plot, it intuitively reveals the most likely reasons that cause failures for each tracker in the specific environment. In other words, the deeper color of the attribute plots shows that this challenging factor is more prominent in the experimental environment, and the algorithm performs poorly in facing this challenge.

For attribute plots (Fig. 11C1–C3), in the static attributes, the object scale and motion blur has a greater impact on tracking performance; in the dynamic attributes, the fast motion has a more significant effect on success rates. Here are some possible reasons: (1) Target with abnormal scale will challenge trackers, since the evaluation indicators are mainly based on IoU, while trackers are usually hard to accurately fit the tiny target bounding-box. (2) Blur is usually caused by fast motion, which may vary the target appearance information and affect the tracking process. (3) Fast motion not only may cause target blur, but also may lead to tracking failure since the target location between the continuous frames may be huge changed.

4.2.2 Experiments in R-OPE Mechanism

Figure 14 shows the performance of trackers under the R-OPE mechanism. The R-OPE mechanism allows re-initialization after failure and avoids the wastage of subsequent sequences. Thus, the success score in Fig. 14D1–D3 are higher than OPE mechanism (Fig. 11A1–A3).

The robust plots (Fig. 14G1–G3) visually display the relationship between the longest successfully tracked subsequence and the restart times. By comparing trackers' performance on different tracking tasks via robust plots, we can summarize that:

- **Most SOTA algorithms can complete tracking with rare re-initialization in the short-term tracking.** Furthermore, for GOT-10k (Huang et al., 2021) with a shorter average length, the SOTA trackers can track the entire video without re-initializations (Fig. 30f). The rare restart times and slight score differences under the OPE and R-OPE mechanisms illustrate that existing SOTA trackers can keep robust tracking ability in most short-term sequences.
- **Trackers are still easy to fail in long-term tracking and global instance tracking task.** G2 in Fig. 14 demonstrates that most excellent algorithms can continuously track 1000 to 1800 frames and then fail due to the influence of challenging attributes. The longest successfully tracked sub-sequence length is still far from the length interval (1k to 10k frames) of the long-term tracking task represented in Fig. 13. G3 in Fig. 14 shows that even if the algorithm restarts dozens of times, it is still difficult to re-locate the object quickly when the shot switching occurs again in the global instance tracking task. Thus, the robustness of existing trackers under longer sequences can be further improved.

Please refer to "Appendices E, F, and G" of the appendices for the initial experimental results under each benchmark.

4.3 Experiments in Challenging Space

Ten challenge attribute spaces are utilized for conducting experiments. The evaluation of performance under the OPE mechanism is based on three chosen indicators: precision plot, normalized precision plot, and success plot.

To better compare the performance changes under different sub-spaces, we plot the algorithm scores in Fig. 15. Since the sequence length is an essential factor affecting the results, while the sequence lengths in challenging spaces are quite diverse, we recalculate the original results (H1, I1, J1) by using sequence length as the weight and generate the weighted result (H2, I2, J2). Here, we select the four trackers (GRM (Gao et al., 2023), Unicorn (Yan et al., 2022), KeepTrack (Mayer et al., 2021), and SiamRCNN (Voigtlaender et al., 2020)) as representative models to exhibit the results, since they are excellent trackers with different model structures. For detailed results of the 23 trackers, please refer to Table 7 to 9 (based on original indicators) and Table 10 to 12 for weighted results.

Influence of Task Constraints.

Obviously, all trackers have the worst performance on VideoCube (Hu et al., 2023), indicating that the GIT task is challenging for even SOTA algorithms. Besides, compared to the challenging factors that have widely existed in each subtask, new challenges included by broadening the task constraints (i.e., relocating the target in shot-switching) have a more significant influence on tracking performance. It also indicates that the correct modeling task is vital in domain development.

Influence of Challenging Factors.

Most algorithms perform poorly in challenging spaces (orange area in Fig. 15). However, some trackers score relatively high on initial results (H1, I1, J1) – one possible reason is multiple challenging spaces contain many short sub-sequences extracted from GOT-10k (Huang et al., 2021). The weighted results decrease (H2, I2, J3) indicates that most challenging factors (e.g., fast motion, abnormal ratio, and abnormal scale) significantly impact the tracking performance; existing algorithms are still required to enhance the tracking robustness under challenging situations.

The four represent trackers are based on different model architectures. GRM (Gao et al., 2023) is a transformer-based tracker with a generalized relation modeling method.

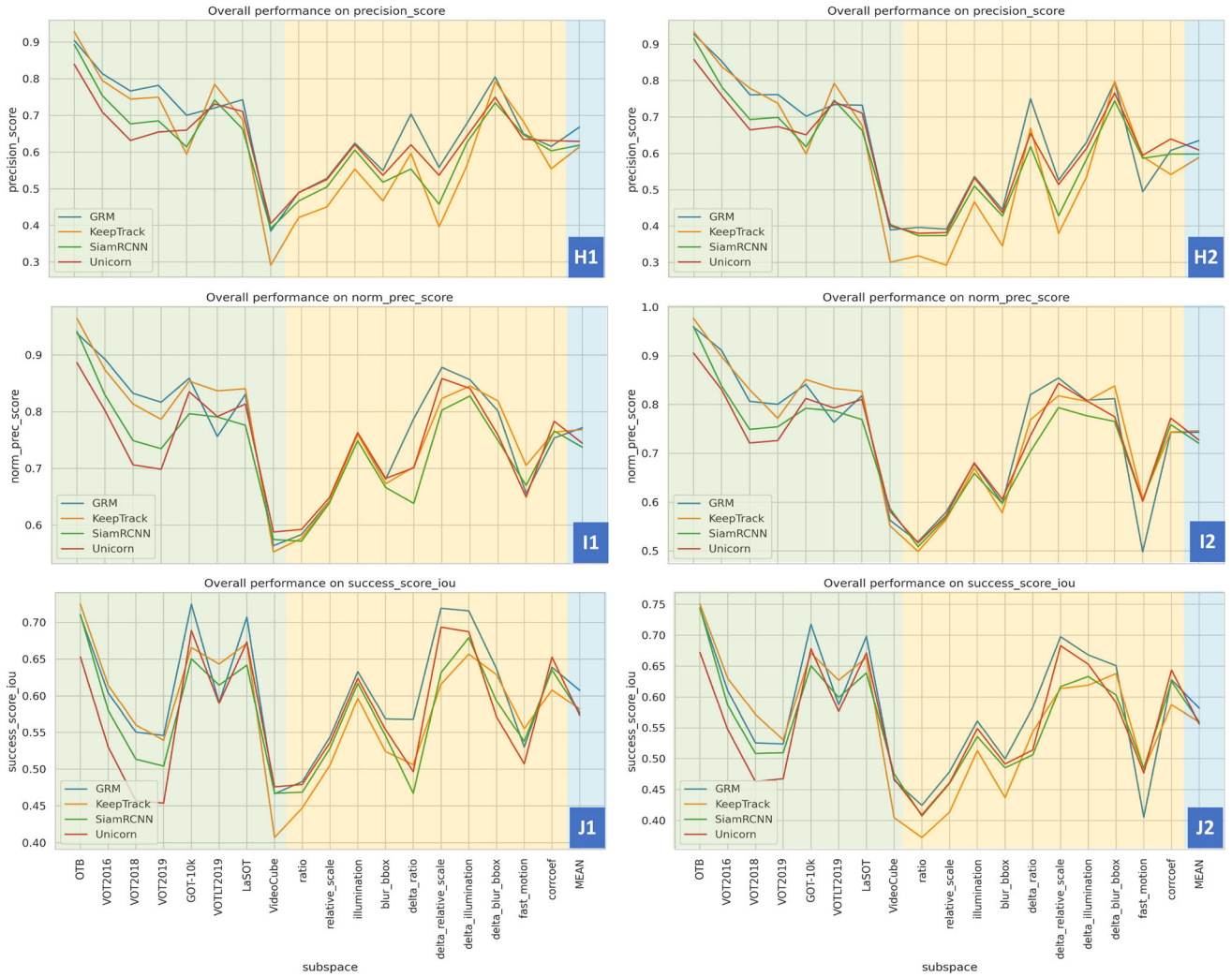


Fig. 15 Experiments in all sub-spaces with OPE mechanism, represented by GRM (Gao et al., 2023), Unicorn (Yan et al., 2022), KeepTrack (Mayer et al., 2021), and SiamRCNN (Voigtlaender et al., 2020). The Green and orange backgrounds represent the performance in normal space and challenge space respectively. The blue background represents the average performance over all sub-spaces. Results are calculated by precision plots (**H1–H2**), normalized precision plots (**I1–I2**), and success plots (**J1–J2**). The left column (**H1, I1, J1**) is generated by original indicators, while the right column (**H2, I2, J2**) is weighted by sequences' length

It performs well in most sub-spaces than other trackers, indicating that the flexible relation modeling method (i.e., by selecting appropriate search tokens to interact with template tokens) can provide powerful tracking ability. The SiamRCNN (Voigtlaender et al., 2020) selects the siamese network as the basic model structure and combines a two-stage scheme with a new trajectory-based dynamic planning algorithm. Besides, the re-detection mechanism in SiamR-CNN helps it to accomplish stable tracking in the abruption of appearance or motion information. KeepTrack (Mayer et al., 2021) model is improved by the DiMP series ((Danelljan et al., 2019; Bhat et al., 2019; Danelljan et al., 2020)), and has an enhanced ability to discriminate interferers in challenging situations. Specially, although the Unicorn (Yan et al., 2022) method is designed for four different tasks, it achieves great performance on SOTVerse, indicating that it has learned more essential information to effectively execute visual tasks. The combination of different sub-tasks in SOTVerse can help us comprehensively analyze various executors, rather than only comparing the performance of the algorithms on a single dataset with shallow conclusions.

To better illustrate the impact of challenging space on tracking performance, we present an example via Fig. 16. The top 3 trackers in the Drone-13 sequence of LaSOT (Fan et al., 2021) achieve nearly 0.5 IoU score in this sequence, while performing poorly on two sub-sequences that belong to fast-motion challenge, indicating that the space construction rule can effectively dig out high-challenging data in the complete environment.

"Appendix H" of the appendices shows the detailed distribution of challenge sub-spaces, and "Appendix I" shows the

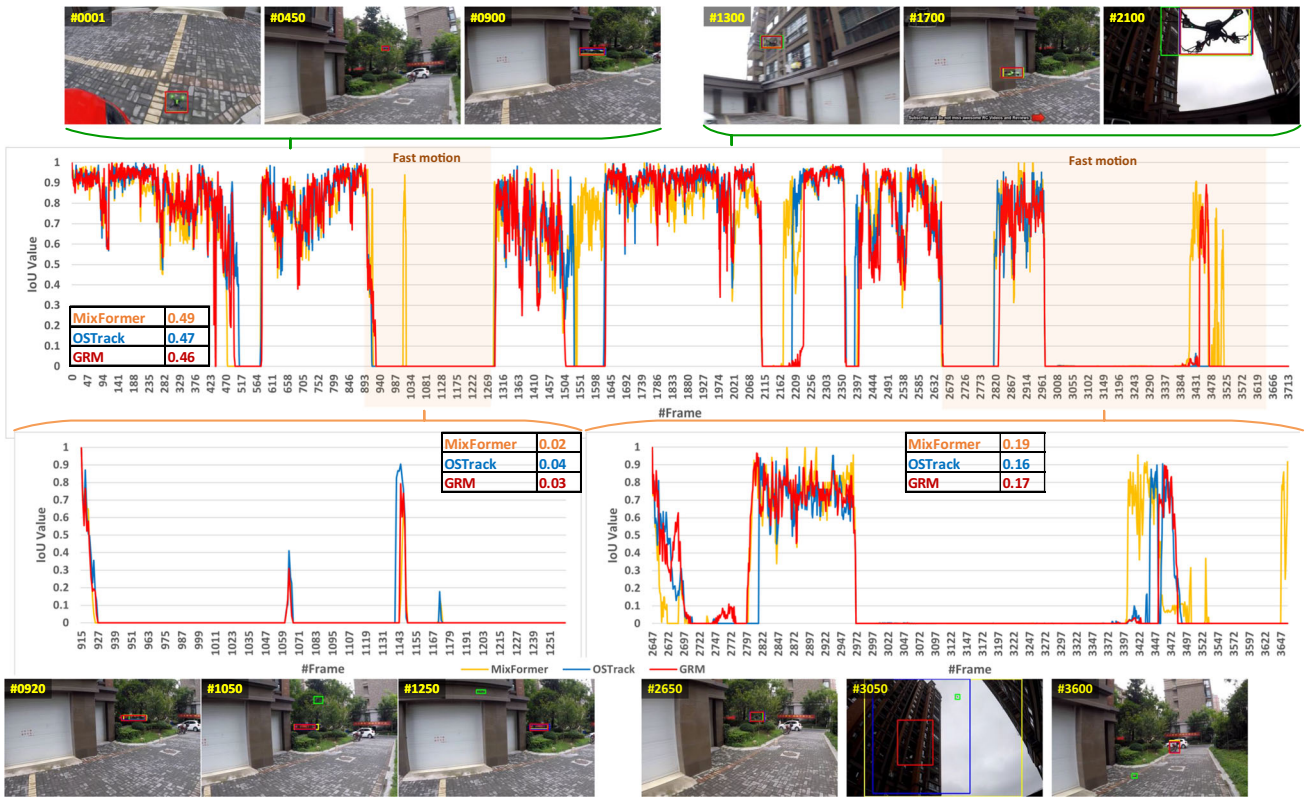


Fig. 16 The impact of challenging space on tracking performance. We take the Drone-13 sequence of LaSOT (Fan et al., 2021) as an example, in which the top 3 algorithms are selected to illustrate the IoU scores with ground-truth (■ green bounding-box represents ground-truth, ■ yellow bounding-box represents MixFormer (Cui et al., 2022), ■ blue bounding-box represents OSTrack (Ye et al., 2022), ■ red bounding-box represents GRM (Gao et al., 2023)). Based on the challenging space construction rule, two sub-sequences in Drone-13 belong to fast-motion. Compared with the complete sequence in normal space, the performance of the three algorithms in challenging space has decreased significantly (Color figure online)

specific experimental results. Please refer to "Appendices H and I" of the appendices for the initial experimental results.

5 Conclusion

This paper first proposes a 3E Paradigm to describe computer vision tasks by three components (i.e., environment, evaluation, and executor), then construct representative benchmarks as SOTVerse with 12.56 million frames. SOTVerse contains a comprehensive and user-defined environment and a thoroughgoing evaluation scheme, allowing users to customize tasks according to research purposes. We also conduct extensive experiments in the SOTVerse and conduct a performance analysis on various executors.

In future work, we will continue to maintain the platform to support users to continuously enrich the content of the SOTVerse through *interaction* and *expansion* functions, and create new sub-spaces according to their respective research purposes. Additionally, we encourage users to expand into new visual or other domain tasks and create new metaverse spaces following our defined paradigm. We welcome researchers to join our platform with their newly created metaverse spaces accepted by the research community, and together promote the promotion of the metaverse paradigm to form broader research outcomes.

Data Availability All data will be made available on reasonable request.

Code Availability The toolkit and experimental results will be made publicly available.

Declarations

Conflict of interest All authors declare no conflicts of interest.

Appendix A: Subsets of Challenging Space

Due to some repeated sequences in VOT2016 (Kristan et al., 2016), VOT2018 (Kristan et al., 2018), and VOT2019 (Kristan et al., 2019), we have removed duplicates from the three datasets when constructing the challenge space, ensuring that the constructed subspace does not contain any repeated sequences (Fig. 17). Specifically, we carefully examined

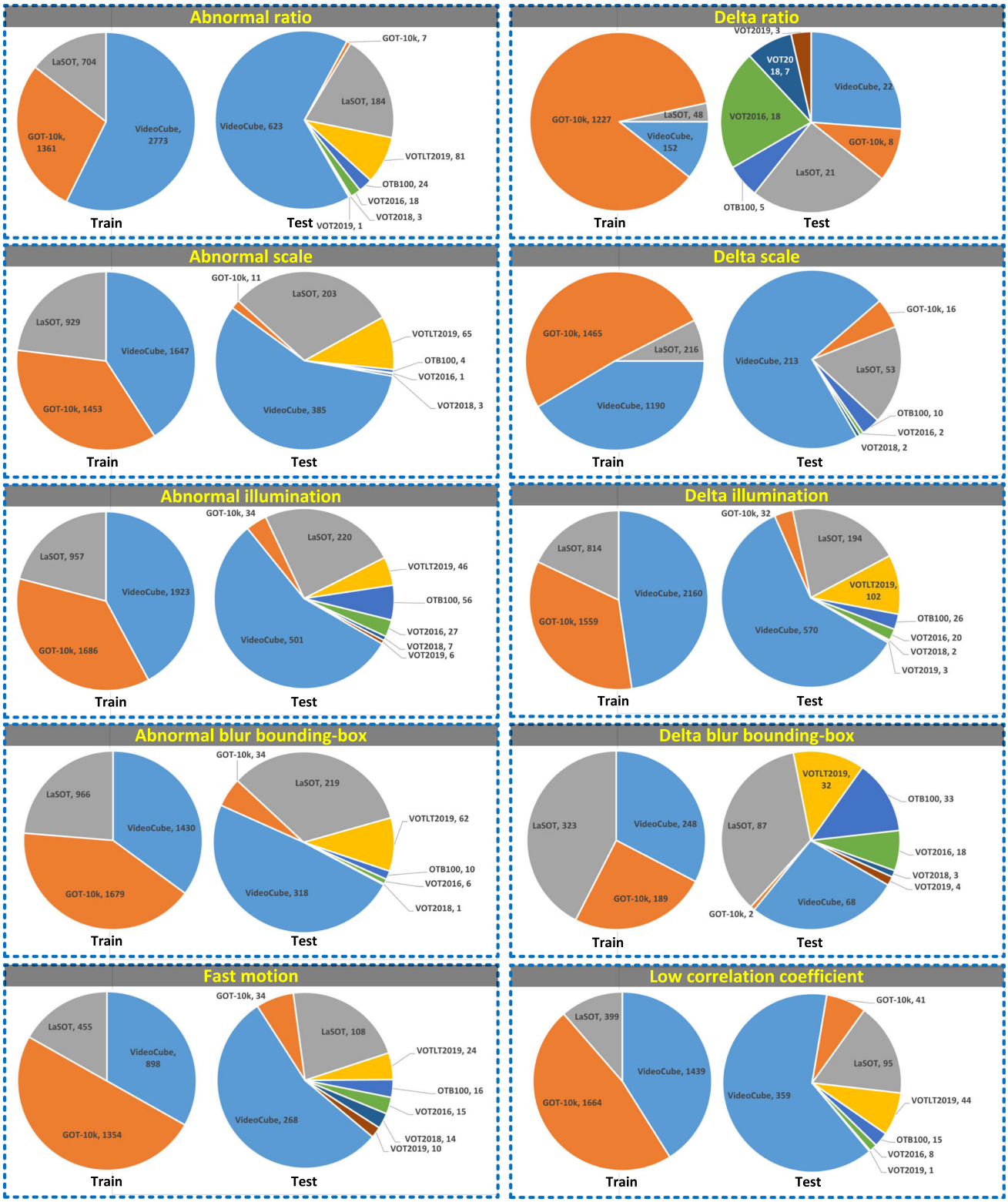


Fig. 17 Distribution of subsets in challenging space

Table 5 The non-repetitive sequences from the three VOT datasets (VOT2016 (Kristan et al., 2016), VOT2018 (Kristan et al., 2018), and VOT2019 (Kristan et al., 2019))

Sequence	Data source	Sequence	Data source	Sequence	Data source	Sequence	Data source
agility	VOT2019	dinosaur	VOT2016	hand2	VOT2019	racing	VOT2016
ants1	VOT2018	dribble	VOT2019	handball1	VOT2016	road	VOT2016
ants3	VOT2018	drone_across	VOT2018	handball2	VOT2016	rowing	VOT2019
bag	VOT2016	drone_flip	VOT2018	helicopter	VOT2016	shaking	VOT2016
ball1	VOT2016	drone1	VOT2018	iceskater1	VOT2016	sheep	VOT2016
ball2	VOT2016	fernando	VOT2016	iceskater2	VOT2016	singer1	VOT2016
ball3	VOT2019	fish1	VOT2016	lamb	VOT2019	singer2	VOT2016
basketball	VOT2016	fish2	VOT2016	leaves	VOT2016	singer3	VOT2016
birds1	VOT2016	fish3	VOT2016	marathon	VOT2019	soccer1	VOT2016
birds2	VOT2016	fish4	VOT2016	marching	VOT2016	soccer2	VOT2016
blanket	VOT2016	flamingo1	VOT2018	matrix	VOT2016	soldier	VOT2016
bmx	VOT2016	frisbee	VOT2018	monkey	VOT2019	sphere	VOT2016
bolt1	VOT2016	girl	VOT2016	motocross1	VOT2016	surfing	VOT2019
bolt2	VOT2016	glove	VOT2016	motocross2	VOT2016	tiger	VOT2016
book	VOT2016	godfather	VOT2016	nature	VOT2016	traffic	VOT2016
butterfly	VOT2016	graduate	VOT2016	octopus	VOT2016	tunnel	VOT2016
car1	VOT2016	gymnastics1	VOT2016	pedestrian1	VOT2016	wheel	VOT2019
car2	VOT2016	gymnastics2	VOT2016	pedestrian2	VOT2016	wiper	VOT2016
conduction1	VOT2018	gymnastics3	VOT2016	polo	VOT2019	zebrafish1	VOT2018
crabs1	VOT2018	gymnastics4	VOT2016	rabbit	VOT2016		
crossing	VOT2016	hand	VOT2016	rabbit2	VOT2019		

the sequences of VOT2016, VOT2018, and VOT2019 and retained only the non-duplicated ones. After the selection process, a total of 82 sequences remained out of the original 180 sequences, as illustrated in the Table 5. Out of these, 60 sequences belong to VOT2016, 10 sequences belong to VOT2018, and 12 sequences belong to VOT2019.

Appendix B: Relationship of Dynamic Attributes

The last row in Fig. 18 indicates that variations of the other five dynamic attributes will change the corrcoef. In addition, compared with the other five dynamic attributes, corrcoef can better comprehensively reflect the dynamic variations in the video sequence. Thus, it can be used as an indicator of variation degree in the tracking process.

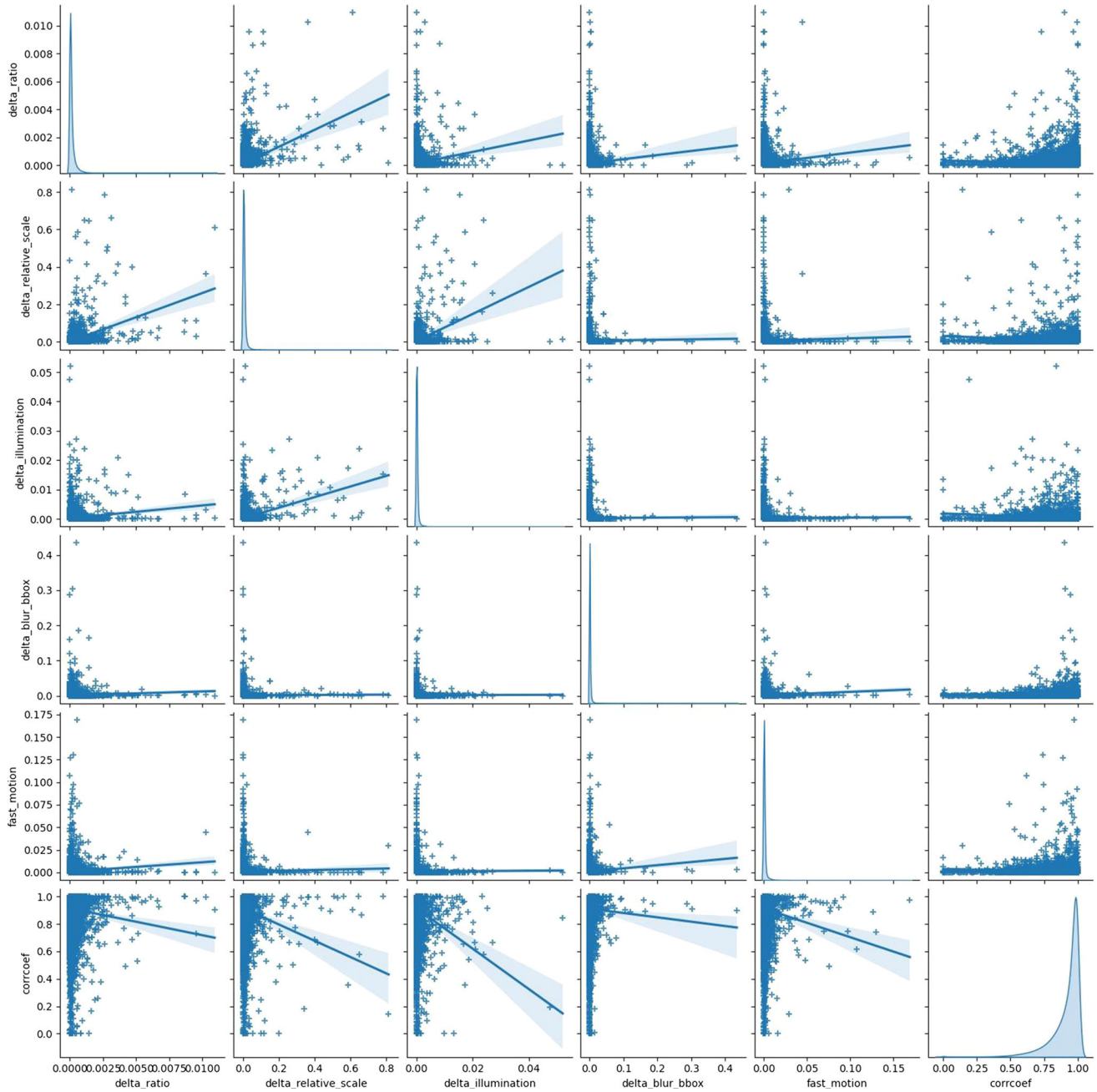


Fig. 18 Relationship of dynamic attributes in SOTVerse

Appendix C: An Example of Attribute Plot

As shown in Fig. 19, sub-figure (a) is from *car-6* in LaSOT (Fan et al., 2021) dataset. Here, we select six algorithms (GRM (Gao et al., 2023), Unicorn (Yan et al., 2022), and OSTrack (Ye et al., 2022) representing the latest state-of-the-art algorithms; while ECO (Danelljan et al., 2017), SiamFC (Bertinetto et al., 2016), and KCF (Henriques et al., 2014) representing classical algorithms) as representatives to generate the attribute plot. Among them, sub-figure (b) employs the previous method, which calculates the proportions based on the fail frames ($\mathcal{A}_f(\cdot)$). In contrast, sub-figure (c) calculates the proportion of each attribute based on the success frames ($\mathcal{A}_s(\cdot)$). Sub-figure (d) represents the difference between sub-figures (b) and (c) and serves as the updated attribute plot $\mathcal{A}(\cdot)$ (i.e., $\mathcal{A}(\cdot) = \mathcal{A}_f(\cdot) - \mathcal{A}_s(\cdot)$).

Clearly, the utilization of the previous calculation method (b) highlights a substantial correlation between the fail frames and the challenging factor of *blur bounding-box*. However, upon closer examination of sub-figure (c), it is

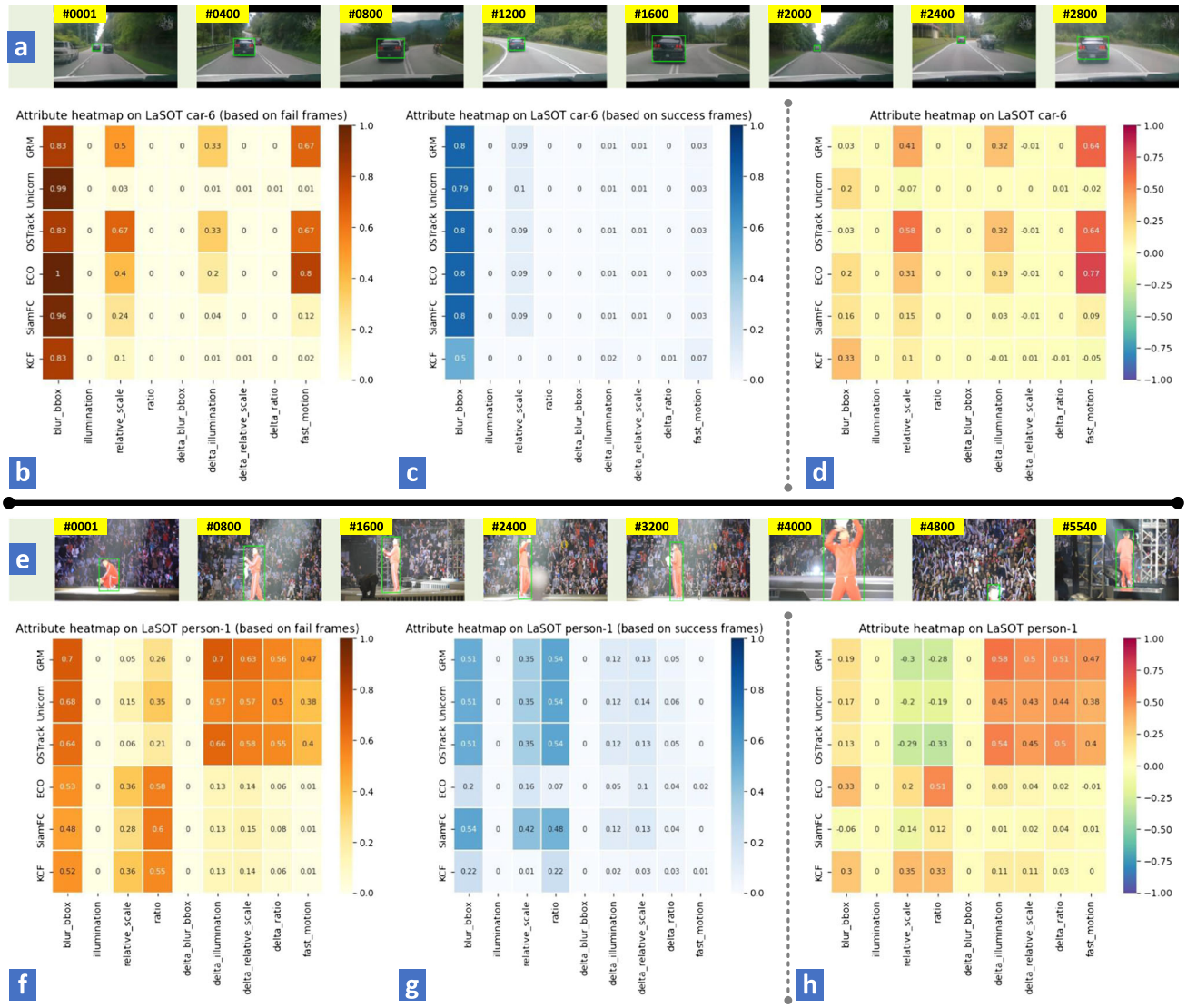


Fig. 19 An example of the calculation process of the attribute plot

evident that *blur bounding-box* remains consistently prevalent across all successful frames. A comparison between (b) and (c) demonstrates that *blur bounding-box* is a widely observed challenging factor in the majority of frames within this sequence. However, it does not serve as the primary cause of algorithm failure. Sub-figure (d) offers a more precise depiction of the challenging factors that contribute to algorithm failure. For instance, in the case of GRM, its failure primarily results from the *fast motion* of the target.

Sub-figures (e–h) are from the *person-1* sequence in LaSOT dataset, analyzed using the same process as (a–d). For the GRM, Unicorn, and OSTrack methods, the most challenging factors in this sequence are a series of dynamic attributes (top right corner of sub-figure (h)), including *variations in illumination, scale, ratio, and fast movement*. Moreover, within sub-figure (h), negative value regions indicate that the algorithms excel in these attributes, making them more likely to achieve successful target tracking in most cases. For instance, the GRM, Unicorn, and OSTrack methods exhibit strong tracking capabilities on the static attributes of *abnormal scale and ratio* within this sequence.

Appendix D: Comprehensive Experimental Results

All experiments are performed on a server with 4 NVIDIA TITAN RTX GPUs and a 64 Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz. We use the parameters provided by the original authors.

Table 6 The model architectures and URLs of open-sourced algorithms used in this work

Tracker	Architecture	URL
KCF (Henriques et al., 2014)	CF	https://github.com/uoip/KCFpy
ECO (Danelljan et al., 2017)	CF+CNN	https://github.com/visionml/pytracking
SiamFC (Bertinetto et al., 2016)	SNN	https://github.com/huanglianghua/siamfc-pytorch
SiamRPN (Li et al., 2018)	SNN	https://github.com/huanglianghua/siamrpn-pytorch
DaSiamRPN (Zhu et al., 2018)	SNN	https://github.com/foolwood/DaSiamRPN
SiamRPN++ (Li et al., 2019)	SNN	https://github.com/PengBoXiangShang/SiamRPN_plus_plus_PyTorch
SPLT (Yan et al., 2019)	SNN	https://github.com/iiau-tracker/SPLT
SiamDW (Zhang & Peng, 2019)	SNN	https://github.com/researchmm/TracKit
SiamCAR (Guo et al., 2020)	SNN	https://github.com/ohhhyeahhh/SiamCAR
SiamFC++ (Xu et al., 2020)	SNN	https://github.com/MegviiDetection/video_analyst
Ocean (Zhang et al., 2020)	SNN	https://github.com/researchmm/TracKit
SiamRCNN (Voigtlaender et al., 2020)	SNN	https://github.com/VisualComputingInstitute/SiamR-CNN
ATOM (Danelljan et al., 2019)	SNN+CF	https://github.com/visionml/pytracking
DiMP (Bhat et al., 2019)	SNN+CF	https://github.com/visionml/pytracking
PrDiMP (Danelljan et al., 2020)	SNN+CF	https://github.com/visionml/pytracking
SuperDiMP (Danelljan et al., 2020)	SNN+CF	https://github.com/visionml/pytracking
KeepTrack (Mayer et al., 2021)	SNN+CF	https://github.com/visionml/pytracking
MixFormer (Cui et al., 2022)	Transformer	https://github.com/MCG-NJU/MixFormer
OSTrack (Ye et al., 2022)	Transformer	https://github.com/botaoye/OSTrack
GRM (Gao et al., 2023)	Transformer	https://github.com/Little-Podi/GRM
KYS (Bhat et al., 2020)	Custom networks	https://github.com/visionml/pytracking
GlobalTrack (Huang et al., 2020)	Custom networks	https://github.com/huanglianghua/GlobalTrack
Unicorn (Yan et al., 2022)	Custom networks	https://github.com/MasterBin-IIAU/Unicorn

CF correlation filter, *CNN* convolutional neural network, *SNN* Siamese neural network

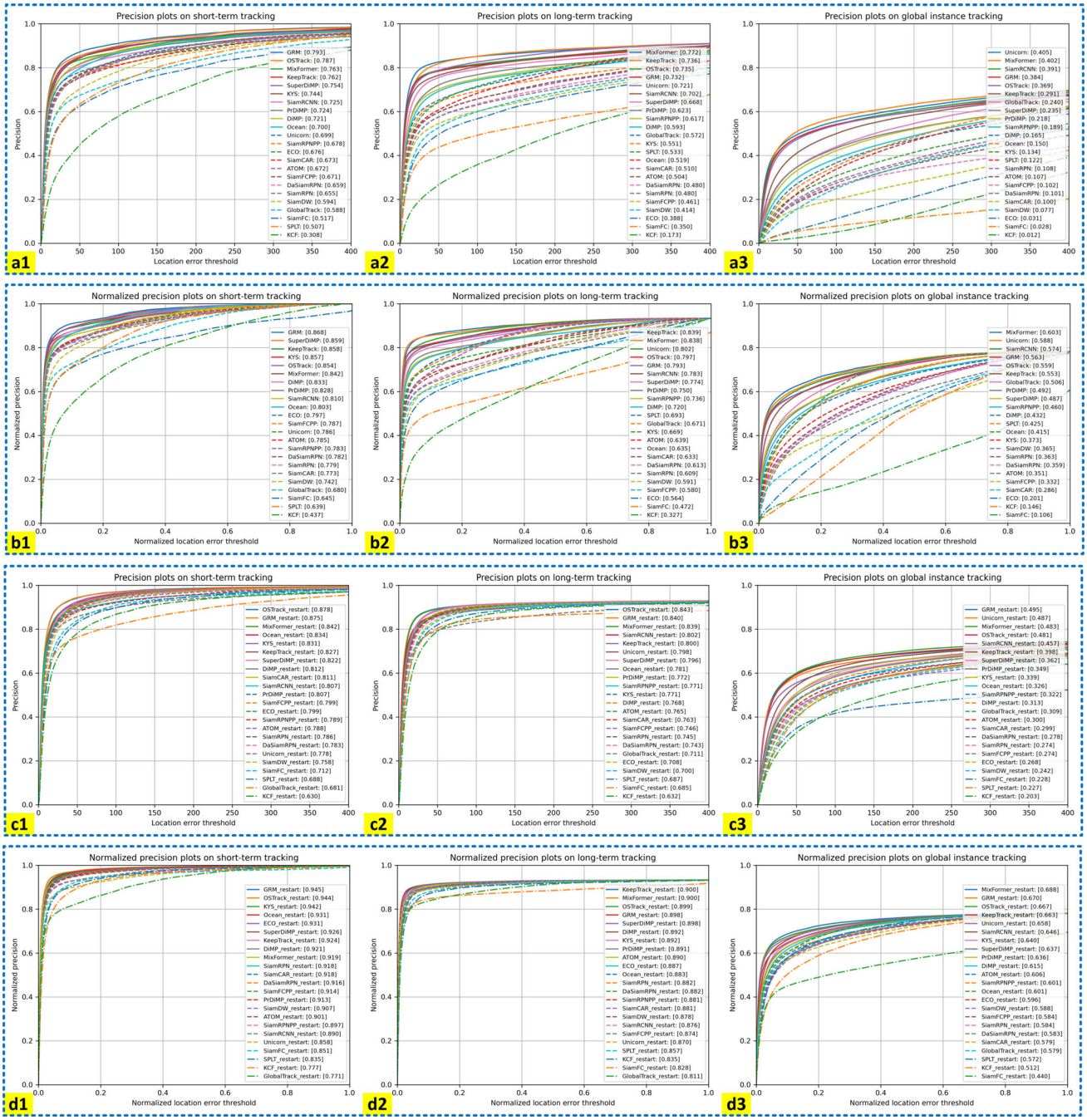


Fig. 20 Experiments in normal space. Three columns represent the results in the short-term tracking task (left), long-term tracking task (middle), and global instance tracking task (right). Each task is evaluated by precision plots in OPE (**a1–a3**), normalized precision plots in OPE (**b1–b3**), precision plots in R-OPE (**c1–c3**), normalized precision plots in R-OPE (**d1–d3**)

Table 7 Performance of 23 representative trackers on all sub-spaces, based on precision score (Color table online)

Trackers	Normal space								Challenging space										Mean
	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	
KCF	0.510	0.374	0.287	0.226	0.141	0.185	0.160	0.012	0.194	0.181	0.226	0.177	0.227	0.103	0.224	0.467	0.237	0.192	0.229
ECO	0.922	0.748	0.706	0.661	0.344	0.407	0.370	0.031	0.286	0.255	0.261	0.283	0.505	0.161	0.364	0.693	0.485	0.333	0.434
SiamFC	0.750	0.583	0.515	0.479	0.259	0.375	0.325	0.028	0.235	0.219	0.308	0.229	0.387	0.110	0.290	0.623	0.399	0.273	0.355
SiamRPN	0.856	0.752	0.687	0.632	0.346	0.498	0.462	0.108	0.314	0.303	0.412	0.319	0.487	0.229	0.405	0.661	0.417	0.363	0.458
DaSiamRPN	0.856	0.759	0.701	0.637	0.341	0.506	0.454	0.101	0.312	0.300	0.409	0.316	0.484	0.228	0.403	0.667	0.413	0.364	0.458
SiamRPNPP	0.840	0.748	0.703	0.680	0.419	0.701	0.533	0.189	0.355	0.364	0.464	0.381	0.507	0.278	0.481	0.685	0.529	0.434	0.516
SPLT	0.734	0.577	0.487	0.425	0.313	0.663	0.402	0.122	0.279	0.257	0.350	0.281	0.377	0.200	0.351	0.590	0.375	0.325	0.395
SiamDW	0.863	0.683	0.579	0.536	0.308	0.455	0.374	0.077	0.280	0.249	0.357	0.277	0.476	0.160	0.337	0.666	0.429	0.324	0.413
SiamCAR	0.866	0.739	0.674	0.664	0.424	0.520	0.501	0.100	0.317	0.312	0.427	0.344	0.514	0.239	0.434	0.685	0.480	0.397	0.480
SiamFCPP	0.850	0.754	0.667	0.636	0.448	0.466	0.456	0.102	0.306	0.320	0.401	0.314	0.551	0.264	0.400	0.678	0.478	0.411	0.472
Ocean	0.845	0.756	0.706	0.687	0.507	0.519	0.518	0.150	0.357	0.366	0.464	0.365	0.536	0.328	0.481	0.700	0.481	0.440	0.511
SiamRCNN	0.893	0.754	0.677	0.685	0.615	0.742	0.663	0.391	0.468	0.505	0.606	0.518	0.554	0.459	0.627	0.734	0.648	0.604	0.619
ATOM	0.872	0.704	0.680	0.662	0.441	0.508	0.500	0.107	0.248	0.236	0.346	0.347	0.514	0.246	0.435	0.583	0.543	0.421	0.466
DiMP	0.890	0.791	0.725	0.703	0.495	0.624	0.561	0.165	0.365	0.370	0.477	0.383	0.569	0.302	0.483	0.758	0.580	0.472	0.539
PrDiMP	0.893	0.765	0.733	0.701	0.530	0.661	0.585	0.218	0.379	0.399	0.507	0.425	0.560	0.334	0.518	0.762	0.617	0.498	0.560
SuperDiMP	0.914	0.795	0.767	0.732	0.563	0.687	0.650	0.235	0.400	0.435	0.531	0.441	0.568	0.378	0.558	0.783	0.648	0.531	0.590
KeepTrack	0.927	0.795	0.745	0.750	0.594	0.785	0.688	0.291	0.422	0.451	0.554	0.468	0.597	0.396	0.565	0.792	0.684	0.555	0.614
MixFormer	0.888	0.791	0.729	0.747	0.658	0.803	0.741	0.402	0.487	0.531	0.619	0.551	0.673	0.525	0.669	0.821	0.719	0.619	0.665
OSTrack	0.897	0.798	0.760	0.774	0.705	0.734	0.736	0.369	0.486	0.517	0.620	0.550	0.709	0.541	0.672	0.807	0.665	0.611	0.664
GRM	0.904	0.814	0.766	0.782	0.701	0.721	0.743	0.384	0.490	0.529	0.625	0.550	0.704	0.559	0.679	0.805	0.651	0.616	0.668
KYS	0.878	0.799	0.792	0.744	0.506	0.573	0.529	0.134	0.360	0.362	0.460	0.374	0.576	0.303	0.482	0.757	0.571	0.463	0.537
GlobalTrack	0.782	0.624	0.534	0.518	0.482	0.618	0.526	0.240	0.357	0.371	0.470	0.390	0.466	0.294	0.459	0.628	0.519	0.464	0.486
Unicorn	0.839	0.709	0.632	0.655	0.660	0.732	0.711	0.405	0.490	0.526	0.621	0.537	0.620	0.537	0.644	0.750	0.635	0.631	0.630

(1) e_1-OTB2015 (Wu et al., 2015). e_2-VOT2016 (Kristan et al., 2016). e_3-VOT2018 (Kristan et al., 2018). e_4-VOT2019 (Kristan et al., 2019). e_5-GOT-10k (Huang et al., 2021). e_6-VOTLT2019 (Kristan et al., 2019). e_7-LaSOT (Fan et al., 2021). e_8-VideoCube (Hu et al., 2023). c_1-abnormal ratio. c_2-abnormal scale. c_3-abnormal illumination. c_4-blur bounding-box. c_5-delta ratio. c_6-delta scale. c_7-delta illumination. c_8-delta blur bounding-box. c_9-fast motion. c_{10}-low correlation coefficient. (2) The background color of cells indicates the score, with red indicating a low score and green indicating a high score

Table 8 Performance of 23 representative trackers on all sub-spaces, based on normalized precision score (Color table online)

Trackers	Normal space								Challenging space										Mean
	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	
KCF	0.601	0.488	0.371	0.286	0.437	0.292	0.361	0.146	0.372	0.423	0.465	0.388	0.413	0.581	0.567	0.499	0.269	0.457	0.412
ECO	0.945	0.821	0.775	0.696	0.748	0.506	0.622	0.201	0.482	0.524	0.477	0.529	0.654	0.750	0.761	0.728	0.526	0.639	0.632
SiamFC	0.796	0.700	0.611	0.543	0.574	0.424	0.520	0.106	0.357	0.399	0.486	0.388	0.519	0.378	0.513	0.655	0.437	0.482	0.494
SiamRPN	0.908	0.830	0.762	0.686	0.709	0.547	0.671	0.363	0.496	0.548	0.647	0.548	0.625	0.754	0.746	0.707	0.463	0.635	0.647
DaSiamRPN	0.906	0.836	0.774	0.690	0.702	0.557	0.669	0.359	0.497	0.546	0.640	0.547	0.625	0.755	0.746	0.718	0.458	0.634	0.648
SiamRPNPP	0.879	0.812	0.760	0.718	0.746	0.758	0.714	0.460	0.525	0.582	0.677	0.603	0.647	0.769	0.787	0.720	0.572	0.694	0.690
SPLT	0.807	0.666	0.564	0.488	0.673	0.745	0.642	0.425	0.486	0.542	0.627	0.556	0.499	0.729	0.737	0.642	0.422	0.635	0.605
SiamDW	0.920	0.804	0.679	0.606	0.703	0.533	0.649	0.365	0.495	0.542	0.629	0.535	0.639	0.738	0.744	0.707	0.473	0.625	0.633
SiamCAR	0.899	0.808	0.735	0.696	0.728	0.576	0.689	0.286	0.484	0.534	0.648	0.549	0.619	0.727	0.753	0.716	0.506	0.640	0.644
SiamFCPP	0.896	0.843	0.745	0.688	0.764	0.513	0.646	0.332	0.493	0.547	0.645	0.541	0.660	0.772	0.748	0.689	0.500	0.664	0.649
Ocean	0.892	0.832	0.771	0.726	0.793	0.579	0.691	0.415	0.514	0.569	0.671	0.576	0.654	0.808	0.784	0.727	0.503	0.674	0.677
SiamRCNN	0.942	0.829	0.749	0.735	0.797	0.791	0.776	0.574	0.571	0.639	0.749	0.666	0.638	0.803	0.828	0.749	0.670	0.766	0.737
ATOM	0.926	0.795	0.741	0.703	0.761	0.572	0.706	0.351	0.429	0.491	0.586	0.576	0.633	0.769	0.784	0.617	0.573	0.685	0.650
DiMP	0.937	0.872	0.796	0.760	0.800	0.691	0.749	0.432	0.539	0.595	0.709	0.611	0.703	0.784	0.801	0.773	0.606	0.713	0.715
PrDiMP	0.938	0.849	0.798	0.745	0.811	0.728	0.773	0.492	0.554	0.618	0.733	0.646	0.678	0.818	0.823	0.778	0.638	0.730	0.730
SuperDiMP	0.962	0.878	0.837	0.792	0.826	0.749	0.800	0.487	0.559	0.630	0.735	0.649	0.691	0.818	0.836	0.810	0.672	0.738	0.748
KeepTrack	0.965	0.874	0.814	0.787	0.854	0.837	0.841	0.553	0.576	0.642	0.760	0.673	0.703	0.823	0.845	0.819	0.705	0.764	0.768
MixFormer	0.917	0.862	0.792	0.786	0.850	0.836	0.840	0.603	0.599	0.664	0.774	0.698	0.771	0.875	0.873	0.827	0.727	0.772	0.782
OSTrack	0.927	0.859	0.810	0.799	0.872	0.766	0.828	0.559	0.584	0.643	0.757	0.683	0.780	0.874	0.859	0.802	0.672	0.753	0.768
GRM	0.939	0.892	0.832	0.817	0.859	0.756	0.830	0.563	0.583	0.643	0.760	0.681	0.788	0.878	0.856	0.801	0.656	0.754	0.772
KYS	0.927	0.885	0.857	0.801	0.812	0.630	0.709	0.373	0.527	0.577	0.688	0.593	0.709	0.794	0.803	0.765	0.599	0.705	0.709
GlobalTrack	0.825	0.694	0.594	0.557	0.731	0.663	0.679	0.506	0.512	0.575	0.674	0.597	0.563	0.756	0.751	0.651	0.544	0.687	0.642
Unicorn	0.886	0.802	0.706	0.698	0.835	0.792	0.813	0.588	0.592	0.649	0.763	0.682	0.701	0.859	0.841	0.758	0.649	0.783	0.744

(1) e_1-OTB2015 (Wu et al., 2015). e_2-VOT2016 (Kristan et al., 2016). e_3-VOT2018 (Kristan et al., 2018). e_4-VOT2019 (Kristan et al., 2019). e_5-GOT-10k (Huang et al., 2021). e_6-VOTLT2019 (Kristan et al., 2019). e_7-LaSOT (Fan et al., 2021). e_8-VideoCube (Hu et al., 2023). c_1-abnormal ratio. c_2-abnormal scale. c_3-abnormal illumination. c_4-blur bounding-box. c_5-delta ratio. c_6-delta scale. c_7-delta illumination. c_8-delta blur bounding-box. c_9-fast motion. c_{10}-low correlation coefficient. (2) The background color of cells indicates the score, with red indicating a low score and green indicating a high score

Table 9 Performance of 23 representative trackers on all sub-spaces, based on success score (Color table online)

Trackers	Normal space								Challenging space										Mean
	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	
KCF	0.380	0.262	0.201	0.170	0.231	0.137	0.175	0.058	0.235	0.231	0.285	0.226	0.228	0.298	0.325	0.323	0.168	0.253	0.232
ECO	0.690	0.524	0.469	0.407	0.424	0.328	0.378	0.100	0.321	0.334	0.268	0.342	0.389	0.422	0.480	0.496	0.352	0.407	0.396
SiamFC	0.579	0.419	0.352	0.315	0.332	0.285	0.319	0.052	0.228	0.256	0.315	0.247	0.307	0.195	0.317	0.454	0.282	0.296	0.308
SiamRPN	0.654	0.549	0.486	0.433	0.451	0.406	0.479	0.222	0.354	0.384	0.461	0.383	0.401	0.478	0.516	0.506	0.320	0.440	0.440
DaSiamRPN	0.653	0.554	0.493	0.435	0.447	0.414	0.473	0.215	0.355	0.382	0.457	0.380	0.403	0.480	0.516	0.511	0.316	0.439	0.440
SiamRPNPP	0.648	0.557	0.501	0.471	0.490	0.562	0.528	0.304	0.381	0.420	0.498	0.432	0.428	0.503	0.565	0.525	0.407	0.491	0.484
SPLT	0.560	0.433	0.359	0.302	0.434	0.526	0.432	0.257	0.335	0.360	0.430	0.376	0.333	0.460	0.494	0.441	0.290	0.425	0.403
SiamDW	0.657	0.499	0.409	0.365	0.408	0.328	0.374	0.127	0.319	0.334	0.413	0.330	0.388	0.427	0.467	0.483	0.313	0.401	0.391
SiamCAR	0.660	0.542	0.466	0.443	0.492	0.419	0.487	0.142	0.335	0.367	0.454	0.377	0.413	0.445	0.515	0.523	0.366	0.443	0.438
SiamFCPP	0.647	0.551	0.469	0.436	0.526	0.382	0.474	0.205	0.348	0.393	0.461	0.383	0.431	0.506	0.525	0.499	0.367	0.480	0.449
Ocean	0.652	0.562	0.503	0.473	0.560	0.429	0.511	0.262	0.369	0.415	0.483	0.413	0.437	0.520	0.550	0.537	0.375	0.485	0.474
SiamRCNN	0.711	0.580	0.514	0.504	0.650	0.615	0.642	0.467	0.469	0.528	0.617	0.545	0.467	0.632	0.679	0.593	0.538	0.635	0.577
ATOM	0.675	0.536	0.492	0.458	0.528	0.419	0.511	0.207	0.299	0.319	0.411	0.407	0.430	0.498	0.554	0.456	0.424	0.492	0.451
DiMP	0.696	0.599	0.534	0.502	0.589	0.514	0.571	0.296	0.400	0.448	0.532	0.454	0.477	0.553	0.598	0.584	0.463	0.541	0.520
PrDiMP	0.703	0.590	0.544	0.504	0.614	0.549	0.596	0.335	0.416	0.464	0.558	0.486	0.480	0.565	0.617	0.597	0.499	0.558	0.538
SuperDiMP	0.719	0.609	0.565	0.531	0.646	0.571	0.641	0.357	0.432	0.494	0.577	0.503	0.487	0.604	0.650	0.619	0.527	0.588	0.562
KeepTrack	0.725	0.615	0.560	0.539	0.666	0.643	0.671	0.407	0.447	0.506	0.596	0.524	0.506	0.616	0.657	0.629	0.555	0.608	0.582
MixFormer	0.699	0.597	0.533	0.532	0.699	0.657	0.702	0.489	0.490	0.552	0.635	0.575	0.555	0.705	0.721	0.653	0.587	0.647	0.613
OSTrack	0.702	0.589	0.544	0.542	0.729	0.600	0.700	0.461	0.484	0.541	0.629	0.570	0.567	0.714	0.717	0.640	0.543	0.638	0.606
GRM	0.709	0.604	0.550	0.546	0.725	0.591	0.707	0.467	0.483	0.544	0.633	0.569	0.568	0.719	0.716	0.637	0.530	0.639	0.608
KYS	0.686	0.608	0.576	0.528	0.601	0.474	0.542	0.251	0.393	0.435	0.516	0.442	0.487	0.561	0.599	0.581	0.454	0.535	0.515
GlobalTrack	0.625	0.490	0.411	0.381	0.562	0.502	0.527	0.360	0.393	0.439	0.520	0.453	0.407	0.530	0.567	0.505	0.430	0.529	0.479
Unicorn	0.653	0.530	0.457	0.454	0.689	0.590	0.673	0.476	0.479	0.536	0.624	0.554	0.497	0.693	0.687	0.571	0.507	0.653	0.574

(1) e_1-OTB2015 (Wu et al., 2015). e_2-VOT2016 (Kristan et al., 2016). e_3-VOT2018 (Kristan et al., 2018). e_4-VOT2019 (Kristan et al., 2019). e_5-GOT-10k (Huang et al., 2021). e_6-VOTLT2019 (Kristan et al., 2019). e_7-LaSOT (Fan et al., 2021). e_8-VideoCube (Hu et al., 2023). c_1-abnormal ratio. c_2-abnormal scale. c_3-abnormal illumination. c_4-blur bounding-box. c_5-delta ratio. c_6-delta scale. c_7-delta illumination. c_8-delta blur bounding-box. c_9-fast motion. c_{10}-low correlation coefficient. (2) The background color of cells indicates the score, with red indicating a low score and green indicating a high score

Table 10 Performance of 23 representative trackers on all sub-spaces, based on precision score, weighted by sequences' length (Color table online)

Trackers	Normal space								Challenging space										Mean
	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	
KCF	0.519	0.373	0.290	0.253	0.121	0.120	0.144	0.011	0.077	0.049	0.120	0.040	0.256	0.092	0.153	0.290	0.119	0.167	0.177
ECO	0.926	0.782	0.719	0.640	0.337	0.345	0.360	0.032	0.135	0.081	0.208	0.088	0.607	0.154	0.312	0.639	0.283	0.328	0.387
SiamFC	0.781	0.628	0.539	0.513	0.261	0.289	0.319	0.031	0.116	0.073	0.212	0.083	0.423	0.118	0.240	0.522	0.223	0.280	0.314
SiamRPN	0.867	0.780	0.699	0.648	0.339	0.528	0.456	0.116	0.176	0.136	0.320	0.176	0.564	0.219	0.372	0.621	0.256	0.363	0.424
DaSiamRPN	0.857	0.782	0.710	0.665	0.334	0.535	0.445	0.110	0.178	0.133	0.308	0.176	0.562	0.219	0.365	0.634	0.239	0.365	0.423
SiamRPNPP	0.863	0.802	0.737	0.713	0.399	0.706	0.532	0.199	0.237	0.194	0.371	0.249	0.563	0.268	0.449	0.643	0.374	0.422	0.485
SPLT	0.776	0.633	0.527	0.502	0.296	0.684	0.400	0.134	0.191	0.131	0.286	0.171	0.449	0.197	0.330	0.606	0.269	0.351	0.385
SiamDW	0.899	0.762	0.644	0.611	0.292	0.509	0.372	0.087	0.153	0.103	0.259	0.136	0.564	0.161	0.314	0.621	0.266	0.333	0.394
SiamCAR	0.856	0.742	0.658	0.658	0.424	0.505	0.492	0.103	0.174	0.130	0.313	0.179	0.575	0.227	0.388	0.575	0.302	0.391	0.427
SiamFCPP	0.864	0.805	0.737	0.714	0.435	0.399	0.446	0.105	0.179	0.148	0.296	0.171	0.624	0.259	0.348	0.610	0.270	0.408	0.434
Ocean	0.836	0.810	0.727	0.705	0.490	0.557	0.521	0.155	0.217	0.198	0.360	0.229	0.599	0.329	0.446	0.640	0.278	0.422	0.473
SiamRCNN	0.916	0.782	0.693	0.699	0.618	0.747	0.663	0.404	0.374	0.374	0.511	0.428	0.619	0.429	0.585	0.745	0.587	0.598	0.598
ATOM	0.878	0.739	0.749	0.716	0.455	0.495	0.480	0.112	0.156	0.126	0.271	0.176	0.589	0.231	0.387	0.510	0.349	0.406	0.435
DiMP	0.883	0.818	0.729	0.724	0.507	0.643	0.553	0.177	0.236	0.191	0.371	0.228	0.652	0.285	0.444	0.715	0.424	0.463	0.502
PrDiMP	0.898	0.788	0.726	0.712	0.533	0.622	0.578	0.227	0.263	0.231	0.409	0.293	0.623	0.319	0.478	0.702	0.469	0.486	0.520
SuperDiMP	0.924	0.827	0.763	0.739	0.574	0.707	0.641	0.242	0.290	0.257	0.432	0.307	0.651	0.368	0.521	0.758	0.505	0.515	0.557
KeepTrack	0.934	0.838	0.779	0.738	0.598	0.793	0.676	0.300	0.318	0.292	0.467	0.345	0.669	0.380	0.536	0.798	0.591	0.542	0.589
MixFormer	0.913	0.840	0.744	0.746	0.667	0.808	0.737	0.412	0.389	0.378	0.537	0.451	0.719	0.486	0.628	0.836	0.622	0.621	0.641
OSTrack	0.920	0.837	0.752	0.755	0.699	0.746	0.721	0.375	0.388	0.374	0.530	0.436	0.754	0.501	0.623	0.788	0.510	0.601	0.628
GRM	0.928	0.852	0.761	0.762	0.702	0.734	0.733	0.389	0.396	0.391	0.537	0.446	0.750	0.527	0.631	0.796	0.495	0.609	0.636
KYS	0.845	0.818	0.817	0.734	0.506	0.551	0.523	0.142	0.220	0.182	0.348	0.207	0.662	0.277	0.425	0.704	0.404	0.450	0.490
GlobalTrack	0.798	0.644	0.553	0.543	0.478	0.635	0.529	0.253	0.250	0.222	0.386	0.283	0.506	0.272	0.438	0.592	0.476	0.474	0.463
Unicorn	0.858	0.759	0.665	0.674	0.651	0.742	0.711	0.399	0.380	0.382	0.533	0.438	0.655	0.515	0.614	0.766	0.595	0.640	0.610

(1) e_1-OTB2015 (Wu et al., 2015). e_2-VOT2016 (Kristan et al., 2016). e_3-VOT2018 (Kristan et al., 2018). e_4-VOT2019 (Kristan et al., 2019). e_5-GOT-10k (Huang et al., 2021). e_6-VOTLT2019 (Kristan et al., 2019). e_7-LaSOT (Fan et al., 2021). e_8-VideoCube (Hu et al., 2023). c_1-abnormal ratio. c_2-abnormal scale. c_3-abnormal illumination. c_4-blur bounding-box. c_5-delta ratio. c_6-delta scale. c_7-delta illumination. c_8-delta blur bounding-box. c_9-fast motion. c_{10}-low correlation coefficient. (2) The background color of cells indicates the score, with red indicating a low score and green indicating a high score

Table 11 Performance of 23 representative trackers on all sub-spaces, based on normalized precision score, weighted by sequences' length (Color table online)

Trackers	Normal space								Challenging space										Mean
	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	
KCF	0.629	0.487	0.371	0.326	0.441	0.202	0.342	0.138	0.219	0.248	0.294	0.195	0.453	0.538	0.421	0.309	0.134	0.408	0.342
ECO	0.944	0.828	0.750	0.662	0.753	0.444	0.617	0.193	0.295	0.328	0.389	0.299	0.736	0.709	0.655	0.674	0.301	0.589	0.565
SiamFC	0.830	0.733	0.619	0.586	0.587	0.329	0.510	0.108	0.207	0.191	0.348	0.190	0.549	0.322	0.435	0.557	0.241	0.469	0.434
SiamRPN	0.913	0.836	0.747	0.695	0.700	0.569	0.661	0.359	0.363	0.426	0.537	0.419	0.697	0.736	0.688	0.662	0.283	0.615	0.606
DaSiamRPN	0.902	0.837	0.756	0.711	0.690	0.577	0.658	0.356	0.371	0.428	0.526	0.424	0.695	0.741	0.683	0.678	0.265	0.620	0.607
SiamRPNPP	0.902	0.851	0.775	0.751	0.725	0.759	0.704	0.460	0.431	0.475	0.575	0.500	0.695	0.760	0.733	0.704	0.404	0.665	0.659
SPLT	0.856	0.724	0.604	0.582	0.668	0.747	0.631	0.422	0.407	0.461	0.547	0.466	0.573	0.732	0.689	0.652	0.306	0.639	0.595
SiamDW	0.952	0.847	0.709	0.671	0.694	0.580	0.640	0.363	0.372	0.435	0.520	0.420	0.719	0.728	0.677	0.672	0.297	0.595	0.605
SiamCAR	0.895	0.786	0.689	0.684	0.729	0.561	0.679	0.275	0.328	0.356	0.509	0.367	0.681	0.670	0.686	0.634	0.318	0.628	0.582
SiamFCPP	0.909	0.859	0.779	0.754	0.749	0.439	0.633	0.320	0.359	0.416	0.520	0.395	0.726	0.766	0.654	0.634	0.280	0.639	0.602
Ocean	0.890	0.861	0.767	0.742	0.775	0.604	0.689	0.403	0.411	0.482	0.564	0.471	0.715	0.795	0.733	0.695	0.294	0.652	0.641
SiamRCNN	0.960	0.838	0.749	0.755	0.792	0.787	0.769	0.587	0.509	0.568	0.659	0.597	0.704	0.794	0.777	0.766	0.602	0.759	0.721
ATOM	0.931	0.810	0.790	0.762	0.771	0.555	0.687	0.336	0.345	0.410	0.472	0.411	0.701	0.731	0.695	0.529	0.363	0.661	0.609
DiMP	0.937	0.875	0.769	0.762	0.801	0.699	0.740	0.433	0.434	0.478	0.600	0.474	0.768	0.767	0.746	0.747	0.439	0.692	0.676
PrDiMP	0.947	0.847	0.766	0.751	0.801	0.674	0.763	0.486	0.465	0.524	0.628	0.538	0.737	0.820	0.771	0.734	0.483	0.722	0.692
SuperDiMP	0.976	0.886	0.809	0.782	0.831	0.760	0.790	0.486	0.476	0.530	0.637	0.534	0.765	0.817	0.796	0.806	0.519	0.712	0.717
KeepTrack	0.977	0.898	0.831	0.772	0.852	0.833	0.827	0.551	0.499	0.564	0.670	0.578	0.769	0.818	0.807	0.838	0.606	0.744	0.746
MixFormer	0.943	0.884	0.777	0.776	0.841	0.835	0.833	0.608	0.533	0.593	0.698	0.620	0.803	0.839	0.826	0.860	0.626	0.764	0.759
OSTrack	0.951	0.881	0.784	0.781	0.850	0.772	0.812	0.557	0.520	0.577	0.676	0.598	0.816	0.842	0.812	0.803	0.514	0.738	0.738
GRM	0.959	0.911	0.807	0.801	0.842	0.764	0.818	0.563	0.519	0.580	0.679	0.598	0.820	0.855	0.809	0.812	0.498	0.743	0.743
KYS	0.899	0.874	0.857	0.772	0.797	0.598	0.699	0.370	0.402	0.456	0.570	0.433	0.776	0.766	0.729	0.720	0.420	0.672	0.656
GlobalTrack	0.841	0.696	0.595	0.584	0.732	0.672	0.678	0.505	0.434	0.509	0.594	0.526	0.603	0.751	0.718	0.621	0.494	0.682	0.624
Unicorn	0.906	0.831	0.721	0.726	0.813	0.793	0.810	0.581	0.516	0.572	0.681	0.606	0.736	0.844	0.808	0.775	0.602	0.772	0.727

(1) e_1-OTB2015 (Wu et al., 2015). e_2-VOT2016 (Kristan et al., 2016). e_3-VOT2018 (Kristan et al., 2018). e_4-VOT2019 (Kristan et al., 2019). e_5-GOT-10k (Huang et al., 2021). e_6-VOTLT2019 (Kristan et al., 2019). e_7-LaSOT (Fan et al., 2021). e_8-VideoCube (Hu et al., 2023). c_1-abnormal ratio. c_2-abnormal scale. c_3-abnormal illumination. c_4-blur bounding-box. c_5-delta ratio. c_6-delta scale. c_7-delta illumination. c_8-delta blur bounding-box. c_9-fast motion. c_{10}-low correlation coefficient. (2) The background color of cells indicates the score, with red indicating a low score and green indicating a high score

Table 12 Performance of 23 representative trackers on all sub-spaces, based on success score, weighted by sequences' length (Color table online)

Trackers	Normal space								Challenging space										Mean
	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	
KCF	0.386	0.267	0.210	0.192	0.223	0.095	0.162	0.054	0.118	0.105	0.157	0.090	0.244	0.266	0.229	0.181	0.084	0.215	0.182
ECO	0.698	0.527	0.462	0.398	0.427	0.273	0.367	0.094	0.176	0.178	0.228	0.168	0.436	0.398	0.402	0.434	0.201	0.362	0.346
SiamFC	0.621	0.438	0.367	0.350	0.345	0.219	0.313	0.052	0.123	0.108	0.216	0.109	0.319	0.176	0.267	0.379	0.156	0.297	0.270
SiamRPN	0.676	0.563	0.487	0.453	0.454	0.408	0.471	0.222	0.239	0.264	0.367	0.273	0.455	0.468	0.467	0.468	0.199	0.424	0.409
DaSiamRPN	0.668	0.566	0.493	0.462	0.448	0.415	0.464	0.216	0.244	0.263	0.358	0.273	0.457	0.471	0.464	0.475	0.186	0.425	0.408
SiamRPNPP	0.684	0.590	0.518	0.502	0.487	0.545	0.525	0.307	0.296	0.322	0.415	0.344	0.464	0.499	0.521	0.498	0.290	0.474	0.460
SPLT	0.603	0.467	0.385	0.366	0.432	0.493	0.427	0.257	0.267	0.278	0.361	0.296	0.380	0.461	0.449	0.428	0.213	0.422	0.388
SiamDW	0.699	0.531	0.435	0.413	0.403	0.327	0.363	0.124	0.192	0.186	0.290	0.190	0.437	0.414	0.414	0.407	0.182	0.372	0.354
SiamCAR	0.669	0.527	0.436	0.433	0.500	0.388	0.480	0.139	0.197	0.190	0.333	0.219	0.455	0.386	0.451	0.442	0.232	0.416	0.383
SiamFCPP	0.669	0.565	0.492	0.476	0.526	0.316	0.468	0.198	0.236	0.264	0.357	0.265	0.479	0.503	0.449	0.448	0.209	0.460	0.410
Ocean	0.657	0.577	0.500	0.481	0.554	0.414	0.511	0.257	0.272	0.312	0.393	0.315	0.474	0.522	0.504	0.484	0.219	0.464	0.440
SiamRCNN	0.744	0.587	0.508	0.510	0.651	0.599	0.639	0.476	0.407	0.460	0.536	0.485	0.506	0.617	0.633	0.603	0.484	0.625	0.560
ATOM	0.697	0.546	0.530	0.505	0.541	0.383	0.496	0.201	0.215	0.242	0.319	0.264	0.472	0.466	0.479	0.394	0.272	0.453	0.415
DiMP	0.706	0.599	0.518	0.514	0.596	0.497	0.565	0.298	0.308	0.334	0.440	0.336	0.526	0.544	0.547	0.555	0.338	0.522	0.486
PrDiMP	0.725	0.587	0.524	0.510	0.617	0.492	0.591	0.334	0.329	0.357	0.462	0.384	0.512	0.564	0.565	0.551	0.381	0.545	0.502
SuperDiMP	0.743	0.613	0.547	0.533	0.654	0.567	0.634	0.356	0.355	0.388	0.487	0.403	0.534	0.607	0.610	0.615	0.411	0.562	0.534
KeepTrack	0.750	0.630	0.572	0.530	0.671	0.627	0.664	0.405	0.372	0.414	0.513	0.437	0.544	0.614	0.619	0.638	0.482	0.587	0.559
MixFormer	0.736	0.615	0.526	0.527	0.700	0.644	0.698	0.491	0.428	0.480	0.567	0.508	0.572	0.672	0.676	0.676	0.507	0.641	0.592
OSTrack	0.736	0.596	0.518	0.519	0.718	0.595	0.687	0.458	0.423	0.473	0.557	0.497	0.587	0.683	0.670	0.644	0.418	0.624	0.578
GRM	0.744	0.610	0.525	0.524	0.718	0.587	0.698	0.465	0.424	0.479	0.561	0.500	0.583	0.698	0.668	0.651	0.405	0.628	0.582
KYS	0.678	0.602	0.578	0.516	0.597	0.431	0.535	0.249	0.285	0.316	0.414	0.307	0.537	0.542	0.532	0.538	0.321	0.511	0.472
GlobalTrack	0.648	0.493	0.414	0.402	0.566	0.480	0.528	0.359	0.314	0.355	0.444	0.382	0.428	0.521	0.532	0.468	0.394	0.523	0.458
Unicorn	0.672	0.547	0.463	0.468	0.678	0.576	0.671	0.468	0.409	0.461	0.549	0.491	0.514	0.683	0.653	0.591	0.477	0.644	0.556

(1) e_1-OTB2015 (Wu et al., 2015). e_2-VOT2016 (Kristan et al., 2016). e_3-VOT2018 (Kristan et al., 2018). e_4-VOT2019 (Kristan et al., 2019). e_5-GOT-10k (Huang et al., 2021). e_6-VOTLT2019 (Kristan et al., 2019). e_7-LaSOT (Fan et al., 2021). e_8-VideoCube (Hu et al., 2023). c_1-abnormal ratio. c_2-abnormal scale. c_3-abnormal illumination. c_4-blur bounding-box. c_5-delta ratio. c_6-delta scale. c_7-delta illumination. c_8-delta blur bounding-box. c_9-fast motion. c_{10}-low correlation coefficient. (2) The background color of cells indicates the score, with red indicating a low score and green indicating a high score

Appendix E: Experiments in Short-Term Tracking

E.1 Experiments in OTB (Wu et al., 2015)

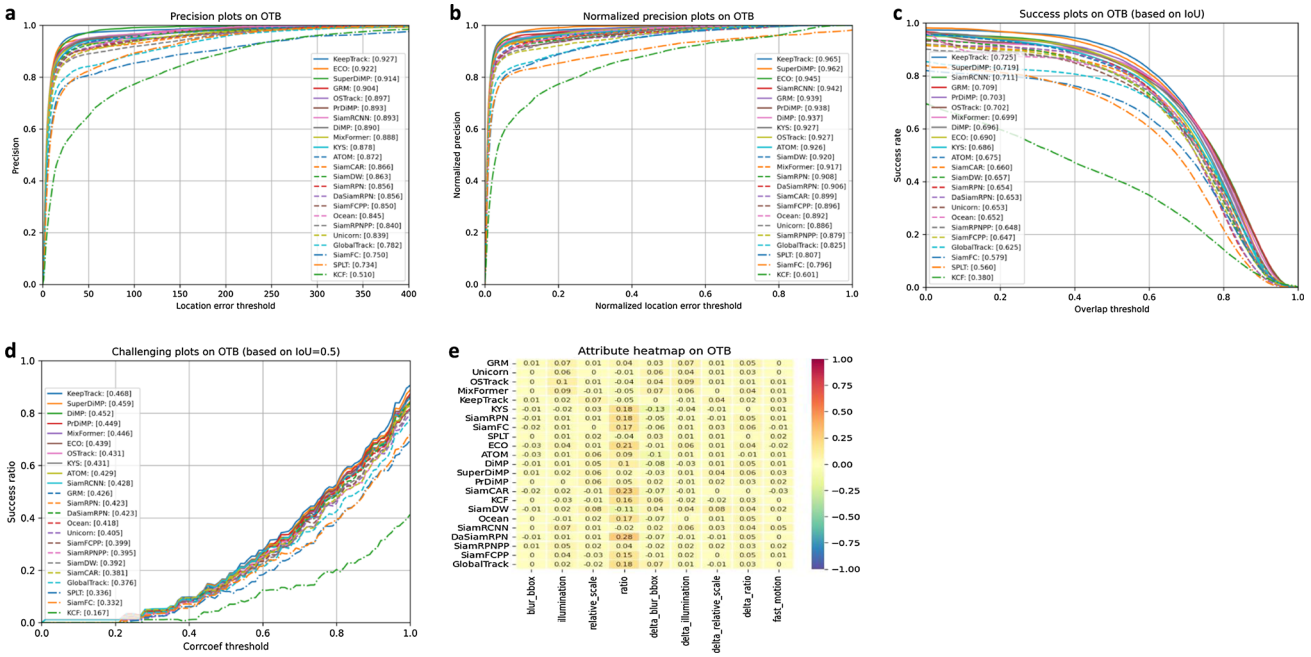


Fig. 21 Experiments in OTB (Wu et al., 2015) with OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, and **e** attribute plot

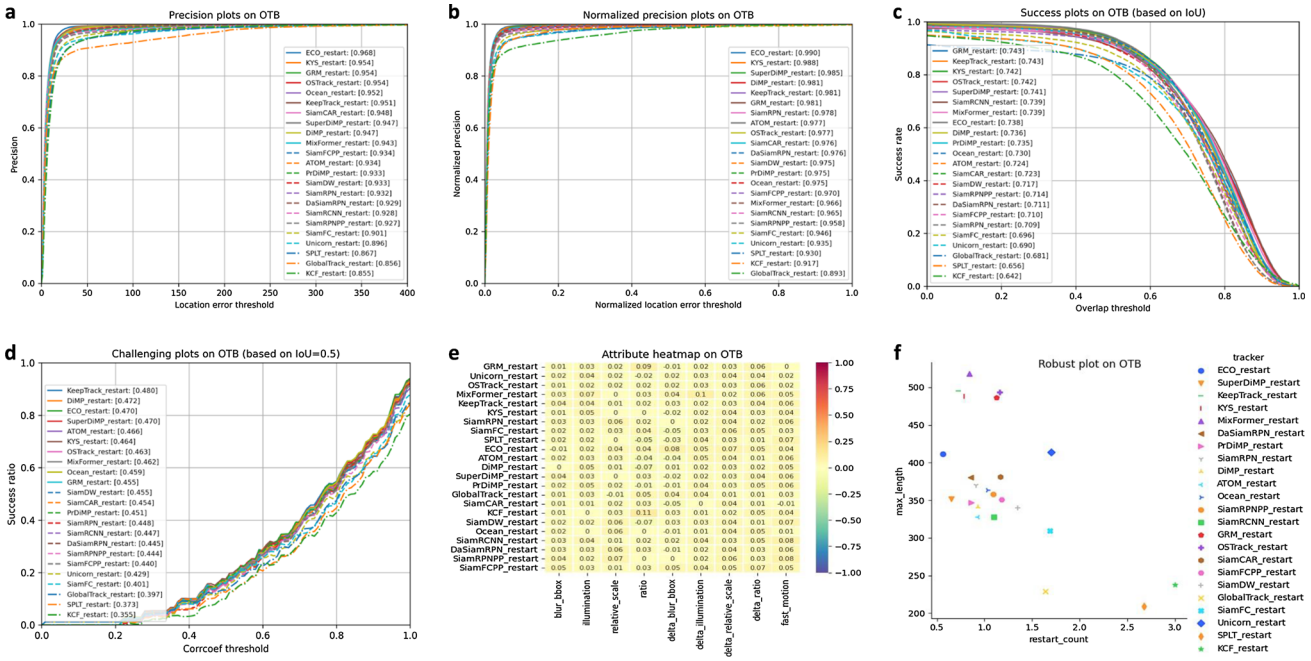


Fig. 22 Experiments in OTB (Wu et al., 2015) with R-OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, **e** attribute plot, and **f** robust plot

E.2 Experiments in VOT2016 (Kristan et al., 2016)

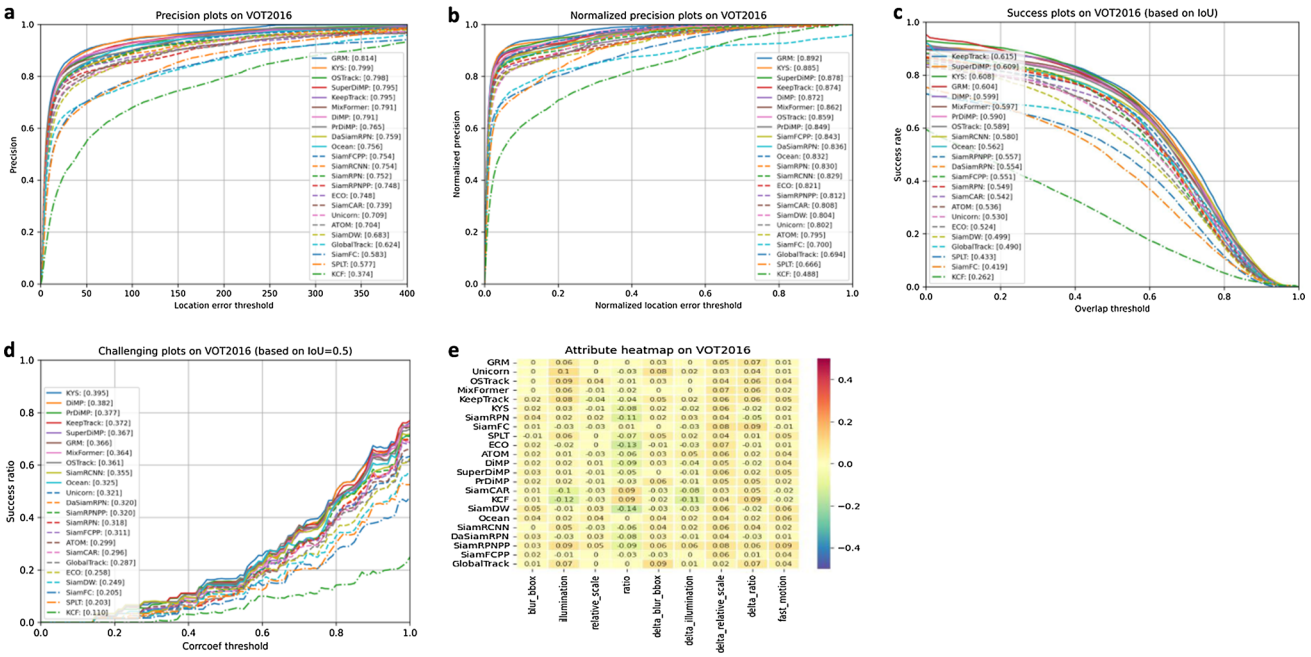


Fig. 23 Experiments in VOT2016 (Kristan et al., 2016) with OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, and **e** attribute plot

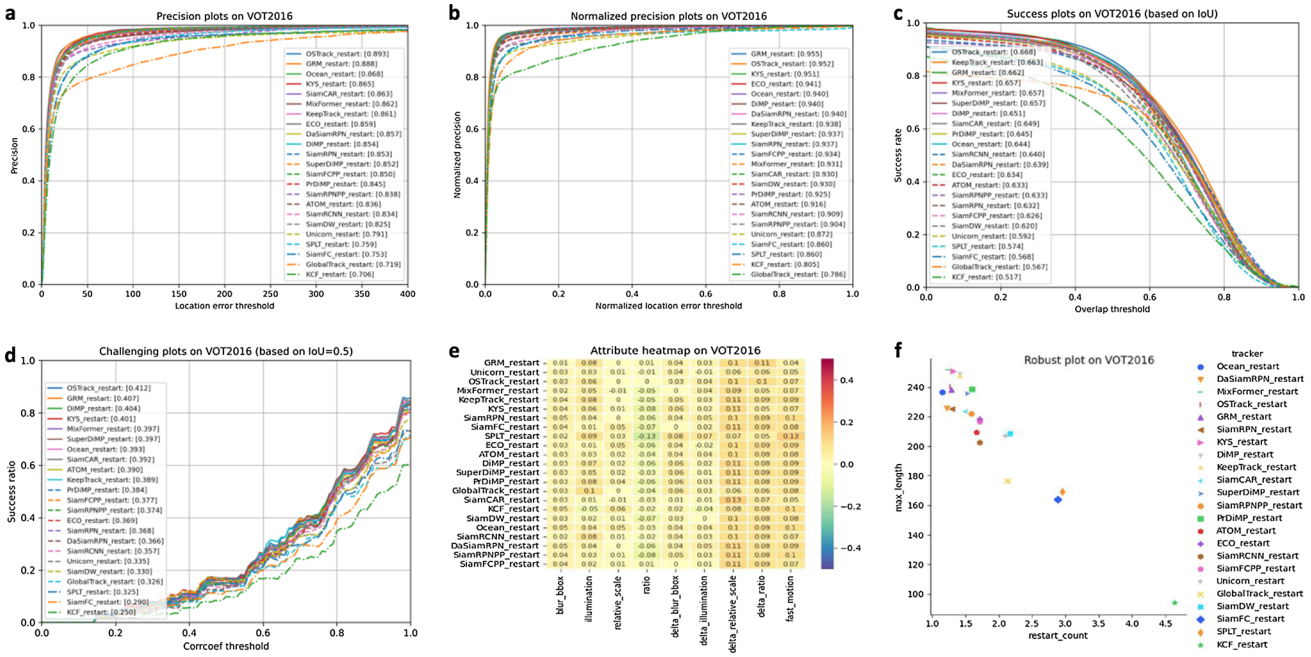


Fig. 24 Experiments in VOT2016 (Kristan et al., 2016) with R-OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, **e** attribute plot, and **f** robust plot

E.3 Experiments in VOT2018 (Kristan et al., 2018)

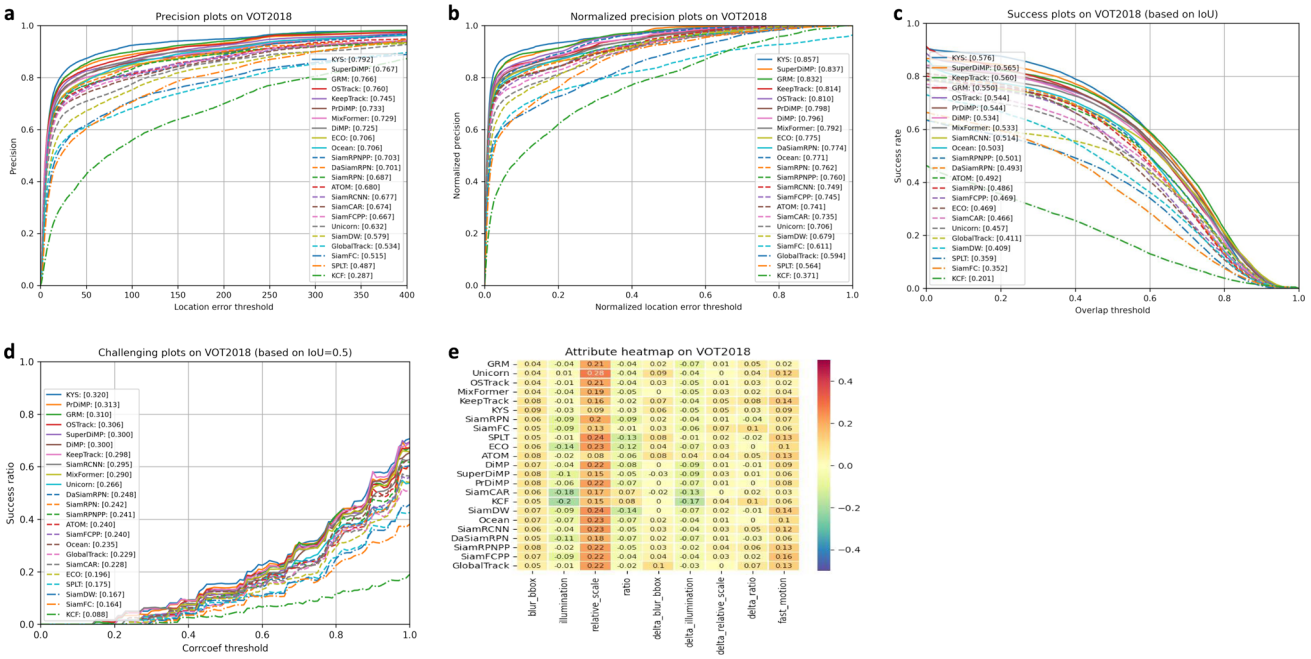


Fig. 25 Experiments in VOT2018 (Kristan et al., 2018) with OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, and **e** attribute plot

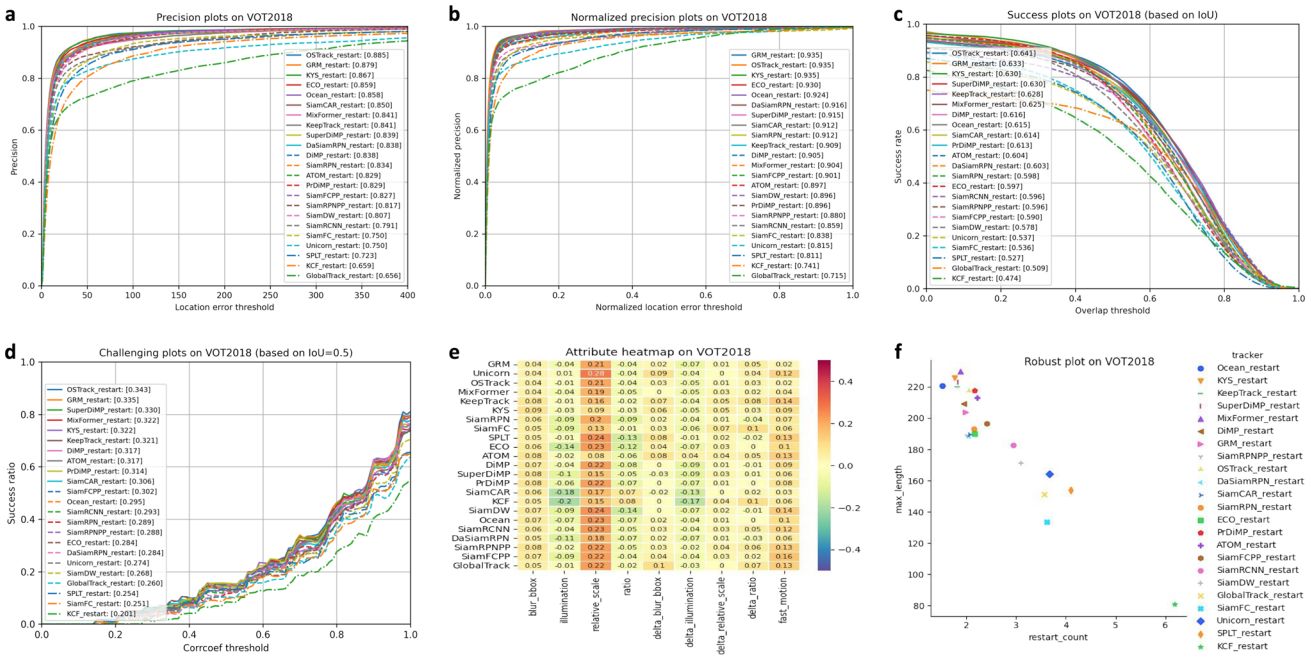


Fig. 26 Experiments in VOT2018 (Kristan et al., 2018) with R-OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, **e** attribute plot, and **f** robust plot

E.4 Experiments in VOT2019 (Kristan et al., 2019)

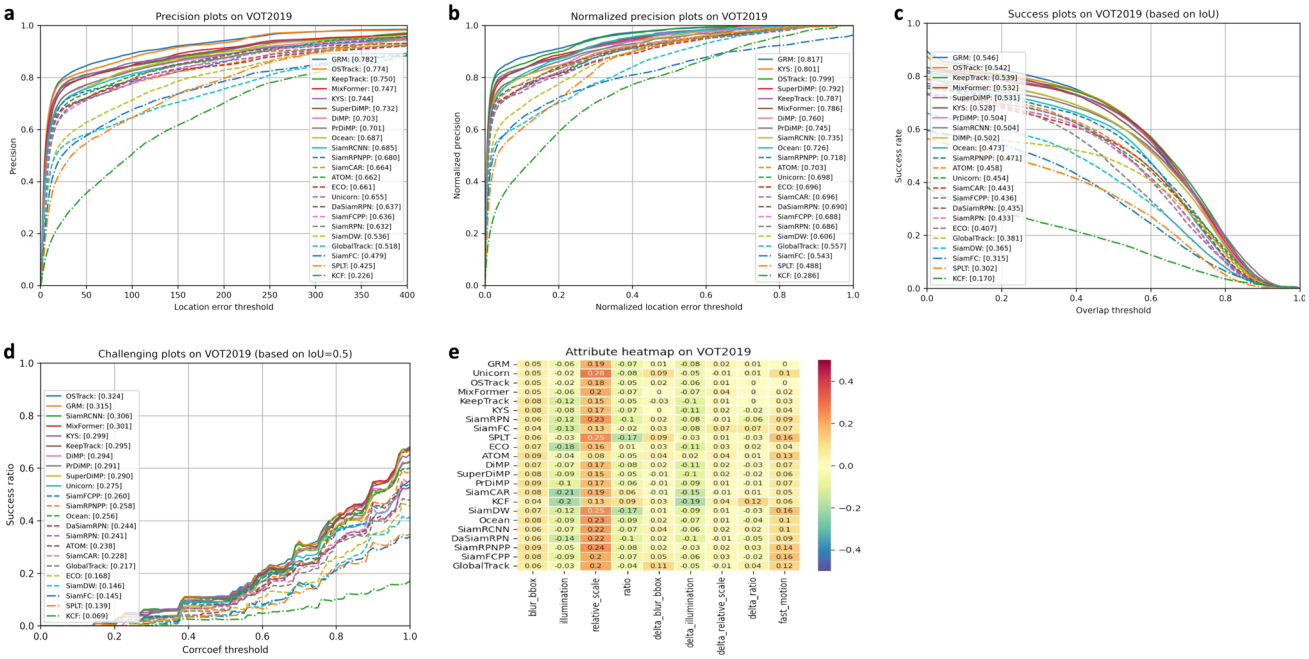


Fig. 27 Experiments in VOT2019 (Kristan et al., 2019) with OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, and **e** attribute plot

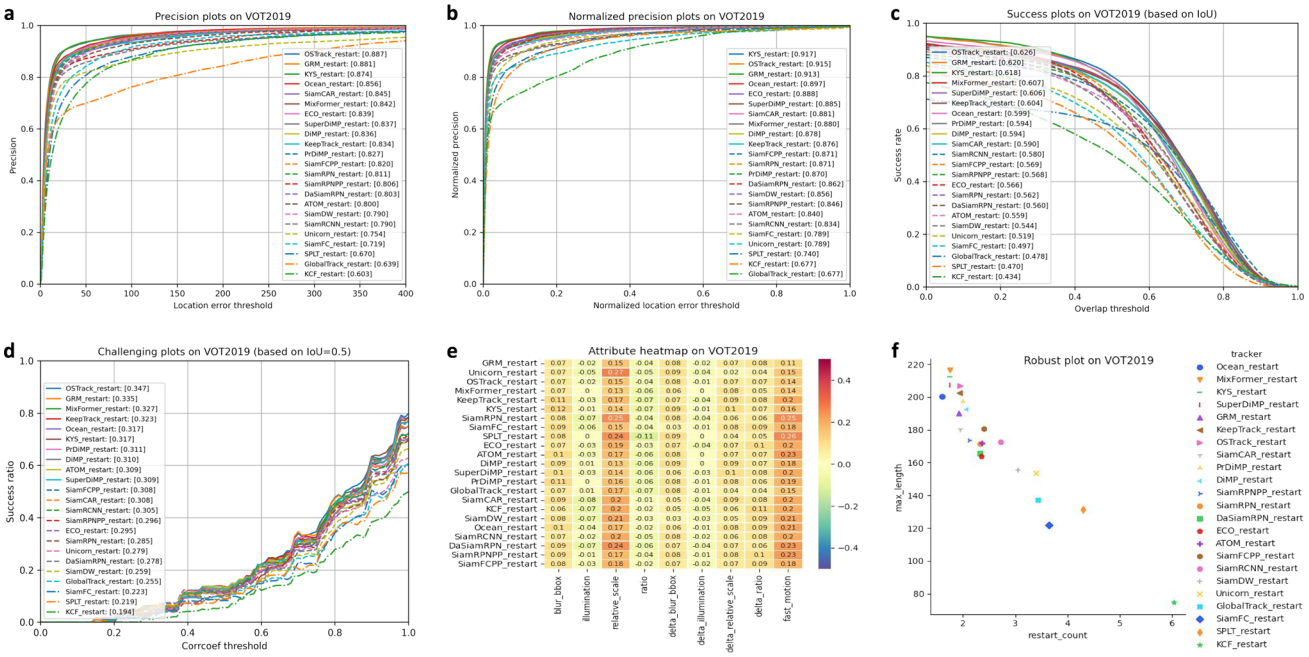


Fig. 28 Experiments in VOT2019 (Kristan et al., 2019) with R-OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, **e** attribute plot, and **f** robust plot

E.5 Experiments in GOT-10k (Huang et al., 2021)

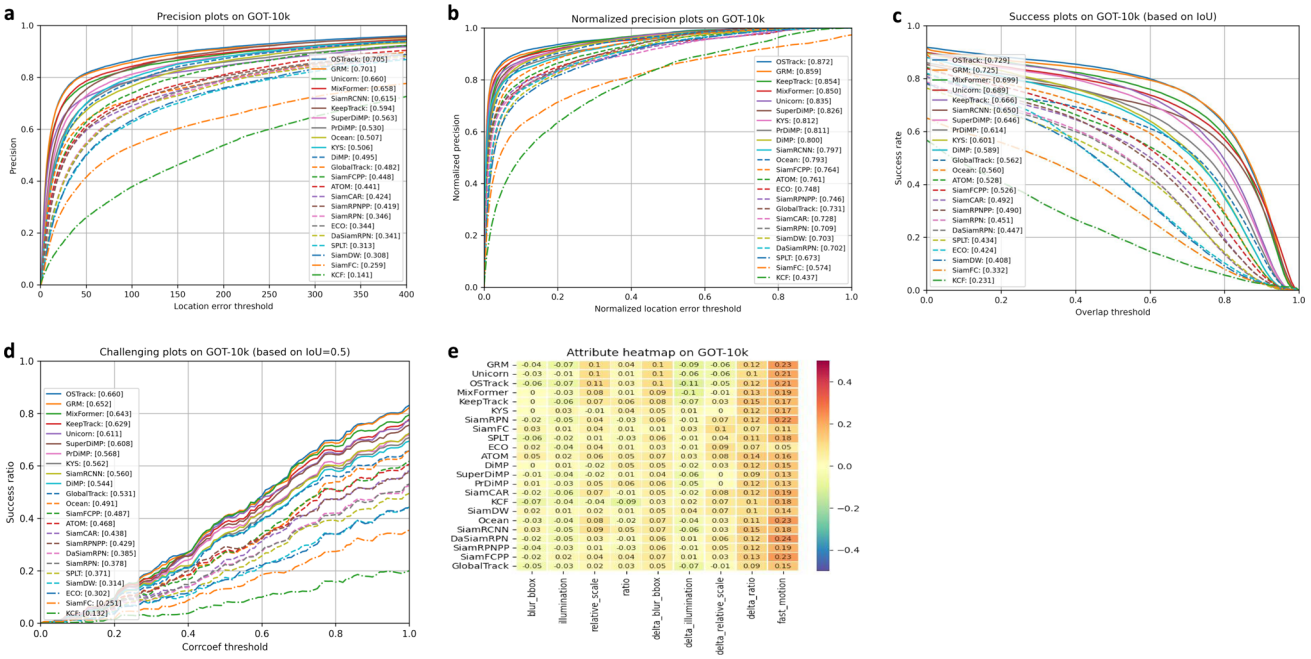


Fig. 29 Experiments in GOT-10k (Huang et al., 2021) with OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, and **e** attribute plot

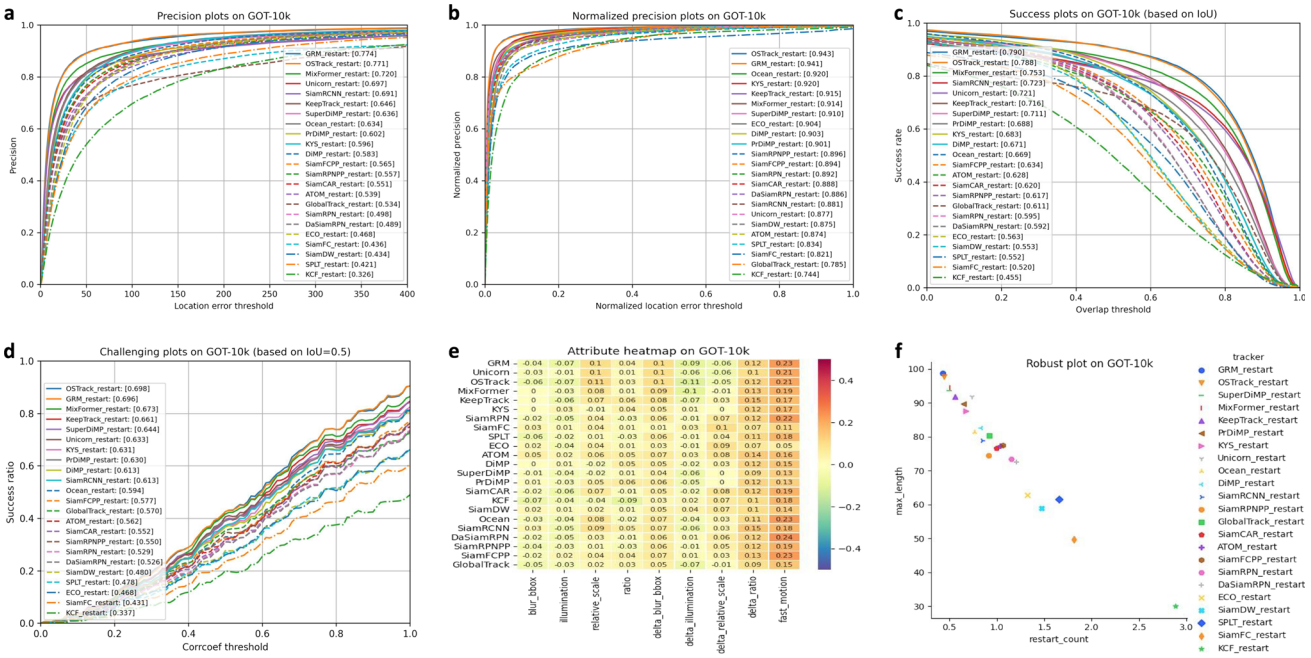


Fig. 30 Experiments in GOT-10k (Huang et al., 2021) with R-OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, **e** attribute plot, and **f** robust plot

Appendix F: Experiments in Long-term Tracking

F.1 Experiments in VOTLT2019 (Kristan et al., 2019)

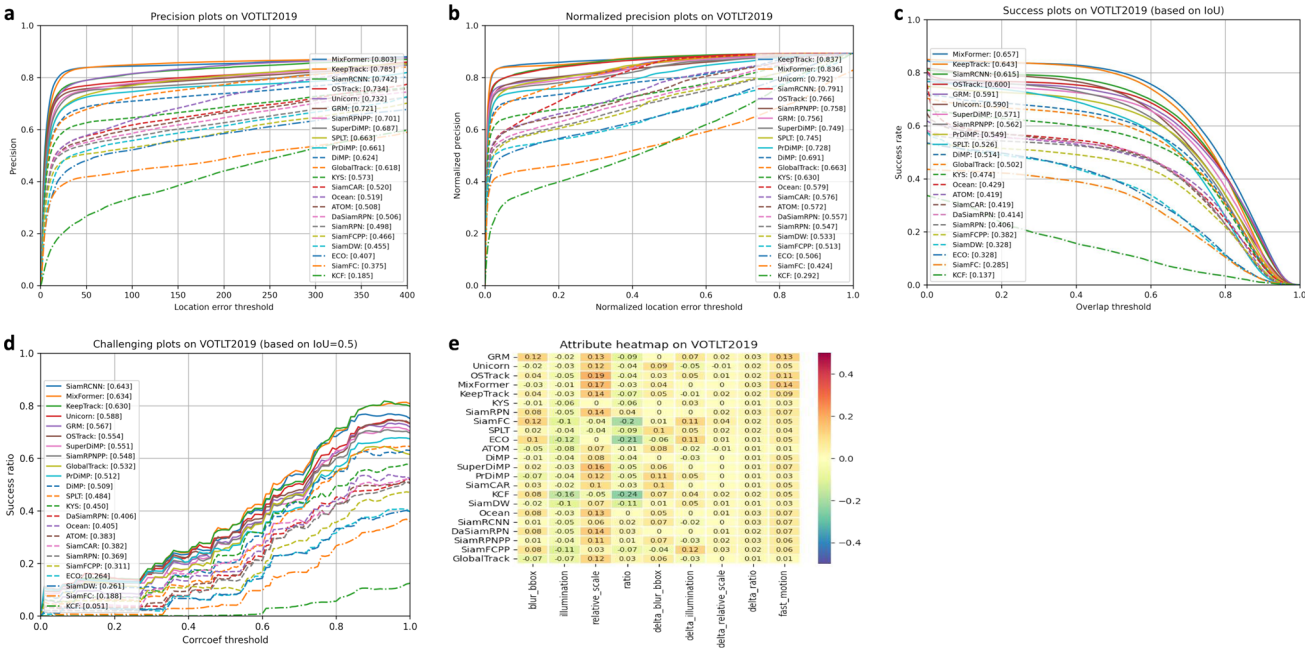


Fig. 31 Experiments in VOTLT2019 (Kristan et al., 2019) with OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, and **e** attribute plot

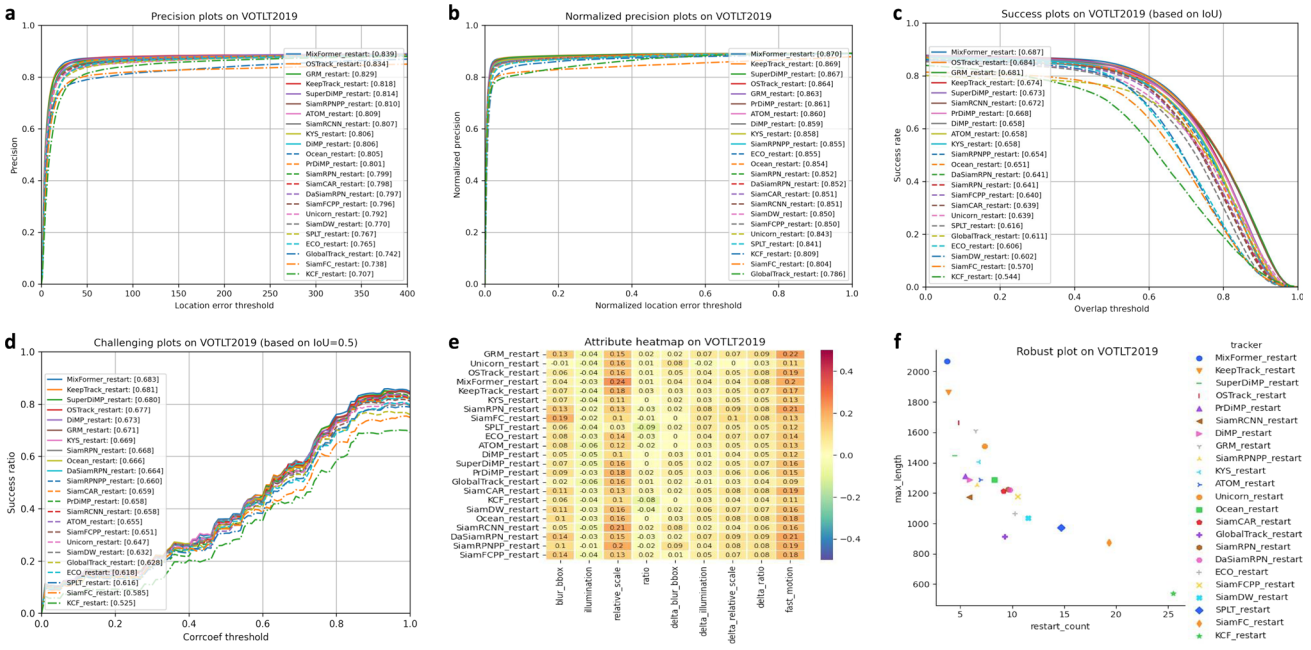


Fig. 32 Experiments in VOTLT2019 (Kristan et al., 2019) with R-OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, **e** attribute plot, and **f** robust plot

F.2 Experiments in LaSOT (Fan et al., 2021)

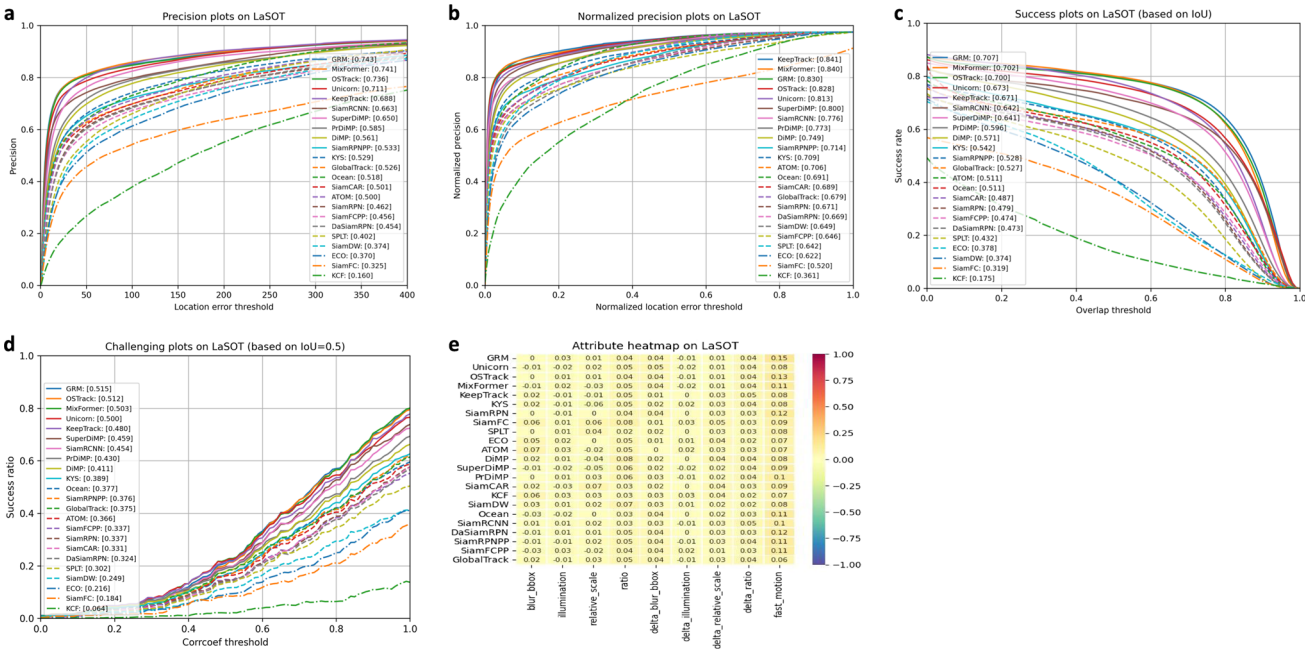


Fig. 33 Experiments in LaSOT (Fan et al., 2021) with OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, and **e** attribute plot

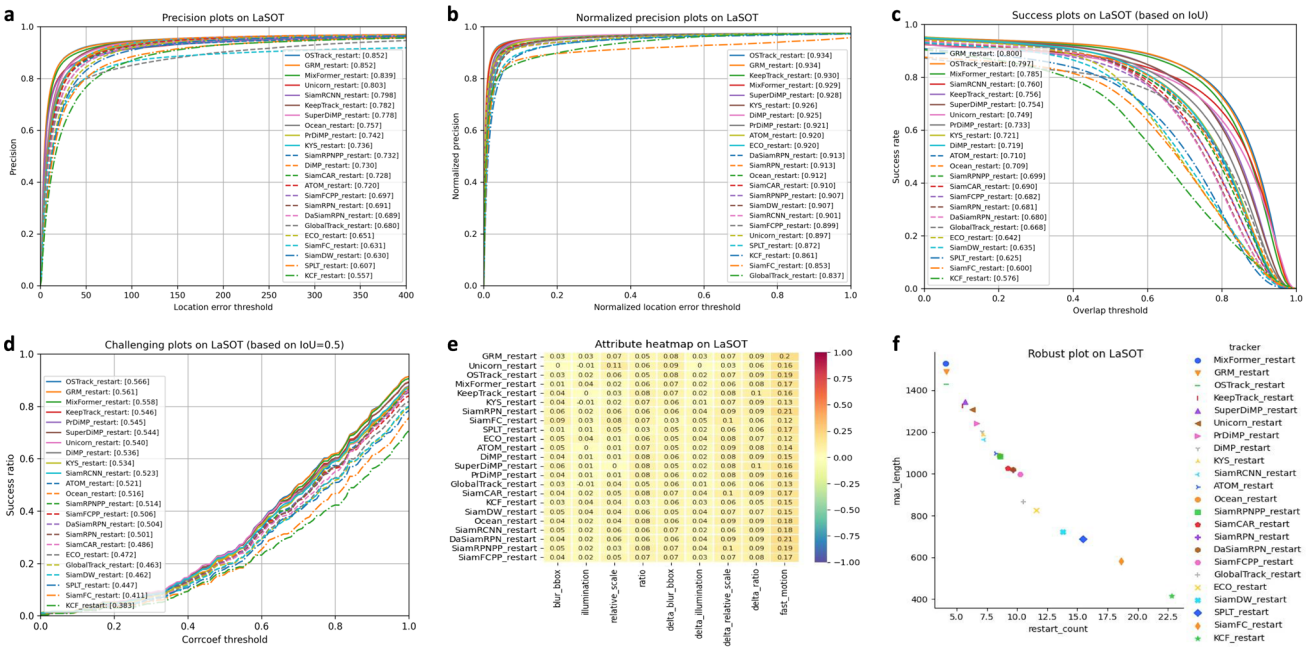


Fig. 34 Experiments in LaSOT (Fan et al., 2021) with R-OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, **e** attribute plot, and **f** robust plot

Appendix G: Experiments in Global Instance Tracking

G.1 Experiments in VideoCube (Hu et al., 2023)

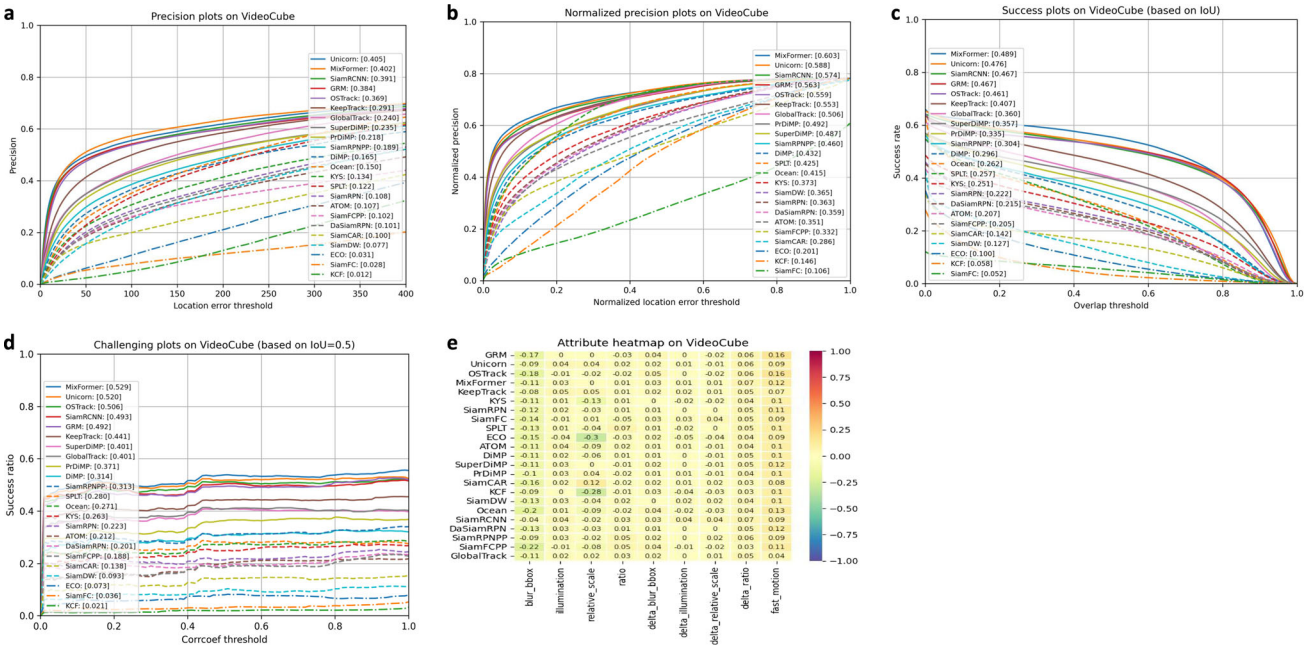


Fig. 35 Experiments in VideoCube (Hu et al., 2023) with OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, and **e** attribute plot

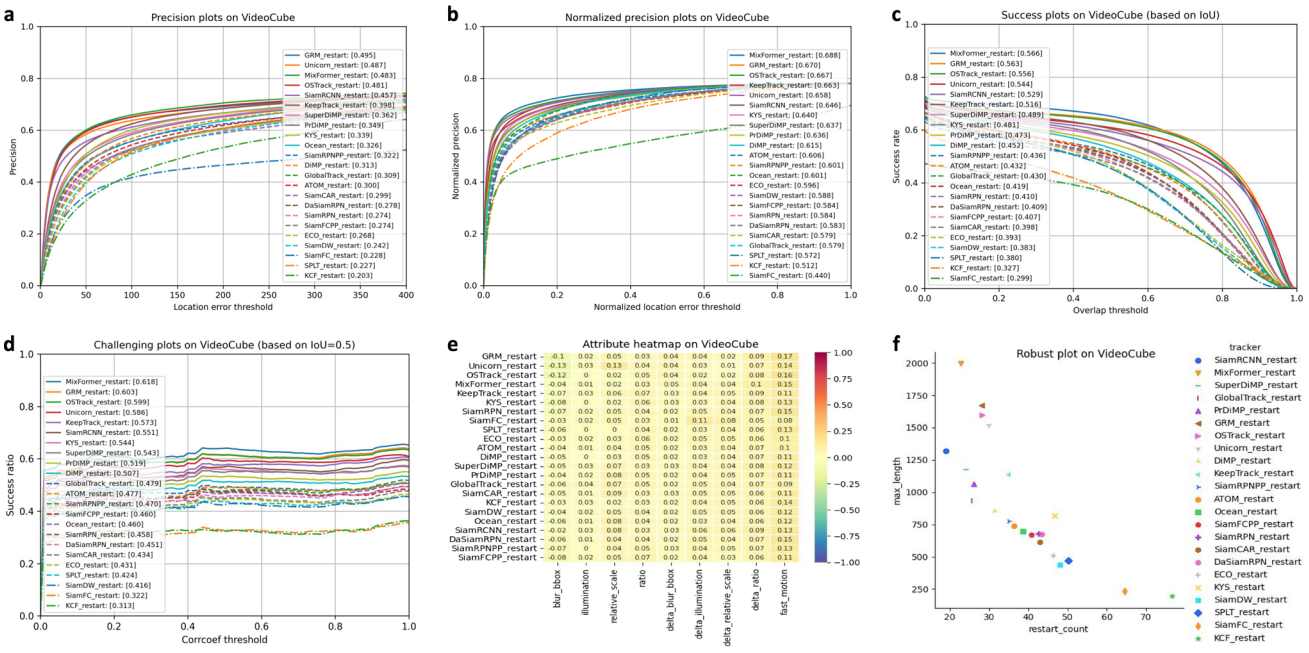


Fig. 36 Experiments in VideoCube (Hu et al., 2023) with R-OPE mechanisms, evaluated by **a** precision plot, **b** normalized precision plot, **c** success plot, **d** challenging plot, **e** attribute plot, and **f** robust plot

Appendix H: The Composition of Challenging Space

H.1 Abnormal Ratio

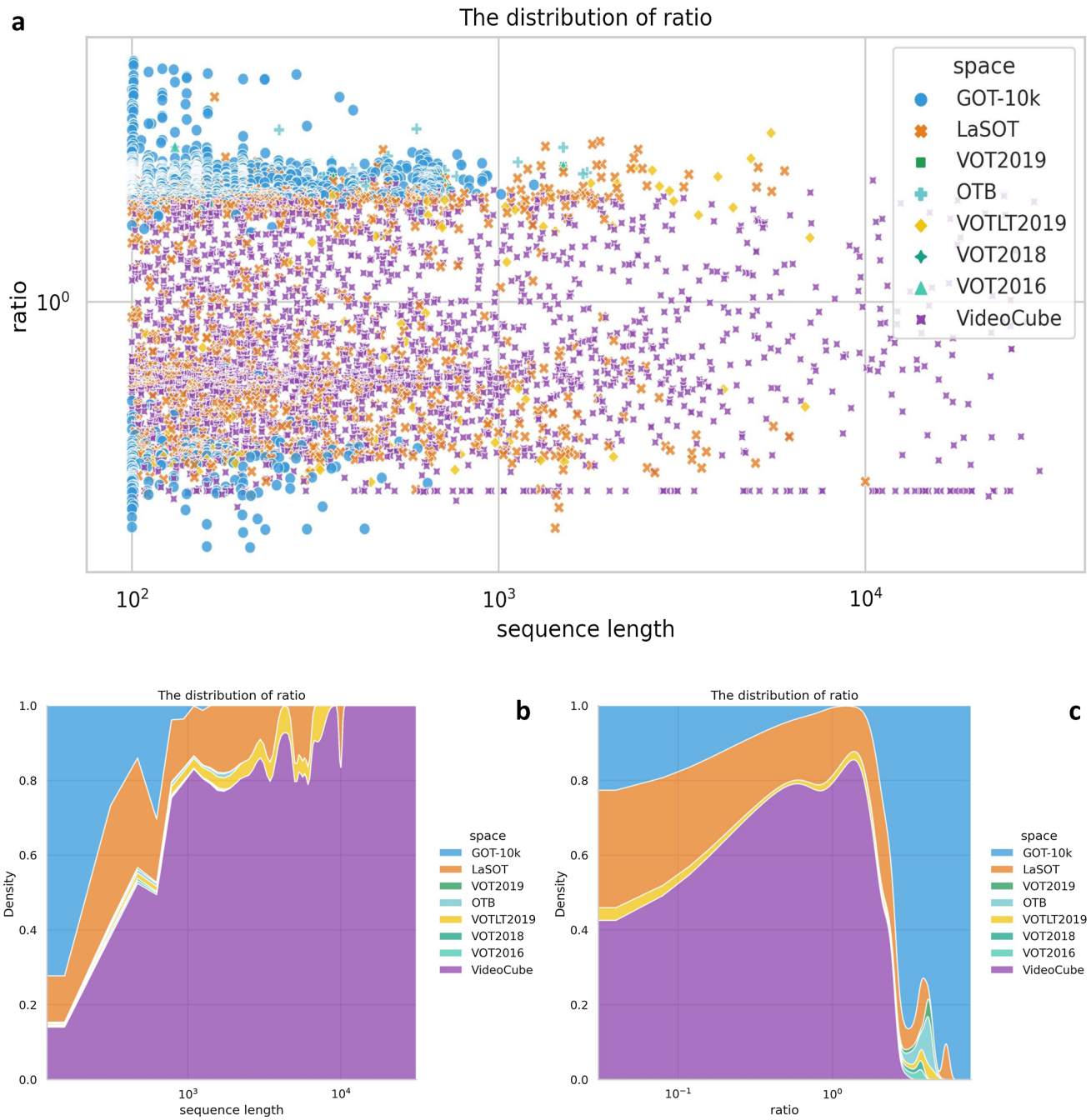


Fig. 37 The composition of abnormal ratio space. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

H.2 Abnormal Scale

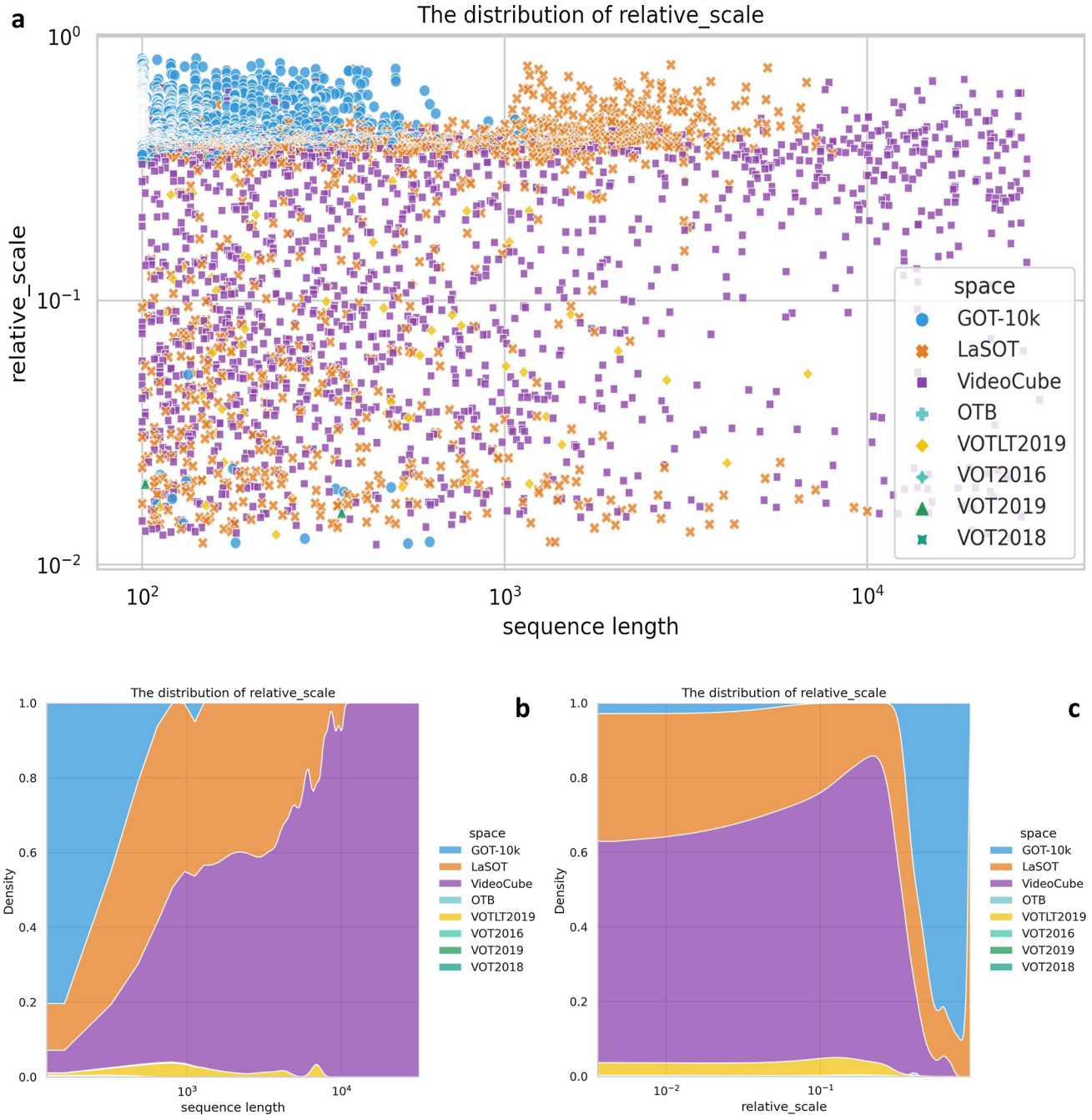


Fig. 38 The composition of abnormal scale space. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

H.3 Abnormal Illumination

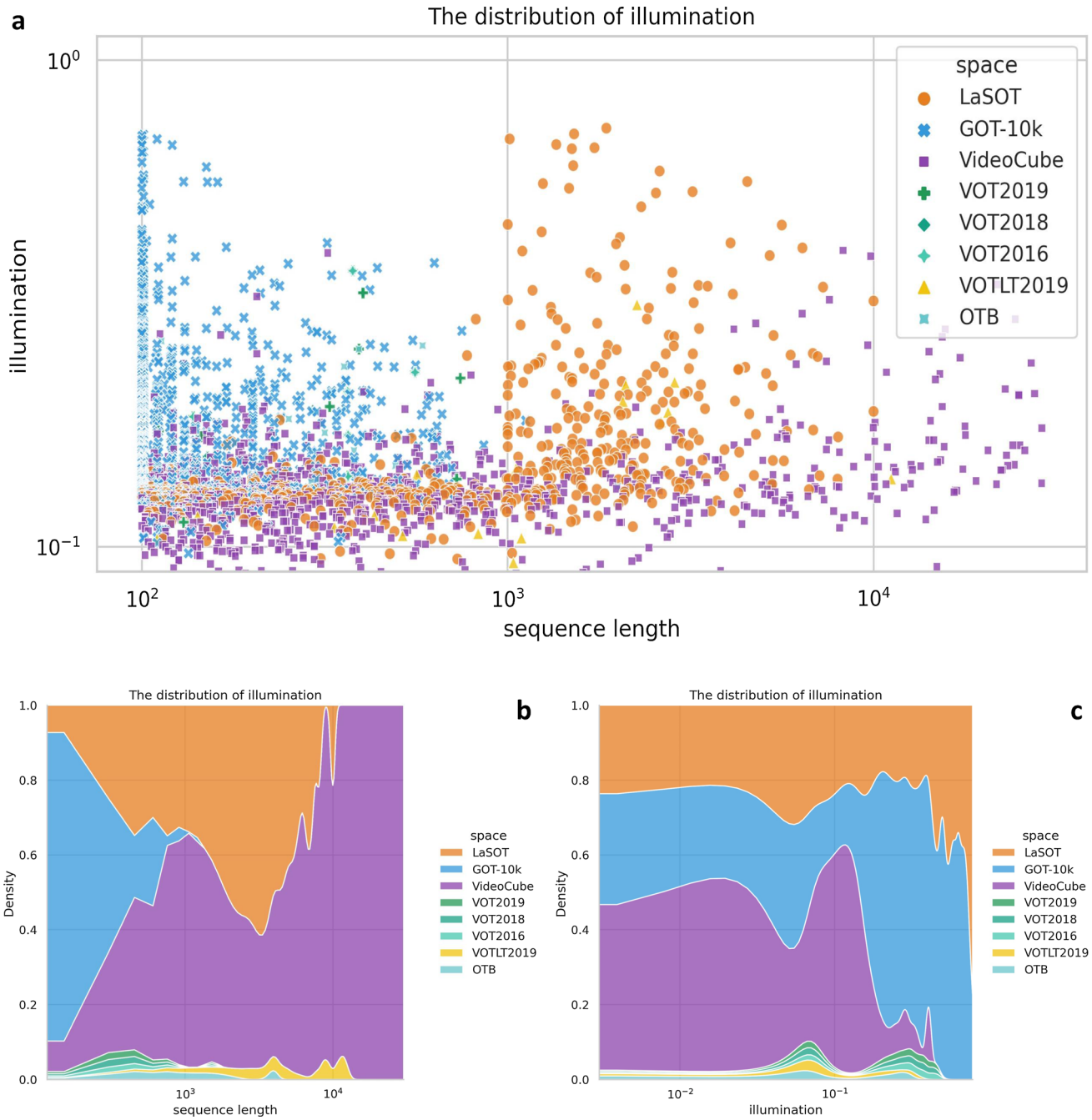


Fig. 39 The composition of abnormal illumination space. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

H.4 Blur Bounding-box

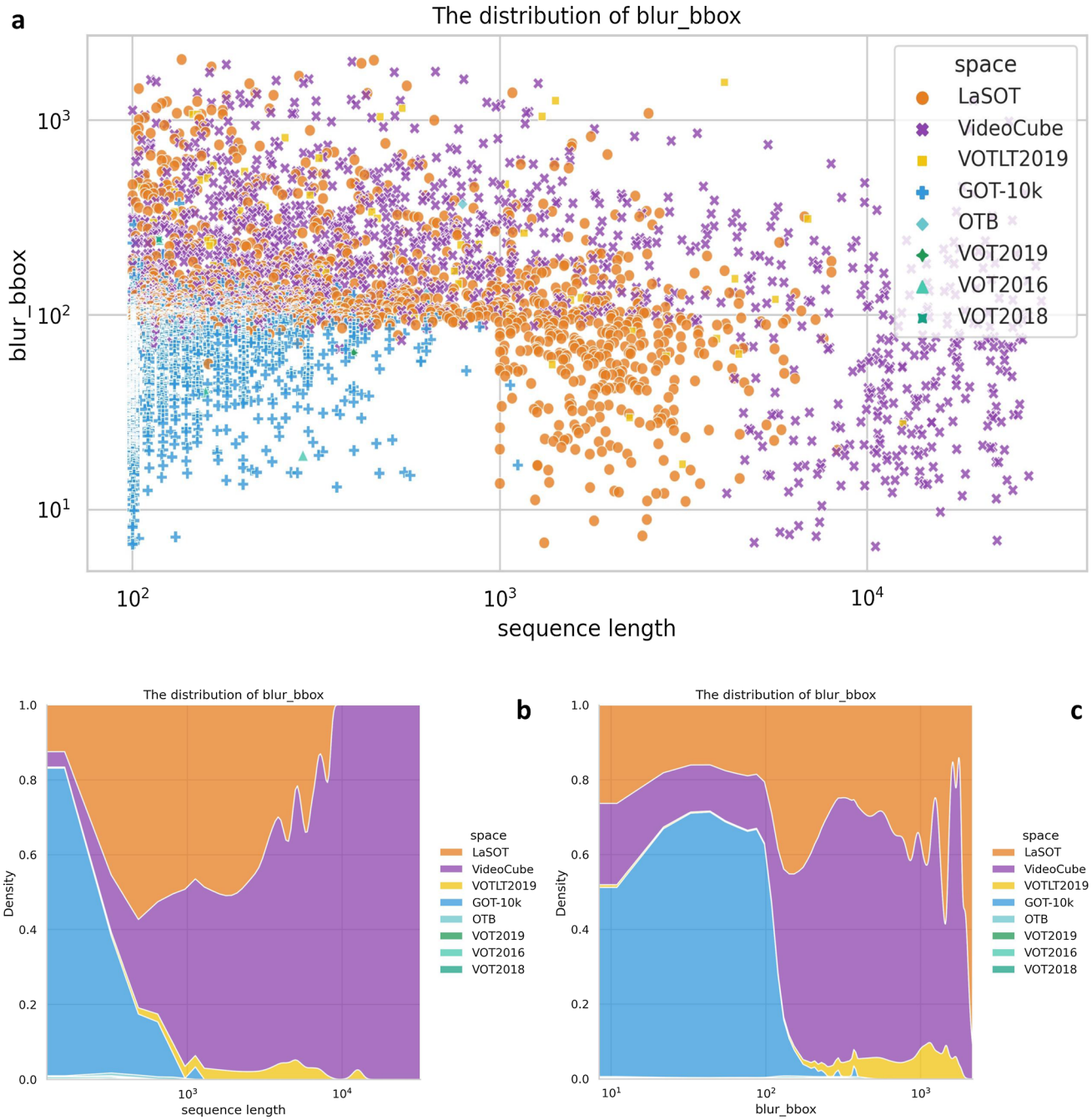


Fig. 40 The composition of blur bounding-box space. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

H.5 Delta Ratio

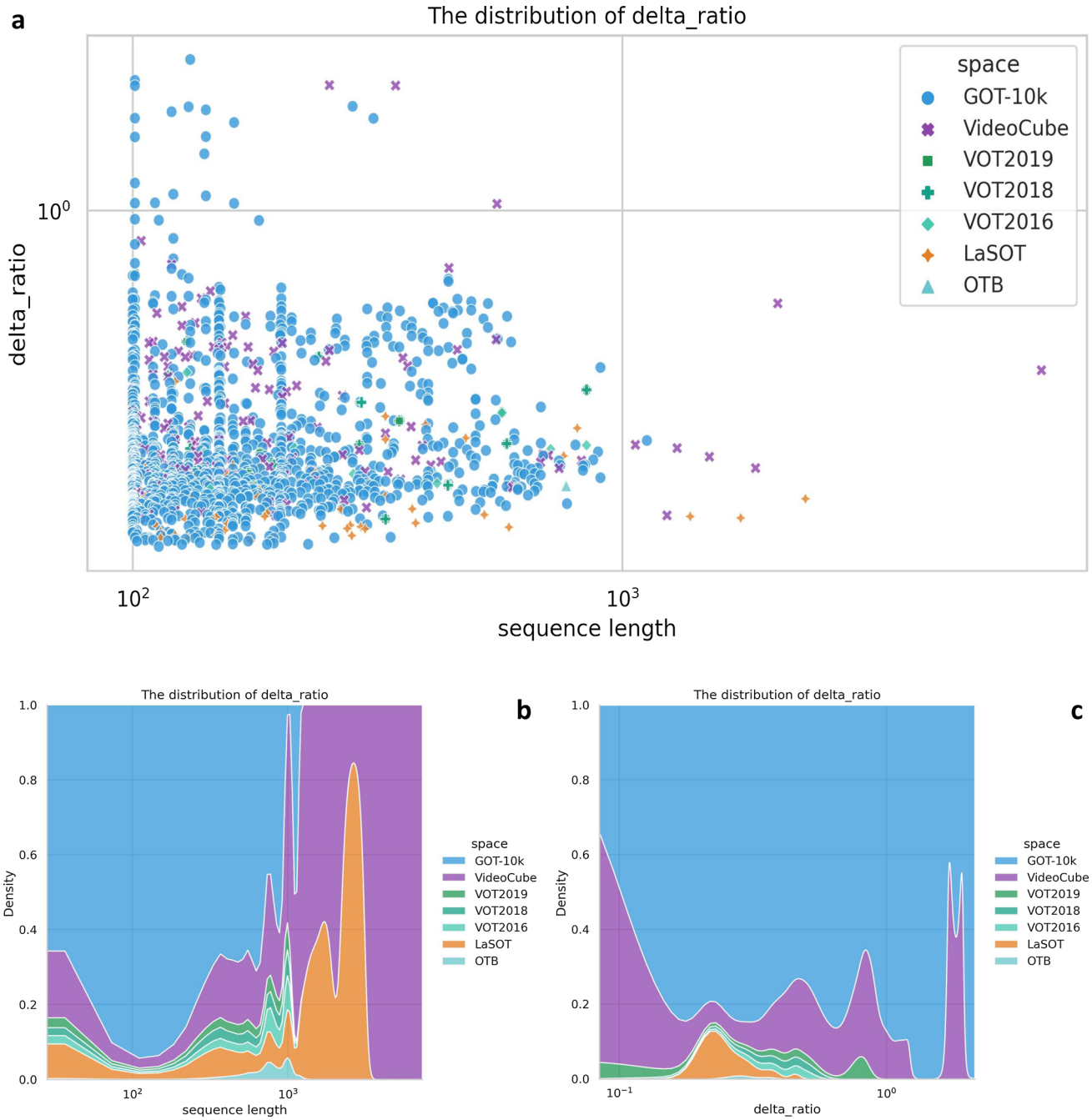


Fig. 41 The composition of delta ratio space. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

H.6 Delta Scale

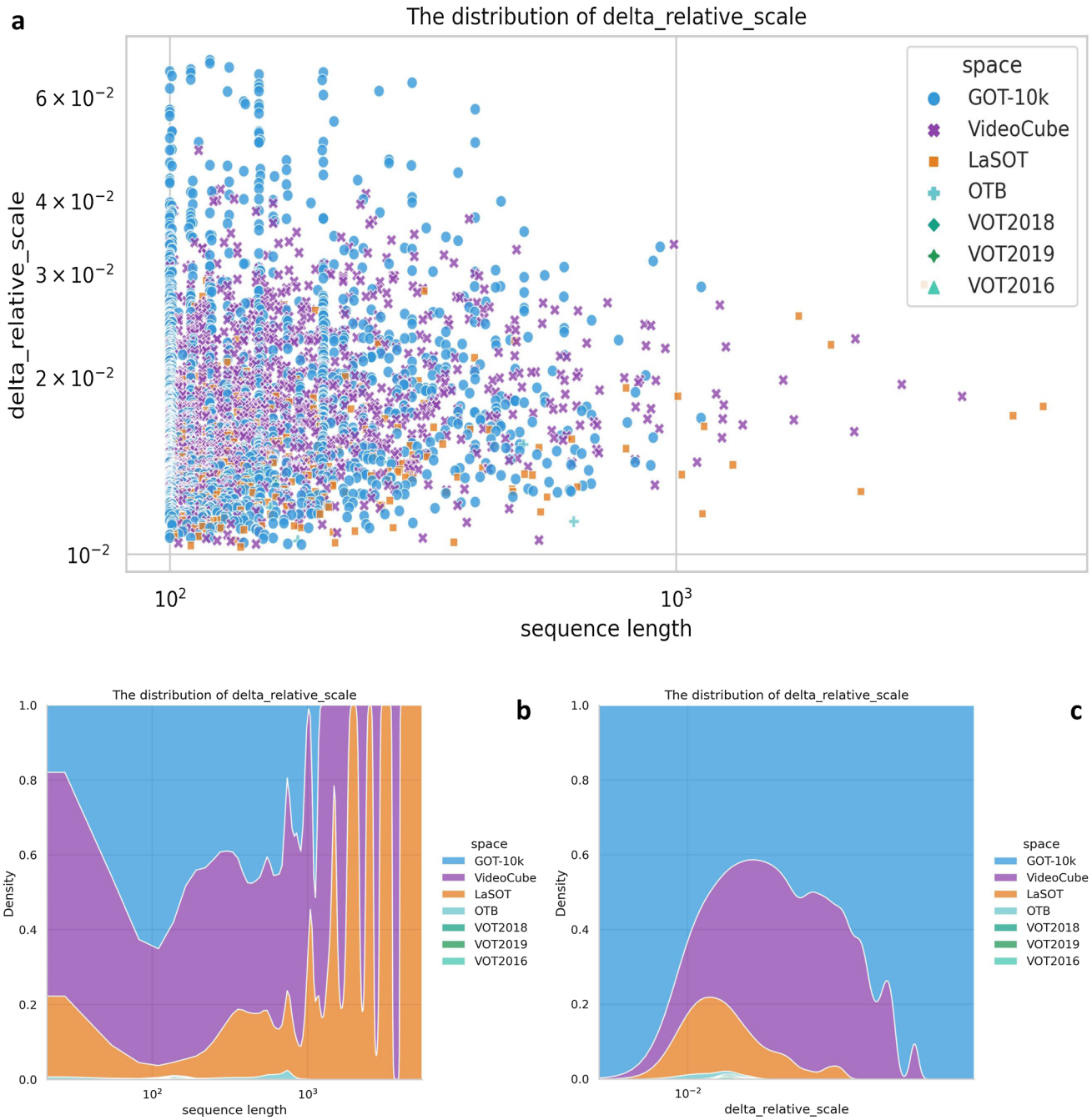


Fig. 42 The composition of delta scale space. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

H.7 Delta Illumination

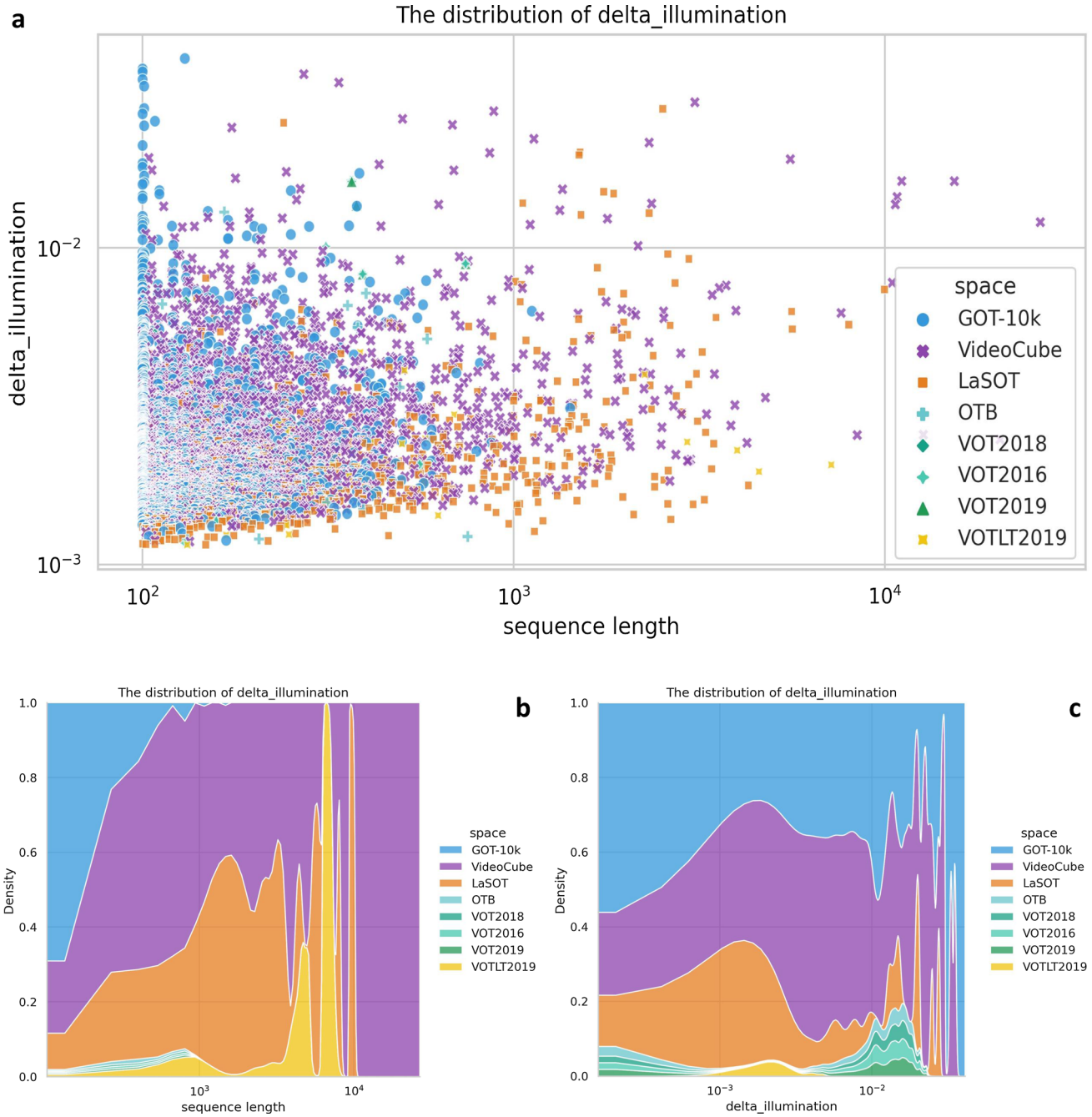


Fig. 43 The composition of delta illumination. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

H.8 Delta Blur Bounding-Box

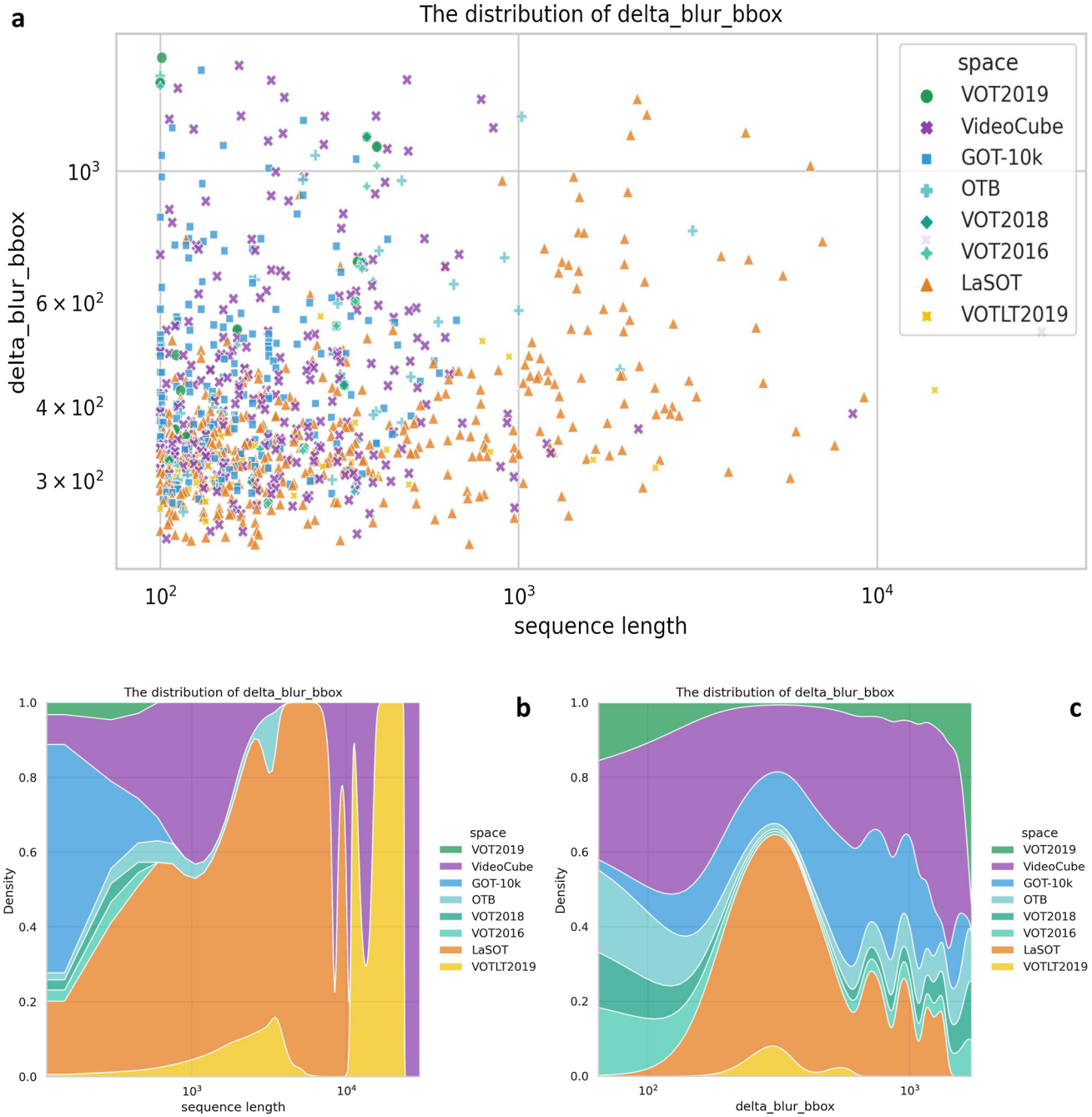


Fig. 44 The composition of delta blur bounding-box. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

H.9 Fast Motion

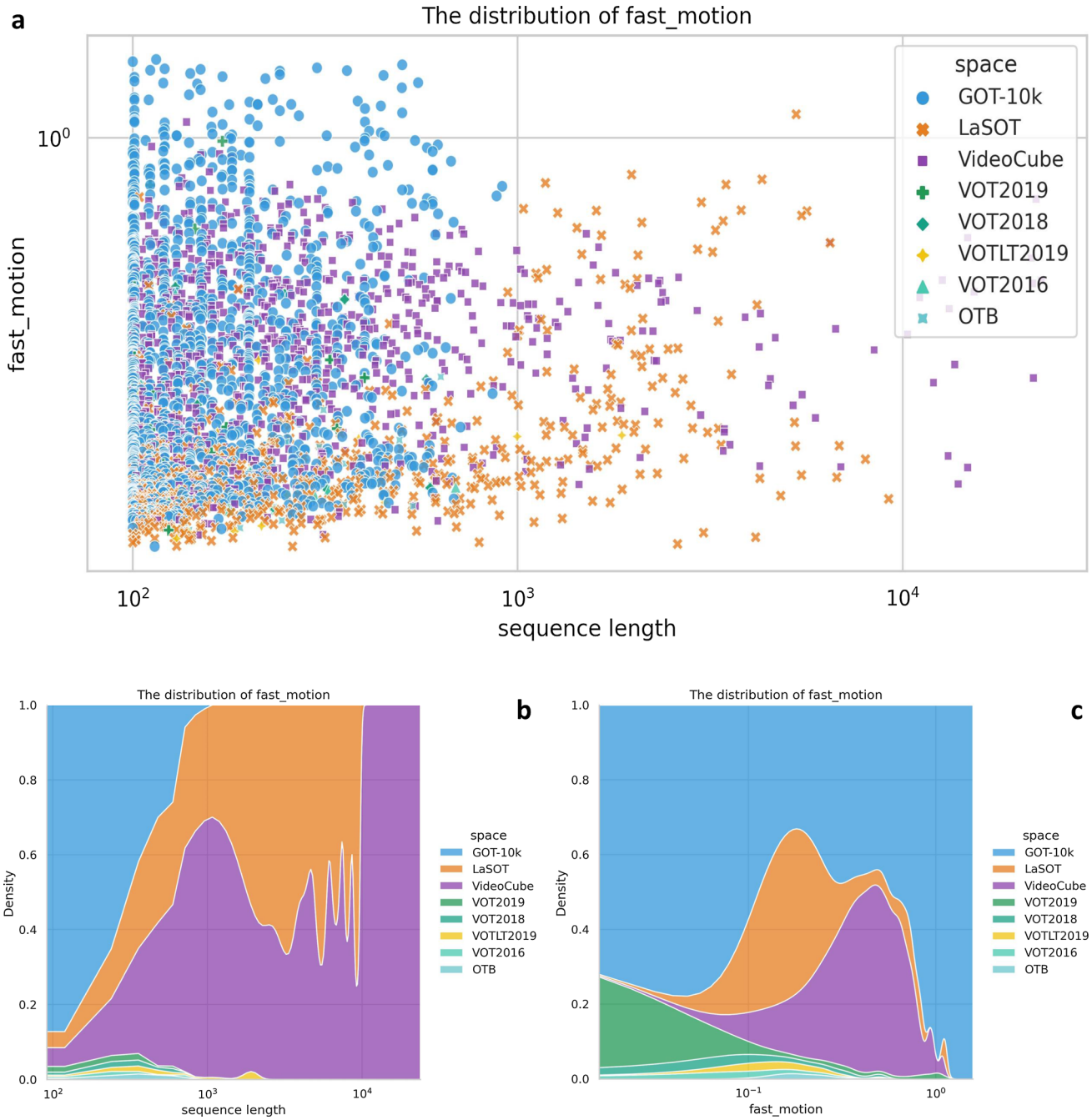


Fig. 45 The composition of fast motion. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

H.10 Low Correlation Coefficient

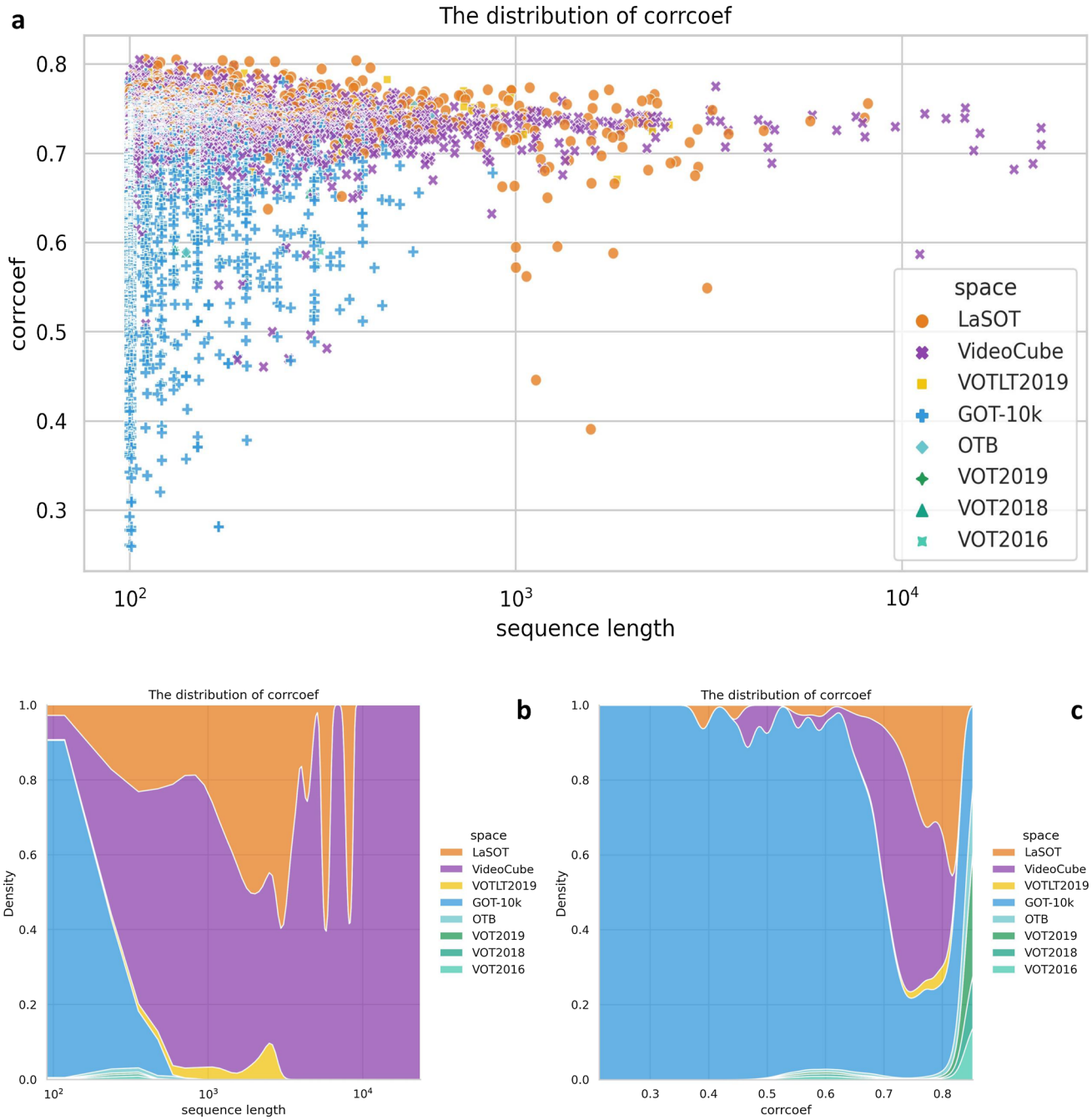
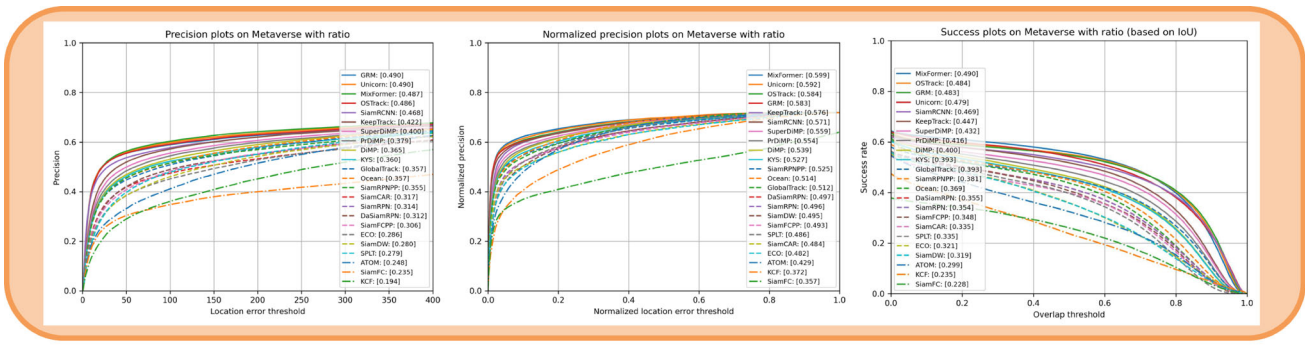


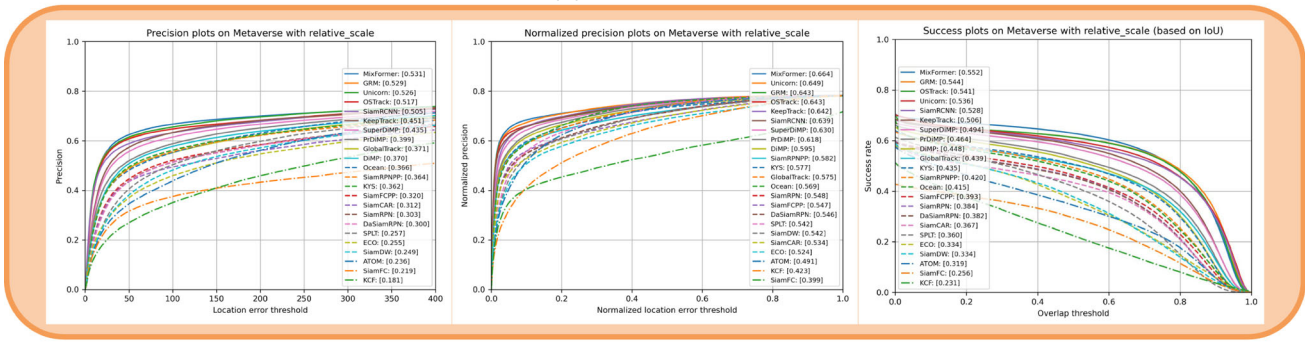
Fig. 46 The composition of low correlation coefficient. **a** The distribution of attribute values and sequence lengths, each point representing a sub-sequence. **b** The distribution of sequence lengths. **c** The distribution of attribute values

Appendix I: Experiments in Challenging Space

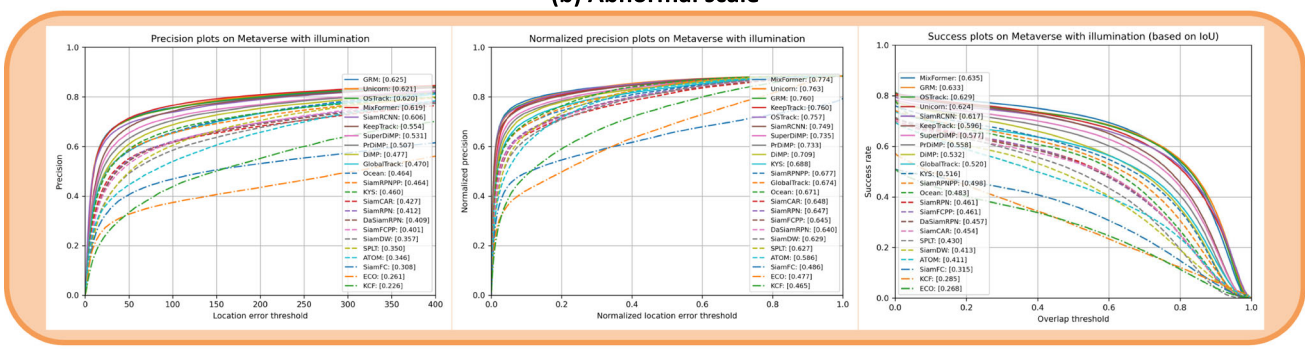
I.1 Static Attributes



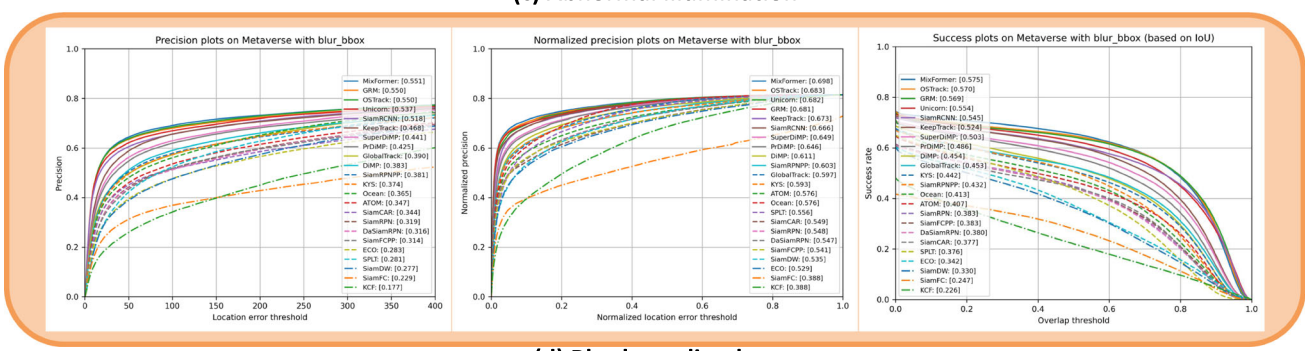
(a) Abnormal ratio



(b) Abnormal scale



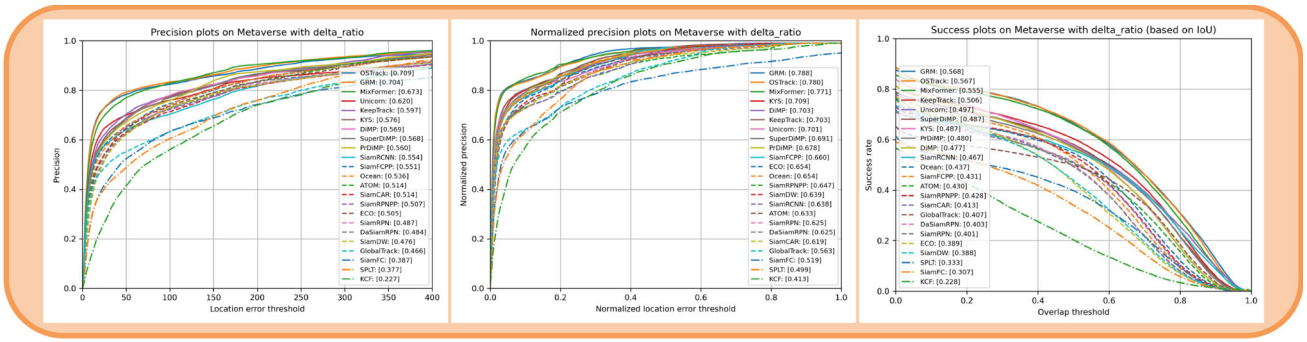
(c) Abnormal illumination



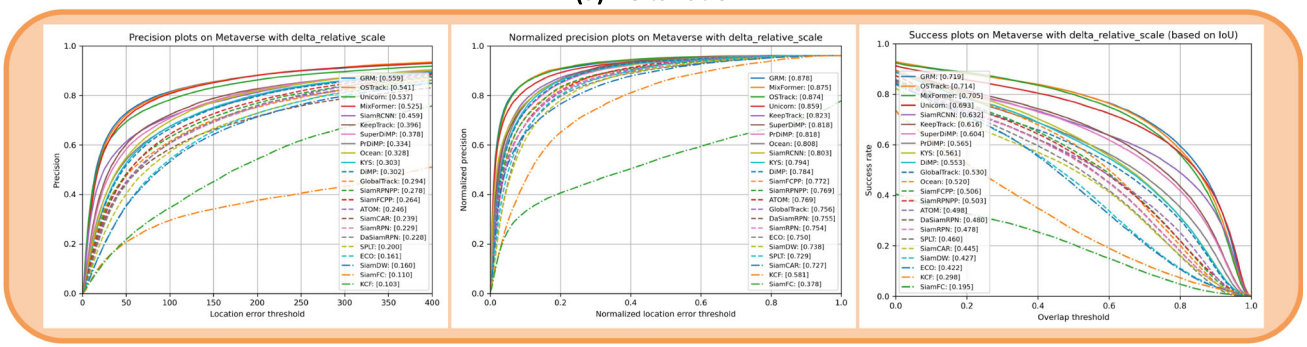
(d) Blur bounding-box

Fig. 47 Experiments in challenging space with static attributes. **a–d** The tracking results in different challenging factors. Each task is evaluated by precision plot, normalized precision plot, and success plot with OPE mechanism

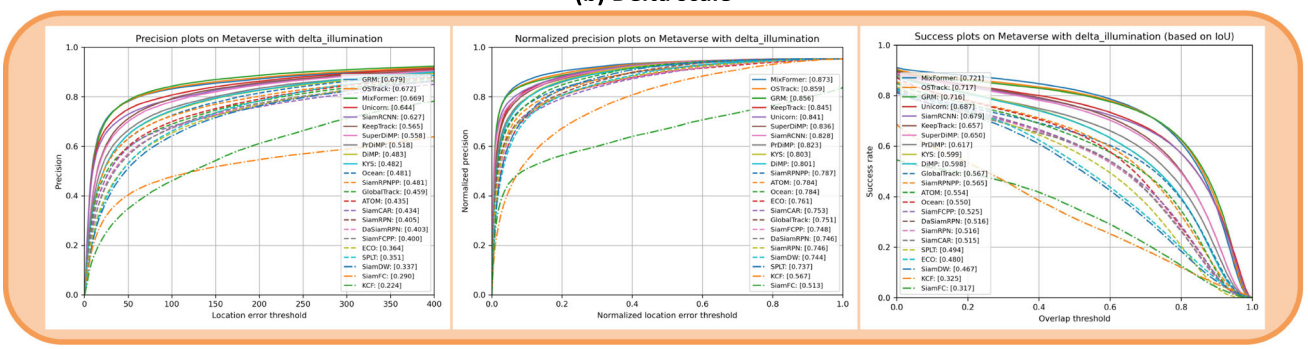
I.2 Dynamic Attributes



(a) Delta ratio

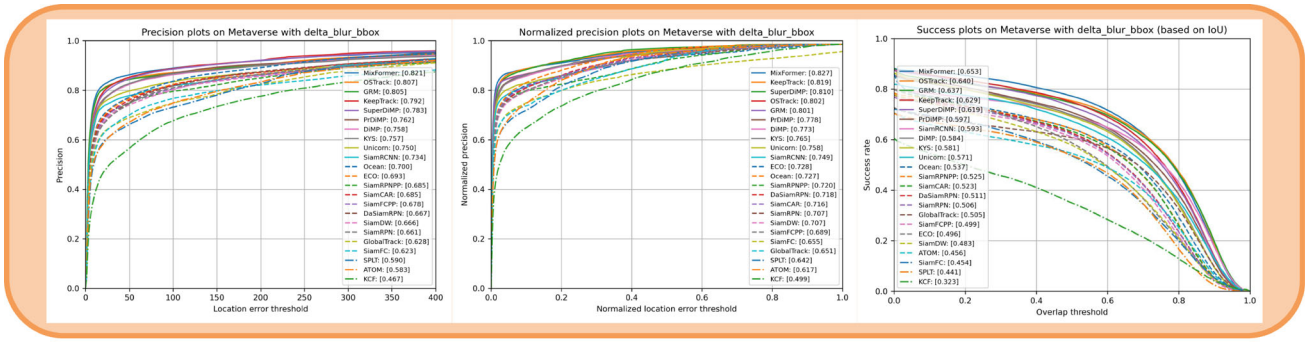


(b) Delta scale

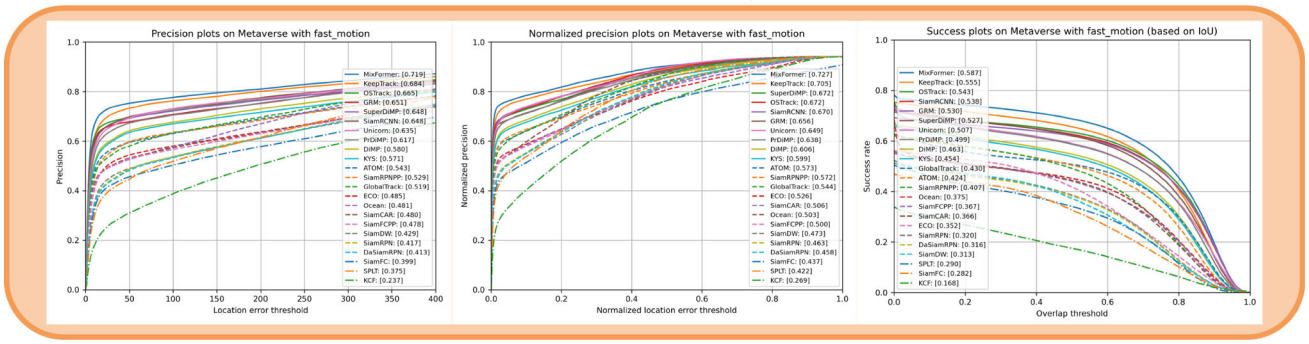


(c) Delta illumination

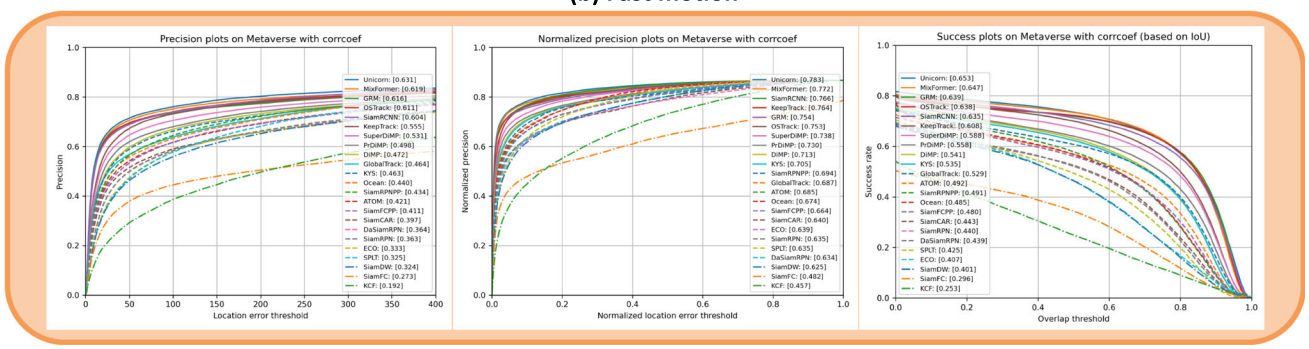
Fig. 48 Experiments in challenging space with dynamic attributes. **a–c** The tracking results in different challenging factors. Each task is evaluated by precision plot, normalized precision plot, and success plot with OPE mechanism



(a) Delta blur bounding-box



(b) Fast motion



(c) Correlation coefficient

Fig. 49 Experiments in challenging space with dynamic attributes. **a–c** The tracking results in different challenging factors. Each task is evaluated by precision plot, normalized precision plot, and success plot with OPE mechanism

References

Abu Alhaija, H., Mustikovela, S. K., Mescheder, L., Geiger, A., & Rother, C. (2018). Augmented reality meets computer vision: Efficient data generation for urban driving scenes. *International Journal of Computer Vision, 126*(9), 961–972.

Beals, R., Mayyasi, A., Templeton, A., & Johnston, W. (1971). The relationship between basketball shooting performance and certain visual attributes. *American Journal of Optometry and Archives of American Academy of Optometry, 48*(7), 585–590.

Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., & Torr, P. H. (2016). Fully-convolutional Siamese networks for object tracking. In *European conference on computer vision* (pp. 850–865). Springer.

Bhat, G., Danelljan, M., Gool, L. V., & Timofte, R. (2019). Learning discriminative model prediction for tracking. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 6182–6191).

Bhat, G., Danelljan, M., Gool, L. V., & Timofte, R. (2020). Know your surroundings: Exploiting scene information for object tracking. In *European conference on computer vision* (pp. 205–221). Springer.

Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review, 94*(2), 115.

Burg, A. (1966). Visual acuity as measured by dynamic and static tests: A comparative evaluation. *Journal of Applied Psychology, 50*(6), 460.

Čehovin, L., Leonardis, A., & Kristan, M. (2016). Visual object tracking performance measures revisited. *IEEE Transactions on Image Processing, 25*(3), 1261–1274.

Chu, Q., Ouyang, W., Li, H., Wang, X., Liu, B., & Yu, N. (2017). Online multi-object tracking using CNN-based single object tracker with spatial-temporal attention mechanism. In *2017 IEEE international conference on computer vision (ICCV)* (pp. 4846–4855). https://doi.org/10.1109/ICCV.2017.518

Ciaparrone, G., Sanchez, F. L., Tabik, S., Troiano, L., Tagliaferri, R., & Herrera, F. (2019). Deep learning in video multi-object tracking: A survey. *Neurocomputing, 381*, 61–88.

Collins, R. T. (2003). Mean-shift blob tracking through scale space. In *Proceedings of the 2003 IEEE computer society conference on computer vision and pattern recognition, 2003* (Vol. 2, p. 234). IEEE.

Collins, R., Zhou, X., & Teh, S. K. (2005). An open source tracking testbed and evaluation web site. In *IEEE international workshop on performance evaluation of tracking and surveillance* (Vol. 2, p. 35).

Cook, D. J. (2012). How smart is your home. *Science, 335*(6076), 1579–1581.

Cui, Y., Jiang, C., Wang, L., & Wu, G. (2022). Mixformer: End-to-end tracking with iterative mixed attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 13608–13618).

Danelljan, M., Bhat, G., Khan, F. S., & Felsberg, M. (2019). Atom: Accurate tracking by overlap maximization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4660–4669).

Danelljan, M., Bhat, G., Shahbaz Khan, F., & Felsberg, M. (2017). Eco: Efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6638–6646).

Danelljan, M., Gool, L. V., & Timofte, R. (2020). Probabilistic regression for visual tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 7183–7192).

Dendorfer, P., Osep, A., Milan, A., Schindler, K., Cremers, D., Reid, I., Roth, S., & Leal-Taixé, L. (2021). MOTChallenge: A benchmark for single-camera multiple target tracking. *International Journal of Computer Vision, 129*(4), 845–881.

Dunnhofer, M., Furnari, A., Farinella, G. M., & Micheloni, C. (2023). Visual object tracking in first person vision. *International Journal of Computer Vision, 131*(1), 259–283.

Dupeyroux, J., Serres, J. R., & Viollet, S. (2019). AntBot: A six-legged walking robot able to home like desert ants in outdoor environments. *Science Robotics, 4*(27), eaau0307.

Esteva, A., Chou, K., Yeung, S., Naik, N., Madani, A., Mottaghi, A., Liu, Y., Topol, E., Dean, J., & Socher, R. (2021). Deep learning-enabled medical computer vision. *NPJ Digital Medicine, 4*(1), 5.

Fan, H., Bai, H., Lin, L., Yang, F., Chu, P., Deng, G., Yu, S., Huang, M., Liu, J., & Xu, Y. (2021). LaSOT: A high-quality large-scale single object tracking benchmark. *International Journal of Computer Vision, 129*(2), 439–461.

Ferryman, J., & Shahrokni, A. (2009). PETS2009: Dataset and challenge. In *2009 twelfth IEEE international workshop on performance evaluation of tracking and surveillance* (pp. 1–6). IEEE.

Finlayson, G. D., & Trezzi, E. (2004). Shades of gray and colour constancy. In *The twelfth color imaging conference 2004* (pp. 37–41).

Fisher, R. B. (2004). The PETS04 surveillance ground-truth data sets. In *Proceedings of the 6th IEEE international workshop on performance evaluation of tracking and surveillance* (pp. 1–5).

Gao, S., Zhou, C., & Zhang, J. (2023). Generalized relation modeling for transformer tracking. arXiv preprint arXiv:2303.16580

Gauglitz, S., Höllerer, T., & Turk, M. (2011). Evaluation of interest point detectors and feature descriptors for visual tracking. *International Journal of Computer Vision, 94*(3), 335–360.

Geuther, B. Q., Deats, S. P., Fox, K. J., Murray, S. A., Braun, R. E., White, J. K., Chesler, E. J., Lutz, C. M., & Kumar, V. (2019). Robust mouse tracking in complex environments using neural networks. *Communications Biology, 2*(1), 124.

Godec, M., Roth, P. M., & Bischof, H. (2013). Hough-based tracking of non-rigid objects. *Computer Vision and Image Understanding, 117*(10), 1245–1256.

Guo, D., Wang, J., Cui, Y., Wang, Z., & Chen, S. (2020). SiamCAR: Siamese fully convolutional classification and regression for visual tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 6269–6277).

Han, B., Comaniciu, D., Zhu, Y., & Davis, L. S. (2008). Sequential kernel density approximation and its application to real-time visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30*(7), 1186–1197.

Held, D., Guillory, D., Rebsamen, B., Thrun, S., & Savarese, S. (2016). A probabilistic framework for real-time 3D segmentation using spatial, temporal, and semantic cues. https://doi.org/10.15607/RSS.2016.XII.024

Henriques, J. F., Caseiro, R., Martins, P., & Batista, J. (2014). High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(3), 583–596.

Huang, L., Zhao, X., & Huang, K. (2020). GlobalTrack: A simple and strong baseline for long-term tracking. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, pp. 11037–11044).

Huang, L., Zhao, X., & Huang, K. (2021). GOT-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 43*(5), 1562–1577. https://doi.org/10.1109/TPAMI.2019.2957464

Hu, S., Zhao, X., Huang, L., & Huang, K. (2023). Global instance tracking: Locating target more like humans. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 45*(1), 576–592. https://doi.org/10.1109/TPAMI.2022.3153312

Kiani Galoogahi, H., Fagg, A., Huang, C., Ramanan, D., & Lucey, S. (2017). Need for speed: A benchmark for higher frame rate object tracking. In *Proceedings of the IEEE international conference on computer vision* (pp. 1125–1134).

Kim, J., Misu, T., Chen, Y.-T., Tawari, A., & Canny, J. (2019). Grounding human-to-vehicle advice for self-driving vehicles. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10591–10599).

Kohl, P., Coffey, B., Reichow, A., Thompson, W., & Willer, P. (1991). A comparative study of visual performance in jet fighter pilots and non-pilots. *Journal of Behavioral Optometry, 5*(2), 123–126.

Kong, Y., & Fu, Y. (2022). Human action recognition and prediction: A survey. *International Journal of Computer Vision, 130*(5), 1366–1401.

Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Čehovin Zajc, L., Vojir, T., Bhat, G., Lukezic, A., Eldesokey, A. (2018). The sixth visual object tracking vot2018 challenge results. In *Proceedings of the European conference on computer vision (ECCV) workshops*.

Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Čehovin, L., Vojir, et al. (2017). The visual object tracking VOT2017 challenge results, 1949–1972. https://doi.org/10.1109/ICCVW.2017.230

Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Kämäräinen, J.-K., Danelljan, M., Zajc, L. Č., Lukežič, A., & Drbohlav, O. (2020). The eighth visual object tracking vot2020 challenge results. In *European conference on computer vision* (pp. 547–601). Springer.

Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Pflugfelder, R., Kamarainen, J.-K., Čehovin Zajc, L., Drbohlav, O., Lukezic, A., & Berg, A. (2019). The seventh visual object tracking vot2019 challenge results. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*.

Kristan, M., Pflugfelder, R., Leonardis, A., Matas, J., Porikli, F., Cehovin, L., Nebehay, G., Fernandez, G., Vojir, T., Gatt, A., Khajenezhad, A., Salahledin, A., Soltani-Farani, A., Zarezade, A., Petrosino, A., Milton, A., Bozorgtabar, B., Li, B., Chan, C. S., Heng, C., Ward, D., Kearney, D., Monekosso, D., Karaimer, H. C., Rabiee, H. R., Zhu, J., Gao, J., Xiao, J., Zhang, J., Xing, J., Huang, K., Lebeda, K., Cao, L., Maresca, M.E., Lim, M. K., El Helw, M., Felsberg, M., Remagnino, P., Bowden, R., Goecke, R., Stolkin,

R., Lim, S.Y., Maher, S., Poullot, S., Wong, S., Satoh, S., Chen, W., Hu, W., Zhang, X., Li, Y., & Niu, Z. (2013). The visual object tracking vot2013 challenge results. In *2013 IEEE international conference on computer vision workshops* (pp. 98–111). https://doi.org/10.1109/ICCVW.2013.20

Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Zajc, L. Č, et al. (2016). *The visual object tracking VOT2016 challenge results*. Springer.

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

Kwon, J., & Lee, K. M. (2009). Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping Monte Carlo sampling. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 1208–1215). IEEE.

Land, M. F., & McLeod, P. (2000). From eye movements to actions: How batsmen hit the ball. *Nature Neuroscience, 3*(12), 1340–1345.

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature, 401*(6755), 788–791.

Li, B., Wu, W., Wang, Q., Zhang, F., Xing, J., & Yan, J. (2019). Siamrpn++: Evolution of Siamese visual tracking with very deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4282–4291).

Li, B., Yan, J., Wu, W., Zhu, Z., & Hu, X. (2018). High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8971–8980).

Liang, P., Blasch, E., & Ling, H. (2015). Encoding color information for visual tracking: Algorithms and benchmark. *IEEE Transactions on Image Processing, 24*(12), 5630–5644. https://doi.org/10.1109/TIP.2015.2482905

Li, A., Lin, M., Wu, Y., Yang, M.-H., & Yan, S. (2015). NUS-PRO: A new visual tracking challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 38*(2), 335–349.

Liu, Q., He, Z., Li, X., & Zheng, Y. (2019). PTB-TIR: A thermal infrared pedestrian tracking benchmark. *IEEE Transactions on Multimedia, 22*(3), 666–675.

Lukežič, A., Zajc, L. Č, Vojíř, T., Matas, J., & Kristan, M. (2020). Performance evaluation methodology for long-term single-object tracking. *IEEE Transactions on Cybernetics, 51*, 6305–6318.

M, J. W. (1962). The effect of relative motion on visual acuity. *Survey of Ophthalmology, 7*, 83–116.

Mayer, C., Danelljan, M., Paudel, D. P., & Van Gool, L. (2021). Learning target candidate association to keep track of what not to track. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 13444–13454).

McLeod, P., Reed, N., & Dienes, Z. (2003). How fielders arrive in time to catch the ball. *Nature, 426*(6964), 244–245.

Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM, 38*(11), 39–41.

Moudgil, A., & Gandhi, V. (2018). Long-term visual object tracking benchmark. In *Asian conference on computer vision* (pp. 629–645).

Mueller, M., Smith, N., & Ghanem, B. (2016). A benchmark and simulator for UAV tracking. In *European conference on computer vision* (pp. 445–461). Springer.

Muller, M., Bibi, A., Giancola, S., Alsubaihi, S., & Ghanem, B. (2018). TrackingNet: A large-scale dataset and benchmark for object tracking in the wild. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 300–317).

Nejhum, S. S., Ho, J., & Yang, M.-H. (2008). Visual tracking with histograms and articulating blocks. In *2008 IEEE conference on computer vision and pattern recognition* (pp. 1–8). IEEE.

Pech-Pacheco, J.L., Cristobal, G., Chamorro-Martinez, & J., Fernandez-Valdivia, J. (2000). Diatom autofocusing in brightfield microscopy: A comparative study. In *Proceedings 15th international conference on pattern recognition*. ICPR-2000 (Vol. 3, pp. 314–317).

Ramakrishnan, S. K., Jayaraman, D., & Grauman, K. (2021). An exploration of embodied visual exploration. *International Journal of Computer Vision, 129*(5), 1616–1649.

Real, E., Shlens, J., Mazzocchi, S., Pan, X., & Vanhoucke, V. (2017). Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5296–5305).

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., & Bernstein, M. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision, 115*(3), 211–252.

Team, O. E. L., Stooke, A., Mahajan, A., Barros, C., Deck, C., Bauer, J., Sygnowski, J., Trebacz, M., Jaderberg, M., Mathieu, M., et al. (2021). Open-ended learning leads to generally capable agents. arXiv preprint arXiv:2107.12808

Valmadre, J., Bertinetto, L., Henriques, J. F., Tao, R., Vedaldi, A., Smeulders, A. W., Torr, P. H., & Gavves, E. (2018). Long-term tracking in the wild: A benchmark. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 670–685).

Voigtlaender, P., Luiten, J., Torr, P. H., & Leibe, B. (2020). Siam R-CNN: Visual tracking by re-detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 6578–6588).

Wang, S., Zhou, Y., Yan, J., & Deng, Z. (2018). Fully motion-aware network for video object detection. In V. Ferrari, M. Hebert, C. Sminchisescu, & Y. Weiss (Eds.), *Computer Vision—ECCV 2018* (pp. 557–573). Springer.

Wu, Y., Lim, J., & Yang, M.-H. (2013). Online object tracking: A benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2411–2418).

Wu, Y., Lim, J., & Yang, M.-H. (2015). Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(09), 1834–1848.

Xu, Y., Wang, Z., Li, Z., Yuan, Y., & Yu, G. (2020). Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, pp. 12549–12556).

Yan, B., Jiang, Y., Sun, P., Wang, D., Yuan, Z., Luo, P., & Lu, H. (2022). Towards grand unification of object tracking. In *Computer vision—ECCV 2022: 17th European conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXI* (pp. 733–751). Springer.

Yan, B., Zhao, H., Wang, D., Lu, H., & Yang, X. (2019). 'Skimming-perusal' tracking: A framework for real-time and robust long-term tracking. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 2385–2393).

Ye, B., Chang, H., Ma, B., Shan, S., & Chen, X. (2022). Joint feature learning and relation modeling for tracking: A one-stream framework. In *Computer vision—ECCV 2022: 17th European conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII* (pp. 341–357). Springer.

Yoon, J. H., Lee, C.-R., Yang, M.-H., & Yoon, K.-J. (2019). Structural constraint data association for online multi-object tracking. *International Journal of Computer Vision, 127*(1), 1–21.

Zhang, Z., & Peng, H. (2019). Deeper and wider Siamese networks for real-time visual tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4591–4600).

Zhang, G., & Vela, P. A. (2015). Good features to track for visual slam. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1373–1382).

Zhang, Z., Peng, H., Fu, J., Li, B., & Hu, W. (2020). Ocean: Object-aware anchor-free tracking. In *European conference on computer vision* (pp. 771–787). Springer.

Zhu, Z., Wang, Q., Li, B., Wu, W., Yan, J., Hu, W. (2018). Distractor-aware siamese networks for visual object tracking. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 101–117).