Exploring 3 R's of Long-term Tracking: Re-detection, Recovery and Reliability

Shyamgopal Karthik Abhinav Moudgil Vineet Gandhi
Center for Visual Information Technology, Kohli Center on Intelligent Systems
IIIT-Hyderabad, India
{shyamgopal.karthik@research., abhinav.moudgil@research., vgandhi@}iiit.ac.in

Abstract

Recent works have proposed several long term tracking benchmarks and highlight the importance of moving towards long-duration tracking to bridge the gap with application requirements. The current evaluation methodologies, however, do not focus on several aspects that are crucial in a long term perspective like Re-detection, Recovery, and Reliability. In this paper, we propose novel evaluation strategies for a more in-depth analysis of trackers from a long-term perspective. More specifically, (a) we test re-detection capability of the trackers in the wild by simulating virtual cuts, (b) we investigate the role of chance in the recovery of tracker after failure and (c) we propose a novel metric allowing visual inference on the ability of a tracker to track contiguously (without any failure) at a given accuracy. We present several original insights derived from an extensive set of quantitative and qualitative experiments.

1. Introduction

Visual tracking is a fundamental problem in computer vision and has rapidly progressed in the recent past with the onset of deep learning. However, progress is still far from matching practitioner needs, which demands consistent and reliable long-duration tracking. Interestingly, most existing works evaluate their performance on datasets consisting of multiple short clips. For instance, the most commonly used OTB dataset has an average length of about 20 seconds [34] per clip. Work by Moudgil and Gandhi [23] observed a sharp performance drop when the trackers were evaluated on long sequences. Following works [10, 32, 21] also make similar observations and suggest that we need alternate ways to evaluate and analyze long term tracking performance.

Based on these works [23, 10, 32, 21] we hypothesize that three properties are crucial for an improved long term tracking performance. First is the ability to re-detect the target if it is lost. *Re-detection* is crucial to handle situations where the target object goes out of the frame and reappears. It is also essential to re-initiate tracking when the target object is lost due to occlusions or momentary tracking failures. The second key aspect is the ability of the tracker to distinguish between the actual target and distractor or background clutter. This aspect is vital for consistency in tracking as well as for *recovery* from failures. Figure 1 illustrates an example where chance plays a crucial role in recovery. We believe that scrutinizing the nature of failures and recoveries will aid improved tracking performance. The third key aspect is *Reliability*, which connects to the ability for consistent and contiguous tracking. Contiguity suggests the ability of the tracker to track for a long duration without any failure. Consistency indicates the accuracy of tracking over time. Tracking in the long-duration video allows us to study factors like a slow accumulation of error which is difficult to observe in short sequences. Several applications like video surveillance or virtual camera simulation from static camera [12] require precise tracking for long time. Surprisingly, none of the current evaluation strategies focus on these three

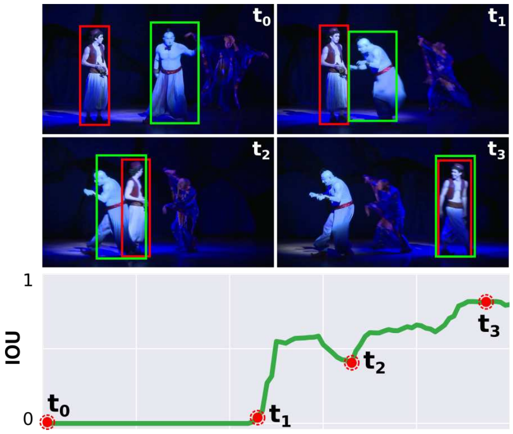

Figure 1. A typical example of a chance based recovery in Alladin sequence from TLP [23] dataset. SiamRPN (green) is tracking the incorrect object and has zero overlap with the target (red) in the start. It switches to tracking the target when they pass through each other. We study such chance based recoveries in long-term setting both qualitatively and quantitatively. Best viewed in colour.

crucial aspects of *Re-detection*, *Recovery*, and *Reliability*.

For instance, the most prevalent metrics are Success and Precision plots, which measure the number of frames with Intersection Over Union (IoU) greater than a threshold and the mean distance from the center of the ground truth respectively. Both these metrics do not reflect anything specific about the 3R's. Recent work by Lukezic *et al*. [21] studied the efficacy of the search region expansion strategy of different trackers. However, the evaluation is performed in a synthetic experimental setup (designed by padding with gray values) and may not be an indicator of performance in real-world scenarios. Valmadre *et al*. [32] improves the evaluation strategy by explicitly handling the cases where the target is not visible/absent from the frame. Other recent efforts [23, 10] identify the aforementioned key issues, but they do not provide any way to evaluate these properties comprehensively.

In this work, we propose two novel evaluation metrics focused on the re-detection ability and the aspect of continuous and consistent long term tracking. Furthermore, we present more in-depth insights into the failure and recovery of different trackers, explicitly addressing the role of distractors. Since shorter sequences are inappropriate to address these concerns, we use the Track Long and Prosper (TLP) [23] dataset for the experiments. The main advantage of TLP is that the average sequence length is longest among other densely annotated datasets [14, 10, 21]. Long duration videos present cases of multiple failures and recoveries for each video, which allows for a deeper analysis. Our contributions include:

1. We propose a novel way to quantitatively evaluate the re-detection abilities of a tracker by simulating cuts (an abrupt transition from a frame to another) in original videos. We propose a method to search challenging locations for placing the cut by maximizing the distance between the ground truth bounding boxes in the frame before and after the cut. Different trackers are then evaluated on their ability to recover/re-detect, and the time they take to recover.

2. We formally study the chance factor in recoveries post-failure. We analyze the role of distractors in failures and recoveries and the co-incidences which aid tracking. For example, it often happens in long sequences that tracker loses the target at some location and freezes there. If by chance the target passes the same location (after a while), the tracker starts tracking it again. Our study aims to quantify such behavior.

3. We propose 3D Longest Subsequence Measure (3D-LSM), as a novel metric for quantifying tracking performance. It measures the longest contiguous sequence successfully tracked at a given precision and allowed failure tolerance. The 3D-LSM allows for a direct visual interpretation of tracking results in the form of a 2D image.

2. Related Work

Tracking Datasets: There are a large variety of datasets for object tracking. Most commonly used datasets are OTB50[34] and OTB100[33]. They consist of short videos from generic real-world scenarios. ALOV300[30] increased diversity by including 300 short sequences. The average video length in ALOV dataset is only 9 seconds. NFS[11] dataset included sequences recorded at high frame rate (240fps). UAV[24] introduced a dataset from sequences shot from an aerial vehicle. Moudgil and Gandhi[23] TLP dataset with 50 sequences, focusing on long-duration tracking (significantly increasing length of individual sequences). LTB35[21] and OxUvA[32] then followed emphasizing the need to focus on long term tracking. LaSOT[10] significantly increased the size of the dataset with over 1000 sequences. GOT10k[14] then followed by proposing a dataset with 10000 sequences, including objects from 563 different classes.

Tracking Methods: We list some notable attempts which led to significant progress in long term tracking. Collins *et al*. [6] proposed the idea of using the neighborhood around the ground truth for discriminative feature learning. This idea was later formalized into tracking by detection frameworks [1]. Kalal *et al*. [15] proposed TLD framework of learning detector from initial tracking, maintaining the confidence of local tracking based on feature point tracks, and switching to detection in low confidence scenarios. TLD tracker was one of the first attempts to elegantly handle the re-detection problem, which is crucial for long term tracking. The consistency aspect was then improved by employing an ensemble of classifiers [36]. These methods maintain several weak classifiers, often initiated at different checkpoints to account for appearance variations of the target.

Another popular direction is Discriminative Correlation Filter (DCF) based tracking [4, 9]. These methods exploit the properties of circular correlation (efficiently performed in Fourier domain) for training a regressor in a sliding-window fashion. Recent progress in DCF's is driven by integrating multi-resolution shallow and deep features maps to learn the correlation filters [8, 3, 31]. Another fundamental contribution is the use of Siamese networks for visual object tracking [2, 13]. The GOTURN tracker [13] uses the Siamese architecture to directly regress the bounding box locations given two cropped images from previous and current frames. The SiamFC tracker [2] transforms the exemplar image and the large search image using the same function and outputs a map by computing similarity in the transformed domain. These efforts [2, 13] do not include any on-

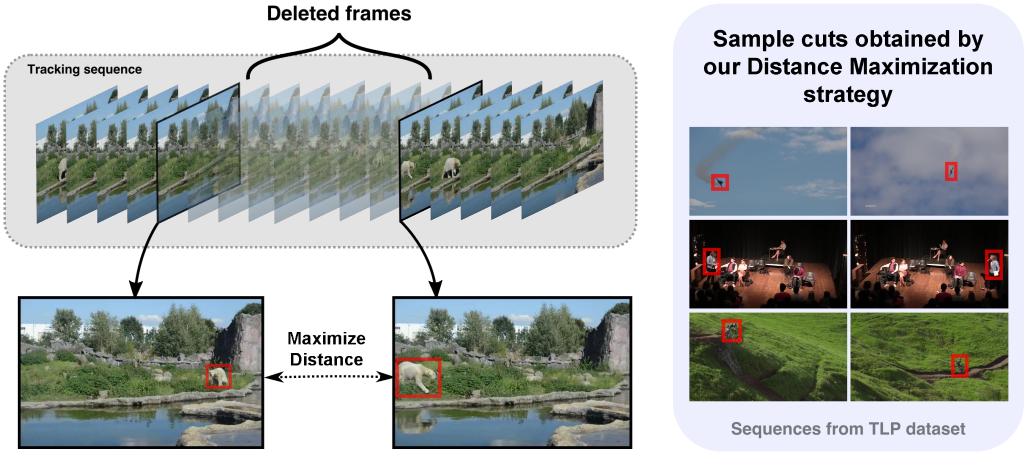
Deleted frames
Tracking sequence
Maximize Distance
Sample cuts obtained by our Distance Maximization strategy
Sequences from TLP dataset

Figure 2. A cut is introduced by removing a set of contiguous frames from a tracking sequence. This introduces a sudden change of position of the ground truth object as shown in the left diagram. The red bounding box shows the position of the target object, before and after the cut. We maximize the amount of target shift by minimizing the distance between these bounding boxes. We evaluate the trackers ability to re-detect the object after the cut. Few more examples from TLP dataset with simulated cuts are shown on the right.

line updates and are extremely efficient in terms of computation. The Siamese framework was further augmented by employing Region Proposal Networks (RPN)[18, 19] which significantly improves the accurate prediction of scale and aspect ratio of the bounding boxes.

Another pioneering effort came from Nam *et al.* [25], who introduced the idea of treating the tracking problem as classifying candidate windows sampled around the previous target region. Several recent efforts have explored the variations of the Tracking Learning Detection (TLD) framework. Nebehay *et al.* [26] proposed a mechanism to drift by filtering outlier correspondences. A combination of short term CF tracker with additional components (e.g., an explicit re-detection module) have been explored [20, 22]. Zhang *et al.* [37] employed an offline trained regression network as the short-term component and an online-trained verification network to detect tracking failure and start image wide detection. Yan *et al.* [35] show significant computational improvements by replacing the online verification network with an offline trained Siamese verification network.

Tracking Metrics: Early works relied on the precision metric[1, 34] for quantifying the tracking performance, which computes the pixel distance between the center of the ground truth and the prediction. This was convenient since it required only annotating the center of the target and not the whole bounding box. However, since this does not account for the scale and aspect ratio, the success metric[34] was introduced. It measures the percentage of frames where the Intersection Over Union (IoU) of the predicted and ground truth bounding boxes is more than a threshold. Failure rate [16] was then introduced to address the continuity and consistency aspect of tracking. In failure rate measure, a manual operator reinitializes the tracker upon every failure. The number of required manual interventions per frame is recorded as the quantitative measure. However, due to the need of manual interventions, it is unscalable for long sequences (in large datasets). For a more detailed review and analysis of metrics for short-term tracking, we would refer the reader to work by Cehovin *et al.* [5].

A few evaluation metrics have been proposed targeting long-duration tracking. Valmadre *et al.* [32] introduced True Positive Rate(TPR), True Negative Rate(TNR) and took their geometric mean. To have a single representative metric accounting for the trackers which do not predict absent labels, they proposed a modified metric called maximum geometric mean metric. However, the metric is biased towards the ability of a tracker to predict absent labels.

Lukezic *et al.* [21] introduced tracking recall and precision and used this to give a tracking F1 score. However, their definition of a long term tracker is limited to the ability of a tracker to predict absence, and the proposed metric does not focus on the continuity and consistency aspect of tracking. We believe the ability to track for long-duration consistently even when the target object is always present has been overlooked in these previous efforts [21, 32]. Lukezic *et al.* also proposed an experiment to quantify the re-detection ability of a tracker. However, their experiment mainly focuses on the search strategy with no appearance changes. Here, we seek to quantify the re-detection ability in the wild. Moudgil and Gandhi [23] proposed the Longest Subsequence Measure (LSM), which quantifies the longest contiguous segment successfully tracked in the sequence. Here, we propose an extension of it called 3D-LSM, which allows comparing trackers visually.

3. Re-detection in the Wild

This experiment is designed to quantify a tracker's ability to re-detect the object after it is lost (either because the target goes of the view or due to momentary failures).

Setup: We select a segment from a sequence, and delete it, thereby introducing a cut (illustrated in Figure 2). We evaluate the tracker's performance on the segment after the cut to evaluate the re-detection ability of the tracker. For each sequence from the TLP dataset, we cut a segment that maximizes the L2 norm of the center locations between the target bounding boxes before and after the cut. The duration of the cut is fixed to 300 frames. We empirically find that 300 frames allow the target to move far away from the tracker's search region without significantly varying the other aspects in the scene. Keeping a similar context around the target helps to keep the focus on the re-detection ability (the context can change dramatically in long sequences if the length of the omitted sequence is large). The proposed re-detection scheme is quite general and can be applied even on datasets that do not have target disappearances at all.

Evaluation: In all the experiments the tracker is initialized 100 frames before the cut. We choose 100 frames so that the tracker starts stable tracking before the cut. It also allows trackers with online updates to build a reasonable representation of the target object. We also make sure that there are no critical challenges in this duration of 100 frames such as heavy occlusion, clutter, etc. to avoid tracker failure in these 100 frames. After the cut, the tracker is continued to run on the sequence for another 200 frames and its performance on this segment is evaluated. We define "recovery" when the IoU of the tracker with the target reaches 0.5. To make a relative comparison of the trackers on the re-detection task, we report the following metrics.

1. Total number of sequences (out of the total 50 TLP sequences) in which a tracker is able to recover within the remaining 200 frames.

2. Total number of sequences where the recovery is "quick," i.e., the recovery happens within 30 frames (1 second).

3. Average number of frames a tracker takes to recover successfully.

Tracker	Quick recoveries ↑	Total recoveries ↑	Avg. recovery length (# frames) ↓
SPLT [35]	**20**	**36**	19
FuCoLoT [20]	10	33	55
MBMD [37]	15	27	28
ATOM [7]	12	25	34
CMT [26]	9	14	22
TLD [15]	6	10	**8**
MDNet [25]	5	13	48
ECO [8]	4	7	28
SiamRPN [19]	2	7	39
LCT [22]	2	7	143

Table 1. Results for the re-detection experiment (out of 50 sequences).

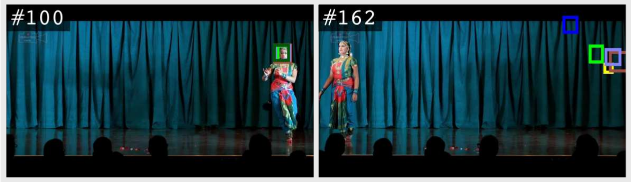

Figure 3. The figure illustrates a simulated cut in the Bharatanatyam sequence from TLP dataset. The cut can be seen as a representation of a situation where the performer exits the stage and enters from another end. None of the 6 trackers was able to recover in this sequence, even with the exact same background and a single target object.

We perform this experiment on TLP dataset with the following trackers: SPLT [35], MBMD [37], FuCoLoT [20], ATOM [7], MDNet [25], SiamRPN [19], ECO [8], CMT [26], LCT [22], and TLD [15]. SPLT, MBMD, FuCoLoT, CMT, LCT and TLD are long-term trackers with explicit re-detection ability; ATOM is the current top performing tracker on the long-term benchmark LaSOT, while MDNet, SiamRPN and ECO are the top performing trackers on other benchmarks [23, 33, 17]. This selection presents all the prevalent tracking approaches: correlation filter based trackers [8, 22, 20], end to end classification with online updates [25], offline trained Siamese trackers with region proposals [19], low level feature tracking with online learned detector [15, 26] and combination of multiple offline/online trained components [7, 37, 35]. The same set of trackers are used in all the following experiments as well.

Results and Discussion: Our results are summarized in Table 1. SPLT gives the best results, followed by FuCoLoT and MBMD. Since the base framework of SPLT and MBMD is the same as SiamRPN, the significant improvements (from SiamRPN to SPLT) can be attributed to the additional verification and re-detection module. An explicit re-detection module also improves CF-based trackers (as seen in FuCoLoT). CMT and TLD dominate in re-detection experiments studied in previous works [21]; however, they give poor results in our experiment. We empirically observe that CMT and TLD fail to adapt to appearance changes that occur before and after the cut, possibly because of the weak appearance model used in the detector. Adapting to appearance changes during re-detection is essential in real-world settings and previous synthetically designed experiments [21] do not account for this aspect. Other trackers like ECO, MDNet, and SiamRPN are limited by their search range and only recover if the target object

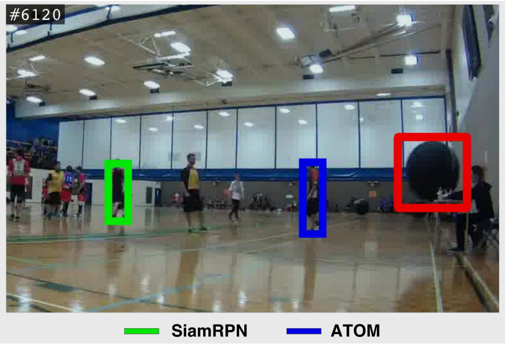

Figure 4. An example from TLP [23] Kinball1 sequence where the tracking target is black ball. Both SiamRPN and ATOM end up tracking objects of totally different class i.e. human which is also significantly different in appearance from the given target.

comes within their search range after the cut. ATOM, on the other hand, uses a larger search area (25 times the area of target object bounding box) and hence recovers more often. Qualitatively, we observe sequences with background clutter or with distractors prove to be the most challenging for all the trackers. Re-detection results are also poor on targets that are small in size.

4. Recovery by Chance

In this section, we investigate the role of chance in tracker recovery post-failure. Interestingly, most of the evaluation metrics [21, 17, 33] do not take this into account, and we believe that to design better long-term trackers, it is essential to scrutinize the nature of recovery. More specifically, we analyze two scenarios that frequently occur in long sequences (a) the tracker starts tracking an alternate object and recovers back when it interacts with the target and (b) tracker freezes somewhere in the background and resumes tracking when the target passes through it.

4.1. Recovery by Tracking Alternate Object

We first investigate the cases when the recovery occurs while tracking an alternate object (distractor). We consider distractors of both the same class as well as other classes. The recovery here occurs only because of the interactions between the objects in the scene. An example of this kind of recovery is illustrated in Fig 1.

However, directly evaluating the role of distractors is challenging because single object tracking benchmarks [33, 23, 17] do not have annotations for multiple objects. We exploit the effectiveness of modern object detectors to resolve this concern. While an object detector would not be accurate enough to be treated as ground truth for bounding boxes for alternate objects, it would still allow us to draw useful insights. Moreover, the results may vary when a different object detector is being used. Hence, the evaluation presented in this section is not intended to serve as a metric. Nonetheless, it presents important insights into the role of distractors in tracking performance, which is further highlighted by qualitative results presented in the supplementary material.

We select 16 out of the 50 sequences from the TLP dataset where distractors are present, and the target interacts with them. We run YOLOv3 [28, 27] on these sequences to obtain all object annotations. We compute and study the following aspects:

- The percentage of frames where the tracker is tracking (IoU ≥ 0.5) an alternate object and has zero overlap with the target (averaged over the selected 16 sequences).

- The recoveries that occur while the tracker is tracking an alternate object (IoU with alternate object ≥ 0.5). We define recovery if the IoU with the ground truth becomes nonzero and maintains a non zero value for the next 60 frames. We present the number of recoveries per sequence for each tracker.

- The performance drop that occurs if we zero out the performance after the first instance of such a recovery.

Results and Discussion: The results are shown in Table 2. SPLT, MBMD, ATOM, and SiamRPN track an incorrect object for more than 13% of the frames on average in a sequence, which is an exceedingly high number. The behavior possibly stems from the nature of their design which looks for "objectness" i.e., the potential bounding boxes in the neighborhood. SPLT and ATOM despite employing hard negative mining strategies while training are prone to tracking alternate objects. Most trackers are highly susceptible to intraclass variations like the color, pose, clothing, etc. and keep on confusing on cases like two boxers in a ring or two nearby cars on a highway. The confusion among different classes is also observed (Fig 4). Interestingly, the trackers which perform online model updates (MDNet, FuColoT, ECO, CMT and TLD) are less susceptible to track an alternate object.

In the last two columns of Table 2 we present the success metric of the listed trackers on the selected 16 sequences and the reduced performance computed by setting the IoU scores to zero after the first chance-based recovery. The reduced performance is indicative of the worst-case performance, i.e., if a chance-based recovery never happened. We observe a significant drop in the case of SPLT, MBMD, ATOM, and SiamRPN. The performance drop for other trackers is also significant in the context of their overall tracking performance (for example, TLD's performance drops by more than 35%).

Tracker	Mean % of frames where alternate object was tracked	Avg. no. of Recoveries	Original Performance (Success Metric)	Reduced Performance (Success Metric)
SPLT [35]	20.99%	8.43	35.99	18.16
MBMD [37]	19.08%	6.0	32.79	15.35
ATOM [7]	13.76%	5.5	31.30	18.59
SiamRPN [19]	14.95%	5.18	43.69	25.15
FuCoLoT [20]	1.61%	1.12	23.14	19.18
MDNet [25]	1.58%	0.68	40.62	38.33
CMT [26]	2.78%	0.56	8.72	7.97
ECO [8]	3.81%	1.5	22.25	19.59
LCT [22]	1.39%	0.75	11.12	10.33
TLD [15]	0.58%	0.18	6.94	4.21

Table 2. Results for the analysis of distractor enabled recoveries.

Tracker	Avg. no. of recoveries	Avg. no. of chances	Sequences with static recoveries	Performance on sequences with static recoveries (Success Metric)	Reduced performance on sequences with static recoveries (Success Metric)
SPLT [35]	0.26	1.24	5	29.53	5.00
MBMD [37]	0.18	1.24	4	15.62	10.18
ATOM [7]	0.94	7.98	11	20.27	10.53
SiamRPN [19]	0.64	2.28	9	39.90	21.18
MDNet [25]	3.14	15.66	13	15.05	10.58
FuCoLoT [20]	2.7	6.1	17	10.60	4.25
CMT [26]	5.26	10.78	21	8.61	5.57
ECO [8]	3.88	24.62	20	9.70	6.42
LCT [22]	3.34	7.14	20	10.05	7.44
TLD [15]	2.54	5.26	16	7.25	2.34

Table 3. Results for the analysis of static recoveries.

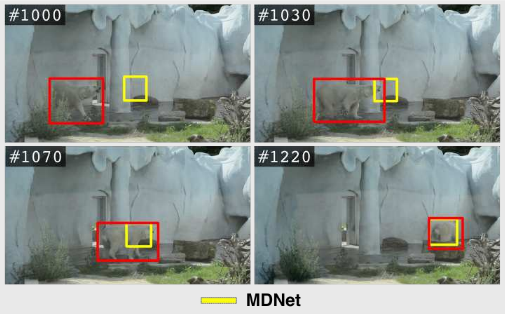

Figure 5. An example of a recovery where the tracker does not move at all, but the ground truth (red) falls right into the tracker's (yellow) prediction

4.2. Recovery With No Motion

The second type of recoveries we study is when the tracker is stationary, and the target passes through it, and then the tracking resumes. An example of such a recovery is illustrated in Figure 5. Here, the recovery can be attributed to chance, because the target, fortunately, moved into the tracker (the tracker recovers even though it had no idea where the target was).

We first formalize the notion of the tracker being "stationary." A tracker is said to be stationary if the IoU of the current prediction (at time t) is more than 0.5 with each of the previous 200 predictions and the IoU with the target is zero. This definition ensures that the tracker is frozen somewhere in the background, after accounting for minor noisy movements. We further define "static recovery," i.e. the recovery which happens when the tracker is stationary (IoU between the tracker and target goes from zero to non-zero and remains non-zero for next 60 frames). We then compute the following:

- The average number of static recoveries per sequence in the dataset.
- The average number of chances i.e., number of times when the tracker was stationary, and the target came towards it leading to a non zero IoU (even for a single frame).
- The impact of static recoveries on the tracking performance i.e., the reduced success metric by ignoring the performance after the first static recovery in each sequence. However, here we report the performance drops only on the sequences where static recovery occurs (which differs for each tracker). The point of reporting these performance drops is not to give a metric, but to understand the worst-case impact of such recoveries on the tracking performance.

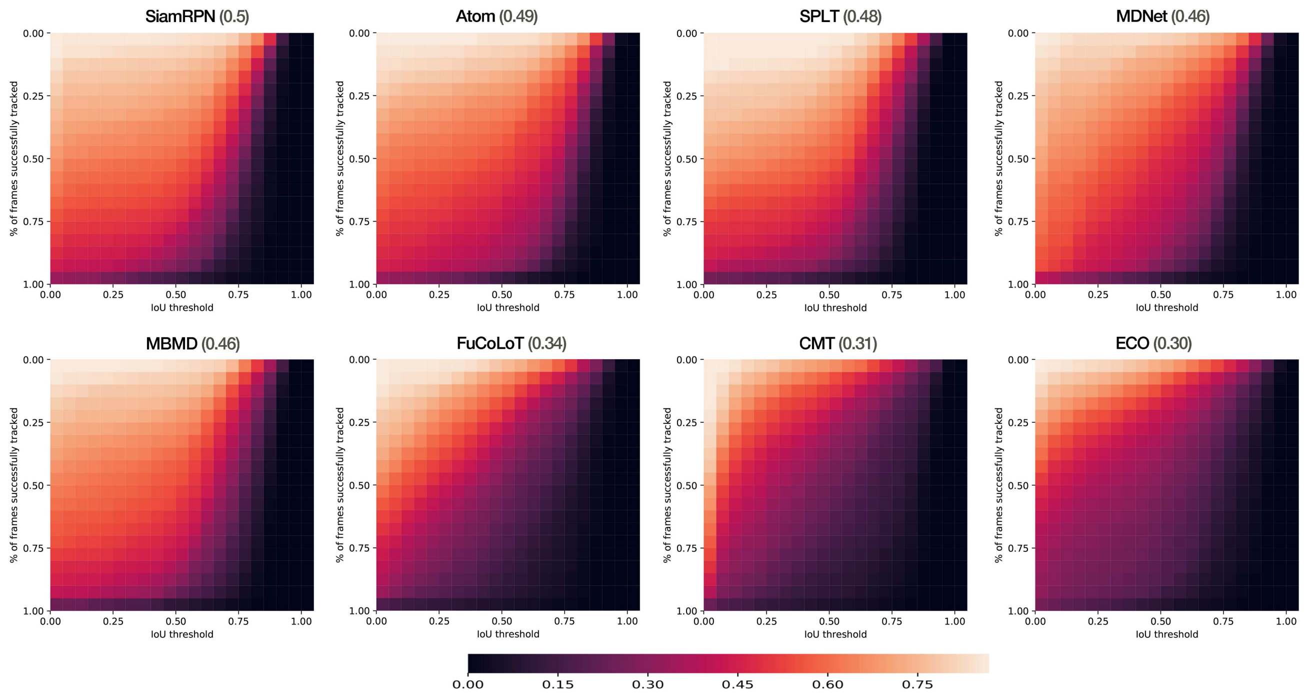

Figure 6. 3D-LSM visualizations for the evaluated trackers. 3D-LSM metric is also reported for each tracker (on top).

Results and Discussion: The results are summarized in Table 3. The first two columns present the average number of static recoveries per sequence and the number of chances it got (averaged over all 50 sequences). The third column presents the number of sequences for each tracker which have static recoveries (the experiments are performed on all 50 sequences of the dataset; however, not all sequences have static recoveries). The last two columns present the success metric before and after accounting for the chance based recoveries (averaged only over the sequences with static recoveries, which is different for each tracker). Our observations are as follows:

1. Trackers that perform online model updates (ECO, CMT, TLD) are prone to freezing very often. This occurs even in the case of trackers like MDNet and FuCoLoT, which only perform conservative model updates (when confident). Model updates could enable the tracker to adapt to the background and hence causing the tracker to freeze.

2. The experiment is quantifying cases when the tracker has failed, and the tracker predictions have frozen entirely. Despite having a very strict definition that gives the benefit of the doubt to the trackers, we still observe that a lot of trackers freeze.

3. We observe a complementary nature of recoveries. Predominantly offline trained trackers tend to look for objectness and can track an alternate object altogether. Due to the interactions between the objects, the tracker recovers. The second class of trackers which perform online model updates can sometimes lose the discriminative ability between the target and background and can freeze while tracking the background. The recovery occurs when the target passes through the tracker.

4. The performance drop in SPLT, MBMD, ATOM and SiamRPN is significant after accounting for the performance due to chance. This also indicates that they make good use of the chances they get.

5. Reliability in Long-term Tracking

Practically, trackers are reliable to use in long-term applications if the human effort to fix the incorrect tracker predictions is minimal. The human effort is a function of the precision required for the application at hand. A tracker which gives contiguous segments of precise tracking would be easier to correct by re-initializing on failures. However, it will take a lot of mental burden to correct a tracker whose IoU fluctuates intermittently. Moudgil and Gandhi [23] made an effort to quantify the reliability aspect and proposed the Longest Subsequence Measure (LSM) metric. In this section, we address some of limitations of LSM metric and extend it in a more general sense. We also present a visual interpretation of trackers which could aid the practitioner to pick appropriate trackers conditioned on their specific needs.

Tracker	Success Metric at IoU 0.5
SPLT [35]	**52.74**
SiamRPN [19]	51.52
MBMD [37]	48.12
ATOM [7]	47.51
MDNet [25]	42.27
FuCoLoT [20]	21.99
CMT [26]	20.81
ECO [8]	21.94
TLD [15]	13.90
LCT [22]	8.75

Table 4. Success Metric for the trackers on entire TLP dataset.

Preliminaries: LSM [23] computes the ratio of the length of the longest successfully tracked continuous subsequence to the total length of the sequence. A subsequence is marked as successfully tracked, if $x\%$ of frames within it have IoU $>$ 0.5, where x is a slack parameter. A representative LSM score per tracker is computed by fixing the slack parameter x to 0.95 (tracking 95% of the sub-sequence successfully).

We believe that the choice of thresholds for IoU (0.5) and slack x (0.95) in LSM does not provide a fair and complete perspective. For example, a tracker that has IoU slightly lesser than 0.5 would be penalized harshly due to binary IoU thresholding at 0.5. Prior work [29] has also shown that human annotators cannot often distinguish between IoU scores of 0.3 and 0.5. In [23], the authors also present LSM plots by fixing IoU to 0.5 and varying the slack. However, such plots fails to give a holistic perspective on the simultaneous effect of changing both the IoU and the slack.

Extending LSM: We present a 3D-LSM metric, which captures the effect of both precision (IoU) and failure tolerance in a connected manner. The 3D-LSM metrics is the mean of a matrix, computed by varying both the slack and the IoU parameters. Each entry in the matrix measures the longest contiguous sub-sequence (normalized) successfully tracked by fixing the IoU and slack parameters (for instance if the slack is 0.95 and IoU is 0.3, then we find the longest sub-sequence where 95% of the frames are tracked with IoU greater than 0.3). Basically, each entry in the matrix is the LSM value computed at a specific slack and IoU threshold. In current experiment we vary both slack and IoU thresholds at a rate of 0.05 from 0.05 to 1, resulting in a 20×20 matrix. One major benefit of the proposed metric is that it can be visualized as an image and makes way for a direct visual interpretation. It would aid non-expert practitioners to compare several trackers by visual inference.

Results and Discussion: The 3D-LSM visualization results for the evaluated trackers on the TLP dataset are shown in Figure 6. SiamRPN, ATOM, SPLT, MDNet, and MBMD give better performance in comparison to the other five trackers. SPLT and MBMD are built upon the SiamRPN as the base network, and though they improve other aspects like re-detection, the reliability aspect reduces marginally. Another interesting observation is that while ECO outperforms SiamRPN on short term benchmarks like OTB100, it performs significantly worse in the presented long term setting. The reliability aspect of trackers like CMT is quite low, possibly due to drift in feature tracks. FuCoLoT was designed as a long term tracker; however, it performs poorly on the reliability aspect. MDNet performs well on the reliability aspect owing to its online updates.

The 3D-LSM plots allow direct visual inferences: (a) brighter plots indicate better performance. We can observe how the images get darker when moving from SiamRPN to ECO. (b) Contours formed in more reliable trackers tend to stretch towards the bottom right corner. Compare SiamRPN and ECO, for instance; we can see that the shape of the contour inverts. (c) The practitioners need lies in the bottom right corner (i.e., low failure tolerance and high IoU), and most trackers are pitch black in the area. This highlights the significant challenges and opportunities which lie ahead in the area of visual object tracking to meet the application requirements.

6. Summary and Conclusion

In this paper, we touch upon the three crucial aspects of Re-detection, Recovery, and Reliability (3R's) for long term tracking. These aspects are not explicit in existing evaluation metrics, which makes it difficult to reason out the poor or effective performance of a particular tracker in the long term setting. The 3R analysis is aimed to bridge this gap and can categorically highlight the shortcomings of different tracking algorithms. It helps us reason out the overall performance of the tracker as well (Table 4). For instance, trackers like CMT and FuCoLoT are specifically designed for long term setting and have an explicit re-detection module; however, they lack reliability and end up giving a poor overall performance.

Hence, definitions that restrict long term trackers to only the algorithms with re-detection capabilities [21] are limited and ignore the Recovery and Reliability aspects. Even trackers like MDNet (without explicit re-detection) give a reasonable overall performance in long term context, owing to high reliability. Recently proposed SPLT tracker gives the best overall performance (Table 4); however, it only gives a marginal improvement over the base SiamRPN network. 3R analysis shows that SPLT improves on the re-detection aspect; however, compromises on reliability and also ends up tracking an alternate object often. Similar, specific insights can be drawn for other trackers as well and can aid in studying their strengths and weaknesses. Overall we believe 3R analysis paves the way for designing better tracking algorithms in the future.

References

[1] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 983–990. IEEE, 2009. 2, 3

[2] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr. Fully-convolutional siamese networks for object tracking. In *European conference on computer vision*, pages 850–865. Springer, 2016. 2

[3] G. Bhat, J. Johnander, M. Danelljan, F. Shahbaz Khan, and M. Felsberg. Unveiling the power of deep tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 483–498, 2018. 2

[4] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui. Visual object tracking using adaptive correlation filters. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2544–2550. IEEE, 2010. 2

[5] L. Čehovin, M. Kristan, and A. Leonardis. Is my new tracker really better than yours? In *IEEE Winter Conference on Applications of Computer Vision*, pages 540–547. IEEE, 2014. 3

[6] R. T. Collins, Y. Liu, and M. Leordeanu. Online selection of discriminative tracking features. *IEEE transactions on pattern analysis and machine intelligence*, 27(10):1631–1643, 2005. 2

[7] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg. Atom: Accurate tracking by overlap maximization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4660–4669, 2019. 4, 6, 8

[8] M. Danelljan, G. Bhat, F. Shahbaz Khan, and M. Felsberg. Eco: efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6638–6646, 2017. 2, 4, 6, 8

[9] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg. Learning spatially regularized correlation filters for visual tracking. In *Proceedings of the IEEE international conference on computer vision*, pages 4310–4318, 2015. 2

[10] H. Fan, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, H. Bai, Y. Xu, C. Liao, and H. Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5374–5383, 2019. 1, 2

[11] H. K. Galoogahi, A. Fagg, C. Huang, D. Ramanan, and S. Lucey. Need for speed: A benchmark for higher frame rate object tracking. *arXiv preprint arXiv:1703.05884*, 2017. 2

[12] V. Gandhi, R. Ronfard, and M. Gleicher. Multi-clip video editing from a single viewpoint. In *Proceedings of the 11th European Conference on Visual Media Production*, page 9. ACM, 2014. 1

[13] D. Held, S. Thrun, and S. Savarese. Learning to track at 100 fps with deep regression networks. In *European Conference on Computer Vision*, pages 749–765. Springer, 2016. 2

[14] L. Huang, X. Zhao, and K. Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *arXiv preprint arXiv:1810.11981*, 2018. 2

[15] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE transactions on pattern analysis and machine intelligence*, 34(7):1409–1422, 2011. 2, 4, 6, 8

[16] M. Kristan, S. Kovacic, A. Leonardis, and J. Pers. A two-stage dynamic model for visual tracking. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(6):1505–1520, 2010. 3

[17] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Cehovin Zajc, T. Vojir, G. Bhat, A. Lukezic, A. Eldesokey, et al. The sixth visual object tracking vot2018 challenge results. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 4, 5

[18] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. *arXiv preprint arXiv:1812.11703*, 2018. 3

[19] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu. High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8971–8980, 2018. 3, 4, 6, 8

[20] A. Lukežič, L. Č. Zajc, T. Vojíř, J. Matas, and M. Kristan. Fucolot–a fully-correlational long-term tracker. In *Asian Conference on Computer Vision*, pages 595–611. Springer, 2018. 3, 4, 6, 8

[21] A. Lukežič, L. Č. Zajc, T. Vojíř, J. Matas, and M. Kristan. Now you see me: evaluating performance in long-term visual tracking. *arXiv preprint arXiv:1804.07056*, 2018. 1, 2, 3, 4, 5, 8

[22] C. Ma, X. Yang, C. Zhang, and M.-H. Yang. Long-term correlation tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5388–5396, 2015. 3, 4, 6, 8

[23] A. Moudgil and V. Gandhi. Long-term visual object tracking benchmark. *arXiv preprint arXiv:1712.01358*, 2017. 1, 2, 3, 4, 5, 7, 8

[24] M. Mueller, N. Smith, and B. Ghanem. A benchmark and simulator for uav tracking. In *European conference on computer vision*, pages 445–461. Springer, 2016. 2

[25] H. Nam and B. Han. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4293–4302, 2016. 3, 4, 6, 8

[26] G. Nebehay and R. Pflugfelder. Clustering of static-adaptive correspondences for deformable object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2784–2791, 2015. 3, 4, 6, 8

[27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 5

[28] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 5

[29] O. Russakovsky, L.-J. Li, and L. Fei-Fei. Best of both worlds: human-machine collaboration for object annotation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2121–2131, 2015. 8

[30] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual tracking: An experimental survey. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1442–1468, 2013. 2

[31] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. Torr. End-to-end representation learning for correlation filter based tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2805–2813, 2017. 2

[32] J. Valmadre, L. Bertinetto, J. F. Henriques, R. Tao, A. Vedaldi, A. W. Smeulders, P. H. Torr, and E. Gavves. Long-term tracking in the wild: A benchmark. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 670–685, 2018. 1, 2, 3

[33] Y. Wu, J. Lim, and M. Yang. Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1834–1848, Sept. 2015. 2, 4, 5

[34] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 1, 2, 3

[35] B. Yan, H. Zhao, D. Wang, H. Lu, and X. Yang. 'skimming-perusal' tracking: A framework for real-time and robust long-term tracking. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 3, 4, 6, 8

[36] J. Zhang, S. Ma, and S. Sclaroff. Meem: robust tracking via multiple experts using entropy minimization. In *European conference on computer vision*, pages 188–203. Springer, 2014. 2

[37] Y. Zhang, D. Wang, L. Wang, J. Qi, and H. Lu. Learning regression and verification networks for long-term visual tracking. *arXiv preprint arXiv:1809.04320*, 2018. 3, 4, 6, 8