FuCoLoT – A Fully-Correlational Long-Term Tracker

Alan Lukežič[1], Luka Čehovin Zajc[1], Tomáš Vojíř[2], Jiří Matas[2], and Matej Kristan[1]

[1] Faculty of Computer and Information Science, University of Ljubljana, Slovenia
[2] Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic
alan.lukezic@fri.uni-lj.si

Abstract. We propose FuCoLoT – a <u>Fu</u>lly <u>Co</u>rrelational <u>Lo</u>ng-term <u>T</u>racker. It exploits the novel DCF constrained filter learning method to design a detector that is able to re-detect the target in the whole image efficiently. FuCoLoT maintains several correlation filters trained on different time scales that act as the detector components. A novel mechanism based on the correlation response is used for tracking failure estimation. FuCoLoT achieves state-of-the-art results on standard short-term benchmarks and it outperforms the current best-performing tracker on the long-term UAV20L benchmark by over 19%. It has an order of magnitude smaller memory footprint than its best-performing competitors and runs at 15fps in a single CPU thread.

1 Introduction

The computer vision community has recently witnessed significant activity and advances of model-free short-term trackers [40, 22] which localize a target in a video sequence given a single training example in the first frame. Current short-term trackers [12, 28, 1, 37, 14] localize the target moderately well even in the presence of significant appearance and motion changes and they are robust to short-term occlusions. Nevertheless, any adaptation at an inaccurate target position leads to gradual corruption of the visual model, drift and irreversible failure. Another major source of failures of short-term trackers are significant occlusion and target disappearance from the field of view. These problems are addressed by long-term trackers which combine a short-term tracker with a detector that is capable of reinitializing the tracker.

A long-term tracker development is complex as it entails: (i) the design of the two core components - the short term tracker and the detector, (ii) an algorithm for their interaction including the estimation of tracking and detection uncertainty, and (iii) the model adaptation strategy. Initially, memoryless displacement estimators like the flock-of-trackers [20] and the flow calculated at keypoints [34] were considered. Later, methods applied keypoint detectors [34, 19, 35, 31], but these require large and sufficiently well textured targets. Cascades of classifiers [20, 30] and more recently deep feature object detection systems [15]
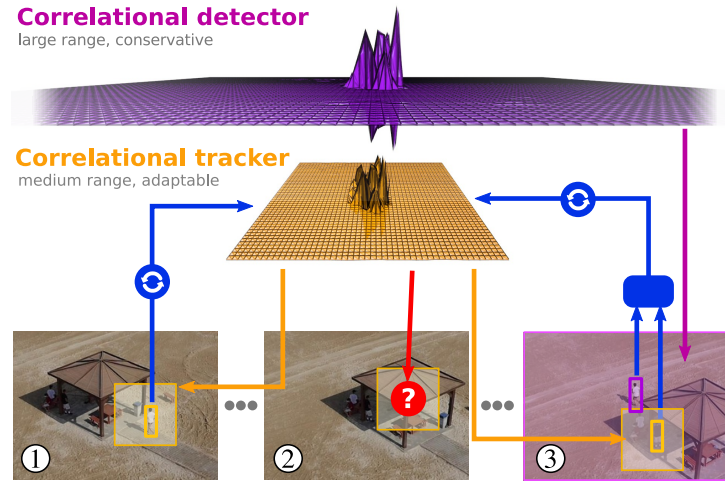
Fig. 1. The FuCoLoT tracker framework: A short-term component of FuCoLoT tracks a visible target (1). At occlusion onset (2), the localization uncertainty is detected and the detection correlation filter is activated in parallel to short-term component to account for the two possible hypotheses of uncertain localization. Once the target becomes visible (3), the detector and short-term tracker interact to recover its position. Detector is deactivated once the localization uncertainty drops.

have been proposed to deal with diverse targets. The drawback is in the significant increase of computational complexity and the subsequent reduction in the range of possible applications. Recent long-term trackers either train the detector on the first frame only [34, 15], thus losing the opportunity to learn target appearance variability or adapt the detector [30, 19] and becoming prone to failure due to learning from incorrect training examples.

The paper introduces FuCoLoT - a novel <u>Fu</u>lly <u>C</u>orrelational <u>Lo</u>ng-term <u>T</u>racker. FuCoLoT is the first long-term tracker that exploits the novel discriminative correlation filter (DCF) learning method based on the ADMM optimization that allows to control the support of the discriminative filter. The method was first used in CSRDCF [28] to limit the DCF filter support to the object segmentation and to avoid problems with shapes not well approximated by a rectangle.

The first contribution of the paper is the observation, and its use, that the ADMM optimization allows DCF to search in an area with size unrelated to the object, e.g. in the whole image. The decoupling of the target and the search region sizes allows implementing the detector as a DCF. Both the short-term tracker and the detector of FuCoLoT are DCFs operating efficiently on the same representation, making FuCoLoT "fully correlational". For some time, DCFs have been the state-of-the-art in short-term tracking, topping a num-

ber of recent benchmarks [40, 23, 24, 23, 22]. However, with the standard learning algorithm [18], a correlation filter cannot be used for detection because of two reasons: (i) the dominance of the background in the search regions which necessarily has the same size as the target model and (ii) the effects of the periodic extension on the borders. Only recently theoretical breakthroughs [10, 28, 21] allowed constraining the non-zero filter response to the area covered by the target.

As a second contribution, FuCoLoT uses correlation filters trained on *different time scales* as a detector to achieve resistance to occlusions, disappearance, or short-term tracking problems of different duration. Both the detectors and its short-term tracker is implemented by a CSRDCF core [28], see Figure 1.

The estimation of the relative confidence of the detectors and the short-term tracker, as well as of the localization uncertainty, is facilitated by the fact that both the detector and the short-term tracker output the same representation - the correlation response. We show that this leads to a simple and effective method that controls their interaction. As another contribution, a stabilizing mechanism is introduced that enables the detector to recover from model contamination.

Extensive experiments show that the proposed FuCoLoT tracker by far outperforms all trackers on a long-term benchmark and achieves excellent performance even on short-term benchmarks. FuCoLoT has a small memory footprint, does not require GPUs and runs at 15 fps on a CPU since both the detectors and the short-term tracker enjoy efficient implementation through FFT.

2   Related work

We briefly overview the most related short-term DCFs and long-term trackers.

Short-term DCFs. Since their inception as the MOSSE tracker [3], several advances have made discriminative correlation filters the most widely used methodology in short-term tracking [22]. Major boosts in performance followed introduction of kernels [18], multi-channel formulations [11, 16] and scale estimation [8, 27]. Hand-crafted features have been recently replaced with deep features trained for classification [12, 9] as well as features trained for localization [37]. Another line of research lead to constrained filter learning approaches [10, 28] that allow learning a filter with the effective size smaller than the training patch.

Long-term trackers. The long-term trackers combine a short-term tracker with a detector – an architecture first proposed by Kalal et al. [20] and now commonly used in long-term trackers. The seminal work of Kalal et al. [20] proposes a memory-less flock of flows as a short-term tracker and a template-based detector run in parallel. They propose a P-N learning approach in which the short-term tracker provides training examples for the detector and pruning events are used to reduce contamination of the detector model. The detector is implemented as a cascade to reduce the computational complexity.

Another paradigm was pioneered by Pernici et al. [35]. Their approach casts localization as local keypoint descriptors matching with a weak geometrical model. They propose an approach to reduce contamination of the keypoints

model that occurs at adaptation during occlusion. Nebehay et al. [34] have shown that a keypoint tracker can be utilized even without updating and using pairs of correspondences in a GHT framework to track deformable models. Maresca and Petrosino [31] have extend the GHT approach by integrating various descriptors and introducing a conservative updating mechanism. The keypoint methods require a large and well textured target, which limits their application scenarios.

Several methods achieve long-term capabilities by careful model updates and detection of detrimental events like occlusion. Grabner et al. [17] proposed an online semi-supervised boosting method that combines a prior and online-adapted classifiers to reduce drifting. Chang et al. [5] apply log-polar transform for tracking failure detection. Kwak et al. [26] proposed occlusion detection by decomposing the target model into a grid of cells and learning an occlusion classifier for each cell. Beyer et al. [2] proposed a Bayes filter for target loss detection and re-detection for multi-target tracking.

Recent long-term trackers have shifted back to the tracker-detector paradigm of Kalal et al. [20], mainly due to availability of DCF trackers [18] which provide a robust and fast short-term tracking component. Ma et al. [30, 29] proposed a combination of KCF tracker [18] and a random ferns classifier as a detector. Similarly, Hong et al. [19] combine a KCF tracker with a SIFT-based detector which is also used to detect occlusions.

The most extreme example of using a fast tracker and a slow detector is the recent work of Fan and Ling [15]. They combine a DSST [8] tracker with a CNN detector [36] which verifies and potentially corrects proposals of the short-term tracker. The tracker achieved excellent results on the challenging long-term benchmark [32], but requires a GPU, has a huge memory footprint and requires parallel implementation with backtracking to achieve a reasonable runtime.

3   Fully correlational long-term tracker

In the following we describe the proposed long-term tracking approach based on constrained discriminative correlation filters. The constrained DCF is overviewed in Section 3.1, Section 3.2 overviews the short-term component, Section 3.3 describes detection of tracking uncertainty, Section 3.4 describes the detector and the long-term tracker is described in Section 3.5.

3.1   Constrained discriminative filter formulation

FuCoLoT is based on discriminative correlation filters. Given a search region of size $W \times H$ a set of N_d feature channels $\mathbf{f} = \{\mathbf{f}_d\}_{d=1}^{N_d}$, where $\mathbf{f}_d \in \mathcal{R}^{W \times H}$, are extracted. A set of N_d correlation filters $\mathbf{h} = \{\mathbf{h}_d\}_{d=1}^{N_d}$, where $\mathbf{h}_d \in \mathcal{R}^{W \times H}$, are correlated with the extracted features and the object position is estimated as the location of the maximum of the weighted correlation responses

$$\mathbf{r} = \sum\nolimits_{d=1}^{N_d} w_d(\mathbf{f}_d \star \mathbf{h}_d), \tag{1}$$

where $\star$ represents circular correlation, which is efficiently implemented by a Fast Fourier Transform and $\{w_d\}_{d=1}^{N_d}$ are channel weights. The target scale can be efficiently estimated by another correlation filter trained over the scale-space [8].

We apply the recently proposed filter learning technique (CSRDCF [28]), which uses the alternating direction method of multipliers (ADMM [4]) to constrain the learned filter support by a binary segmentation mask. In the following we provide a brief overview of the learning framework and refer the reader to the original paper [28] for details.

Constrained learning. Since feature channels are treated independently, we will assume a single feature channel (i.e., $N_d = 1$) in the following. A channel feature $\mathbf{f}$ is extracted from a learning region and a fast segmentation method [25] is applied to produce a binary mask $\mathbf{m} \in \{0, 1\}^{W \times H}$ that approximately separates the target from the background. Next a filter of the same size as the training region is learned, with support constrained by the mask $\mathbf{m}$. The discriminative filter $\mathbf{h}$ is learned by introducing a dual variable $\mathbf{h}_c$ and minimizing the following augmented Lagrangian

$$\mathcal{L}(\hat{\mathbf{h}}_c, \mathbf{h}, \hat{\mathbf{l}}|\mathbf{m}) = \|\mathrm{diag}(\hat{\mathbf{f}})\overline{\hat{\mathbf{h}}_c} - \hat{\mathbf{g}}\|^2 + \frac{\lambda}{2}\|\mathbf{m} \odot \mathbf{h}\|^2 + \tag{2}$$
$$2[\hat{\mathbf{l}}^H \mathcal{F}(\mathbf{h}_c - \mathbf{m} \odot \mathbf{h})]_{\mathrm{Re}} + \mu\|\mathcal{F}(\mathbf{h}_c - \mathbf{m} \odot \mathbf{h})\|^2,$$

where $\mathbf{g}$ is a desired output, $\hat{\mathbf{l}}$ is a complex Lagrange multiplier, $\hat{(\cdot)} = \mathcal{F}(\cdot)$ denotes Fourier transformed variable, $[\cdot]_{\mathrm{Re}}$ is an operator that removes the imaginary part and μ is a non-negative real number. The solution is obtained via ADMM [4] iterations of two closed-form solutions:

$$\hat{\mathbf{h}}_c^{i+1} = \left(\hat{\mathbf{f}} \odot \overline{\hat{\mathbf{g}}} + (\mu\mathcal{F}(\mathbf{m} \odot \mathbf{h}^i) - \hat{\mathbf{l}}^i)\right) \odot^{-1} \left(\overline{\hat{\mathbf{f}}} \odot \hat{\mathbf{f}} + \mu^i\right), \tag{3}$$

$$\mathbf{h}^{i+1} = \mathbf{m} \odot \mathcal{F}^{-1}\left[\hat{\mathbf{l}}^i + \mu^i\hat{\mathbf{h}}_c^{i+1}\right]/\left(\frac{\lambda}{2D} + \mu^i\right), \tag{4}$$

where $\mathcal{F}^{-1}(\cdot)$ denotes the inverse Fourier transform. In the case of multiple channels, the approach independently learns a single filter per channel. Since the support of the learned filter is constrained to be smaller than the learning region, the maximum response on the training region reflects the reliability of the learned filter [28]. These values are used as per-channel weights w_d in (1) for improved target localization.

Note that the constrained learning [28] estimates a filter implicitly padded with zeros to match the learning region size. In contrast to the standard approach to filter learning like e.g., [18] and multiplying with a mask post-hoc, the padding is explicitly enforced during learning, resulting in an increased filter robustness. We make an observation that adding or removing the zeros at filter borders keeps the filter unchanged, thus correlation on an arbitrary large region via FFT is possible by zero padding the filter to match the size. These properties make the constrained learning an excellent candidate to train the short-term component (Section 3.4) as well as the detector (Section 3.4) in a long-term tracker.
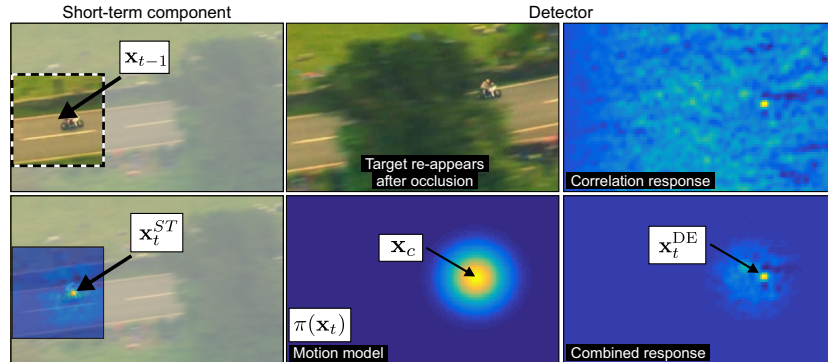
Fig. 2. The short-term component (left) estimates the target location at the maximum response of its DCF within a search region centered at the estimate in the previous frame. The detector (right) estimates the target location as the maximum in the whole image of the response of its DCF multiplied by the motion model $\pi(\mathbf{x}_t)$. If tracking fails, the prior $\pi(\mathbf{x}_t)$ spreads.

3.2   The short-term component

The CSRDCF [28] tracker is used as the short-term component in FuCoLoT. The short-term component is run within a search region centered on the target position predicted from the previous frame. The new target position hypothesis $\mathbf{x}_t^{\mathrm{ST}}$ is estimated as the location of the maximum of the correlation response between the short-term filter $\mathbf{h}_t^{\mathrm{ST}}$ and the features extracted from the search region (see Figure 2, left).

The visual model of the short-term component $\mathbf{h}^{\mathrm{ST}}$ is updated by a weighted running average

$$\mathbf{h}_{t+1}^{\mathrm{ST}} = (1-\eta)\mathbf{h}_t^{\mathrm{ST}} + \eta\tilde{\mathbf{h}}_t^{\mathrm{ST}}, \tag{5}$$

where $\mathbf{h}_t^{\mathrm{ST}}$ is the correlation filter used to localize the target, $\tilde{\mathbf{h}}_t^{\mathrm{ST}}$ the filter estimated by constrained filter learning (Section 3.1) in the current frame, and η is the update factor.

3.3   Tracking uncertainty estimation

Tracking uncertainty estimation is crucial for minimizing short-term visual model contamination as well as for activating target re-detection after events like occlusion. We propose a self-adaptive approach for tracking uncertainty based on the maximum correlation response.

Confident localization produces a well expressed local maximum in the correlation response $\mathbf{r}_t$, which can be measured by the peak-to-sidelobe ratio $\mathrm{PSR}(\mathbf{r}_t)$ [3] and by the peak absolute value $\max(\mathbf{r}_t)$. Empirically, we observed that multiplying the two measures leads to improved performance, therefore the localiza-
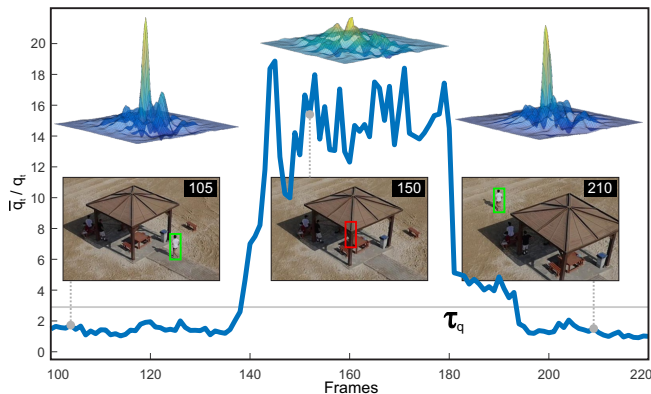
Fig. 3. Localization uncertainty measure (7), reflects the correlation response peak strength relative to its past values. The measure rises fast during occlusion and drops immediately afterwards.

tion quality is defined as the product

$$q_t = \mathrm{PSR}(\mathbf{r}_t) \cdot \max(\mathbf{r}_t). \qquad (6)$$

The following observations were used in design of tracking uncertainty (or failure) detection: (i) relative value of the localization quality q_t depends on target appearance changes and is only a weak indicator of tracking uncertainty, and (ii) events like occlusion occur on a relatively short time-scale and are reflected in a significant reduction of the localization quality. Let $\bar{q}_t$ be the average localization quality computed over the recent N_q confidently tracked frames. Tracking is considered uncertain if the ratio between $\bar{q}_t$ and q_t exceeds a predefined threshold τ_q, i.e.,

$$\bar{q}_t / q_t > \tau_q. \qquad (7)$$

In practice, the ratio between the average and current localization quality significantly increases during occlusion, indicating a highly uncertain tracking, and does not require threshold fine-tuning (an example is shown in Figure 3).

3.4   Target loss recovery

A visual model not contaminated by false training examples is desirable for reliable re-detection after a long period of target loss. The only known certainly uncontaminated filter is the one learned at initialization. However, for a short-term occlusions, the most recent uncontaminated model would likely yield a better detection. While contamination of the short-term visual model (Section 3.2) is reduced by the long-term system (Section 3.5), it cannot be prevented. We thus maintain as set of several filters $\mathcal{H}^{\mathrm{DE}} = \{\mathbf{h}_i^{\mathrm{DE}}\}_{i \in 1,\dots N_{\mathrm{DE}}}$ updated at different *temporal scales* to deal with potential model contamination.

The filters updated frequently learn recent appearance changes, while the less frequently updated filters increase robustness to learning during potentially undetected tracking failure. In our approach, one of the filters is never updated (the initial filter), which guarantees full recovery from potential contamination of the updated filters if a view similar to the initial training image appears. The i-th filter is updated every n_i^{DE} frames similarly as the short-term filter:

$$\mathbf{h}_i^{\mathrm{DE}} = (1 - \eta)\mathbf{h}_i^{\mathrm{DE}} + \eta\tilde{\mathbf{h}}_t^{\mathrm{ST}}. \tag{8}$$

A random-walk motion model is added as a principled approach to modeling the growth of the target search region size. The target position prior $\pi(\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_t; \mathbf{x}_c, \Sigma_t)$ at time-step t is a Gaussian with a diagonal covariance $\Sigma_t = \mathrm{diag}(\sigma_{xt}^2, \sigma_{yt}^2)$ centered at the last confidently estimated position $\mathbf{x}_c$. The variances in the motion model gradually increase with the number of frames Δ_t since the last confident estimation, i.e., $[\sigma_{xt}, \sigma_{yt}] = [x_w, x_h]\alpha_s^{\Delta_t}$, where α_s is scale increase parameter, x_w and x_h are the target width and height, respectively.

During target re-detection, a filter is selected from $\mathcal{H}^{\mathrm{DE}}$ and correlated with features extracted from the entire image. The detected position $\mathbf{x}_t^{\mathrm{DE}}$ is estimated as the location maximum of the correlation response multiplied with the motion prior $\pi(\mathbf{x}_t)$ as shown in Figure 2 (right). For implementation efficiency only a single filter is evaluated on each image. The algorithm cycles through all filters in $\mathcal{H}^{\mathrm{DE}}$ and all target size scales $\mathcal{S}^{\mathrm{DE}}$ in subsequent images until the target is detected. In practice this means that all filters are evaluated approximately within a second of the sequence (Section 4.1).

3.5   Tracking with FuCoLoT

The FuCoLoT integrates the short-term component (Section 3.2), the uncertainty estimator (Section 3.3) and target recovery (Section 3.4) as follows.

Initialization. The long-term tracker is initialized in the first frame and the learned initialization model $\mathbf{h}_1^{\mathrm{ST}}$ is stored. In the remaining frames, N_{DE} visual models are maintained at different time-scales for target localization and detection $\{\mathbf{h}_i^{\mathrm{DE}}\}_{i \in 1, \dots N_{\mathrm{DE}}}$, where the model updated at every frame is the short-term visual model, i.e., $\mathbf{h}_t^{\mathrm{ST}} = \mathbf{h}_{N_{\mathrm{DE}}}^{\mathrm{DE}}$, and the model that is never updated is equal to the initialization model, i.e., $\mathbf{h}_1^{\mathrm{DE}} = \mathbf{h}_1^{\mathrm{ST}}$.

Localization. A tracking iteration at frame t starts with the target position $\mathbf{x}_{t-1}$ from the previous frame, a tracking quality score q_{t-1} and the mean $\bar{q}_{t-1}$ over the recent N_q confidently tracked frames. A region is extracted around $\mathbf{x}_{t-1}$ in the current image and the correlation response is computed using the short-term component model $\mathbf{h}_{t-1}^{\mathrm{ST}}$ (Section 3.2). Position $\mathbf{x}_t^{\mathrm{ST}}$ and localization quality q_t^{ST} (6) are estimated from the correlation response $\mathbf{r}_t^{\mathrm{ST}}$. When tracking is confident at $t-1$, i.e., the uncertainty (7) $\bar{q}_t/q_t$ was smaller than τ_q, only the short-term component is run. Otherwise the detector (Section 3.4) is activated as well to address potential target loss. The detector filter $\mathbf{h}_i^{\mathrm{DE}}$ is chosen from the sequence of stored detectors $\mathcal{H}^{\mathrm{DE}}$ and correlated with the features extracted from the entire image. The detection hypothesis $\mathbf{x}_t^{\mathrm{DE}}$ is obtained as the location

of the maximum of the correlation multiplied by the motion model $\pi(\mathbf{x}_t)$, while the localization quality q_t^{DE} (6) is computed only on the correlation response.

Update. In case the detector has not been activated, the short-term position is taken as the final target position estimate. Alternatively, both position hypotheses, i.e., the position estimated by the short-term component as well as the position estimated by the detector, are considered. The final target position is estimated as the one with higher quality score, i.e.,

$$(\mathbf{x}_t, q_t) = \begin{cases} (\mathbf{x}_t^{\mathrm{ST}}, q_t^{\mathrm{ST}}) \; ; \; q_t^{\mathrm{ST}} \geq q_t^{\mathrm{DE}} \\ (\mathbf{x}_t^{\mathrm{DE}}, q_t^{\mathrm{DE}}) \; ; \; \text{otherwise} \end{cases}. \tag{9}$$

If the estimated position is reliable (7), a constrained filter $\tilde{\mathbf{h}}_t^{\mathrm{ST}}$ is estimated according to [28] and the short-term component (5) and detector (8) are updated. Otherwise the models are not updated, i.e., $\eta = 0$ in (5) and (8).

4   Experiments

4.1   Implementation details

We use the same standard HOG [7] and colornames [39, 11] features in the short-term component and in the detector. All the parameters of the CSRDCF filter learning are the same as in [28], including filter learning rate $\eta = 0.02$ and regularization $\lambda = 0.01$. We use 5 filters in the detector, updated with the following frequencies $\{0, 1/250, 1/50, 1/10, 1\}$ and size scale factors $\{0.5, 0.7, 1, 1.2, 1.5, 2\}$.

The random-walk motion model region growth parameter was set to $\alpha_s = 1.05$. The uncertainty threshold was set to $\tau_q = 2.7$ and the parameter "recent frames" was $N_q = 100$. The parameters did not require fine tuning and were kept constant throughout all experiments. Our Matlab implementation runs at 15 fps on OTB100 [40], 8 fps on VOT16 [23] and 6 fps on UAV20L [32] dataset. The experiments were conducted on an Intel Core i7 3.4GHz standard desktop.

4.2   Evaluation on a long-term benchmark

The long-term performance of the FuCoLoT is analyzed on the recent long-term benchmark UAV20L [32] that contains results of 11 trackers on 20 long term sequences with average sequence length 2934 frames. To reduce clutter in the plots we include top-performing tracker SRDCF [10] and all long-term trackers in the benchmark, i.e., MUSTER [19] and TLD [20]. We add the most recent state-of-the-art long-term trackers CMT [34], LCT [30], and PTAV [15] in the analysis, as well as the recent state-of-the-art short-term DCF trackers CSRDCF [28], CCOT [12] and CNN-based MDNet [33] and SiamFC [1].

Results in Figure 4 show that on benchmark, FuCoLoT by far outperforms all top baseline trackers as well as all the recent long-term state-of-the-art. In particular FuCoLoT outperforms the recent long-term correlation filter LCT [30] by 102% in precision and 106% in success measures. The FuCoLoT also outperforms the currently best-performing published long-term tracker PTAV [15] by
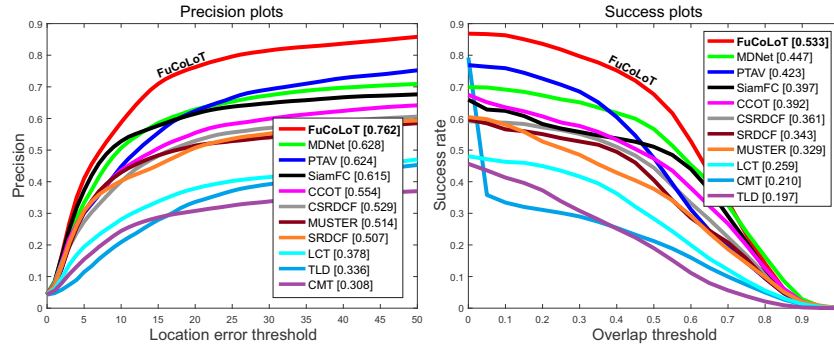
Fig. 4. UAV20L [32] benchmark results. The precision plot (left) and the success plot (right).

Tracker	Scale Var.	Aspect Change	Low Res.	Fast Motion	Full Occ.	Partial Occ.	Out-of-View	Back. Clutter	Illum. Var.	Viewp. Change	Camera Motion	Similar Object
FuCoLoT	① **0.526**	① **0.500**	① **0.472**	① **0.400**	① **0.460**	① **0.526**	① **0.488**	① **0.538**	① **0.513**	① **0.499**	① **0.528**	① **0.573**
MDNet	② 0.438	③ 0.385	③ 0.357	② 0.383	0.188	② 0.419	② 0.439	0.149	③ 0.403	② 0.444	② 0.432	② 0.525
PTAV	③ 0.416	② 0.410	② 0.390	③ 0.349	② 0.357	③ 0.415	③ 0.389	② 0.435	② 0.430	③ 0.418	③ 0.420	0.426
SiamFC	0.383	0.328	0.242	0.264	③ 0.237	0.364	0.386	③ 0.239	0.371	0.356	0.383	0.436
CCOT	0.378	0.322	0.277	0.275	0.183	0.368	0.352	0.188	0.382	0.330	0.380	③ 0.463
CSRDCF	0.346	0.293	0.232	0.194	0.210	0.339	0.326	0.227	0.359	0.308	0.348	0.403
SRDCF	0.332	0.270	0.228	0.197	0.170	0.320	0.329	0.156	0.295	0.303	0.327	0.397
MUSTER	0.314	0.275	0.278	0.206	0.200	0.305	0.309	0.230	0.242	0.318	0.307	0.342
LCT	0.244	0.201	0.183	0.112	0.151	0.244	0.249	0.156	0.232	0.225	0.245	0.283
CMT	0.208	0.169	0.139	0.199	0.134	0.173	0.184	0.104	0.146	0.212	0.187	0.203
TLD	0.193	0.196	0.159	0.235	0.154	0.201	0.212	0.111	0.167	0.188	0.202	0.225

Table 1. Tracking performance (AUC measure) for fourteen tracking attributes and eleven trackers on the UAV20L [32].

over 22% and 26% in precision and success measures, respectively. This is an excellent result especially considering that FuCoLoT does not apply deep features and backtracking like PTAV [15] and that it runs in near-realtime on a single thread CPU. The FuCoLoT outperforms the second-best tracker on UAV20L, MDNet [33] which uses pre-trained network and runs at cca. 1fps, by 21% in precision and 19% in success measure.

Table 1 shows tracking performance in terms of the AUC measure for the twelve attributes annotated in the UAV20L benchmark. The FuCoLoT is the top performing tracker across all attributes, including full occlusion and out-of-view, where it outperforms the second-best PTAV and MDNet by 29% and 11%, respectively. These attributes focus on the long-term tracker capabilities since they require target re-detection.

Figure 5 shows qualitative tracking examples for the FuCoLoT and four state-of-the-art trackers: PTAV [15], CSRDCF [28], MUSTER [19] and TLD [20]. In all these sequences the target becomes fully occluded at least once. FuCoLot is the only tracker that is able to successfully track the target throughout Group2,
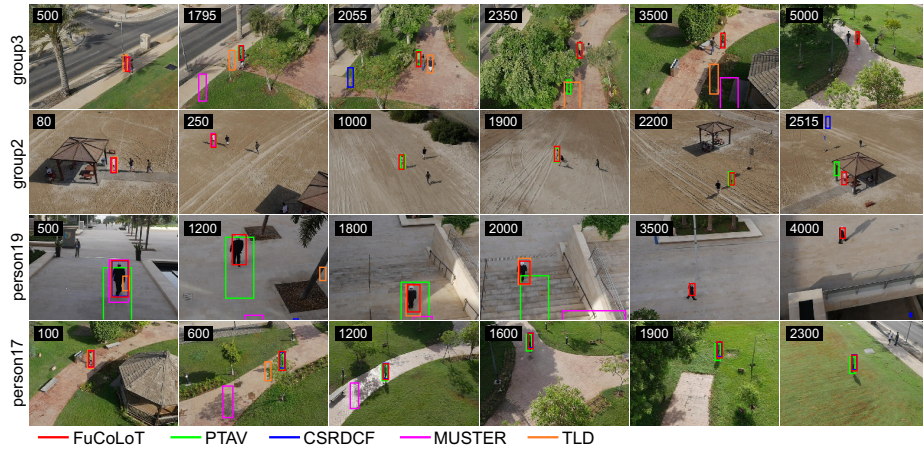
Fig. 5. Qualitative results of the FuCoLoT and four state-of-the-art trackers on four sequences from [32]. The selected sequences contain full occlusions which highlight the re-detection ability of a tracker.

Group3 and Person19 sequences, which shows the strength of the proposed correlation filter based detector. In Person 17 sequence, the occlusion is shorter, thus the short-term CSRDCF [28] and long-term PTAV [15] are able to track as well.

4.3   Re-detection capability experiment

In the original UAV20L [32] dataset, the target disappears and reappears only 39 times, resulting in only 4% of frames with the target absent. This setup does not significantly expose the target re-detection capability of the tested trackers. To address this, we have cropped the images in all sequences to 40% of their size around the center at target initial position. An example of the dataset modification is shown in Figure 6. This modification increased the target disappearance/reappearance to 114 cases, and the target is absent in 34% of the frames.

The top six trackers from Section 4.2 and a long-term baseline TLD [20] were re-evaluated on the modified dataset (results in Table 2). The gap between the FuCoLoT and the other trackers is further increased. FuCoLoT outperforms the second-best MDNet [33] by 30% and the recent long-term state-of-the-art tracker PTAV [15] by 47%. Note that FuCoLoT outperforms all CNN-based trackers using only hand-crafted features, which speaks in favor of the highly efficient architecture.
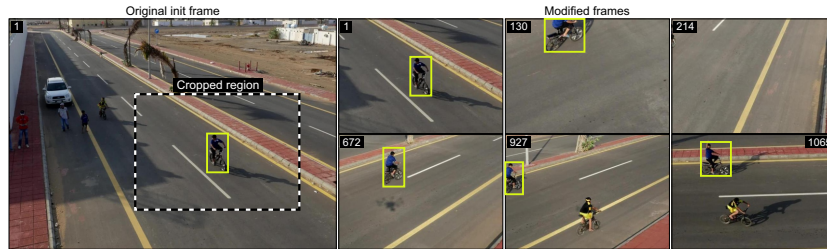
Fig. 6. Re-detection experiment - an example of the modification of a sequence. Yellow bounding-boxes denote the ground-truth position of the target. The target leaves the field-of-view more frequently in the dataset with cropped images.

Tracker	FuCoLoT	MDNet	CCOT	PTAV	SiamFC	TLD	CSRDCF
AUC	**0.314**	0.240	0.220	0.213	0.205	0.160	0.158

Table 2. Re-detection experiment on the UAV20L [32] dataset with images cropped to increase the number of times the target leaves and re-enters the field of view.

4.4   Ablation study

Several modifications of the FuCoLoT detector were tested to expose the contributions of different parts in our architecture. Two variants used the filter extracted at initialization in the detector with a single scale detection ($FuCoLoT_{D^0S^1}$) and multiple scale detection ($FuCoLoT_{D^0S^M}$) and one variant used the most recent filter from the short-term tracker in the detector with multiple scale detection ($FuCoLoT_{D^{ST}S^M}$). The results are summarized in Table 3.

In single-scale detection, the most recent short-term filter ($FuCoLoT_{D^{ST}S^1}$) marginally improves the performance of the ($FuCoLoT_{D^0S^1}$) and achieves 0.499 AUC. The performance improves to 0.505 AUC by adding multiple scales search in the detector ($FuCoLoT_{D^{ST}S^M}$) and further improves to 0.533 AUC when considering filters with variable temporal updating in the detector (FuCoLoT). For reference, all FuCoLoT variants significantly outperform the FuCoLOT short-term tracker without our detector, i.e. the CSRDCF [28] tracker.

4.5   Performance on short-term benchmarks

For completeness, we first evaluate the performance of FuCoLoT on the popular short-term benchmarks: OTB100 [40], and VOT2016 [23]. A standard no-reset evaluation (OPE [40]) is applied to focus on long-term behavior: a tracker is initialized in the first frame and left to track until the end of the sequence.

Tracking quality is measured by precision and success plots. The success plot shows all threshold values, the proportion of frames with the overlap between the predicted and ground truth bounding boxes as greater than a threshold.

Tracker	FuCoLoT$_{D^0S^1}$	FuCoLoT$_{D^{ST}S^1}$	FuCoLoT$_{D^{ST}S^M}$	FuCoLoT	CSRDCF [28]
AUC	0.489	0.499	0.505	**0.533**	0.361

Table 3. Ablation study of the FuCoLoT tracker on UAV20L [32].
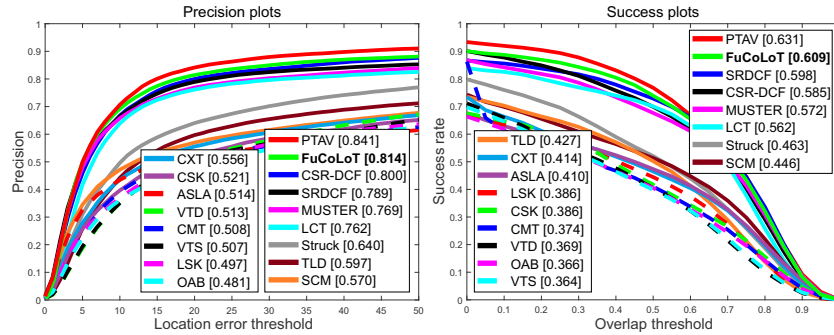

Fig. 7. OTB100 [40] benchmark results. The precision plot (left) and the success plot (right).

The results are summarized by areas under these plots which are shown in the legend. The precision plots in Figures 7 and 8 show a similar statistics computed from the center error. The results in the legends are summarized by percentage of frames tracked with an center error less than 20 pixels.

The benchmarks results have some long-term trackers and the most recent PTAV [15] – the currently best-performing published long-term tracker. Note that PTAV applies preemptive tracking with backtracking and requires future frames to predict position of the tracked object which limits its applicability.

OTB100 [40] contains results of 29 trackers evaluated on 100 sequences with average sequence length of 589 frames. We show only the results for top-performing recent baselines, and recent top-performing state-of-the-art trackers SRDCF [10], MUSTER [19], LCT [30] PTAV [15] and CSRDCF [28].

The FuCoLoT ranks among the top on this benchmark (Figure 7) outperforming all baselines as well as state-of-the-art SRDCF, CSRDCF and MUSTER. Using only handcrafted features, the FuCoLoT achieves comparable performance to the PTAV [15] which uses deep features for re-detection and backtracking.

VOT2016 [23] is a challenging recent short-term tracking benchmark which evaluates 70 trackers on 60 sequences with the average sequence length of 358 frames. The dataset was created using a methodology that selected sequences which are difficult to track, thus the target appearance varies much more than in other benchmarks. In the interest of visibility, we show only top-performing trackers on no-reset evaluation, i.e., SSAT [33, 23], TCNN [33, 23], CCOT [12], MDNetN [33, 23], GGTv2 [13], MLDF [38], DNT [6], DeepSRDCF [10], SiamRN [1]
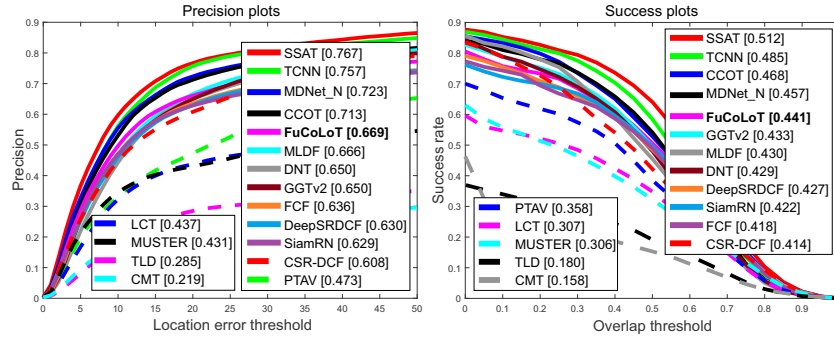
Fig. 8. VOT2016 [23] benchmark results. The precision plot (left) and the success plot (right).

and FCF [23]. We add CSRDCF [28] and the long-term trackers TLD [20], LCT [30], MUSTER [19], CMT [34] and PTAV [15].

The FuCoLoT is ranked fifth (Figure 8) according to the tracking success measure, outperforming 65 trackers, including trackers with deep features, CSR-DCF and PTAV. Note that four trackers with better performance than FuCoLoT (SSAT, TCNN, CCOT and MDNetN) are computationally very expensive CNN-based trackers. They are optimized for accurate tracking on short sequences, without an ability for re-detection. The FuCoLoT outperforms all long-term trackers on this benchmark (TLD, CMT, LCT, MUSTER and PTAV).

5   Conclusion

A fully-correlational long-term tracker – FuCoLot – was proposed. FuCoLoT is the first long-term tracker that exploits the novel DCF constrained filter learning method [28]. The constrained filter learning based detector is able to re-detect the target in the whole image efficiently. FuCoLoT maintains several correlation filters trained on different time scales that act as the detector components. A novel mechanism based on the correlation response quality is used for tracking uncertainty estimation which drives interaction between the short-term component and the detector.

On the UAV20L long-term benchmark [32] FuCoLoT outperforms the best method by over 19%. Experimental evaluation on short-term benchmarks [40, 23] showed state-of-the-art performance. The Matlab implementation running at 15 fps will be made publicly available.

References

1. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.: Fully-convolutional siamese networks for object tracking (2016)
2. Beyer, L., Breuers, S., Kurin, V., Leibe, B.: Towards a principled integration of multi-camera re-identification and tracking through optimal bayes filters. CoRR **abs/1705.04608** (2017)
3. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: Comp. Vis. Patt. Recognition. pp. 2544–2550 (2010)
4. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine Learning **3**(1), 1–122 (2011)
5. Chang, H.J., Park, M.S., Jeong, H., Choi, J.Y.: Tracking failure detection by imitating human visual perception. In: Proc. Int. Conf. Image Processing. pp. 3293–3296 (2011)
6. Chi, Z., Li, H., Lu, H., Yang, M.H.: Dual deep network for visual tracking. IEEE Trans. Image Proc. **26**(4), 2005–2015 (2017)
7. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Comp. Vis. Patt. Recognition. vol. 1, pp. 886–893 (June 2005)
8. Danelljan, M., Häger, G., Khan, F.S., Felsberg, M.: Discriminative scale space tracking. IEEE Trans. Pattern Anal. Mach. Intell. **39**(8), 1561–1575 (2017)
9. Danelljan, M., Bhat, G., Shahbaz Khan, F., Felsberg, M.: Eco: Efficient convolution operators for tracking. In: Comp. Vis. Patt. Recognition. pp. 6638–6646 (2017)
10. Danelljan, M., Hager, G., Shahbaz Khan, F., Felsberg, M.: Learning spatially regularized correlation filters for visual tracking. In: Int. Conf. Computer Vision. pp. 4310–4318 (2015)
11. Danelljan, M., Khan, F.S., Felsberg, M., van de Weijer, J.: Adaptive color attributes for real-time visual tracking. pp. 1090–1097 (2014)
12. Danelljan, M., Robinson, A., Khan, F.S., Felsberg, M.: Beyond correlation filters: learning continuous convolution operators for visual tracking. In: Proc. European Conf. Computer Vision. pp. 472–488 (2016)
13. Du, D., Qi, H., Wen, L., Tian, Q., Huang, Q., Lyu, S.: Geometric hypergraph learning for visual tracking. IEEE Trans. on Cyber. **47**(12), 4182–4195 (2017)
14. Du, D., Wen, L., Qi, H., Huang, Q., Tian, Q., Lyu, S.: Iterative graph seeking for object tracking. IEEE Trans. Image Proc. **27**(4), 1809–1821 (2018)
15. Fan, H., Ling, H.: Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking. In: Int. Conf. Computer Vision. pp. 5486–5494 (2017)
16. Galoogahi, H.K., Sim, T., Lucey, S.: Multi-channel correlation filters. In: Int. Conf. Computer Vision. pp. 3072–3079 (2013)
17. Grabner, H., Leistner, C., Bischof, H.: Semi-supervised on-line boosting for robust tracking. In: Proc. European Conf. Computer Vision. pp. 234–247 (2008)
18. 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 (2015)
19. Hong, Z., Chen, Z., Wang, C., Mei, X., Prokhorov, D., Tao, D.: Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking. In: Comp. Vis. Patt. Recognition. pp. 749–758 (June 2015)
20. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. IEEE Trans. Pattern Anal. Mach. Intell. **34**(7), 1409–1422 (July 2012)

21. Kiani Galoogahi, H., Sim, T., Lucey, S.: Correlation filters with limited boundaries. In: Comp. Vis. Patt. Recognition. pp. 4630–4638 (2015)
22. Kristan, M., Matas, J., Leonardis, A., Vojir, T., Pflugfelder, R., Fernandez, G., Nebehay, G., Porikli, F., Cehovin, L.: A novel performance evaluation methodology for single-target trackers. IEEE Trans. Pattern Anal. Mach. Intell. (2016)
23. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Čehovin, L., Vojir, T., Häger, G., Lukežič, A., et al. Fernandez, G.: The visual object tracking vot2016 challenge results. In: Proc. European Conf. Computer Vision (2016)
24. Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Čehovin, L., Fernandez, G., Vojir, T., Häger, G., Nebehay, G., et al. Pflugfelder, R.: The visual object tracking vot2015 challenge results. In: Int. Conf. Computer Vision (2015)
25. Kristan, M., Perš, J., Sulič, V., Kovačič, S.: A graphical model for rapid obstacle image-map estimation from unmanned surface vehicles. In: Proc. Asian Conf. Computer Vision. pp. 391–406 (2014)
26. Kwak, S., Nam, W., Han, B., Han, J.H.: Learning occlusion with likelihoods for visual tracking. In: Int. Conf. Computer Vision. pp. 1551–1558 (2011)
27. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. In: Proc. European Conf. Computer Vision. pp. 254–265 (2014)
28. Lukežič, A., Vojíř, T., Čehovin Zajc, L., Matas, J., Kristan, M.: Discriminative correlation filter with channel and spatial reliability. In: Comp. Vis. Patt. Recognition. pp. 6309–6318 (2017)
29. Ma, C., Huang, J.B., Yang, X., Yang, M.H.: Adaptive correlation filters with long-term and short-term memory for object tracking. Int. J. Comput. Vision $\mathbf{126}$(8), 771–796 (2018)
30. Ma, C., Yang, X., Zhang, C., Yang, M.H.: Long-term correlation tracking. In: Comp. Vis. Patt. Recognition. pp. 5388–5396 (2015)
31. Maresca, M.E., Petrosino, A.: Matrioska: A multi-level approach to fast tracking by learning. In: Proc. Int. Conf. Image Analysis and Processing. pp. 419–428 (2013)
32. Mueller, M., Smith, N., Ghanem, B.: A benchmark and simulator for uav tracking. In: Proc. European Conf. Computer Vision. pp. 445–461 (2016)
33. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: Comp. Vis. Patt. Recognition. pp. 4293–4302 (June 2016)
34. Nebehay, G., Pflugfelder, R.: Clustering of static-adaptive correspondences for deformable object tracking. In: Comp. Vis. Patt. Recognition. pp. 2784–2791 (2015)
35. Pernici, F., Del Bimbo, A.: Object tracking by oversampling local features. IEEE Trans. Pattern Anal. Mach. Intell. $\mathbf{36}$(12), 2538–2551 (2013)
36. Tao, R., Gavves, E., Smeulders, A.W.M.: Siamese instance search for tracking. In: Comp. Vis. Patt. Recognition. pp. 1420–1429 (2016)
37. Valmadre, J., Bertinetto, L., Henriques, J., Vedaldi, A., Torr, P.H.S.: End-to-end representation learning for correlation filter based tracking. In: Comp. Vis. Patt. Recognition. pp. 2805–2813 (2017)
38. Wang, L., Ouyang, W., Wang, X., Lu, H.: Visual tracking with fully convolutional networks. In: Int. Conf. Computer Vision. pp. 3119–3127 (Dec 2015)
39. van de Weijer, J., Schmid, C., Verbeek, J., Larlus, D.: Learning color names for real-world applications. IEEE Trans. Image Proc. $\mathbf{18}$(7), 1512–1523 (July 2009)
40. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell. $\mathbf{37}$(9), 1834–1848 (Sept 2015)