The Thermal Infrared Visual Object Tracking VOT-TIR2016 Challenge Results

Michael Felsberg[1(✉)], Matej Kristan[2], Jiři Matas[3], Aleš Leonardis[4], Roman Pflugfelder[5], Gustav Häger[1], Amanda Berg[1,6], Abdelrahman Eldesokey[1], Jörgen Ahlberg[1,6], Luka Čehovin[2], Tomáš Vojíř[3], Alan Lukežič[2], Gustavo Fernández[5], Alfredo Petrosino[20], Alvaro Garcia-Martin[22], Andrés Solís Montero[25], Anton Varfolomieiev[15], Aykut Erdem[12], Bohyung Han[21], Chang-Ming Chang[23], Dawei Du[9], Erkut Erdem[12], Fahad Shahbaz Khan[1], Fatih Porikli[7,8,19], Fei Zhao[9], Filiz Bunyak[24], Francesco Battistone[20], Gao Zhu[8], Guna Seetharaman[17], Hongdong Li[7,8], Honggang Qi[9], Horst Bischof[11], Horst Possegger[11], Hyeonseob Nam[18], Jack Valmadre[26], Jianke Zhu[28], Jiayi Feng[9], Jochen Lang[25], Jose M. Martinez[22], Kannappan Palaniappan[24], Karel Lebeda[27], Ke Gao[24], Krystian Mikolajczyk[14], Longyin Wen[23], Luca Bertinetto[26], Mahdieh Poostchi[24], Mario Maresca[20], Martin Danelljan[1], Michael Arens[10], Ming Tang[9], Mooyeol Baek[21], Nana Fan[13], Noor Al-Shakarji[24], Ondrej Miksik[26], Osman Akin[12], Philip H.S. Torr[26], Qingming Huang[9], Rafael Martin-Nieto[22], Rengarajan Pelapur[24], Richard Bowden[27], Robert Laganière[25], Sebastian B. Krah[10], Shengkun Li[23], Shizeng Yao[24], Simon Hadfield[27], Siwei Lyu[23], Stefan Becker[10], Stuart Golodetz[26], Tao Hu[9], Thomas Mauthner[11], Vincenzo Santopietro[20], Wenbo Li[16], Wolfgang Hübner[10], Xin Li[13], Yang Li[28], Zhan Xu[28], and Zhenyu He[13]

[1] Linköping University, Linköping, Sweden
michael.felsberg@liu.se
[2] University of Ljubljana, Ljubljana, Slovenia
[3] Czech Technical University, Prague, Czech Republic
[4] University of Birmingham, Birmingham, England
[5] Austrian Institute of Technology, Seibersdorf, Austria
[6] Termisk Systemteknik AB, Linköping, Sweden
[7] ARC Centre of Excellence for Robotic Vision, Canberra, Australia
[8] Australian National University, Canberra, Australia
[9] Chinese Academy of Sciences, Beijing, China
[10] Fraunhofer IOSB, Karlsruhe, Germany
[11] Graz University of Technology, Graz, Austria
[12] Hacettepe University, Ankara, Turkey
[13] Harbin Institute of Technology, Harbin, China
[14] Imperial College London, London, UK
[15] Kyiv Polytechnic Institute, Kiev, Ukraine
[16] Lehigh University, Bethlehem, USA
[17] Naval Research Lab, Washington, D.C., USA
[18] NAVER Corp., Seongnam, South Korea
[19] Data61/CSIRO, Alexandria, Australia

© Springer International Publishing Switzerland 2016
G. Hua and H. Jégou (Eds.): ECCV 2016 Workshops, Part II, LNCS 9914, pp. 824–849, 2016.
DOI: 10.1007/978-3-319-48881-3_55

[20] Parthenope University of Naples, Naples, Italy
[21] POSTECH, Pohang, South Korea
[22] Universidad Autónoma de Madrid, Madrid, Spain
[23] University at Albany, Albany, USA
[24] University of Missouri, Columbia, USA
[25] University of Ottawa, Ottawa, Canada
[26] University of Oxford, Oxford, England
[27] University of Surrey, Guildford, England
[28] Zhejiang University, Hangzhou, China

Abstract. The Thermal Infrared Visual Object Tracking challenge 2016, VOT-TIR2016, aims at comparing short-term single-object visual trackers that work on thermal infrared (TIR) sequences and do not apply pre-learned models of object appearance. VOT-TIR2016 is the second benchmark on short-term tracking in TIR sequences. Results of 24 trackers are presented. For each participating tracker, a short description is provided in the appendix. The VOT-TIR2016 challenge is similar to the 2015 challenge, the main difference is the introduction of new, more difficult sequences into the dataset. Furthermore, VOT-TIR2016 evaluation adopted the improvements regarding overlap calculation in VOT2016. Compared to VOT-TIR2015, a significant general improvement of results has been observed, which partly compensate for the more difficult sequences. The dataset, the evaluation kit, as well as the results are publicly available at the challenge website.

Keywords: Performance evaluation · Object tracking · Thermal IR · VOT

1 Introduction

Visual tracking is sometimes considered a solved task, but many applied projects show that robust and accurate object tracking in the visual domain is highly challenging. Thus, tracking has attracted significant attention in review papers from the past two decades, e.g. [1–3] and is subject of a constantly high number (∼40 papers annually) of accepted papers in high profile conferences, such as ICCV, ECCV, and CVPR. In recent years, several performance evaluation methodologies have been established in order to assess and understand the advancements made by this large number (a few hundred) of publications. One of the pioneers for building a common ground in tracking performance evaluation is PETS [4], followed-up more recently by the Visual Object Tracking (VOT) challenges [5–7] and the Object Tracking Benchmarks [8,9].

Thermal cameras have several advantages compared to cameras for the visual spectrum: They are able to operate in total darkness, they are robust to illumination changes and shadow effects, and they reduce privacy intrusion. Historically, thermal cameras have delivered low-resolution and noisy images and were mainly

used for tracking point targets or small objects against colder backgrounds. Thus applications had often been restricted to military purposes, whereas today, thermal cameras are commonly used in civilian applications, e.g., cars and surveillance systems. Increasing image quality and decreasing price and size allow exploration of new application areas [10], often requiring methods for tracking of extended dynamic objects, also from moving platforms.

Tracking on thermal infrared (TIR) imagery has thus become an emerging niche and evaluation or comparison of methods is required. This has been addressed by VOT-TIR2015, the first TIR short-term tracking challenge [11]. This challenge resembles the VOT challenge, in the sense that the VOT-TIR challenge considers single-camera, single-target, model-free, and causal trackers, applied to short-term tracking. It has been featured as a sub-challenge to VOT2015, organized in conjunction with ICCV2015.

Since the first challenge attracted a significant number of submissions and due to required improvements of the dataset, a second VOT-TIR challenge has been initiated in conjunction with VOT2016 [12] and ECCV2016: VOT-TIR2016. The present paper summarizes this challenge, the submissions, and the obtained results. The aim of this work is to give guidance for future applications in the TIR domain and to trigger further development of methods, similar to the boosting of visual tracking methods caused by the VOT challenges. Likewise VOT2016, the dataset, the evaluation kit, as well as the results are publicly available at the challenge website http://votchallenge.net.

1.1 Related Work

In contrast to the large number of benchmarks that exist in the area of visual tracking (cf. the VOT2016 results paper [12] for several examples), TIR tracking offers few options for evaluation. For tracking in RGB sequences, the most closely related approach is obviously the VOT2016 challenge [12], as well as those of previous years [5–7].

An evaluation resembling VOT is offered by the online tracking benchmark (OTB) by Wu et al. [8,9], which is however based on different measures of performance. Trackers are compared using a precision score (the percentage of frames where the estimated bounding box is within some fixed distance to the ground truth) and a success score (the area under the curve of number of frames where the overlap is greater than some fixed percentage). This area has been shown to be equivalent to the average overlap [13,14] and is computed without restarting a failed tracker as done in VOT. For further comparisons with the VOT evaluation we refer to [7,12,15].

For TIR sequences, basically two challenges have been organized in the past. Within the series of workshops on Performance Evaluation of Tracking and Surveillance (PETS) [4], thermal infrared challenges have been organized on two occasions, 2005 and 2015. The PETS challenges addressed multiple research areas such as detection, multi-camera/long-term tracking, and behavior (threat) analysis.

In contrast, the VOT-TIR2015 challenge has focused on the problem of short-term tracking only. The challenge has been based on a newly collected dataset (LTIR) [16], as available datasets for evaluation of tracking in thermal infrared had become outdated. The lack of an accepted evaluation dataset leads often to comparisons on proprietary datasets. This and inconsistent performance measures make it difficult to systematically assess the advancement of the field. Thus, VOT-TIR2015 made use of the well-established VOT methodology [11].

The challenge had 20 participating methods and the following observations were made: (i) The relative ranking of methods differed significantly from the visual domain, which justifies a separate TIR challenge. For instance, the EDFT-based ABCD tracker [17] performed very well on VOT-TIR2015, but only moderately on VOT2015 (despite that EDFT [18] was among the top three in VOT2013). (ii) The recent progress of tracking methodology rendered the LTIR dataset being too simple for observing a significant spread of performance: the benchmark was basically saturated, at least for the top-performing methods. Thus, for the VOT-TIR2016 challenge, some of the easiest sequences from LTIR have been removed and new sequences that have been contributed by the community have been added. Furthermore and in parallel to VOT2016, the bounding box overlap estimation is constrained to the image region [12].

1.2 The VOT-TIR2016 Challenge

Similar to VOT-TIR2015, the VOT-TIR2016 challenge targets specific trackers that are required to be: (i) Causal – sequence frames have to be processed in sequential order; (ii) Short-term – trackers are not required to handle reinitialization; (iii) Model-free – pre-built models of object appearances are not allowed.

The performance of participating trackers is measured using the VOT2016 evaluation toolkit[1]. The toolkit runs the experiment in a standardized way and stores the output bounding boxes. If a tracker fails, it is re-initialized and the evaluation is continued after some few frames delay. Tracking results are analyzed using the VOT2015 evaluation methodology [7], but without rotating bounding boxes.

The rules are as always in VOT: Only a single set of results may be submitted per tracker and binaries are required for result verification. User-adjustable parameters need to be constant for all sequences and different sets of parameters do not constitute new trackers. Detecting specific sequences for choosing parameters or training networks on similar, tracking-specific datasets is not allowed. Further details regarding participation rules are available from the challenge homepage[2].

Compared to VOT2016 [12], VOT-TIR2016 is still using a simpler annotation and no fully automatic selection of sequences (as in VOT2014 [6]). The LTIR dataset (the Linköping Thermal IR dataset) [16] has been extended by a public

[1] https://github.com/vicoslab/vot-toolkit.
[2] http://www.votchallenge.net/vot2016/participation.html.

call for contributions and replacing simple LTIR sequences with community-provided sequences. A detailed description of the sequences can be found in Sect. 2.

Section 3 briefly summarizes the performance measures and evaluation methodology that resembles VOT2016 [12]. Since top-performing methods showed hardly any failures, no OTB-like no-reset experiments have been performed as done in VOT2016. Instead, a ranking comparison similar to the one in VOT-TIR2015 and a sequence difficulty analysis have been performed.

The results and their analysis are presented in Sect. 4 together with recommendations regarding trackers and a meta analysis of the challenge itself. Finally, conclusions are drawn in Sect. 5. In addition, short descriptions of all evaluated trackers can be found in Appendix A together with references to the original publications.

2 The VOT-TIR2016 Dataset

The dataset used in VOT-TIR2016 is a modification of the LTIR, the Linköping Thermal IR dataset [16], denoted LTIR2016. Sequences contained in the dataset were collected from nine different sources using ten different types of sensors. The included sequences originate from industry, universities, a research institute and two EU projects. The average sequence length is 740 frames and resolutions range from 305×225 to 1920×480 pixels.

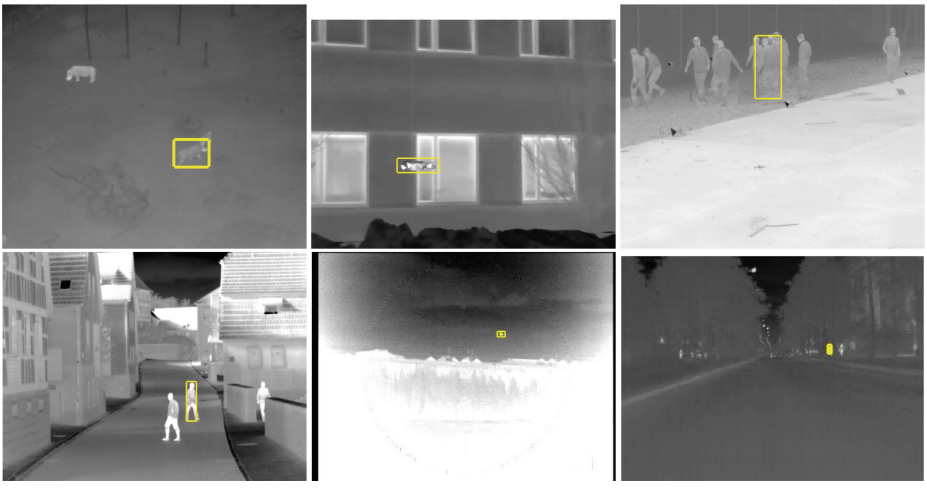

Fig. 1. Snapshots from six sequences (*Running_rhino, Quadrocopter, Crowd, Street, Bird, Trees2*) included in the LTIR2016 dataset as used in VOT-TIR2016. The ground truth bounding boxes are shown in yellow. (Color figure online)

Although some sequences in the LTIR dataset are available with 16-bit dynamic range, we only use 8-bit pixel values in the VOT-TIR2016 challenge.

This choice is motivated by the fact that several of the submitted methods cannot deal with 16-bit data. There are sequences recorded outdoors in different weather conditions and sequences recorded indoors with artificial illumination and heat sources.

Example frames from six sequences are shown in Fig. 1. Compared to VOT-TIR2015, the sequences *Crossing*, *Horse*, and *Rhino_behind_tree* have been removed. The newly added sequences are *Bird*, *Boat1*, *Boat2*, *Car2*, *Dog*, *Excavator*, *Ragged*, and *Trees2*.

In contrast to the novel annotation approach in VOT2016 [12], all benchmark annotations have been done manually in accordance with the VOT2013 annotation process [19]. Exactly one object within each sequence is annotated throughout the sequence with a bounding box that encloses the object entirely. The bounding box is allowed to vary in size but not to rotate. In addition to the bounding box annotations, local attributes are annotated frame-wise and global attributes are annotated sequence-wise.

Some attributes from VOT had to be changed or modified for VOT-TIR:

Changed attributes: Dynamics change and *temperature change* have been introduced instead of *illumination change* and *object color change*. Several cameras convert an internal constant 16-bit range into an adaptively changing 8-bit range. *Dynamics change* indicates whether the dynamic range is fixed during the sequence or not. *Temperature change* refers to changes in the thermal signature of the object during the sequence.

Modified attributes: Blur indicates blur due to motion, high humidity, rain or water on the lens instead of defocussing.

Based on the modified attribute set, the following local and global attributes are annotated:

Local attributes: The per-frame annotated local attributes are: *motion change*, *camera motion*, *dynamics change*, *occlusion*, and *size change*. The attributes are used to evaluate the performance of tracking methods on frames with specific attributes. The attributes allow also weighting the evaluation process, e.g., pool by attribute.

Global attributes: The per-sequence global attributes are: *Dynamics change*, *temperature change*, *blur*, *camera motion*, *object motion*, *background clutter*, *size change*, *aspect ratio change*, *object deformation*, and *scene complexity*.

3 Performance Measures and Evaluation Methodology

The performance measures as well as evaluation methodology for VOT-TIR2016 are identical to the ones for VOT2016, except for the OTB-like average overlap and the practical difference evaluation. Therefore, only a brief summary is given below and for details the reader is referred to [12].

Similar to VOT2016, the two weakly correlated performance measures, accuracy (A) and robustness (R), are used due to their high level of interpretability [13,14]. The accuracy measurement is computed from the overlap between the predicted bounding box and the ground truth, restricted to the image region, while the robustness measurement counts the number of tracking failures. If tracking has failed, the tracker is re-initialized with a delay of five frames. In order to reduce biased accuracy assessment, the overlap measure is continued with a further delay of ten frames.

The two primary measures A and R are fused in the expected average overlap (EAO), which is an estimator of the expected average overlap of a tracker on a new sequence of typical length. The EAO curve is given by the bounding-box-overlap averaged over a set of sequences of certain length, plotted over the sequence length N_s [7]. The EAO measure is obtained by integrating the EAO curve over an interval of typical sequence lengths of 223 to 509 frames. Overlap calculations, re-initialization, definition of a failure, and the computation of the EAO measure are further explained in [12].

As in VOT-TIR2015, the performance measures are only evaluated in the baseline experiment and we did not consider the region noise experiment for the same reasons as before [11]: Results hardly differed, experiments need more time, and reproducibility of results requires to store the seed.

4 Analysis and Results

4.1 Submitted Trackers

As in VOT-TIR2015 [11], 24 trackers were included in the VOT-TIR2016 challenge. Among them, 21 trackers were submitted to the challenge and 3 trackers were added by the VOT Committee (DSST, the VOT2014 winner, SRDCFir, which achieved the highest EAO score in VOT-TIR2015, and NCC as baseline).

The committee has used the submitted binaries/source code for result verification. All methods are briefly described below and references to the original papers are given in the Appendix A where available. All 24 VOT-TIR2016 participating trackers also participated in the VOT2016 challenge.[3]

One tracker, EBT (A.2), uses object proposals [20] for object position generation or scoring. One tracker is based on a Mean Shift tracker extension [21], PKLTF (A.5). MAD (A.4) and LOFT-Lite (A.16) are fusion based trackers. DAT (A.8) is based on tracking-by-detection learning.

Eight trackers can be classified as part-based trackers: BDF (A.3), BST (A.14), DPCF (A.1), DPT (A.20), FCT (A.15), GGTv2 (A.7), LT-FLO (A.19), and SHCT (A.12).

Seven trackers are based on the method of discriminative correlation filters (DCFs) [22,23] with various sets of image features: DSST2014 (A.22), MvCF

[3] Here, we consider SRDCF/SRDCFir and Staple/Staple-TIR being the same, despite the fact that the TIR versions use slightly different feature vectors, see Appendices A.24 and A.13.

(A.6), NSAMF (A.10), sKCF (A.17), SRDCFir (A.24), Staple-TIR (A.13), and STAPLE+ (A.11).

One tracker applies convolutional neural network (CNN) features instead of standard features, deepMKCF (A.9), and two trackers are entirely based on CNNs, TCNN (A.21) and MDNet-N (A.18). Finally, one tracker was the basic normalized cross correlation tracker NCC (A.23).

4.2 Results

The results are collected in AR-rank and AR-raw plots, pooled by sequence and averaged by attribute, c.f. Fig. 2. The sequence-pooled AR-rank plot is obtained by concatenating the results from all sequences and creating a single rank list. The attribute-normalized AR-rank plot is created by ranking the trackers over each attribute and averaging the rank lists.

The AR-raw plots are constructed without ranking. The A-values correspond to the average overlap for the whole dataset (pooled) or the attribute-normalized average overlap. The R-values correspond to the likelihood that on $S = 100$ frames the tracking will not fail (pooled over dataset or attribute-normalized). The raw values and the ranks for the pooled results are given in Table 1.

Three trackers are either very accurate or very robust (closest to the upper or right border of rank/AR plots): NCC (A.23), Staple-TIR (A.13), and EBT (A.2). Three trackers combine good accuracy and good robustness (upper right corner of rank/AR plots): MDNet-N (A.18), SRDCFir (A.24), and TCNN (A.21).

The top accuracy of NCC comes at the cost of a very high failure rate. Due to the frequent re-initializations, the NCC results are very accurate. The excellent robustness of EBT is achieved by a strategy to enlarge the predicted bounding boxes in cases of low tracking confidence. This implies some penalty on the accuracy so that EBT only achieves moderate average overlap.

The three trackers that combine good robustness and accuracy as well as further well-performing trackers are based on CNNs (TCNN, MDNet-N) and DCFs (SRDCFir, Staple-TIR, STAPLE+). SHCT combines DCFs with a part-based model and deepMKCF combines DCFs with deep features. Hence, the top-performing methods are mostly based on deep learning or DCFs.

The robustness ranks with respect to the visual attributes are shown in Fig. 3. The top three trackers of the overall assessment, EBT, SRDCFir, and TCNN, are also mostly among the top robustness ranks for the different visual attributes (exceptions SRDCFir on Dynamics_change & Occlusion and TCNN on Motion_change). The top ranks are sometimes shared with other well-performing methods: Camera_motion FCT; Dynamics_change DPT, MDNet-N, and SHCT; Empty DPT and Staple-TIR; Motion_change SHCT and STAPLE+; Occlusion MDNet-N; Size_change deepMKCF, MDNet-N, SHCT, and Staple-TIR.

The overall criterion *expected average overlap* (EAO), see Fig. 4, confirms the top-performance of SRDCFir, EBT, and TCNN. The EAO curves show that SRDCFir is consistently better than EBT in the range of typical sequence lengths. Hence, SRDCFir gives the best overall performance exactly as in the previous challenge [11]. Still, EBT is the best performing tracker submitted to

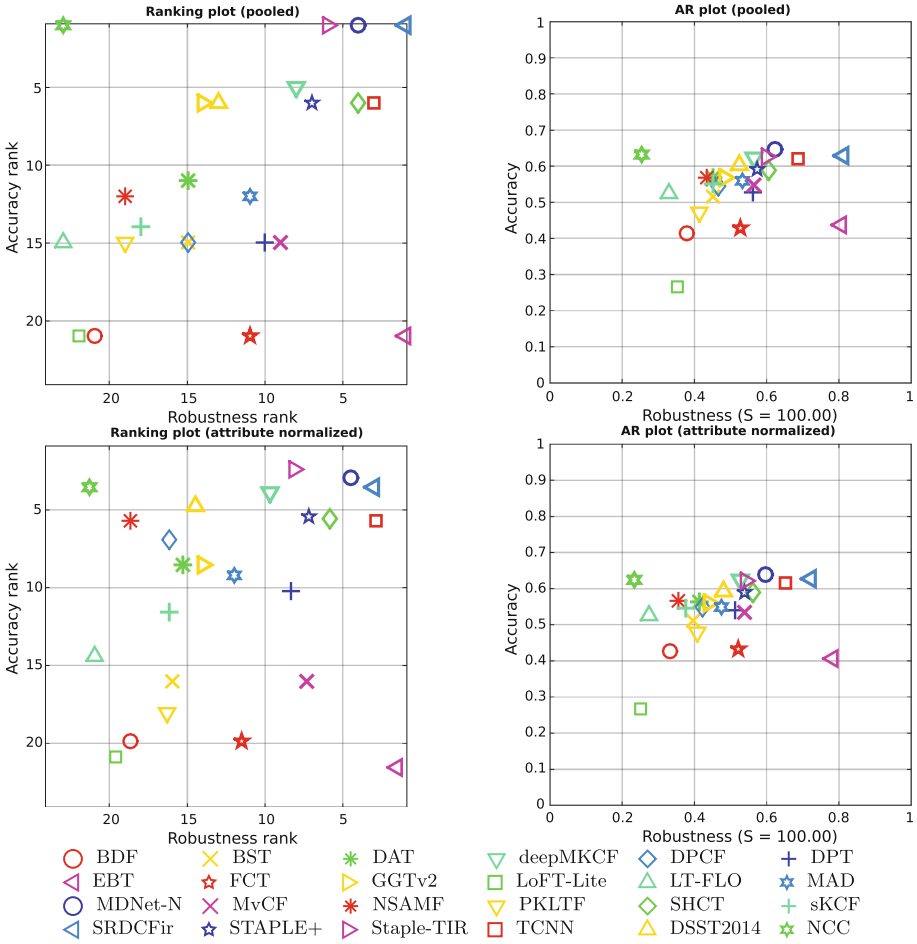

Fig. 2. The AR rank plots and AR raw plots generated by sequence pooling (upper) and by attribute normalization (below).

VOT-TIR2016. Regarding the EAO measure, TCNN is clearly inferior to the two top-ranked methods. The fact that EBT is better than TCNN regarding the EAO measure despite that it is inferior regarding accuracy (c.f. Fig. 2), underpins the importance of robustness for the expected average overlap measure.

Apart from tracking accuracy A, robustness R, and expected average overlap EAO, the tracking speed is also crucial in many realistic tracking applications. We therefore also visualize the EAO values with respect to the tracking speed measured in EFO units in Fig. 4. The vertical dashed line indicates the real-time speed (equivalent to approximately 20fps). Among the three top-performing trackers, SRDCFir comes closest to real-time performance. The top-performing

Table 1. The table shows the expected average overlap (EAO), the accuracy and robustness (S = 100) pooled values (A, R), the ranks for A and R, the tracking speed (EFO), and implementation details (M is Matlab, C is C or C++, M/C means Matlab with mex). Trackers marked with * have been verified by the committee.

	Tracker	EAO	A	R	A_{rank}	R_{rank}	EFO	Impl.
1.	**SRDCFir***	**0.364**	0.63	**0.82**	**1**	**1**	2.48	D M/C
2.	**EBT***	*0.340*	0.43	*0.81*	21	**1**	1.99	D C
3.	**TCNN***	0.287	0.62	0.69	6	*3*	0.76	S M/C
4.	**Staple-TIR***	0.264	0.63	0.60	**1**	6	14.25	D M/C
5.	**SHCT***	0.263	0.59	0.61	6	4	0.91	D M/C
6.	**MDNet-N***	0.243	**0.65**	0.63	**1**	4	0.61	S M/C
7.	**STAPLE+***	0.241	0.59	0.58	6	7	16.70	D M/C
8.	**DSST2014***	0.236	0.60	0.53	6	13	11.29	D M
9.	**MvCF***	0.231	0.55	0.57	15	9	27.83	D M
10.	**DPT***	0.219	0.53	0.57	15	10	11.40	D M/C
11.	**deepMKCF**	0.213	0.62	0.57	*5*	8	2.36	S M/C
12.	**MAD***	0.211	0.56	0.54	12	11	12.54	D C
13.	**GGTv2***	0.197	0.57	0.49	6	14	0.93	S M/C
14.	**NSAMF***	0.192	0.57	0.44	12	19	26.27	D M/C
15.	**DPCF***	0.191	0.54	0.47	15	15	2.73	D M/C
16.	**sKCF***	0.188	0.55	0.46	14	18	*135.64*	D C
17.	**FCT***	0.186	0.43	0.53	21	11	116.33	D C
18.	**LT-FLO**	0.163	0.52	0.33	15	23	2.16	S M/C
19.	**DAT***	0.162	0.57	0.46	11	15	15.71	D M
20.	**NCC***	0.160	*0.63*	0.26	**1**	23	59.49	D M
21.	**BDF***	0.147	0.41	0.38	21	21	**189.41**	D C
22.	**PKLTF***	0.141	0.47	0.42	15	19	45.99	D C
23.	**BST***	0.140	0.51	0.46	15	15	9.66	S C
24.	**LoFT-Lite***	0.107	0.26	0.36	21	22	1.30	D M/C

tracker in terms of EAO among the trackers that exceed the real-time threshold is MvCF (A.6).

4.3 TIR-Specific Analysis and Results

Likewise VOT-TIR2015, we analyze the effect of the differences between RGB sequences and TIR sequences on the ranking of the trackers [11]. For this purpose, the joint ranking for VOT and VOT-TIR is generated for all VOT-TIR trackers (see Footnote 3), c.f. Fig. 5. The dashed lines are the margin of a rank-change by more than three positions. Any change of rank within this margin is considered insignificant and only eight trackers change their rank by more than three positions.

The most dramatic change occurs for BST (A.14), which ranks 23 in VOT-TIR, but 35 (out of 70) in VOT, corresponding to rank 14 within the set of 24

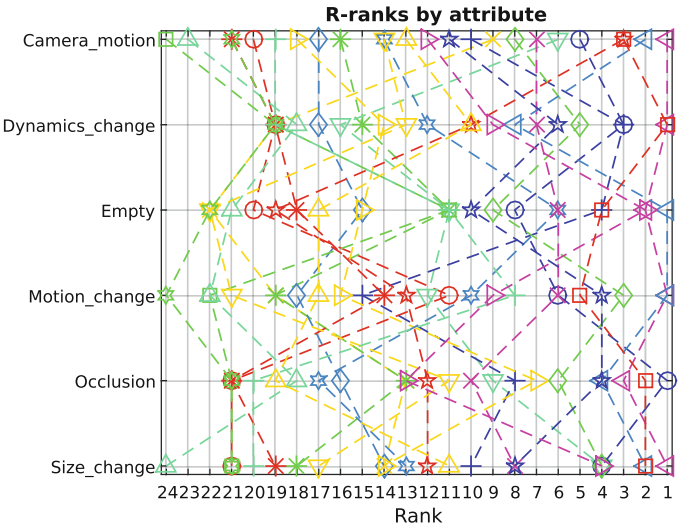

Fig. 3. Robustness plots with respect to the visual attributes. See Fig. 2 for legend.

trackers. Other trackers that perform significantly worse in VOT-TIR are DAT (A.8, 19 vs. 31/12) and GGTv2 (A.7, 13 vs. 19/8).

On the other hand, DSST2014 (A.22, 8 vs. 43/16), MvCF (A.6, 9 vs. 42/15), SRDCF(ir) (A.24, 1 vs. 17/7), LT-FLO (A.19, 18 vs. 62/22), and NCC (A.23, 20 vs. 70/24) perform significantly better on VOT-TIR than on VOT according to the relative ranking.

Similar as for the overall performance, it is difficult to identify a systematic correlation between improvement and type of tracking methods. Tracking methods that do not rely on color (e.g. DSST2014, SRDCFir, NCC) are likely to perform better on TIR sequences than color-based methods (e.g. DAT, GGTv2).

Also the size of targets differ between VOT (larger) and VOT-TIR (smaller) and scale variations need to be modeled (e.g. DSST2014, MvCF, SRDCFir). It is also believed that the tuning of input features is highly relevant for changes of performance. Methods that are highly tuned for VOT2016 and applied to VOT-TIR2016 as they are, are more likely to perform inferior compared to methods that use specific TIR-suited features, e.g. SRDCFir (A.24). In general, HOG features seem to be highly suitable for TIR.

Finally, the dramatic difference in ranking for BST need to be investigated further, as it cannot be explained by previous arguments.

One limitation of VOT-TIR2015 has been the saturation of results: several of the LTIR sequences are so simple to track that hardly any of the participating methods failed on them [11]. Therefore, the three easiest sequences have been removed and eight new sequences have been added, c.f. Sect. 2. In the difficulty analysis 2015, only three sequences were considered challenging and twelve were easy.

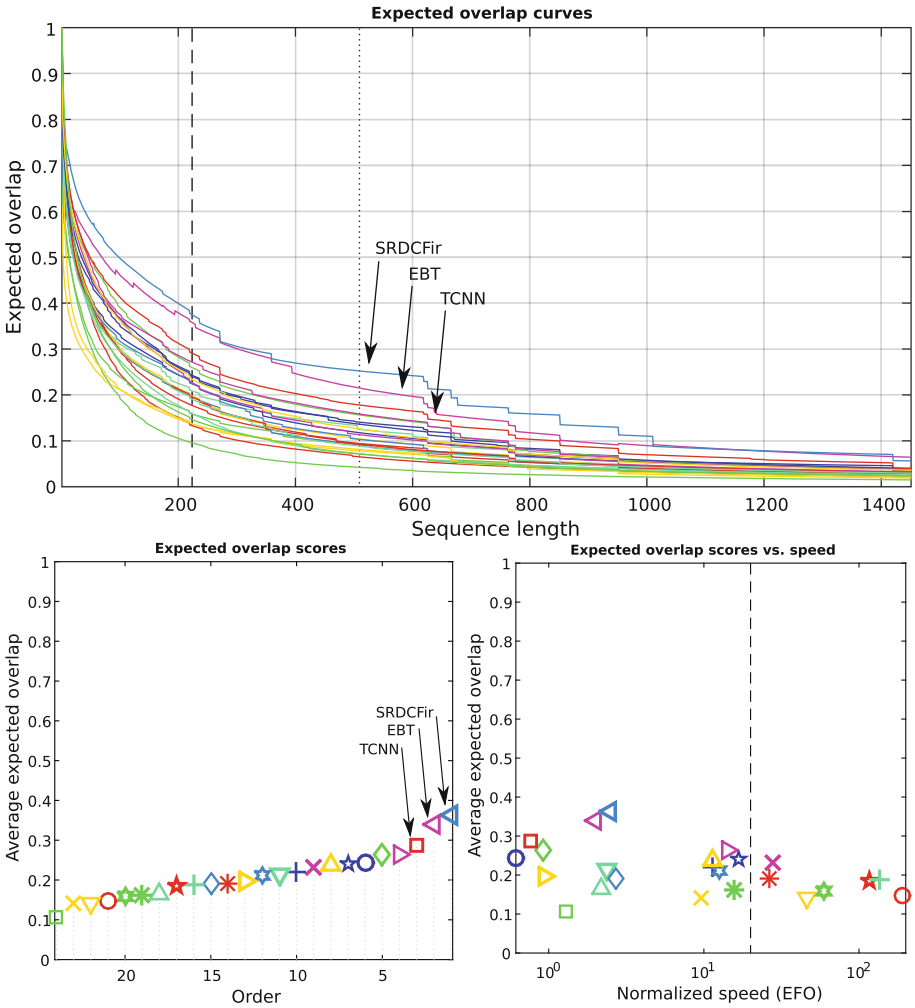

Fig. 4. Expected average overlap curve (above), expected average overlap graph (below left) with trackers ranked from right to left, and expected average overlap scores w.r.t. the tracking speed in EFO units (below right). The right-most tracker in the EAO-graph is the top-performing according to the VOT-TIR2016 expected average overlap values. See Fig. 2 for legend. The vertical lines in the upper plot show the range of typical sequence lengths. The dashed vertical line in the lower right plot denotes the estimated real-time performance threshold of 20 EFO units.

If A_f is the average number of trackers that failed per frame and M_f is the maximum number of trackers that failed at a single frame, sequences with $A_f \leq 0.04$ and $M_f \leq 7$ are considered easy and sequences with $A_f \geq 0.06$ and $M_f \geq 14$ are considered challenging. In the extended dataset, eight sequences are challenging and nine are easy (c.f. Table 2). The average difficulty score

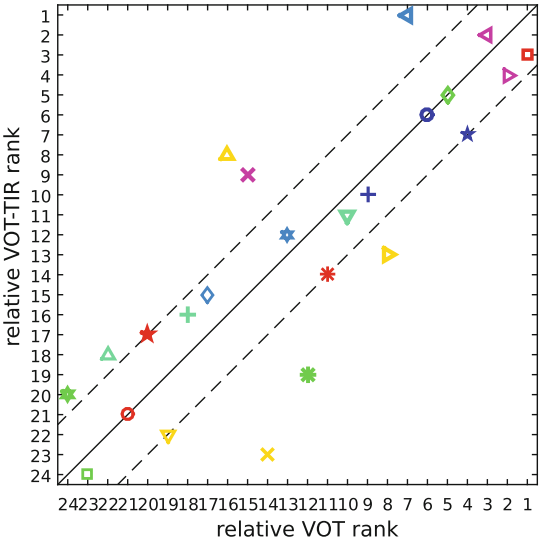

Fig. 5. Comparison of relative ranking of the 24 VOT-TIR trackers in VOT. See Fig. 2 for legend

(1.0 hardest, 5.0 easiest) is reduced from 4.0 (easy) to 3.3 (intermediate), which means that the new dataset is significantly more challenging than LTIR. This also shows in the EAO score of SRDCFir, which has been significantly higher in VOT-TIR2015 (0.70 vs. 0.364) [11].

Table 2. Difficulty analysis of sequences from VOT-TIR2015 and 2016. A score smaller than 3 means *challenging*, a score larger or equal four means *easy*. Mean difficulty VOT-TIR2015: 4.0, VOT-TIR2016: 3.3.

	VOT-TIR	Crowd	Quadrocopter	Quadrocopter2	Garden	Mixed_distractors	Saturated	Selma	Street	Birds	Crouching	Jacket	Hiding	Car	Crossing	Depthwise_crossing	Horse	Rhino_behind_tree	Running_rhino	Soccer	Trees	Bird	Boat1	Boat2	Car2	Dog	Excavator	Ragged	Trees2
2015	2.0	2.5	2.5	3.0	3.0	3.5	3.5	3.5	4.0	4.0	4.0	4.5	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	–	–	–	–	–	–	–	–	–
2016	2.0	2.5	1.5	2.0	3.5	4.5	3.5	3.5	4.5	3.5	4.0	5.0	4.5	–	5.0	–	–	4.5	4.5	3.5	1.5	4.0	3.0	2.0	3.5	3.0	2.5	1.5

A major limitation of the current evaluation methodology used in VOT-TIR2016 is caused by the criterion of a failure: A failure is reported if the ground truth bounding box and the predicted bounding box do not overlap [5]. As a result, trackers that systematically overestimate the size of the tracked target in case of low confidence, are highly likely to never drop the target at the cost of a low accuracy A, c.f. Fig. 6.

Fig. 6. Example from sequence *Boat2*: A report of failure is avoided by increasing the predicted bounding box to the whole image.

If a tracker succeeds to estimate the confidence for successful tracking well and increases the bounding box only in those cases, a very low failure rate can be obtained at the cost of still acceptable accuracy. The joint measure of EAO score will then be superior to methods that have much better accuracy, but slightly more failures.

In order to limit the effect of arbitrarily large bounding boxes, we suggest to modify the failure test in the following way: We require the overlap to be above the quantization level if we rescale the intersection with the ratio of the bounding boxes. Let A_t^G and A_t^T be the ground truth and predicted bounding boxes, respectively. Let further $|A_t|$ be the size of the bounding box in pixels. The criterion for successful tracking currently used is

$$\frac{|A_t^G \cap A_t^T|}{|A_t^G \cup A_t^T|} > 0 \tag{1}$$

and the suggested new criterion reads

$$|A_t^G \cap A_t^T| \frac{|A_t^G|}{|A_t^T|} > \frac{1}{2} \ . \tag{2}$$

Since the rules of VOT-TIR2016 cannot be changed retrospectively, we will not provide any results according to the new criterion within VOT-TIR2016.

5 Conclusions

The VOT-TIR2016 challenge has received 21 submissions and compared in total 24 trackers, which is a successful continuation of the first challenge. The extended dataset is significantly more challenging such that the results of the challenge give a better guidance to future research within TIR tracking than VOT-TIR2015.

The best overall performance has been achieved by SRDCFir, followed by EBT, as best performing submitted method, and TCNN. The analysis of

results shows that the performance of some trackers differ significantly between VOT2016 and VOT-TIR2016. However, to be top-ranked in VOT-TIR2016 requires a strong result in VOT2016. Modeling of scale-variations and suitable features are necessary to achieve top results. The strongest two tracking methodologies within the benchmark are CNN-based and DCF-based trackers, where several trackers are among the top-performers.

For future challenges, the annotation and evaluation need to be adapted to the current VOT standard: multiple annotations and rotating bounding boxes. The failure criterion might need to be modified as suggested. Also challenges with mixed sequences (RGB and TIR) might be interesting to perform.

Acknowledgments. This work was supported in part by the following research programs and projects: Slovenian research agency research programs P2-0214, P2-0094, Slovenian research agency projects J2-4284, J2-3607, J2-2221 and European Union 7th Framework Programme under grant agreement 257906. J. Matas and T. Vojir were supported by CTU Project SGS13/142/OHK3/2T/13 and by the Technology Agency of the Czech Republic project TE01020415 (V3C – Visual Computing Competence Center). M. Felsberg, G. Häger, and A. Eldesokey were supported by the Wallenberg Autonomous Systems Program WASP, the Swedish Foundation for Strategic Research through the project CUAS, and the Swedish Research Council trough the project EMC[2]. J. Ahlberg and A. Berg were supported by the European Union 7th Framework Programme under grant agreement 312784 (P5) and the Swedish Research Council through the contract D0570301. Some experiments where run on GPUs donated by NVIDIA.

A Submitted Trackers

This appendix contains short descriptions of all trackers from the challenge.

A.1 Deformable Part-based Tracking by Coupled Global and Local Correlation Filters (DPCF)

O. Akin, E. Erdem, A. Erdem, K. Mikolajczyk
oakin25@gmail.com, {erkut, aykut}@cs.hacettepe.edu.tr,
k.mikolajczyk@imperial.ac.uk

DPCF is a deformable part-based correlation filter tracking approach which depends on coupled interactions between a global filter and several part filters. Specifically, local filters provide an initial estimate, which is then used by the global filter as a reference to determine the final result. Then, the global filter provides a feedback to the part filters regarding their updates and the related deformation parameters. In this way, DPCF handles not only partial occlusion but also scale changes. The reader is referred to [24] for details.

A.2 Edge Box Tracker (EBT)

G. Zhu, F. Porikli, H. Li
{gao.zhu, fatih.porikli, hongdong.li}@anu.edu.au

EBT tracker is not limited to a local search window and has ability to probe efficiently the entire frame. It generates a small number of 'high-quality' proposals by a novel instance-specific objectness measure and evaluates them against the object model that can be adopted from an existing tracking-by-detection approach as a core tracker. During the tracking process, it updates the object model concentrating on hard false-positives supplied by the proposals, which help suppressing distractors caused by difficult background clutters, and learns how to re-rank proposals according to the object model. Since the number of hypotheses the core tracker evaluates is reduced significantly, richer object descriptors and stronger detectors can be used. More details can be found in [25].

A.3 Best Displacement Flow (BDF)

M. Maresca, A. Petrosino
mariomaresca@hotmail.it, petrosino@uniparthenope.it

Best Displacement Flow (BDF) is a short-term tracking algorithm based on the same idea of Flock of Trackers [26] in which a set of local tracker responses are robustly combined to track the object. Firstly, BDF performs a clustering to identify the best displacement vector which is used to update the object's bounding box. Secondly, BDF performs a procedure named Consensus-Based Reinitialization used to reinitialize candidates which were previously classified as outliers. Interested readers are referred to [27] for details.

A.4 Median Absolute Deviation Tracker (MAD)

S. Becker, S. Krah, W. Hübner, M. Arens
{*stefan.becker, sebastian.krah, wolfgang.huebner, michael.arens*}*@iosb.fraunhofer.de*

The key idea of the MAD tracker [28] is to combine several independent and heterogeneous tracking approaches and to robustly identify an outlier subset based on the Median Absolute Deviation (MAD) measure. The MAD fusion strategy is very generic and it only requires frame-based target bounding boxes as input and thus can work with arbitrary tracking algorithms. The overall median bounding box is calculated from all trackers and the deviation or distance of a sub-tracker to the median bounding box is calculated using the Jaccard-Index. Further, the MAD fusion strategy can also be applied for combining several instances of the same tracker to form a more robust swarm for tracking a single target. For this experiments the MAD tracker is set-up with a swarm of KCF [23] trackers in combination with the DSST [29] scale estimation scheme. The reader is referred to [28] for details.

A.5 Point-Based Kanade Lukas Tomasi Colour-Filter (PKLTF)

R. Martin-Nieto, A. Garcia-Martin, J. M. Martinez
{*rafael.martinn, alvaro.garcia, josem.martinez*}*@uam.es*

PKLTF [30] is a single-object long-term tracker that supports high appearance changes in the target, occlusions, and is also capable of recovering a target lost during the tracking process. PKLTF consists of two phases: The first one uses the Kanade Lukas Tomasi approach (KLT) [31] to choose the object features (using colour and motion coherence), while the second phase is based on mean shift gradient descent [32] to place the bounding box into the position of the object. The object model is based on the RGB colour and the luminance gradient and it consists of a histogram including the quantized values of the colour components, and an edge binary flag. The interested reader is referred to [30] for details.

A.6 A multi-view model for visual tracking via correlation filters (MvCF)

Z. He, X. Li, N. Fan
zyhe@hitsz.edu.cn, hitlixin@126.com, nanafanhit@gmail.com

The multi-view correlation filter tracker (MvCF tracker) fuses several features and selects the more discriminative features to enhance the robustness. More specifically, for the VOT-TIR dataset, the histogram of oriented gradients (HOG) and gray value features play more important roles in tracking than color features. The combination of the multiple views is conducted by the Kullback-Leibler (KL) divergences. In addition, a simple but effective scale-variation detection mechanism is provided, which strengthens the stability of scale variation tracking.

A.7 Geometric Structure Hyper-Graph based Tracker Version 2 (GGTv2)

T. Hu, D. Du, L. Wen, W. Li, H. Qi, S. Lyu
{*yihouxiang, cvdaviddo, lywen.cv.workbox, wbli.app, honggangqi.cas, heizi.lyu*}*@gmail.com*

GGTv2 is an improvement of GGT [33] by combining the scale adaptive kernel correlation filter [34] and the geometric structure hyper-graph searching framework to complete the object tracking task. The target object is represented by a geometric structure hyper-graph that encodes the local appearance of the target with higher-order geometric structure correlations among target parts and a bounding box template that represents the global appearance of the target. The tracker use HSV colour histogram and LBP texture to calculate the appearance similarity between associations in the hyper-graph. The templates of correlation filter is calculated by HOG and colour name according to [34].

A.8 Distractor Aware Tracker (DAT)

H. Possegger, T. Mauthner, H. Bischof
{*possegger, mauthner, bischof*}*@icg.tugraz.at*

The Distractor Aware Tracker is an appearance-based tracking-by-detection approach. To demonstrate its performance on the VOT-TIR dataset, DAT learns a discriminative model from the grey scale image to distinguish the object from its surrounding region. Additionally, a distractor-aware model term suppresses visually distracting regions whenever they appear within the field-of-view, thus reducing tracker drift. The reader is referred to [35] for details.

A.9 Deep Multi-kernelized Correlation Filter (deepMKCF)

J. Feng, F. Zhao, M. Tang
{*jiayi.feng, fei.zhao, tangm*}*@nlpr.ia.ac.cn*

deepMKCF tracker is the MKCF [36] with deep features extracted by using VGG-Net [37]. deepMKCF tracker combines the multiple kernel learning and correlation filter techniques and it explores diverse features simultaneously to improve tracking performance. In addition, an optimal search technique is also applied to estimate object scales. The multi-kernel training process of deepMKCF is tailored accordingly to ensure tracking efficiency with deep features.

A.10 NSAMF (NSAMF)

Y. Li, J. Zhu
{*liyang89, jkzhu*}*@zju.edu.cn*

NSAMF is an improved version of the previous method SAMF [34]. To further exploit color information, NSAMF employs color probability map, instead of color name, as color based feature to achieve more robust tracking results. In addition, multi-models based on different features are integrated to vote the final position of the tracked target.

A.11 An Improved STAPLE Tracker with Multiple Feature Integration (STAPLE+)

Z. Xu, Y. Li, J. Zhu
xuzhan2012@whu.edu.cn, {*liyang89, jkzhu*}*@zju.edu.cn*

An improved version of STAPLE tracker [38] by integrating multiple features is presented. Besides extracting HOG feature from merely gray-scale image, we also extract HOG feature from color probability map, which can exploit color information better. The final response map is thus a fusion of different features.

A.12 Structure Hyper-graph Based Correlation Filter Tracker (SHCT)

L. Wen, D. Du, S. Li, C.-M. Chang, S. Lyu, Q. Huang
{*lywen.cv.workbox, cvdaviddo, shengkunliluo, mingching, heizi.lyu*}*@gmail.com, qmhuang@jdl.ac.cn*

SHCT tracker constructs a structure hyper-graph model similar to [39] to extract the motion coherence of target parts. The tracker also computes a part confidence map based on the extracted dense subgraphs on the constructed structure hyper-graph, which indicates the confidence score of the part belonging to the target. SHCT uses HSV colour histogram and LBP feature to calculate the appearance similarity between associations in the hyper-graph. Finally, the tracker combines the response maps of correlation filter and structure hyper-graph in a linear way to find the optimal target state (i.e., target scale and location). The templates of correlation filter are calculated by HOG and colour name according to [34]. The appearance models of correlation filter and structure hyper-graph are updated to ensure the tracking performance.

A.13 Sum of Template and Pixel-wise LEarners TIR (Staple-TIR)

L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, P. H. S. Torr
{*luca, jvlmdr*}*@robots.ox.ac.uk, stuart.golodetz@ndcn.ox.ac.uk,* {*ondrej.miksik, philip.torr*}*@eng.ox.ac.uk*

Staple is a tracker that combines two image patch representations that are sensitive to complementary factors to learn a model that is inherently robust to both intensity changes and deformations. To maintain real-time speed, two independent ridge-regression problems are solved, exploiting the inherent structure of each representation. Staple combines the scores of two models in a dense translation search, enabling greater accuracy. A critical property of the two models is that their scores are similar in magnitude and indicative of their reliability, so that the prediction is dominated by the more confident. Staple-TIR uses one-dimensional instead of three-dimensional histograms and has different hyperparameters as Staple Tracker. For more details, we refer the reader to [40].

A.14 Best Structured Tracker (BST)

F. Battistone, A. Petrosino, V. Santopietro
{*battistone.francesco, vinsantopietro*}*@gmail.com, petrosino@uniparthenope.it*

BST is based on the idea of Flock of Trackers [41]: a set of local trackers tracks a little patch of the original target and then the tracker combines their information in order to estimate the resulting bounding box. Each local tracker separately analyzes the features extracted from a set of samples and then classifies them using a structured Support Vector Machine as Struck [41]. Once having predicted local target candidates, an outlier detection process is computed by analyzing the displacements of local trackers. Trackers that have been labeled as outliers are reinitialized. At the end of this process, the new bounding box is calculated using the Convex Hull technique.

A.15 Optical flow clustering tracker (FCT)

A. Varfolomieiev
a.varfolomieiev@kpi.ua

FCT is based on the same idea as the best displacement tracker (BDF) [27]. It uses pyramidal Lucas-Kanade optical flow algorithm to track individual points of an object at several pyramid levels. The results of the point tracking are clustered in the same way as in the BDF [27] to estimate the best object displacement. The initial point locations are generated by the FAST detector [42]. The tracker estimates a scale and an in-plane rotation of the object. These procedures are similar to the scale calculation of the median flow tracker [43], except that the clustering is used instead of median. In case of rotation calculation angles between the respective point pairs are clustered. In contrast to BDF, the FCT does not use consensus-based reinitialization. The current implementation of FCT calculates the optical flow only in the objects region, which is four times larger than the initial bounding box of the object, and thus speeds up the tracker with respect to its previous version [7].

A.16 Likelihood of Features Tracking-Lite (LoFT-Lite)

M. Poostchi, K. Palaniappan, F. Bunyak, G. Seetharaman, R. Pelapur, K. Gao, S. Yao, N. Al-Shakarji

mpoostchi@mail.missouri.edu, {pal, bunyak}@missouri.edu, guna@ieee.org {rvpnc4, kg954, syyh4, nmahyd}@missouri.edu,

LoFT (Likelihood of Features Tracking)-Lite [44] is an appearance based single object tracker that employs a rich set of low level image feature descriptors that account for intensity, edge, shape and motion properties of the target. The feature likelihood maps are computed using sliding window search comparing target and reference feature histograms of intensity, gradient magnitude, gradient orientation, and shape information based on the eigenvalues of the Hessian matrix. Intensity and gradient magnitude normalized cross-correlation likelihood maps are also used to incorporate spatial information. Moreover, for stationary cameras LoFT can take advantage of its flux tensor motion module to robustly estimate the location of moving objects [45]. A parts-based target model is added into LoFT to provide a set of patch-based maximum likelihood maps. This increases tracking robustness to partial occlusions and compensates for orderless nature of histogram-based features. The integral histogram method accelerates computation of the parts-based sliding window histograms [46]. LoFT performs feature fusion using a foreground-background model by comparing the current target appearance with the model inside the search region [47]. LOFT-Lite also incorporates an adaptive orientation-based Kalman prediction update to restrict the search region which reduces sensitivity to abrupt motion changes and decreases computational cost [48].

A.17 Scalable Kernel Correlation Filter with Sparse Feature Integration (sKCF)

A. Solís Montero, J. Lang, R. Laganière
asolismo@uottawa.ca, {jlang, laganier}@eecs.uottawa.ca

sKCF [49] extends the Kernalized Correlation Filter (KCF) framework by introducing an adjustable Gaussian window function and keypoint-based model for scale estimation to deal with the fixed size limitation in the Kernelized Correlation Filter along with some performace enhancements. In the submission, a model learning strategy is introduced to the original sKCF [49] which updates the model only for highly similar KCF responses of the tracked region as to the model. This potentially limits model drift due to temporary disturbances or occlusions. The original sKCF always updates the model in each frame.

A.18 Multi-Domain Convolutional Neural Network Tracker (MDNet-N)

H. Nam, M. Baek, B. Han
{*namhs09, mooyeol, bhhan*}*@postech.ac.kr*

This algorithm is a variation of MDNet [50], which does not pre-train CNNs with other tracking datasets. The network is initialised using the ImageNet [51]. The new classification layer and the fully connected layers within the shared layers are then fine-tuned online during tracking to adapt to the new domain. The online update is conducted to model long-term and short-term appearance variations of a target for robustness and adaptiveness, respectively, and an effective and efficient hard negative mining technique is incorporated in the learning procedure. This experiment result shows that the online tracking framework scheme of MDNet is still effective without multi-domain training.

A.19 Long Term Featureless Object Tracker (LT-FLO)

K. Lebeda, S. Hadfield, J. Matas, R. Bowden
{*k.lebeda, s.hadfield*}*@surrey.ac.uk, matas@cmp.felk.cvut.cz, r.bowden@surrey.ac.uk*

The tracker is based on and extends previous work of the authors on tracking of texture-less objects [52]. It significantly decreases reliance on texture by using edge-points instead of point features. LT-FLO uses correspondences of lines tangent to the edges and candidates for a correspondence are all local maxima of gradient magnitude. An estimate of the frame-to-frame transformation similarity is obtained via RANSAC. When the confidence is high, the current state is learnt for future corrections. On the other hand, when a low confidence is achieved, the tracker corrects its position estimate restarting the tracking from previously stored states. LT-FLO tracker also has a mechanism to detect disappearance of the object, based on the stability of the gradient in the area of projected edge-points. The interested reader is referred to [53] for details.

A.20 Deformable Part Correlation Filter Tracker (DPT)

A. Lukežič, L. Čehovin, M. Kristan
alan.lukezic@fri.uni-lj.si, luka.cehovin@fri.uni-lj.si, matej.kristan@fri.uni-lj.si

DPT is a part-based correlation filter composed of a coarse and mid-level target representations. Coarse representation is responsible for approximate target localization and uses HOG as well as colour features. The mid-level representation is a deformable parts correlation filter with fully-connected parts topology and applies a novel formulation that threats geometric and visual properties within a single convex optimization function. The mid level as well as coarse level representations are based on the kernelized correlation filter from [23]. The reader is referred to [54] for details.

A.21 Tree-Structured Convolutional Neural Network Tracker (TCNN)

H. Nam, M. Baek, B. Han
{*namhs09, mooyeol, bhhan*}*@postech.ac.kr*

TCNN maintains multiple target appearance models based on CNNs in a tree structure to preserve model consistency and handle appearance multi-modality effectively. TCNN tracker consists of two main components, state estimation and model update. When a new frame is given, candidate samples around the target state estimated in the previous frame are drawn, and the likelihood of each sample based on the weighted average of the scores from multiple CNNs is computed. The weight of each CNN is determined by the reliability of the path along which the CNN has been updated in the tree structure. The target state in the current frame is estimated by finding the candidate with the maximum likelihood. After tracking a predefined number of frames, a new CNN is derived from an existing one, which has the highest weight among the contributing CNNs to target state estimation. Interested readers are referred to [55] for details.

A.22 Discriminative Scale Space Tracker (DSST2014)

Authors implementation. Submitted by VOT Committee

The Discriminative Scale Space Tracker (DSST) [29] extends the Minimum Output Sum of Squared Errors (MOSSE) tracker [22] with robust scale estimation. The DSST additionally learns a one-dimensional discriminative scale filter, that is used to estimate the target size. For the translation filter, the intensity features employed in the MOSSE tracker is combined with a pixel-dense representation of HOG-features.

A.23 Normalized Cross-Correlation (NCC)

Submitted by VOT Committee

The NCC tracker is a VOT2016 baseline tracker and follows the very basic idea of tracking by searching for the best match between a static grayscale template and the image using normalized cross-correlation.

A.24 Spatially Regularized Discriminative Correlation Filter Tracker for IR (SRDCFir)

Authors implementation. Submitted by VOT Committee

SRDCFir adapts the SRDCF approach proposed in [56] to thermal infrared data. Standard Discriminative Correlation Filter (DCF) based trackers such as [23,29,57] suffer from the inherent periodic assumption when using circular correlation. The resulting periodic boundary effects leads to inaccurate training samples and a restricted search region. The SRDCF mitigates these problems by introducing a spatial regularization function that penalizes filter coefficients residing outside the target region. This allows the size of the training and detection samples to be increased without affecting the effective filter size. By selecting the spatial regularization function to have a sparse Discrete Fourier Spectrum, the filter is efficiently optimized directly in the Fourier domain. Instead of solving for an approximate filter, as in previous DCF based trackers (e.g. [23,29,57]), the SRDCF employs an iterative optimization based on Gauss-Seidel that converges to the exact filter. The detection step employs a sub-grid location estimation. In addition to the HOG features used in [56], SRDCFir also employs channel coded intensity features. SRDCFir also employs a motion feature channel, computed by thresholding the difference between the current and previous frame. The result is a binary image that indicates if a pixel has changed its value compared to the previous frame. The intensity and motion features are averaged over the 4×4 HOG cells and then concatenated, giving a 43 dimensional feature vector at each cell.

References

1. Gavrila, D.M.: The visual analysis of human movement: a survey. Comp. Vis. Image Underst. **73**(1), 82–98 (1999)
2. Moeslund, T.B., Hilton, A., Kruger, V.: A survey of advances in vision-based human motion capture and analysis. Comp. Vis. Image Underst. **103**(2–3), 90–126 (2006)
3. Li, X., Hu, W., Shen, C., Zhang, Z., Dick, A.R., Van den Hengel, A.: A survey of appearance models in visual object tracking arXiv:1303.4803 [cs.CV] (2013)
4. Young, D.P., Ferryman, J.M.: Pets metrics: On-line performance evaluation service. In: ICCCN 2005 Proceedings of the 14th International Conference on Computer Communications and Networks, pp. 317–324 (2005)
5. Kristan, M., Pflugfelder, R., Leonardis, A., Matas, J., Porikli, F., Cehovin, L., Nebehay, G., Fernández, G., Vojir, T. et al.: The visual object tracking vot2013 challenge results. In: ICCV 2013 Workshops, Workshop on Visual Object Tracking Challenge, pp. 98–111 (2013)
6. Kristan, M., et al.: The visual object tracking vot2014 challenge results. In: Agapito, L., et al. (eds.) ECCV 2014 Workshops. LNCS, vol. 8926, pp. 191–217. Springer, Heidelberg (2014)
7. Kristan, M., Matas, J., Leonardis, A., Felsberg, M., et al.: The visual object tracking vot2015 challenge results. In: ICCV 2015 Workshops, Workshop on Visual Object Tracking Challenge (2015)

8. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: a benchmark. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2013)
9. Wu, Y., Lim, J., Yang, M.: Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1834–1848 (2015)
10. Gade, R., Moeslund, T.B.: Thermal cameras and applications: a survey. Mach. Vis. Appl. **25**(1), 245–262 (2014)
11. Felsberg, M., Berg, A., Häger, G., Ahlberg, J., et al.: The thermal infrared visual object tracking VOT-TIR2015 challenge results. In: ICCV 2015 Workshop Proceedings, VOT 2015 Workshop (2015)
12. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., et al.: The visual object tracking VOT2016 challenge results. In: Jegou, H., Hua, G. (eds.) ECCV 2016 Workshops. LNCS, vol. 9914, pp. 777–823. Springer, Heidelberg (2016)
13. Čehovin, L., Kristan, M., Leonardis, A.: Is my new tracker really better than yours? In: WACV 2014: IEEE Winter Conference on Applications of Computer Vision (2014)
14. Čehovin, L., Leonardis, A., Kristan, M.: Visual object tracking performance measures revisited arXiv:1502.05803 [cs.CV] (2013)
15. Kristan, M., Pflugfelder, R., Leonardis, A., Matas, J., Porikli, F., Cehovin, L., Nebehay, G., Fernandez, G., Vojir, T.: The vot2013 challenge: overview and additional results. In: Computer Vision Winter Workshop (2014)
16. Berg, A., Ahlberg, J., Felsberg, M.: A thermal object tracking benchmark. In: 12th IEEE International Conference on Advanced Video- and Signal-based Surveillance, Karlsruhe, Germany, 25–28 August 2015. IEEE (2015)
17. Berg, A., Ahlberg, J., Felsberg, M.: Channel coded distribution field tracking for thermal infrared imagery. In: IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS) (2016)
18. Felsberg, M.: Enhanced distribution field tracking using channel representations. In: Visual Object Tracking Challenge VOT 2013, In conjunction with ICCV 2013 (2013)
19. Kristan, M., Pflugfelder, R., Leonardis, A., Matas, J., Porikli, F., Čehovin, 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., Helw, M.E., 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.: The Visual Object Tracking VOT2013 challenge results. In: ICCV Workshops, pp. 98–111 (2013)
20. Zitnick, C.L., Dollár, P.: Edge boxes: locating object proposals from edges. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 391–405. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10602-1_26
21. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Trans. Pattern Anal. Mach. Intell. **25**(5), 564–577 (2003)
22. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2010)
23. Henriques, J., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 583–596 (2015)

24. Akin, O., Erdem, E., Erdem, A., Mikolajczyk, K.: Deformable part-based tracking by coupled global and local correlation filters. J. Vis. Commun. Image Represent. **38**, 763–774 (2016)
25. Zhu, G., Porikli, F., Li, H.: Beyond local search: tracking objects everywhere with instance-specific proposals. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
26. Vojir, T., Matas, J.: Robustifying the flock of trackers. In: Computer Vision Winter Workshop, pp. 91–97. IEEE (2011)
27. Maresca, M., Petrosino, A.: Clustering local motion estimates for robust and efficient object tracking. In: Agapito, L., et al. (eds.) ECCV 2014 Workshops. LNCS, vol. 8926, pp. 244–253. Springer, Heidelberg (2014)
28. Becker, S., Krah, S.B., Hübner, W., Arens, M.: Mad for visual tracker fusion. In: SPIE Proceedings Optics and Photonics for Counterterrorism, Crime Fighting, and Defence 9995 (2016, to appear)
29. Danelljan, M., Häger, G., Khan, F.S., Felsberg, M.: Accurate scale estimation for robust visual tracking. In: Proceedings of the British Machine Vision Conference (2014)
30. González, A., Martín-Nieto, R., Bescós, J., Martínez, J.M.: Single object long-term tracker for smart control of a PTZ camera. In: International Conference on Distributed Smart Cameras, pp. 121–126 (2014)
31. Shi, J., Tomasi, C.: Good features to track. In: Computer Vision and Pattern Recognition, pp. 593–600, June 1994
32. Comaniciu, D., Ramesh, V., Meer, P.: Real-time tracking of non-rigid objects using mean shift. Comp. Vis. Patt. Recogn. **2**, 142–149 (2000)
33. Du, D., Qi, H., Wen, L., Tian, Q., Huang, Q., Lyu, S.: Geometric hypergraph learning for visual tracking. In: CoRR (2016)
34. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. In: Agapito, L., et al. (eds.) ECCV 2014 Workshop. LNCS, vol. 8926, pp. 254–265. Springer, Heidelberg (2014)
35. Possegger, H., Mauthner, T., Bischof, H.: In defense of color-based model-free tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2015)
36. Tang, M., Feng, J.: Multi-kernel correlation filter for visual tracking. In: ICCV (2015)
37. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
38. Bertinetto, L., Valmadre, J., Golodetz, S., Miksik, O., Torr, P.: Staple: Complementary learners for real-time tracking arXiv:1512.01355 [cs.CV] (2015)
39. Du, D., Qi, H., Li, W., Wen, L., Huang, Q., Lyu, S.: Online deformable object tracking based on structure-aware hyper-graph. IEEE Trans. Image Process. **25**(8), 3572–3584 (2016)
40. Bertinetto, L., Valmadre, J., Golodetz, S., Miksik, O., Torr, P.H.S.: Staple: complementary learners for real-time tracking. In: CVPR (2016)
41. Vojíř, T., Matas, J.: The enhanced flock of trackers. In: Cipolla, R., Battiato, S., Farinella, G.M. (eds.) Registration and Recognition in Images and Video. SCI, vol. 532, pp. 111–138. Springer, Heidelberg (2014)
42. Rosten, E., Drummond, T.W.: Machine learning for high-speed corner detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006)
43. Kalal, Z., Mikolajczyk, K., Matas, J.: Forward-backward error: Automatic detection of tracking failures. In: Computer Vision and Pattern Recognition (2010)

44. Pelapur, R., Candemir, S., Bunyak, F., Poostchi, M., Seetharaman, G., Palaniappan, K.: Persistent target tracking using likelihood fusion in wide-area and full motion video sequences. In: IEEE Conference on Information Fusion (FUSION), pp. 2420–2427 (2012)
45. Poostchi, M., Aliakbarpour, H., Viguier, R., Bunyak, F., Palaniappan, K., Seetharaman, G.: Semantic depth map fusion for moving vehicle detection in aerial video. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp. 32–40 (2016)
46. Poostchi, M., Palaniappan, K., Bunyak, F., Becchi, M., Seetharaman, G.: Efficient GPU implementation of the integral histogram. In: Park, J.I., Kim, J. (eds.) ACCV 2012 Workshops. LNCS, vol. 7728, pp. 266–278. Springer, Heidelberg (2012)
47. Palaniappan, K., Bunyak, F., Kumar, P., Ersoy, I., Jaeger, S., Ganguli, K., Haridas, A., Fraser, J., Rao, R., Seetharaman, G.: Efficient feature extraction and likelihood fusion for vehicle tracking in low frame rate airborne video. In: IEEE Conference on Information Fusion (FUSION), pp. 1–8 (2010)
48. Pelapur, R., Palaniappan, K., Seetharaman, G.: Robust orientation and appearance adaptation for wide-area large format video object tracking. In: Proceedings of the IEEE Conference on Advanced Video and Signal based Surveillance, pp. 337–342 (2012)
49. Montero, A.S., Lang, J., Laganiere, R.: Scalable kernel correlation filter with sparse feature integration. In: The IEEE International Conference on Computer Vision (ICCV) Workshops, pp. 24–31, December 2015
50. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: CoRR (2015)
51. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database a large-scale hierarchical image database. In: CVPR (2009)
52. Lebeda, K., Matas, J., Bowden, R.: Tracking the untrackable: how to track when your object is featureless. In: Proceedings of ACCV DTCE (2012)
53. Lebeda, K., Hadfield, S., Matas, J., Bowden, R.: Texture-independent long-term tracking using virtual corners. IEEE Trans. Image Process. **25**(1), 359–371 (2016)
54. Lukezic, A., Cehovin, L., Kristan, M.: Deformable parts correlation filters for robust visual tracking. CoRR abs/1605.03720 (2016)
55. Nam, H., Baek, M., Han, B.: Modeling and propagating cnns in a tree structure for visual tracking. CoRR abs/1608.07242 (2016)
56. Danelljan, M., Häger, G., Khan, F.S., Felsberg, M.: Learning spatially regularized correlation filters for visual tracking. In: International Conference on Computer Vision (2015)
57. Danelljan, M., Khan, F.S., Felsberg, M., Van de Weijer, J.: Adaptive color attributes for real-time visual tracking. In: Computer Vision Pattern Recognition (2014)