Hard Occlusions in Visual Object Tracking

Thijs P. Kuipers[(✉)], Devanshu Arya[(✉)], and Deepak K. Gupta[(✉)]

Informatics Institute, University of Amsterdam, Amsterdam, The Netherlands
kuiperthijs@gmail.com, d.arya@ua.ul, gupta.deepak@gmail.com

Abstract. Visual object tracking is among the hardest problems in computer vision, as trackers have to deal with many challenging circumstances such as illumination changes, fast motion, occlusion, among others. A tracker is assessed to be good or not based on its performance on the recent tracking datasets, *e.g.*, VOT2019, and LaSOT. We argue that while the recent datasets contain large sets of annotated videos that to some extent provide a large bandwidth for training data, the hard scenarios such as occlusion and in-plane rotation are still underrepresented. For trackers to be brought closer to the real-world scenarios and deployed in safety-critical devices, even the rarest hard scenarios must be properly addressed. In this paper, we particularly focus on hard occlusion cases and benchmark the performance of recent state-of-the-art trackers (SOTA) on them. We created a small-scale dataset (Dataset can be accessed at https://github.com/ThijsKuipers1995/HTB2020) containing different categories within hard occlusions, on which the selected trackers are evaluated. Results show that hard occlusions remain a very challenging problem for SOTA trackers. Furthermore, it is observed that tracker performance varies wildly between different categories of hard occlusions, where a top-performing tracker on one category performs significantly worse on a different category. The varying nature of tracker performance based on specific categories suggests that the common tracker rankings using averaged single performance scores are not adequate to gauge tracker performance in real-world scenarios.

Keywords: Visual object tracking · Occlusion · Benchmarks · Metrics · Dataset

1 Introduction

Visual object tracking remains a challenging problem, even though it has been studied for several decades. A visual object tracker has to account for many different and varying circumstances. For instance, changes in the illumination may alter the appearance of the target object. The object could also blend in with the background environment, or it might get occluded, resulting in the object, or part of the object, being obstructed from view. Because of all the possible

different circumstances, visual object trackers have to account for, visual tracking is considered a hard problem [1].

Commonly, a tracker is assessed to be good or not based on its performance on established benchmarking datasets such as OTB100 [2], UAV123 [3], VOT2019 [4], GOT-10k [5], LaSOT [6], and TrackingNet [7]. These datasets comprise large sets of video sequences spanning across different challenges of tracking. For every dataset, the performance of a tracker is averaged over all sequences.

For a tracker to be used in real-world scenarios and embedded in safety-critical equipment, it should be able to handle even the rarest and hardest instances of tracking. Therefore, the evaluation datasets should also contain such instances. We argue that this is not yet the case, and scenarios such as occlusion, in-plane rotation, and out-of-plane rotation are still underrepresented in these datasets. Moreover, most performance metrics compute an average score across all sequences, thereby overshadowing the poor performance of the subjected tracker on a certain specific challenge. For a deeper study on these aspects, we tailor the focus of this paper to only hard occlusions.

Occlusion refers to the phenomenon where parts of the target object are blocked from the field of view. For instance, an object can be occluded when it either moves partially or fully out of frame. Occlusion also occurs when another object fully or partially blocks the target object. When the target object is partially blocked, either certain specific features of the target object can disappear, or part of the entire target object appearance will disappear. Unlike other challenges of tracking, learning a distribution for occlusion is hard - no distribution exists in parts of the object that are occluded. This makes occlusion a hard problem. Some methods for handling occlusion do exist, but they are often focused on very specific aspects of tracking such as solely tracking pedestrians [8]. A major problem in evaluating visual object trackers on occlusion is the lack of data containing hard occlusions in current datasets. While the above-mentioned datasets do contain samples of occlusion, they often do not represent the challenging cases of occlusion that can occur when tracking in the wild [2,6,7]. Therefore, the available benchmarks might not accurately evaluate tracker performance on hard occlusions.

This work aims to evaluate a set of current state-of-the-art (SOTA) visual object trackers on the occurrences of hard occlusions. To perform this evaluation, a small dataset containing various samples of hard occlusions is compiled. Our preliminary results show that the performance of the SOTA trackers is, in general, lower on the hard occlusion scenarios. Further, we analyze whether the leading tracker among the ones chosen in this study performs superior on different scenarios of occlusion. Our results reveal interesting insights on whether even the best tracker, decided based on the existing evaluation strategies, could be considered a safe choice for deployment in real-world scenarios, especially for safety-critical devices. Further, it raises the question of whether the current performance metrics, averaging the score over all the sequences, are the right choice to assess the performance of trackers.

2 Related Work

In the following, we present an overview of some previous works that are relevant to this study. First, we present an overview of the recent visual object tracking algorithms, followed by works related to tackling occlusion in tracking. Finally, an overview of different tracking benchmarks is presented.

2.1 Object Tracking

The task of object tracking can refer to tracking multiple objects simultaneously (multi-object tracking) or tracking a single instance of an object (single-object tracking) [7]. This work will only consider *generic* single-object trackers. To perform successful tracking, trackers must be able to create a strong appearance model for the target object and be capable of reliable and fast localization of the target object.

To address the above-mentioned challenge, various methods have been proposed. One such category is Correlation Filters (CF), which forms the basis of several SOTA single-object trackers [7]. CF uses circular correlation, which results in all shifted variants of input samples being implicitly included. This enables the construction of a powerful appearance model with very limited data [9]. Furthermore, CF allows for fast run-times as computations are performed in the Fourier domain [10]. The MOSSE [11] tracker paved the way for more advanced CF-based approaches such as the use of multi-dimensional features [12–14], improved robustness to variations in scale and deformations [15,16], mitigating boundary effects [9,17], and the use of deep convolutional filters [18,19]. The advancements made in CF trackers resulted in large and complex models, which significantly increases the computational cost. ECO [20] improves upon the CF-framework by reducing the model complexity by proposing a factorized convolution operation to increase running speeds.

Another category is deep learning-based trackers. Recurrent Neural Networks have been proposed for tracking [21,22], but do not yield competitive performance compared to the SOTA trackers. MDNet [23] implements a deep convolutional network that is trained offline, and performs Stochastic Gradient Descent during tracking but is not able to operate in real-time. GOTURN [24] utilizes a convolutional network to learn a function between image pairs. SiamFC [25] introduces a fully-connected Siamese architecture to perform general similarity learning. The goal is to learn a similarity function offline, which can then be evaluated during tracking to locate an exemplar image within a larger candidate image. SiamRPN [26] implements a Siamese network that is extended with a Region Proposal Network (RPN) which allows for improved bounding-box predictions. SiamRPN++ [27] improves upon the Siamese architecture by enabling the use of deep networks in Siamese trackers. ATOM [28] improves upon the Siamese trackers by introducing an approach that consists of a target estimation module that is learned offline, and a target classification module that is learned online. DiMP [29] consists of a discriminative model prediction archi-

tecture derived from a discriminative learning loss for visual tracking which can fully exploit background information.

2.2   Occlusion in Tracking

Occlusion remains largely unexplored in generic single-target object tracking. Some tracking architectures that do explicitly handle occlusion have been proposed. ROT [30] utilizes occlusion-aware real-time object tracking by overcoming target model decay which can occur when the target object is being occluded. In [31] SiamRPN++ and SiamFC are equipped with structured dropouts to handle occlusion. Other methods include more experimental strategies such as analyzing the occurrence of occlusion by utilizing spatiotemporal context information [32]. By further analyzing motion constraints and the target reference the strategy allows for better discrimination between the target object and background distractors causing occlusion. Another experimental strategy uses layer-based strategies, extending it by specific background occluding layers [33]. Strategies that focus on handling occlusion have also been proposed in more specific object tracking tasks, such as tracking pedestrians [8] and cars [34]. In [13] a multi-object tracker approach that handles occlusion is proposed, which is built on the idea of object permanence, using a region-level association and object-level localization process to handle long periods of occlusion. In [35] alternative SOTA methods for handling occlusion are presented such as depth-analysis [36,37] and fusion methods such as a Kalman filter for predicting target object motion and location [38,39].

2.3   Tracking Datasets

To evaluate the robustness of single-target visual object trackers, many datasets have been proposed. ALOV300 [1] contains 314 short sequences. ALOV300 does include 14 different attributes, including occlusion. However, it does not differentiate between different kinds of occlusion. OTB [2] is another well-known dataset. The full dataset (OTB100) contains 98 sequences, while OTB50 is a subset of OTB100 containing 51 of the most challenging sequences. OTB offers 11 attributes, including both partial and full occlusion. Since the rise of deep trackers, the demand for large-scale datasets has increased. TrackingNet [7] is introduced to accommodate these demands. TrackingNet consists of over 30 thousand sequences with varying frame rates, resolutions, and lengths. TrackingNet includes 15 attributes, including both partial and full occlusion, as well as out-of-frame occlusion. GOT-10k [5] consists of over 10 thousand sequences. GOT-10k offers an impressive 563 different object classes, and offers a train and test-set with zero overlaps between classes, resulting in more accurate evaluations. GOT-10k offers several attributes, including occlusion. Many of the sequences contained in the above-mentioned datasets are of relatively short duration. However, when tracking in the wild, tracking often occurs for sustained periods. LaSOT [6] introduces a dataset consisting of 1400 longer duration sequences with an average sequence length of over 2500 frames. Similarly to the

previously mentioned datasets, LaSOT includes a variety of attributes, including both partial and full occlusion, and out-of-frame occlusion. UAV123 [3] contains 123 videos captured from a UAV, as well as artificial sequences generated by a flight-simulator.

3   Benchmarking on Hard Occlusions

While most datasets described in Sect. 2.3 contain cases of occlusion, the chosen instances are still very simple and do not account for the hard scenarios. These datasets do not take specific sub-categories within occlusion into account, as often solely general cases of occlusion, such as partial or full, are considered. Furthermore, many of the sequences contained in the datasets are of relatively short duration. As a result, occlusion often occurs for only short amounts of time. These short durations are not enough to accurately assess tracker performance on occlusion. Another issue is that often the occlusions that do occur involve simple cases. The occluded target object often possesses barely any movement relative to the camera, or the target object remains stationary throughout the sequence (see Fig. 1a). The challenging LaSOT [6] dataset does contain more challenging cases of occlusion, including longer durations and more extreme movement of the target object (see Fig. 1b). However, the set of sequences containing these hard occlusions remains very limited.
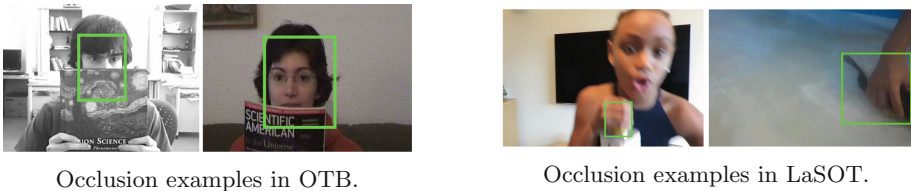

Occlusion examples in OTB.   Occlusion examples in LaSOT.

Fig. 1. Examples of occlusion from the OTB and LaSOT datasets.

Here, we present our Hard Occlusion Benchmark[1] (HOB), a small-scale dataset containing 20 long-duration sequences that encompass a variety of different hard occlusion scenarios. For the sake of demonstration, Fig. 2 shows the first frame of some of the sequences with the corresponding ground-truth. Each sequence is of similar length, with an average of 2760 frames per sequence. Each sequence in HOB is annotated every 15th frame. Despite the lack of fully annotated sequences, with an average of 185 annotated frames per sequence, there exist ample ground-truths to perform an accurate evaluation. Naturally, HOB contains the general cases of hard occlusion, such as *partial occlusion*, *full occlusion*, and *out-of-frame occlusion*. The cases of occlusion occur for long periods and are combined with strong movement and scale-variations of the target object

---

[1] The dataset can be accessed at https://github.com/ThijsKuipers1995/HTB2020.

relative to the camera. Also, these general cases are complemented with more specific attributes, to obtain a more precise evaluation of the SOTA tracker implementations on hard occlusions.

Fig. 2. First frame of the sequences in HOB with ground-truth.

In its current form, HOB dataset comprises the following occlusion types.

- *Full out of frame occlusion (FOC).* The target object moves fully out of the frame for extended periods. The target object may enter the frame at a different location compared to where it exited the frame.
- *Feature occlusion (FO).* Some specific features of the target are omitted from view. It is still possible for the entire target object to be in view in this case.
- *Occlusion by transparent object (OCT).* The target object is being occluded by a transparent object. This means the target object can still be visible through the occluding object, although the occluding object does alter the appearance of the target object.
- *Occlusion by similar object (OCS).* The target object is being occluded by a similar-looking object.

4   Experiments and Evaluations

The current section introduces the set of trackers that are evaluated. Next, an overview of the metrics used for evaluation is given. Finally, the performance of the selected set of trackers is evaluated. The selected set of trackers aims to cover a variety of common state-of-the-art (SOTA) tracking principles. The following

trackers are chosen for the evaluation: ECO [20], SiamFC [25], SiamRPN++ [27], ATOM [28], DiMP [29]. ECO proposes a general framework to improve upon the discriminant correlation filter tracking. SiamFC proposed similarity matching by using a fully connected Siamese network. SiamRPN++ uses a deep network for more sophisticated features, as well as a region proposal network. Three variants of SiamRPN++ are evaluated: the original version using the ResNet50 (r50) [40] (r50) backbone, a version using the shallow AlexNet [41] backbone, and the long-term version which uses the long-term update strategy as described in [42]. ATOM proposes a tracking architecture consisting of dedicated target estimation and target classification components. DiMP [29] proposes a tracking architecture that can utilize both target and background information. Both ATOM and DiMP utilize a memory model to update the appearance model to take into account the change of appearance over time. Solely trackers with publicly available code are used in this work.

4.1   Evaluation Methodology

We perform a One Pass Evaluation (OPE) measuring precision, success rate, area-under-curve, and the least-subsequence-metric [43] on the 20 hard occlusion sequences in HOB. A brief overview of these metrics is presented below.

Precision. When tracking precision, the center localization error is calculated. The center localization error is defined as the Euclidean distance between the center of the ground-truth bounding box and the prediction bounding box. A frame is deemed successfully tracked if the calculated distance is below a certain threshold, say t. While precision does not take the predicted bounding box size into account, it does correctly measure how close the position of the prediction is to the ground truth, which is not always the case when using success rates, as only the overlap between prediction and ground truth is considered. In the case of occlusion, where the target object is not entirely visible, precision can depict to what extend the tracker manages to correctly predict the location of the occluded target object. The issue with using precision as a metric is its sensitiveness to resolution, which in the case of HOB is avoided since every sequence is of the same resolution. The final precision score for each tracker is evaluated using a threshold of 20 pixels such as in [44].

Success Rate. The success rate makes use of the average bounding box overlap, which measures performance by calculating the overlap between ground-truth bounding boxes and prediction bounding boxes. It takes both position accuracy and accuracy of the predicted bounding box size and shape into account. Therefore, the success rate can offer a solid measurement of tracker performance. Bounding box overlap is calculated using the intersection-over-union (IoU) score. Similar to precision, a frame is deemed successfully tracked when the calculated IoU meets a certain threshold, say t. By calculating the success rate at a range of different thresholds, a success plot can be formed. A threshold of $t > 0.5$ is

often used to measure success. However, this does not necessarily represent a successfully tracked frame [2]. Because of this, the area-under-curve (AuC) of the success plot is calculated instead, which takes the entire range of thresholds into account. Furthermore, frames in which the absence of the target object is correctly predicted are given an IoU score of 1.

Least Subsequence Metric. The least-subsequence-metric (LSM) quantifies long term tracking behavior by computing the ratio between the length of the longest continuously "successfully" tracked sequence of frames and the full length of the sequence. A sequence of frames is deemed as successfully tracked if at least a certain percentage p of frames within this is successfully tracked [43]. The representative LSM score is calculated at a threshold of $p = 95\%$ as in [43]. A frame is considered correctly tracked when the IoU of that frame is greater than 0.5. Because LSM calculates the ratio between the longest continuously tracked subsequence and the length of the entire sequence, it can introduce a bias towards extremely long and short sequences. However, all sequences used in this work are of similar length, therefore this is not an issue for accurate evaluation.

4.2   Baseline Dataset

For the sake of comparison with HOB, we use LaSOT as the baseline dataset. LaSOT is a large benchmark that focuses on long-term tracking, and it includes many sequences containing occlusions and out-of-frame occlusions. Due to this reason, it is one of the more difficult tracking benchmarks. Evaluating the selected visual object trackers on LaSOT will, therefore, offer a great baseline for comparing to HOB. HOB is a relatively small dataset containing only 20 sequences. To keep the comparison between HOB and LaSOT fair, only the top 20 occlusion heavy sequences from LaSOT are selected. Furthermore, while LaSOT offers per-frame annotations of ground-truths, HOB contains a ground-truth annotation every 15th frame. Therefore only every 15th frame of LaSOT will be used during the evaluation procedure.

4.3   Overall Performance

Figure 4 and Fig. 5 depicts the predictions for each of the evaluated trackers on four sequences corresponding to the mentioned attributes.

Figure 3 shows the precision, success rate, and LSM of each evaluated tracker on both HOB and LaSOT. In Table 1, the representative scores of each of the metrics are shown. The results show that on average, the performance of the evaluated trackers is worse on HOB compared to LaSOT. SiamRPN++(r50) is the top-performing tracker on HOB on all metrics. SiamRPN++(r50) outperforms SiamRPN++(lt) by a small margin on the AuC metric, and similar observations can be made for the other two metrics as well. This is an interesting

Table 1. Representative scores for precision (t = 20), area under curve (AuC) and LSM (x = 0.95) scores for each of the evaluated trackers on HOB and LaSOT. Best scores are shown in bold.

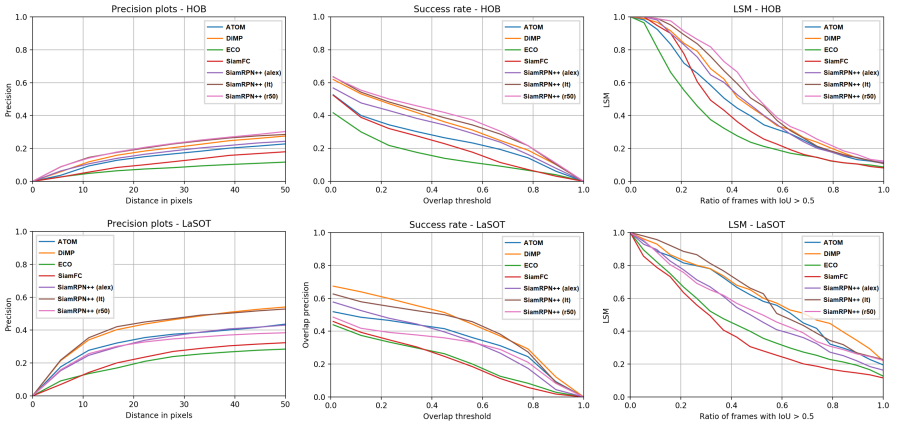| Dataset	Precision		AuC		LSM	
	HOB	LaSOT	HOB	LaSOT	HOB	LaSOT
ATOM	0.142	0.342	0.243	0.317	0.122	0.228
DiMP	0.173	0.421	0.324	**0.391**	0.126	**0.292**
ECO	0.070	0.191	0.149	0.199	0.098	0.166
SiamFC	0.093	0.225	0.205	0.191	0.090	0.135
SiamRPN++ (alex)	0.154	0.320	0.300	0.300	0.125	0.185
SiamRPN++ (lt)	0.192	**0.437**	0.343	0.383	0.127	0.248
SiamRPN++ (r50)	**0.195**	0.318	**0.359**	0.278	**0.133**	0.245

Fig. 3. Overall results on HOB (top) and LaSOT (bottom) on the precision rate, success rate, and LSM (left, middle, and right respectively).

result, as SiamRPN++(lt) is specifically tailored to handling long-term tracking which includes occlusion of the target object. These results imply that even the re-detection module of SiamRPN++(lt) can occasionally drift the tracker model to false targets. This could be attributed to SiamRPN++(lt) re-detecting the wrong target object and sticking to it during long and heavy stretches of occlusion, which would result in lower overall performance. Contrary to the results obtained on HOB, performance on LaSOT seems significantly different. On LaSOT, DiMP is the top-performing tracker on the AuC and LSM metrics, and second-best on precision. Only SiamRPN++(lt) shows comparative performance, and as shown in Table 1. The performance of the remaining trackers is significantly lower.

It is interesting to note that DiMP consistently underperforms compared to both SiamRPN++(r50) and SiamRPN++(lt) on HOB. While HOB dataset

contains only 20 sequences, the sequences of HOB have been chosen with no intended bias towards Siamese and SiamRPN-type trackers. Thus, we would ideally expect DiMP to perform the best, as has been seen on LaSOT. One possible reason for the reduced performance on HOB could be attributed to the fact that the training set of DiMP comprised the training set of LaSOT sequences as well. This could mean that DiMP tends to overfit on objects similar to those observed in the LaSOT training set. Since most other datasets contain similar tracking instances, DiMP performs well on them. On the contrary, the scenarios observed in HOB are quite different, and that leads to reduced performance on this dataset. Another possible reason for the performance decay of DiMP on HOB could be attributed to the bias added to the tracking model due to frequent model updates happening even under the scenarios of occlusion [45]. This is not the case for the SiamRPN++ variants, as they do not perform any model update. Note that Table 1 shows a relatively large difference in precision scores between HOB and LaSOT compared to the AuC scores. This is partly caused by the lower resolution sequences of LaSOT, as precision is sensitive to resolution.

On HOB, ECO is the worst performing tracker in terms of, precision and AuC. On LaSOT, ECO and SiamFC are the worst-performing trackers, with ECO obtaining a slightly higher AuC score. It seems that the discriminative correlation filter approach utilized in ECO is not very well suited for occlusions and long-term tracking in general, as it may not be able to generalize compared to the Siamese based trackers. In the case of SiamFC, its lack of accurate target classification and localization capabilities seems to hamper performance during cases of hard occlusion. This becomes more apparent in the LSM score, where SiamFC and ECO are the lowest-performing trackers, as a low LSM score indicates frequent loss of the target object. ATOM performs consistently worse on HOB compared to the three SiamRPN++ variants. On LaSOT, ATOM performs very similarly to SiamRPN++(r50) and SiamRPN++(alex), generally outperforming them by a slight margin. ATOM also utilizes a model update strategy, which could result in the decay of the appearance model during cases of hard occlusion.

4.4   Attribute Evaluation

While the results from the previous section have shown that most trackers struggle in the presence of hard occlusions, it is of interest to analyze further how different occlusion types affect the overall performance. In this section, we study the trackers for the different categories of occlusion that we have defined earlier. The success plots for each of the categories are shown in Fig. 4. Figure 5 depicts the predictions for each of the evaluated trackers on sequences corresponding to each of the categories.

Full Out-of-Frame Occlusion (FOC). FOC seems to be a very challenging problem for the visual object trackers, with SiamRPN++(lt) being the top-performing tracker in this category. This is most likely attributed to its re-
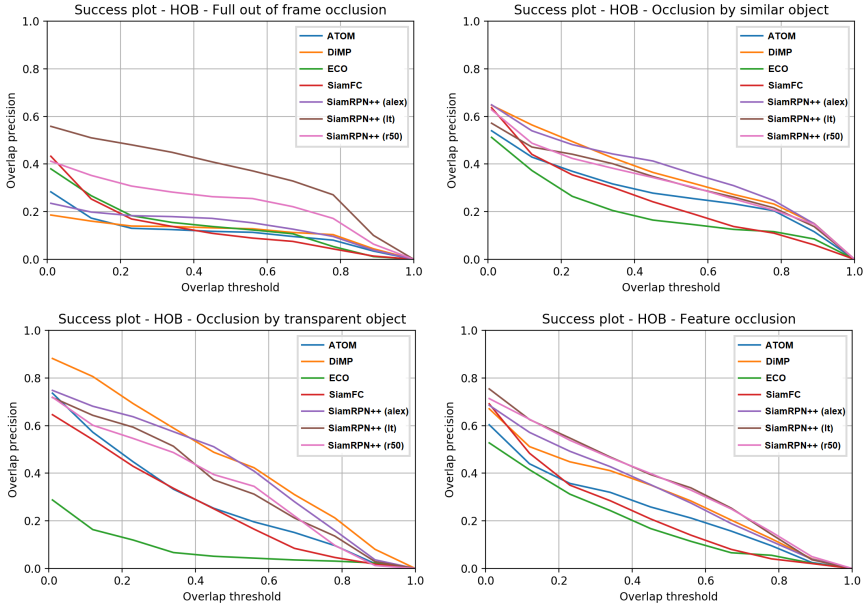
Fig. 4. Success plots for full out of frame occlusion (top left), occlusion by similar object (top right), occlusion by transparent object (bottom left), and feature occlusion (bottom right).

initialization strategy when detecting target object loss. The second-best performing tracker on FOC is SiamRPN++(r50), performing considerably better compared to the rest. Having access to rich features at different scales seems to aid its re-detection capabilities when the target object moves within the localization area. When observing the predictions, only SiamRPN++(lt) seems to consistently be able to re-detect the target object (see Fig. 5a). Overall, the SiamRPN++ variants outperform the other evaluated trackers. DiMP performs considerably worse during FOC. Interestingly, even trackers with weak discriminative power, such as SiamFC and ECO, perform on par with DiMP and ATOM for cases of FOC. ATOM, DiMP, and ECO update their appearance model during tracking. In the case of FOC, this could result in the appearance updating on samples that do not contain the target object causing strong appearance model decay. This is not the case for the Siamese trackers, as their appearance model remains fixed during tracking.

Occlusion by Similar Object (OCS). In the case of OCS, SiamRPN++(alex) has the highest overall performance, while SiamRPN++(r50) performs the worst of the SiamRPN++ trackers. Interestingly, the use of the shallow AlexNet as a backbone results in better performance compared to using the deep ResNet, even outperforming the long-term SiamRPN++ variant. Re-initialising on target loss does not offer an advantage during OCS, as the performance of SiamRPN++(lt) is similar to the performance of SiamRPN++(alex). DiMP is the second-best
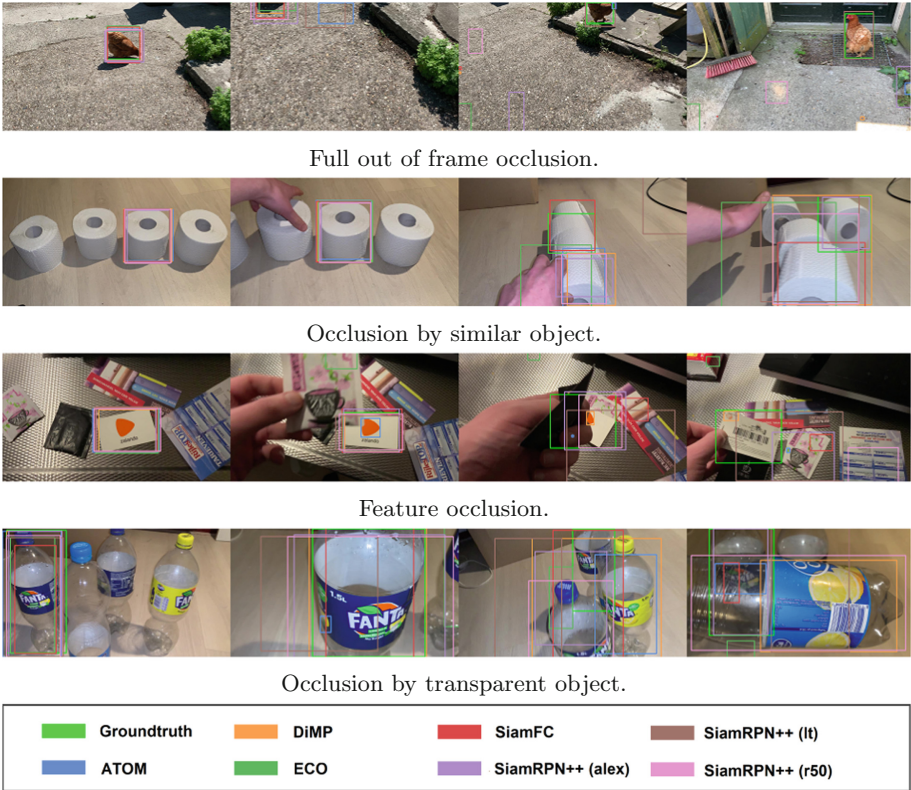
Full out of frame occlusion.

Occlusion by similar object.

Feature occlusion.

Occlusion by transparent object.

	Groundtruth		DiMP		SiamFC		SiamRPN++ (lt)
	ATOM		ECO		SiamRPN++ (alex)		SiamRPN++ (r50)

Fig. 5. Depiction of four frames including prediction and groundtruth bounding-boxes for each the categories *full out-of-frame occlusion, occlusion by similar object, feature occlusion*, and *occlusion by transparent object*.

performing tracker, with ATOM, SiamFC and, ECO being the lowest-performing trackers. ECO performs considerably lower compared to the other trackers. When observing the predictions during OCS, trackers struggle to accurately keep track of the target object (see Fig. 5b).

Feature Occlusion (FO). During FO, SiamRPN++(r50) and SiamRPN++(lt) are the top performing trackers, with near-identical performance. As objects tend to stay at least partially visible in this category, the re-initialization strategy of SiamRPN++(lt) does not offer much benefit in tracking performance. SiamRPN++(lt) and SiamRPN++(r50) are closely followed by SiamRPN++-(alex) and DiMP. Once again, ECO is the worst performing tracker. The results of the FO category are very similar to the overall performance on occlusion as shown in Fig. 3, although on average the trackers seem to perform slightly worse on feature occlusion specifically at higher thresholds.

Occlusion by Transparent Object (OCT). DiMP is the best performing tracker on OCT, SiamRPN++(alex) being a very close second. Both DiMP and Siam-RPN++(alex) perform considerably better than to the other SiamRPN++ variants, similar to the OCS category. SiamRPN++(r50) and SiamRPN++(lt) have very similar performance. In the cases of OCS and OCT, it seems that the ability to generalize for objects that are not seen during training could play an important role. The appearance model predictor implemented in DiMP contains few parameters, leading to better generalization as less overfitting to observed classes occurs during the offline training phase [29]. Likewise, SiamRPN++(alex) using AlexNet contains less parameters compared to SiamRPN++(r50) [27]. The performance of ATOM is considerably lower compared to DiMP, while both use the same IoU maximization based architecture for updating the appearance model, suggesting the appearance model update is of less importance during OCT and OCS. It is interesting to note that when observing the predictions on OCT and FE, DiMP and ATOM tend to strongly focus on striking target object features, as can be observed in Fig. 5c and Fig. 5d.

5   Conclusion

In this work, we presented an evaluation of the current state-of-the-art (SOTA) visual object trackers on hard occlusions. We compiled a small dataset containing sequences that encompass several samples of hard occlusions to assess the performance of these trackers in such occluded scenarios. Furthermore, we evaluated the trackers on a subset of the most occlusion-heavy LaSOT sequences. From the results, we show that on average the trackers perform worse on hard occlusion scenarios, suggesting that occlusion is still a relatively unsolved problem in tracking. While DiMP is the best performing tracker on the LaSOT benchmark, it is consistently outperformed by SiamRPN++ using the ResNet backbone architecture (r50) and its long-term tracking variant (lt) on instances of hard occlusions. Furthermore, we show that the top-performing tracker can vary drastically between specific scenarios of hard occlusions. For example, while DiMP seems the best for handling occlusions caused by semi-transparent objects, it performs the worst for full out-of-frame occlusion scenarios.

The set of results presented in this paper hint towards the fact that even the best performing tracker based on the current benchmark datasets might not be suited for real-world deployment, especially in safety-critical applications, such as self-driving cars. Real-world problems do not promise the presence of a uniform set of challenges, and at any random instance, a different tracking challenge could be the most important. Correspondingly, we focused on the challenge of hard occlusions in this paper, and trackers behaved differently than they did on LaSOT. This implies two important things for future research. First, tracking datasets need to incorporate more instances of difficult tracking challenges. Second, evaluation methodologies need to be designed that give more importance to instances where a certain tracker performs the worst. To summarize on a high-level, a model that handles even the most difficult challenges of tracking sufficiently well should be considered a better visual object tracker.

References

1. Smeulders, A.W., Chu, D.M., Cucchiara, R., Calderara, S., Dehghan, A., Shah, M.: Visual tracking: an experimental survey. IEEE Trans. Pattern Anal. Mach. Intell. **36**(7), 1442–1468 (2013)
2. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell. **37**, 1 (2015)
3. Mueller, M., Smith, N., Ghanem, B.: A benchmark and simulator for UAV tracking. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 445–461. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_27
4. Kristan, M., et al.: The seventh visual object tracking vot2019 challenge results (2019)
5. Huang, L., Zhao, X., Huang, K.: GOT-10k: a large high-diversity benchmark for generic object tracking in the wild. IEEE Trans. Pattern Anal. Mach. Intell. (2019)
6. Fan, H., et al.: LaSOT: a high-quality benchmark for large-scale single object tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5374–5383 (2019)
7. Muller, M., Bibi, A., Giancola, S., Alsubaihi, S., Ghanem, B.: 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 (2018)
8. Noh, J., Lee, S., Kim, B., Kim, G.: Improving occlusion and hard negative handling for single-stage pedestrian detectors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 966–974 (2018)
9. Danelljan, M., Hager, G., Shahbaz Khan, F., Felsberg, M.: Learning spatially regularized correlation filters for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4310–4318 (2015)
10. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2544–2550. IEEE (2010)
11. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters (2010)
12. Danelljan, M., Shahbaz Khan, F., Felsberg, M., Van de Weijer, J.: Adaptive color attributes for real-time visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1090–1097 (2014)
13. Huang, Y., Essa, I.: Tracking multiple objects through occlusions. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), vol. 2, pp. 1051–1058. IEEE (2005)
14. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 583–596 (2014)
15. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8926, pp. 254–265. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16181-5_18
16. Danelljan, M., Häger, G., Khan, F., Felsberg, M.: Accurate scale estimation for robust visual tracking. In: British Machine Vision Conference, Nottingham, September 1–5, 2014. BMVA Press (2014)
17. Kiani Galoogahi, H., Sim, T., Lucey, S.: Correlation filters with limited boundaries. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4630–4638 (2015)

18. Danelljan, M., Robinson, A., Shahbaz Khan, F., Felsberg, M.: Beyond correlation filters: learning continuous convolution operators for visual tracking. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9909, pp. 472–488. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46454-1_29

19. Ma, C., Huang, J.B., Yang, X., Yang, M.H.: Hierarchical convolutional features for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3074–3082 (2015)

20. Danelljan, M., Bhat, G., Shahbaz Khan, F., Felsberg, M.: ECO: efficient convolution operators for tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6638–6646 (2017)

21. Gan, Q., Guo, Q., Zhang, Z., Cho, K.: First step toward model-free, anonymous object tracking with recurrent neural networks. arXiv preprint arXiv:1511.06425 (2015)

22. Kahou, S.E., Michalski, V., Memisevic, R., Pal, C., Vincent, P.: RATM: recurrent attentive tracking model. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1613–1622. IEEE (2017)

23. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4293–4302 (2016)

24. Held, D., Thrun, S., Savarese, S.: Learning to track at 100 FPS with deep regression networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 749–765. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_45

25. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.S.: Fully-convolutional Siamese networks for object tracking. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9914, pp. 850–865. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48881-3_56

26. Li, B., Yan, J., Wu, W., Zhu, Z., Hu, X.: High performance visual tracking with Siamese region proposal network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8971–8980 (2018)

27. Li, B., Wu, W., Wang, Q., Zhang, F., Xing, J., Yan, J.: SiamRPN++: evolution of Siamese visual tracking with very deep networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4282–4291 (2019)

28. Danelljan, M., Bhat, G., Khan, F.S., Felsberg, M.: ATOM: accurate tracking by overlap maximization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4660–4669 (2019)

29. Bhat, G., Danelljan, M., Gool, L.V., Timofte, R.: Learning discriminative model prediction for tracking (2019)

30. Yilmaz, A., Li, X., Shah, M.: Contour-based object tracking with occlusion handling in video acquired using mobile cameras. IEEE Trans. Pattern Anal. Mach. Intell. **26**(11), 1531–1536 (2004)

31. Gupta, D.K., Gavves, E., Smeulders, A.W.: Tackling occlusion in Siamese tracking with structured dropouts. arXiv:2006.16571 (2020)

32. Pan, J., Hu, B.: Robust occlusion handling in object tracking. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE (2007)

33. Zhou, Y., Tao, H.: A background layer model for object tracking through occlusion. In: Proceedings Ninth IEEE International Conference on Computer Vision, pp. 1079–1085. IEEE (2003)

34. Koller, D., Weber, J., Malik, J.: Robust multiple car tracking with occlusion reasoning. In: Eklundh, J.-O. (ed.) ECCV 1994. LNCS, vol. 800, pp. 189–196. Springer, Heidelberg (1994). https://doi.org/10.1007/3-540-57956-7_22

35. Lee, B.Y., Liew, L.H., Cheah, W.S., Wang, Y.C.: Occlusion handling in videos object tracking: a survey. In: IOP Conference Series: Earth and Environmental Science, vol. 18, p. 012020. IOP Publishing (2014)
36. Greenhill, D., Renno, J., Orwell, J., Jones, G.A.: Occlusion analysis: learning and utilising depth maps in object tracking. Image Vis. Comput. **26**(3), 430–441 (2008)
37. Ma, Y., Chen, Q.: Depth assisted occlusion handling in video object tracking. In: Bebis, G., et al. (eds.) ISVC 2010. LNCS, vol. 6453, pp. 449–460. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-17289-2_43
38. Ali, A., Terada, K.: A framework for human tracking using Kalman filter and fast mean shift algorithms. In: 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, pp. 1028–1033. IEEE (2009)
39. Zhao, J., Qiao, W., Men, G.Z.: An approach based on mean shift and Kalman filter for target tracking under occlusion. In: 2009 International Conference on Machine Learning and Cybernetics, vol. 4, pp. 2058–2062. IEEE (2009)
40. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
41. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
42. Kristan, M., et al.: The sixth visual object tracking vot2018 challenge results. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)
43. Moudgil, A., Gandhi, V.: Long-term visual object tracking benchmark. arXiv preprint arXiv:1712.01358 (2017)
44. Wu, Y., Lim, J., Yang, M.: Online object tracking: a benchmark. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2411–2418 (2013)
45. Gavves, E., Tao, R., Gupta, D.K., Smeulders, A.W.M.: Model decay in long-term tracking. arXiv:1908.01603 (2019)