Discriminative Correlation Filter with Channel and Spatial Reliability

Alan Lukežič[1], Tomáš Vojíř[2], Luka Čehovin[1], Jiří Matas[2], 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, luka.cehovin, matej.kristan}@fri.uni-lj.si
{vojirtom, matas}@cmp.felk.cvut.cz

Abstract

Short-term tracking is an open and challenging problem for which discriminative correlation filters (DCF) have shown excellent performance. We introduce the channel and spatial reliability concepts to DCF tracking and provide a novel learning algorithm for its efficient and seamless integration in the filter update and the tracking process. The spatial reliability map adjusts the filter support to the part of the object suitable for tracking. This both allows to enlarge the search region and improves tracking of non-rectangular objects. Reliability scores reflect channel-wise quality of the learned filters and are used as feature weighting coefficients in localization. Experimentally, with only two simple standard features, HoGs and Colornames, the novel CSR-DCF method – DCF with Channel and Spatial Reliability – achieves state-of-the-art results on VOT 2016, VOT 2015 and OTB100. The CSR-DCF runs in real-time on a CPU.

1. Introduction

Short-term visual object tracking is the problem of continuously localizing a target in a video-sequence given a single example of its appearance. It has received significant attention of the computer vision community which is reflected in the number of papers published on the topic and the existence of multiple performance evaluation benchmarks [43, 29, 30, 26, 27, 33, 38, 36]. Diverse factors – occlusion, illumination change, fast object or camera motion, appearance changes due to rigid or non-rigid deformations and similarity to the background – make short-term tracking challenging.

Recent short-term tracking evaluations [43, 29, 30, 26] consistently confirm the advantages of semi-supervised discriminative tracking approaches [17, 1, 18, 4]. In particular, trackers based on the discriminative correlation filter method (DCF) [4, 8, 20, 31, 10] have shown state-of-the-art performance in all standard benchmarks. Discriminative correlation methods learn a filter with a pre-defined response on the training image.

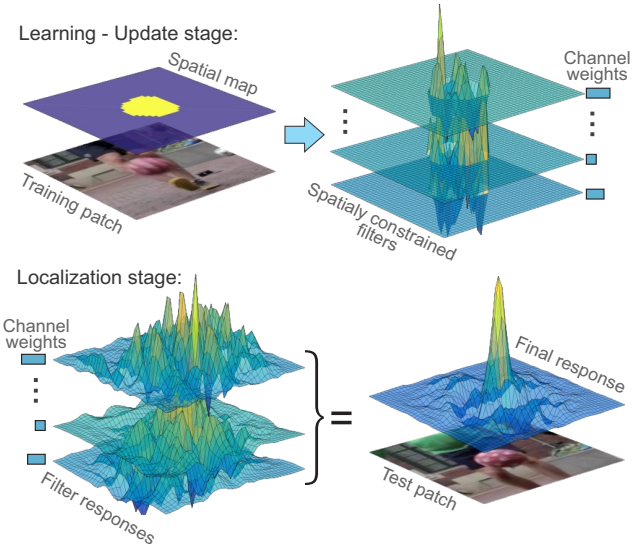

Figure 1: Overview of the proposed CSR-DCF approach. An automatically estimated spatial reliability map restricts the correlation filter to the parts suitable for tracking (top) improving the search range and performance for irregularly shaped objects. Channel reliability weights calculated in the constrained optimization step of the correlation filter learning reduce the noise of the weight-averaged filter response (bottom).

The standard formulation of DCF uses circular correlation which allows to implement learning efficiently by Fast Fourier transform (FFT). However, the FFT requires the filter and the patch size to be equal which limits the detection range. Due to the circularity, the filter is trained on many examples that contain unrealistic, wrapped-around circularly-shifted versions of the target. These windowing problems were recently addressed by state-of-the-art approaches of Galoogahi et al. [24] who propose zero-padding the filter during learning and by Danneljan et al. [10] who introduce

spatial regularization to penalize filter values outside the object boundaries. Both approaches train from image patches larger than the object and thus increase the detection range.

Another limitation of the published DCF methods is the assumption that the target shape is well approximated by an axis-aligned rectangle. For irregularly shaped objects or those with a hollow center, the filter eventually learns the background, which may lead to drift and failure. The same problem appears for approximately rectangular objects in the case of occlusion. The Galoogahi et al. [24] and Danneljan et al. [10] methods both suffer from this problem.

In this paper we introduce the CSR-DCF, the Discriminative Correlation Filter with Channel and Spatial Reliability. The spatial reliability map adapts the filter support to the part of the object suitable for tracking which overcomes both the problems of circular shift enabling an arbitrary search range and the limitations related to the rectangular shape assumption. The spatial reliability map is estimated using the output of a graph labeling problem solved efficiently in each frame. An efficient novel optimization procedure is derived for learning a correlation filter with the support constrained by the spatial reliability map since the standard closed-form solution cannot be generalized to this case. Experiments show that the novel filter optimization procedure outperforms related approaches for constrained learning in DCFs.

Channel reliability is the second novelty the CSR-DCF tracker introduces. The reliability is estimated from the properties of the constrained least-squares solution to filter design. The channel reliability scores are used for weighting the per-channel filter responses in localization (Figure 1). The CSR-DCF shows state-of-the-art performance on standard benchmarks – OTB100 [44], VOT2015 [26] and VOT2016 [26] while running in real-time on a single CPU. The spatial and channel reliability formulation is general and can be used in most modern correlation filters, e.g. those using deep features.

2. Related work

The discriminative correlation filters for object detection date back to the 80's with seminal work of Hester and Casasent [21]. They have been popularized only recently in the tracking community, starting with the Bolme et al. [4] MOSSE tracker published in 2010. Using a grayscale template, MOSSE achieved a state-of-the-art performance on a tracking benchmark [43] at a remarkable processing speed. Significant improvements have been made since and in 2014 the top-performing trackers on a recent benchmark [30] were all from this class of trackers. Improvements of DCFs fall into two categories, application of improved features and conceptual improvements in filter learning.

In the first group, Henriques et al. [20] replaced the grayscale templates by HoG [7], Danelljan et al. [12] proposed multi-dimensional color attributes and Li and Zhu [32] applied feature combination. Recently, the convolutional network features learned for object detection have been applied [35, 11, 13], leading to a performance boost, but at a cost of significant speed reduction.

Conceptually, the first successful theoretical extension of the standard DCF was the kernelized formulation by Henriques et al. [20]. Later, a correlation-filter-based scale adaptation was proposed by Danelljan et al. [8]. Zhang et al. [46] introduced spatio-temporal context learning in the DCFs. Recently, Galoogahi et al. [16] addressed the problems resulting from learning with circular correlation from small patches. They proposed a learning framework that artificially increases the filter size by implicit zero-padding to the right and down. The non-symmetric padding only partially reduces the boundary artefacts in filter learning. Danneljan et al. [10] reformulate the learning cost function to penalize non-zero filter values outside the object bounding box. Performance better than [16] is reported, but the learned filter is still a tradeoff between the correlation response and regularization, and it does not guarantee that filter values are zero outside of object bounding box.

3. Spatially constrained correlation filters

Given a set of N_d channel features $\mathbf{f} = \{\mathbf{f}_d\}_{d=1:N_d}$ and corresponding target templates (filters) $\mathbf{h} = \{\mathbf{h}_d\}_{d=1:N_d}$, where $\mathbf{f}_d \in \mathcal{R}^{d_w \times d_h}$, $\mathbf{h}_d \in \mathcal{R}^{d_w \times d_h}$, the object position $\mathbf{x}$ is estimated by maximizing the probability

$$p(\mathbf{x}|\mathbf{h}) = \sum_{d=1}^{N_d} p(\mathbf{x}|\mathbf{f}_d)p(\mathbf{f}_d). \tag{1}$$

The density $p(\mathbf{x}|\mathbf{f}_d) = [\mathbf{f}_d * \mathbf{h}_d](\mathbf{x})$ is a convolution of a feature map with a learned template evaluated at $\mathbf{x}$ and $p(\mathbf{f}_d)$ is a prior reflecting the channel reliability.

Most correlation filters, e.g., [31, 16, 46], assume independent feature channels. Optimal filters are obtained at learning stage by minimizing the sum of squared differences between the channel-wise correlation outputs and the desired output $\mathbf{g} \in \mathcal{R}^{d_w \times d_h}$,

$$\begin{aligned} &\arg\min_{\mathbf{h}} \sum_{d=1}^{N_d} \|\mathbf{f}_d * \mathbf{h}_d - \mathbf{g}\|^2 + \lambda \sum_{d=1}^{N_d} \|\mathbf{h}_d\|^2 \\ &= \arg\min_{\mathbf{h}} \sum_{d=1}^{N_d} (\|\hat{\mathbf{h}}_d^H \operatorname{diag}(\hat{\mathbf{f}}_d) - \hat{\mathbf{g}}_d\|^2 + \lambda \|\hat{\mathbf{h}}_d\|^2). \end{aligned} \tag{2}$$

The equivalence in (2) follows from the Parsevaal's theorem, the operator $\hat{\mathbf{a}} = \operatorname{vec}(\mathcal{F}[\mathbf{a}])$ is a Fourier transform of $\mathbf{a}$ reshaped into a column vector, i.e., $\mathbf{a} \in \mathcal{R}^{D \times 1}$, with $D = d_w \cdot d_h$, $\operatorname{diag}(\mathbf{a})$ forms a $D \times D$ diagonal matrix from $\mathbf{a}$ and $(\cdot)^H$ is a Hermitian transpose. Minimization of (2) has a closed-form solution by equating the complex gradient of

(2) w.r.t. each channel to zero [4]. Albeit its simplicity, this solution suffers from boundary defects due to input circularity assumption and from assuming all pixels are equally reliable for filter learning. In the following we address this issue by proposing an efficient spatial reliability map construction for correlation filters and propose a new spatially constrained correlation filter learning framework.

3.1. Constructing spatial reliability map

Spatial reliability map $\mathbf{m} \in [0,1]^{d_w \times d_h}$, with elements $m \in \{0,1\}$, indicates the learning reliability of each pixel. Probability of pixel $\mathbf{x}$ being reliable conditioned on appearance $\mathbf{y}$ is specified as

$$p(m=1|\mathbf{y},\mathbf{x}) \propto p(\mathbf{y}|m=1,\mathbf{x})p(\mathbf{x}|m=1)p(m=1). \tag{3}$$

The appearance likelihood $p(\mathbf{y}|m=1,\mathbf{x})$ is computed by Bayes rule from the object foreground/background color models, which are maintained during tracking as color histograms $\mathbf{c} = \{\mathbf{c}^{\mathrm{f}}, \mathbf{c}^{\mathrm{b}}\}$. The prior $p(m=1)$ is defined by the ratio between the region sizes for foreground/background histogram extraction.

Central pixels in axis-aligned approximations of elongated rotating, articulated or deformable object will likely contain the object regardless of deformation. On the other hand, pixels away from the center will equally likely contain object or background. This deformation invariance of central elements reliability is enforced in our approach by defining a weak spatial prior

$$p(\mathbf{x}|m=1) = k(\mathbf{x};\sigma), \tag{4}$$

where $k(\mathbf{x};\sigma)$ is a modified Epanechnikov kernel, $k(r;\sigma) = 1-(r/\sigma)^2$, with size parameter σ equal to the minor bounding box axis and clipped to interval $[0.5, 0.9]$ such that the object prior probability at center is 0.9 and changes to a uniform prior away from the center (Figure 2).

Spatial consistency of labeling $\mathbf{m}$ is enforced by using (3) as unary terms in a Markov random field. An efficient solver [28] is applied to compute maximum a posteriori solution of $\mathbf{m}$. To avoid potentially poor classifications at significant appearance changes, the interior of the object bounding box is set to a uniform distribution if less than a very small percentage of pixels $\alpha_{\min}$ are classified as foreground. The reliability map is morphologically dilated by a small kernel to prevent unwanted masking of object boundaries. Figure 2 shows the likelihood and spatial prior in the unary terms and the final binary reliability map.

3.2. Constrained correlation filter learning

In interest of notation clarity, we assume only a single channel in the following derivation, i.e., $N_d = 1$, and drop the channel index $(\cdot)_d$, since the filter learning is independent across the channels.

The spatial reliability map $\mathbf{m}$ identifies pixels in the filter that should be ignored in learning, i.e., introduces a constraint $\mathbf{h} \equiv \mathbf{m} \odot \mathbf{h}$, which prohibits a closed-form solution of (2). In the following we summarize our solution of this constrained optimization and report the full derivation in the supplementary material.

We introduce a dual variable $\mathbf{h}_c$ and the constraint $\mathbf{h}_c - \mathbf{m} \odot \mathbf{h} \equiv 0$, which leads to the following augmented Lagrangian [5]

$$\mathcal{L}(\hat{\mathbf{h}}_c, \mathbf{h}, \hat{\mathbf{l}}|\mathbf{m}) = \|\hat{\mathbf{h}}_c^H \operatorname{diag}(\hat{\mathbf{f}}) - \hat{\mathbf{g}}\|^2 + \frac{\lambda}{2}\|\mathbf{h}_m\|^2 + \tag{5}$$
$$[\hat{\mathbf{l}}^H(\hat{\mathbf{h}}_c - \hat{\mathbf{h}}_m) + \overline{\hat{\mathbf{l}}^H(\hat{\mathbf{h}}_c - \hat{\mathbf{h}}_m)}] + \mu\|\hat{\mathbf{h}}_c - \hat{\mathbf{h}}_m\|^2,$$

where $\hat{\mathbf{l}}$ is a complex Lagrange multiplier, $\mu > 0$, and we use the following definition $\mathbf{h}_m = (\mathbf{m} \odot \mathbf{h})$ for compact notation. The augmented Lagrangian (5) can be iteratively minimized by the alternating direction method of multipliers [5], which sequentially solves the following subproblems at each iteration:

$$\hat{\mathbf{h}}_c^{i+1} = \arg\min_{\mathbf{h_c}} \mathcal{L}(\hat{\mathbf{h}}_c, \mathbf{h}^i, \hat{\mathbf{l}}^i|\mathbf{m}), \tag{6}$$

$$\mathbf{h}^{i+1} = \arg\min_{\mathbf{h}} \mathcal{L}(\hat{\mathbf{h}}_c^{i+1}, \mathbf{h}, \hat{\mathbf{l}}^i|\mathbf{m}), \tag{7}$$

and the Lagrange multiplier is updated as

$$\hat{\mathbf{l}}^{i+1} = \hat{\mathbf{l}}^i + \mu(\hat{\mathbf{h}}_c^{i+1} - \hat{\mathbf{h}}^{i+1}). \tag{8}$$

The minimizations in (6) have a closed-form solution:

$$\hat{\mathbf{h}}_c^{i+1} = (\hat{\mathbf{f}} \odot \bar{\hat{\mathbf{g}}} + (\mu\hat{\mathbf{h}}_m^i - \hat{\mathbf{l}}^i)) \odot^{-1} (\hat{\mathbf{f}} \odot \bar{\hat{\mathbf{f}}} + \mu^i), \tag{9}$$

$$\mathbf{h}^{i+1} = \mathbf{m} \odot \mathcal{F}^{-1}[\hat{\mathbf{l}}^i + \mu^i\hat{\mathbf{h}}_c^{i+1}]/(\frac{\lambda}{2D} + \mu^i). \tag{10}$$

A standard scheme for updating the constraint penalty μ values [5] is applied, i.e., $\mu^{i+1} = \beta\mu^i$.

Computations of (9,8) are fully carried out in frequency domain, the solution for (10) requires a single inverse FFT and another FFT to compute the $\hat{\mathbf{h}}^{i+1}$. A single optimization iteration thus requires only two calls of the Fourier transform, resulting in a very fast optimization. The computational complexity is that of the Fourier transform, i.e., $\mathcal{O}(D \log D)$. Filter learning is implemented in less than five lines of Matlab code and is summarized in the Algorithm 1.

3.3. Channel reliability estimation

The channel reliability at target localization stage is computed as the product of a *learning* channel reliability measure and a *detection* reliability measure. These are described next.

Minimization of (5) solves a least squares problem averaged over all displacements of the filter on a feature channel. A discriminative feature channel $\mathbf{f}_d$ produces a filter $\mathbf{h}_d$

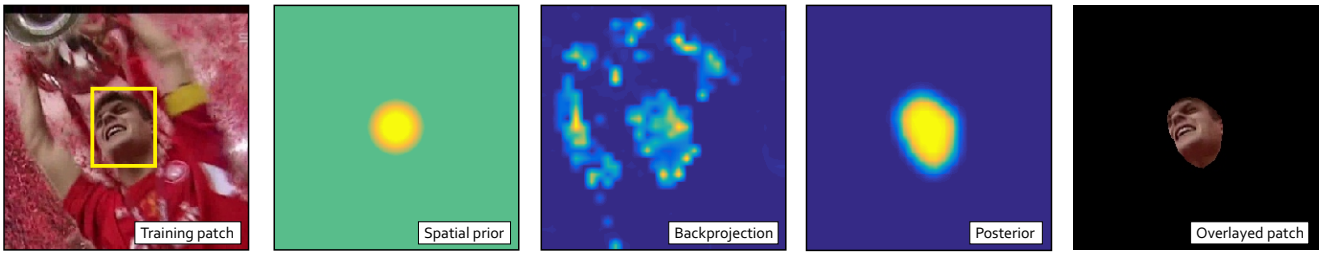

Figure 2: Spatial reliability map construction. From left to right: a training patch with the tracked object bounding box, the spatial prior used as a unary term in the Markov random field optimization, the object log-likelihood according to foreground-background color models, the posterior object probability after Markov random field regularization. The training patch masked with the final binary reliability map.

Algorithm 1 : Constrained filter optimization.

Require:
Image patch features $\mathbf{f}$, ideal correlation response $\mathbf{g}$, binary mask $\mathbf{m}$.
Ensure:
Optimized filter $\widehat{\mathbf{h}}$.
Procedure:
Initialize filter $\widehat{\mathbf{h}}^0$ by $\mathbf{h}_{t-1}$.
Initialize Lagrangian coefficients: $\widehat{\mathbf{l}}^0 \leftarrow$ zeros.
while stop condition **do**
  Calculate $\hat{\mathbf{h}}_c^{i+1}$ from $\hat{\mathbf{h}}^i$ and $\hat{\mathbf{l}}^i$ using (9).
  Calculate $\mathbf{h}^{i+1}$ from $\hat{\mathbf{h}}_c^{i+1}$ and $\hat{\mathbf{l}}^i$ using (10).
  Update the Lagrangian $\hat{\mathbf{l}}^{i+1}$ from $\hat{\mathbf{h}}_c^{i+1}$ and $\mathbf{h}^{i+1}$ (8).
end while

whose output $\mathbf{f}_d * \mathbf{h}_d$ nearly exactly fits the ideal response $\mathbf{g}$. On the other hand, the response is highly noisy on the channel with low discriminative power and a global error reduction in the least squares significantly reduces the maximal response. Thus a straight-forward measure of channel learning reliability $p(\mathbf{f}_d)$ in (1) is the maximum response of a learned channel filter, i.e., $w_d = \zeta \max(\mathbf{f}_d * \mathbf{h}_d)$, where the normalization scalar ζ ensures that $\sum_d w_d = 1$.

At detection stage, the per-channel detection reliability is reflected in the expressiveness of the major mode in the response of each channel. Note that Bolme et al. [4] proposed a similar approach to detect target loss. Our measure is based on the ratio between the second and first major mode in the response map, i.e., $\rho_{\max 2}/\rho_{\max 1}$. Note that this ratio penalizes cases when multiple similar objects appear in the target vicinity since these result in multiple equally expressed modes, even though the major mode accurately depict the target position. To prevent such penalizations, the ratio is clamped by 0.5. Therefore, the per-channel detection reliability is estimated as $w_d^{(\text{det})} = 1 - \min(\rho_{\max 2}/\rho_{\max 1}, \frac{1}{2})$.

3.4. Tracking with channel and spatial reliability

The localization and update steps of the proposed channel and spatial reliability correlation filter trackers (CSR-DCF) proceed as follows. **Localize.** Per-channel responses and the corresponding detection reliability values are computed (Section 3.3) and multiplied with the learning reliability measures from previous time-step $\mathbf{w}_{t-1}$ into channel reliability scores. The object is localized by summing the responses of the learned correlation filters $\mathbf{h}_{t-1}$ weighted by the estimated channel reliability scores. Scale is estimated by a single scale-space correlation filter from Danelljan et al. [8]. **Update.** Foreground/background histograms $\tilde{\mathbf{c}}$ are extracted at the estimated location and updated by an autoregressive scheme with learning rate η_c. The foreground histogram is extracted by a standard Epanechnikov kernel within the estimated object bounding box and the background is extracted from the neighborhood twice the object size. The spatial reliability map (Section 3.1) is constructed, the optimal filters $\tilde{\mathbf{h}}$ are computed by optimizing (5) and the per-channel learning reliability $\tilde{\mathbf{w}} = [\tilde{w}_1, \ldots, \tilde{w}_{N_d}]^T$ is estimated from their responses (Section 3.3). For temporal robustness, the filters and channel learning reliability weights are updated by an autoregressive model with learning rate η. A single tracking iteration is summarized in Algorithm 2.

4. Experimental analysis

This section overviews a comprehensive experimental evaluation of the proposed CSR-DCF tracker. Implementation details are discussed in Section 4.1, Section 4.2 reports comparison of the proposed constrained learning to the related state-of-the-art, ablation study is provided in Section 4.3, performance on three recent benchmarks is reported in Section 4.4, Section 4.5 and Section 4.6. Section 4.7 evaluates the tracking speed.

4.1. Implementation details and parameters

Standard HOG [15] and Colornames [39] features are used in the correlation filter and HSV fore-

Algorithm 2 : The CSR-DCF tracking algorithm.

Require:

Image $\mathbf{I}_t$, object position on previous frame $\mathbf{p}_{t-1}$, scale s_{t-1}, filter $\mathbf{h}_{t-1}$, color histograms $\mathbf{c}_{t-1}$, channel reliability $\mathbf{w}_{t-1}$.

Ensure:

Position $\mathbf{p}_t$, scale s_t and updated models.

Localization and scale estimation:

1: New target location $\mathbf{p}_t$: position of the maximum in correlation between $\mathbf{h}_{t-1}$ and image patch features $\mathbf{f}$ extracted on position $\mathbf{p}_{t-1}$ and weighted by the channel reliability scores (Section 3.3).
2: Using location $\mathbf{p}_t$, estimate new scale s_t.

Update:

3: Extract foreground and background histograms $\tilde{\mathbf{c}}^f$, $\tilde{\mathbf{c}}^b$.
4: Update foreground and background histograms $\mathbf{c}_t^f = (1-\eta_c)\mathbf{c}_{t-1}^f + \eta_c\tilde{\mathbf{c}}^f$, $\mathbf{c}_t^b = (1-\eta_c)\mathbf{c}_{t-1}^b + \eta_c\tilde{\mathbf{c}}^b$.
5: Estimate reliability map $\mathbf{m}$ (Section 3.1).
6: Estimate a new filter $\tilde{\mathbf{h}}$ using $\mathbf{m}$ (Algorithm 1).
7: Estimate channel reliability $\tilde{\mathbf{w}}$ from $\mathbf{h}$ (Section 3.3).
8: Update filter $\mathbf{h}_t = (1-\eta)\mathbf{h}_{t-1} + \eta\tilde{\mathbf{h}}$.
9: Update channel reliability $\mathbf{w}_t = (1-\eta)\mathbf{w}_{t-1} + \eta\tilde{\mathbf{w}}$.

ground/background color histograms with 16 bins per color channel are used in reliability map estimation with parameter $\alpha_{\min} = 0.05$. All the parameters are set to values commonly used in literature [10, 16]. Histogram adaptation rate is set to $\eta_c = 0.04$, correlation filter adaptation rate is set to $\eta = 0.02$, and the regularization parameter is set to $\lambda = 0.01$. The augmented Lagrangian optimization parameters are set to $\mu^0 = 5$ and $\beta = 3$. All parameters have straight-forward interpretation, do not require fine-tuning, and were kept constant throughout all experiments. Our Matlab implementation[1] runs at 13 frames per second on an Intel Core i7 3.4GHz standard desktop.

4.2. Impact of boundary constraint formulation

This section compares our proposed boundary constraints formulation (Section 3) with recent state-of-the-art approaches [10, 16]. In the first experiment, three variants of the standard single-scale HOG-based correlation filter were implemented to emphasize the difference in boundary constraints: the first one uses our channel reliability boundary constraint formulation from Section 3 ($\mathrm{T_{CR}}$) the second one applies the spatial regularization constraint [10] ($\mathrm{T_{SR}}$) and the third one applies the limited boundaries constraint [16] ($\mathrm{T_{LB}}$).

The three variants were compared on the challenging VOT2015 dataset [26] by applying a standard no-reset one-pass evaluation (OTB [43]) and computing the AUC on the success plot. The tracker with our constraint formulation

[1]The CSR-DCF Matlab source code will be made publicly available.

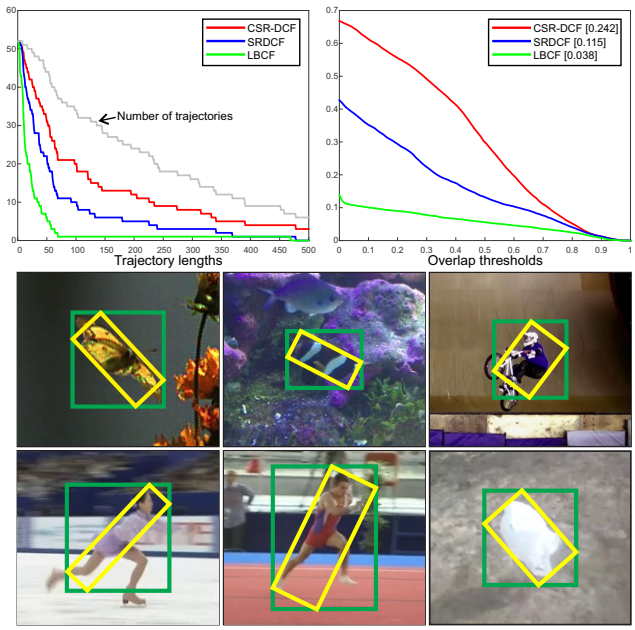

Figure 3: The number of trajectories with tracking successful up to frame Θ_{frm} (upper left), the success plots (upper right) and initialization examples of non-axis-aligned targets (bottom).

$\mathrm{T_{CR}}$ achieved 0.32 AUC, while the alternatives achieved 0.28 ($\mathrm{T_{SR}}$) and 0.16 ($\mathrm{T_{LB}}$). The only difference between these tackers is in the constraint formulation, which indicates superiority of the proposed channel-reliability-based constraints formulation over the recent alternatives [16, 10].

4.2.1 Non-axis-aligned target initialization robustness

The proposed CSR-DCF tracker from Section 3 was compared to the original recent state-of-the-art trackers SRDCF [10] and CFLB [24] that apply alternative boundary constraints. The source code was obtained from the authors and only HoG features were used in all three trackers for a fair comparison. An experiment was designed to evaluate initialization and tracking of non axis-aligned targets, which is the case for most realistic deforming and non-circular objects. Trackers were initialized on frames with non-axis aligned targets and left to track until the sequence end, resulting in a large number of tracking trajectories and summarized by various performance measures.

The VOT2015 dataset [26] contains non-axis-aligned annotations, which allows automatic identification of tracker initialization frames, i.e., frames in which the ground truth bounding box significantly deviates from an axis-aligned approximation. Frames with overlap between ground truth and axis-aligned approximation lower than 0.5 were identified and filtered to obtain a set of initialization frames at

Table 1: Comparison of three most related trackers on non-axis-aligned initialization experiment: weighted average tracking length in frames Γ_{frm} and proportions Γ_{prp}, and weighted average overlaps using the original and axis-aligned ground truth, Φ_{rot} and Φ_{aa}, respectively.

Tracker	Γ_{prp}	Γ_{frm}	Φ_{aa}	Φ_{rot}
CSR-DCF	**0.58**	**221**	**0.31**	**0.24**
SRDCF (ICCV2015)	*0.31*	*95*	*0.16*	*0.12*
LBCF (CVPR2015)	0.12	37	0.06	0.04

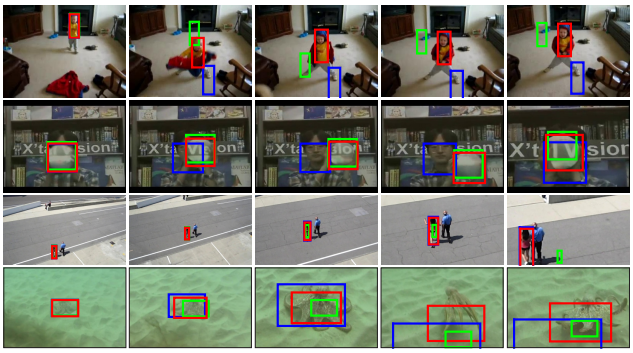

Figure 4: Qualitative results for trackers CSR-DCF (red) tracker, SRDCF (blue) and LBCF (green).

least hundred frames apart – this constraint fits half the typical short-term sequence length [26] and reduces the potential correlation across the initializations (see Figure 3 for examples).

Initialization robustness is estimated by counting the number of trajectories in which the tracker was still tracking (overlap with ground truth greater than 0) Θ_{frm} frames after initialization. Figure 3 shows these values with increasing the threshold Θ_{frm}. The CSR-DCF graph is consistently above the SRDCF and LBCF for all thresholds. The performance is summarized by the average tracking length (number of frames before the overlap drops to zero) weighted by trajectory lengths. The weighted average tracking lengths in frames, Γ_{frm}, and proportions of full trajectory lengths, Γ_{prp}, are shown in Table 1. The CSR-DCF by far outperforms SRDCF and LBCF in all measures indicating a significant robustness at initialization of challenging targets that deviate from axis-aligned templates. This improvement is further confirmed by Figure 3 which shows the OTB success plots [43] calculated on these trajectories and summarized by the AUC values, which are equal to the average overlaps [6]. Table 1 shows the average overlaps computed on the original ground truth on VOT2015 (Φ_{rot}) and on ground truth approximated by the axis-aligned bounding box (Φ_{aa}). Again, the CSR-DCF by far outperforms the competing alternatives SRDCF and LBCF. Tracking examples for the three trackers are shown in Figure 4.

Table 2: Ablation study of CSR-DCF.

Tracker	EAO	R_{av}	A_{av}
CSR-DCF	**0.338**	**0.85**	**0.51**
CuSR-DCF	*0.297*	*1.08*	*0.51*
CSuR-DCF	0.264	1.18	0.49
CuSuR-DCF	0.256	1.33	*0.51*
DCF	0.152	2.85	0.47

4.3. Spatial and channel reliability ablation study

An ablation study on VOT2016 (see Section 4.6 for details of the evaluation protocol) was conducted to evaluate the contribution of spatial and channel reliability measures in our CSR-DCF. Results of the VOT primary measure EAO and two supplementary measures (A,R) are summarized in Table 2. Setting the adaptive channel reliability weights to uniform values (CuSR-DCF) results in 12% performance drop in EAO compared to CSR-DCF. Replacing the adaptive spatial reliability map in CSR-CDF by a constant map with uniform values within the bounding box and zeros elsewhere (CSuR-DCF), results in a 21% drop in EAO. Making both replacements in CSR-DCF (CuSuR-DCF) results in 24% drop. Removing the channel and spatial reliability map reduces our tracker to a standard DCF with a large receptive field – the performance drops by over 50%.

4.4. The OTB100 benchmark [44]

The OTB100 [44] benchmark contains results of 29 trackers evaluated on 100 sequences by a no-reset evaluation protocol. Tracking quality is measured by precision and success plots. Success plot shows portion of frames with the overlap between predicted and ground truth bounding box greater than a threshold with respect to all threshold vales. The precision plot shows similar statistics on the center error. The results are summarized by areas under these plots. To reduce clutter in the graphs, we show here only the results for top-performing recent baselines, i.e., Struck [18], TLD [23], CXT [14], ASLA [45], SCM [42], LSK [34], CSK [19] and results for recent top-performing state-of-the-art trackers SRDCF [10] and MUSTER [22].

The CSR-DCF is ranked top on the benchmark (Figure 5). It significantly outperforms the best performers reported in [44] and outperforms the current state-of-the-art (SRDCF [10] and MUSTER [22]). The average CSR-DCF performance on success plot is slightly lower than SRDCF [9] due to poorer scale estimation, but yields better performance in the average precision (center error). Both, precision and success plot, show that the CSR-DCF tracks on average longer than competing methods.

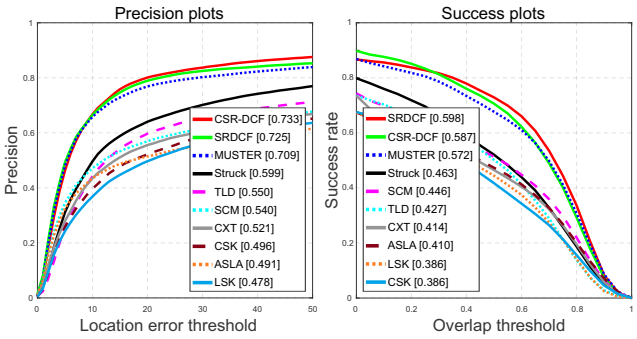

Figure 5: OTB100 [44] benchmark comparison. The precision plot (left) and the success plot (right).

4.5. The VOT2015 benchmark [26]

The VOT2015 [26] benchmark contains results of 63 state-of-the-art trackers evaluated on 60 challenging sequences. In contrast to related benchmarks, the VOT2015 dataset was constructed from 300 sequences by an advanced sequence selection methodology that favors objects difficult to track and maximizes a visual attribute diversity cost function [26]. This makes it arguably the most challenging sequence set available. The VOT methodology [27] resets a tracker upon failure to fully use the dataset. The basic VOT measures are the number of failures during tracking (robustness) and average overlap during the periods of successful tracking (accuracy), while the primary VOT2015 measure is the expected average overlap (EAO) on short-term sequences. The latter can be thought of as the expected no-reset average overlap (AUC in OTB methodology), but with reduced bias and the variance as explained in [26].

Figure 7 shows the VOT EAO plots with the CSR-DCF and the VOT2015 state-of-the-art approaches considering the VOT2016 rules that do not consider trackers learned on video sequences related to VOT to prevent over-fitting. The CSR-DCF outperforms all trackers and achieves a top rank. The CSR-DCF significantly outperforms the related correlation filter trackers like SRDCF [9] as well as trackers that apply computationally-intesive state-of-the-art deep features e.g., deepSRDCF [11] and SO-DLT [41].

4.6. The VOT2016 benchmark [25]

Finally, we compare our tracker on the most recent visual tracking benchmark, VOT2016 [25]. The dataset contains 60 sequences from VOT2015 [26] with improved annotations. The benchmark evaluated a set of 70 trackers which includes the recently published and yet unpublished state-of-the-art trackers. The set is indeed diverse, the top-performing trackers come from various classes e.g., correlation filter methods (CCOT [13], Staple [2], DDC [25]), deep ConvNets (TCNN [25], SSAT[25, 37], MLDF [25, 40], FastSiamnet [3]) and different detection-

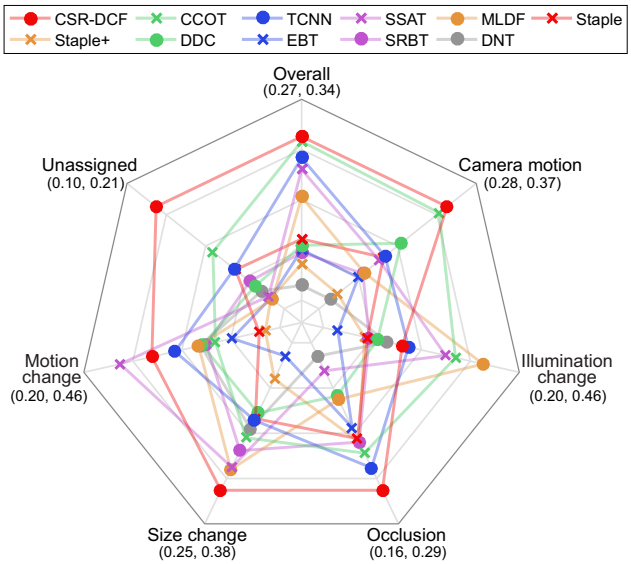

Figure 6: Expected averaged overlap performance on different visual attributes on the VOT2016 [25] benchmark. The proposed CSR-DCF and the top 10 performing trackers from VOT2016 are shown. The scales of visual attribute axes are displayed below the attribute labels.

based approaches (EBT [47], SRBT [25]).

Figure 7 shows the EAO performance on the VOT2016. Our proposed CSR-DCF outperforms all 70 trackers at the EAO score 0.338. The CSR-DCF significantly outperforms correlation filter approaches that do not apply deep ConvNets. Even though the CSR-DCF applies only simple features, it outperforms all trackers that apply computationally intensive deep features.

The VOT2016 [25] dataset is per-frame annotated with visual attributes and allows detailed analysis of per-attribute tracking performance. Figure 6 shows per-attribute plot for ten top-performing trackers on VOT2016 in EAO. The proposed CSR-DCF is consistently ranked among top three trackers on five out of six attributes. In four attributes (size change, occlusion, camera motion, unassigned) the tracker is ranked number one.

4.7. Tracking speed analysis

Tracking speed is an important factor of many real-world tracking problems. Table 3 thus compares several related and well-known trackers (including the best-performing tracker on the VOT2016 challenge) in terms of speed and VOT performance measures. Speed measurements on a single CPU are computed on Intel Core i7 3.4GHz standard desktop.

The proposed CSR-DCF performs on par with the VOT2016 best-performing CCOT [13], which applies deep ConvNets, with respect to VOT measures, while being 20

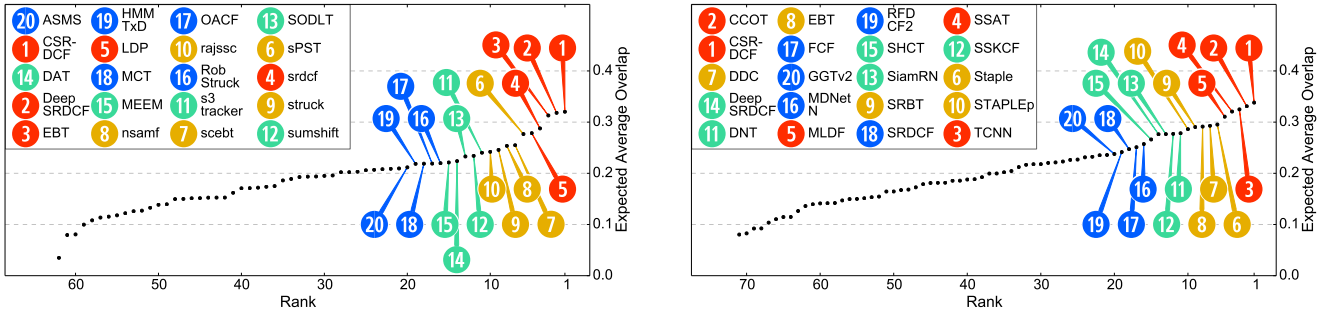

Figure 7: Expected average overlap plot for VOT2015 [26] (left) and VOT2016 [25] (right) benchmarks with the proposed CSR-DCF tracker. Legends are shown only for top performing trackers. The graphs with full legends are provided in supplementary materials.

Table 3: Speed in frames per second (fps) of correlation trackers and Struck – a baseline. The EAO, average accuracy (A_{av}) and average failures (R_{av}) are shown for reference.

Tracker	Published at	EAO	A_{av}	R_{av}	fps
CSR-DCF	This work.	**0.338**	*0.51*	**0.85**	13.0
CCOT	ECCV2016	*0.331*	**0.52**	**0.85**	0.55
CCOT*	ECCV2016	0.274	**0.52**	*1.18*	1.0
SRDCF	ICCV2015	0.247	**0.52**	1.50	7.3
KCF	PAMI2015	0.192	0.48	2.03	**115.7**
DSST	PAMI2016	0.181	0.48	2.52	*18.6*
Struck	ICCV2011	0.142	0.42	3.37	8.5

times faster than the CCOT. The CCOT was modified by replacing the computationally intensive deep features with the same simple features used in CSR-DCF. The resulting tracker, indicated by CCOT*, is still ten times slower than CSR-DCF, while the performance drops by over 15%. The proposed CSR-DCF performs twice as fast as the related SRDCF [9], while achieving approximately 25% better tracking results. The speed of baseline real-time trackers like DSST [8] and Struck [18] is comparable to CSR-DCF, but their tracking performance is significantly poorer. The fastest compared tracker, KCF [20] runs much faster than real-time, but delivers a significantly poorer performance than CSR-DCF.

The experiments show that the CSR-DCF performs at comparably to the state-of-the-art trackers which apply computationally demanding high-dimensional features, but runs considerably faster and delivers top tracking performance among the real-time trackers.

5. Conclusion

The Discriminative Correlation Filter with Channel and Spatial Reliability (CSR-DCF) was introduced. The spatial reliability map adapts the filter support to the part of the object suitable for tracking which overcomes both the problems of circular shift enabling an arbitrary search range and the limitations related to the rectangular shape assumption. A novel efficient spatial map estimation method was proposed and an efficient optimization procedure derived for learning a correlation filter with the support constrained by the spatial reliability map. The second novelty of CSR-DCF is the channel reliability. The reliability is estimated from the properties of the constrained least-squares solution. The channel reliability scores were used for weighting the per-channel filter responses in localization.

Experimental comparison with recent related state-of-the-art boundary-constraints formulations showed significant benefits of using our formulation. The CSR-DCF has state-of-the-art performance on standard benchmarks – OTB100 [44], VOT2015 [26] and VOT2016 [26] while running in real-time on a single CPU. Despite using simple features like HoG and Colornames, the CSR-DCF performs on par with trackers that apply computationally complex deep ConvNet, but is significantly faster.

To the best of our knowledge, the proposed approach is the first of its kind to introduce constrained filter learning with arbitrary spatial reliability map and the use of channel reliabilities. The spatial and channel reliability formulation is general and can be used in most modern correlation filters, e.g. those using deep feature.

References

[1] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(8):1619–1632, Aug. 2011. 1

[2] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr. Staple: Complementary learners for real-time tracking. In *Comp. Vis. Patt. Recognition*, pages 1401–1409, June 2016. 7

[3] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr. Fully-convolutional siamese networks for object

tracking. *arXiv preprint arXiv:1606.09549*, 2016. 7

[4] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui. Visual object tracking using adaptive correlation filters. In *Comp. Vis. Patt. Recognition*, pages 2544–2550. IEEE, 2010. 1, 2, 3, 4

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011. 3

[6] L. Čehovin, A. Leonardis, and M. Kristan. Visual object tracking performance measures revisited. *IEEE Trans. Image Proc.*, 25(3):1261–1274, 2016. 6

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Comp. Vis. Patt. Recognition*, volume 1, pages 886–893, June 2005. 2

[8] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg. Accurate scale estimation for robust visual tracking. In *Proc. British Machine Vision Conference*, pages 1–11, 2014. 1, 2, 4, 8

[9] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg. Learning spatially regularized correlation filters for visual tracking. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 4310–4318, 2015. 6, 7, 8

[10] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg. Learning spatially regularized correlation filters for visual tracking. In *Int. Conf. Computer Vision*, pages 4310–4318, 2015. 1, 2, 5, 6

[11] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg. Convolutional features for correlation filter based visual tracking. In *IEEE International Conference on Computer Vision Workshop (ICCVW)*, pages 621–629, Dec 2015. 2, 7

[12] M. Danelljan, F. S. Khan, M. Felsberg, and J. van de Weijer. Adaptive color attributes for real-time visual tracking. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 1090–1097, 2014. 2

[13] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg. Beyond correlation filters: learning continuous convolution operators for visual tracking. In *Proc. European Conf. Computer Vision*, pages 472–488. Springer, 2016. 2, 7

[14] T. B. Dinh, N. Vo, and G. Medioni. Context tracker: Exploring supporters and distracters in unconstrained environments. In *Comp. Vis. Patt. Recognition*, pages 1177–1184, 2011. 6

[15] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, Sept 2010. 4

[16] H. K. Galoogahi, T. Sim, and S. Lucey. Multi-channel correlation filters. In *Int. Conf. Computer Vision*, pages 3072–3079, 2013. 2, 5

[17] H. Grabner, M. Grabner, and H. Bischof. Real-time tracking via on-line boosting. In *Proc. British Machine Vision Conference*, volume 1, pages 47–56, 2006. 1

[18] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured output tracking with kernels. In *Int. Conf. Computer Vision*, pages 263–270, Washington, DC, USA, 2011. IEEE Computer Society. 1, 6, 8

[19] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, editors, *Proc. European Conf. Computer Vision*, pages 702–715, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. 6

[20] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. High-speed tracking with kernelized correlation filters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(3):583–596, 2015. 1, 2, 8

[21] C. F. Hester and D. Casasent. Multivariant technique for multiclass pattern recognition. *Applied Optics*, 19(11):1758–1761, 1980. 2

[22] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, and D. Tao. Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking. In *Comp. Vis. Patt. Recognition*, pages 749–758, June 2015. 6

[23] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(7):1409–1422, July 2012. 6

[24] H. Kiani Galoogahi, T. Sim, and S. Lucey. Correlation filters with limited boundaries. In *Comp. Vis. Patt. Recognition*, pages 4630–4638, 2015. 1, 2, 5

[25] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojir, G. Häger, A. Lukežič, and G. et al. Fernandez. The visual object tracking vot2016 challenge results. In *Proc. European Conf. Computer Vision*, 2016. 7, 8

[26] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Čehovin, G. Fernandez, T. Vojir, G. Häger, G. Nebehay, and R. et al. Pflugfelder. The visual object tracking vot2015 challenge results. In *Int. Conf. Computer Vision*, 2015. 1, 2, 5, 6, 7, 8

[27] M. Kristan, J. Matas, A. Leonardis, T. Vojir, R. Pflugfelder, G. Fernandez, G. Nebehay, F. Porikli, and L. Cehovin. A novel performance evaluation methodology for single-target trackers. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2016. 1, 7

[28] M. Kristan, J. Perš, V. Sulič, and S. Kovačič. A graphical model for rapid obstacle image-map estimation from unmanned surface vehicles. In *Proc. Asian Conf. Computer Vision*, pages 391–406, 2014. 3

[29] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, F. Porikli, L. Čehovin, G. Nebehay, G. Fernandez, and T. e. a. Vojir. The visual object tracking vot2013 challenge results. In *Vis. Obj. Track. Challenge VOT2013, In conjunction with ICCV2013*, pages 98–111, Dec 2013. 1

[30] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, L. Čehovin, G. Nebehay, T. Vojir, and G. et al. Fernandez. The visual object tracking vot2014 challenge results. In *Proc. European Conf. Computer Vision*, pages 191–217, 2014. 1, 2

[31] Y. Li and J. Zhu. A scale adaptive kernel correlation filter tracker with feature integration. In *Proc. European Conf. Computer Vision*, pages 254–265, 2014. 1, 2

[32] Y. Li and J. Zhu. A scale adaptive kernel correlation filter tracker with feature integration. In *Proc. European Conf. Computer Vision*, pages 254–265, 2014. 2

[33] P. Liang, E. Blasch, and H. Ling. Encoding color information for visual tracking: Algorithms and benchmark. *IEEE Trans. Image Proc.*, 24(12):5630–5644, Dec 2015. 1

[34] B. Liu, J. Huang, L. Yang, and C. Kulikowsk. Robust tracking using local sparse appearance model and k-selection. In *Comp. Vis. Patt. Recognition*, pages 1313–1320, June 2011. 6

[35] C. Ma, J. B. Huang, X. Yang, and M. H. Yang. Hierarchical convolutional features for visual tracking. In *Int. Conf. Computer Vision*, pages 3074–3082, Dec 2015. 2

[36] M. Mueller, N. Smith, and B. Ghanem. A benchmark and simulator for uav tracking. In *Proc. European Conf. Computer Vision*, 2016. 1

[37] H. Nam and B. Han. Learning multi-domain convolutional neural networks for visual tracking. In *Comp. Vis. Patt. Recognition*, pages 4293–4302, June 2016. 7

[38] A. Smeulders, D. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual tracking: An experimental survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(7):1442–1468, July 2014. 1

[39] J. van de Weijer, C. Schmid, J. Verbeek, and D. Larlus. Learning color names for real-world applications. *IEEE Trans. Image Proc.*, 18(7):1512–1523, July 2009. 4

[40] L. Wang, W. Ouyang, X. Wang, and H. Lu. Visual tracking with fully convolutional networks. In *Int. Conf. Computer Vision*, pages 3119–3127, Dec 2015. 7

[41] N. Wang, S. Li, A. Gupta, and D. Yeung. Transferring rich feature hierarchies for robust visual tracking. *CoRR*, abs/1501.04587, 2015. 7

[42] M.-H. Y. Wei Zhong, Huchuan Lu. Robust object tracking via sparsity-based collaborative model. In *Comp. Vis. Patt. Recognition*, pages 1838–1845, 2012. 6

[43] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In *Comp. Vis. Patt. Recognition*, pages 2411–2418, 2013. 1, 2, 5, 6

[44] Y. Wu, J. Lim, and M. H. Yang. Object tracking benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(9):1834–1848, Sept 2015. 2, 6, 7, 8

[45] M.-H. Y. Xu Jia, Huchuan Lu. Visual tracking via adaptive structural local sparse appearance model. In *Comp. Vis. Patt. Recognition*, pages 1822–1829, 2012. 6

[46] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang. Fast visual tracking via dense spatio-temporal context learning. In *Proc. European Conf. Computer Vision*, pages 127–141. Springer International Publishing, 2014. 2

[47] G. Zhu, F. Porikli, and H. Li. Beyond local search: Tracking objects everywhere with instance-specific proposals. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 943–951, June 2016. 7