IGSSTRCF: Importance Guided Sparse Spatio-Temporal Regularized Correlation Filters For Tracking

Monika Jain
QUT, Brisbane, Australia
IIIT, Delhi, India
monikaj@iiitd.ac.in

A V Subramanyam
IIIT, Delhi, India
subramanyam@iiitd.ac.in

Simon Denman
QUT, Brisbane, Australia
s.denman@qut.edu.au

Sridha Sridharan
QUT, Brisbane, Australia
s.sridharan@qut.edu.au

Clinton Fookes
QUT, Brisbane, Australia
c.fookes@qut.edu.au

Abstract

This paper proposes a novel Importance Guided Sparse Spatio-Temporal Regularization based Correlation Filter (IGSSTRCF) tracker. Our formulation explicitly models the variations in the correlation filters and associated spatial weights in successive frames. By imposing a sparsity penalty on these variations, the formulation ensures that only relevant changes are incorporated during updates. This results in more robust filter coefficients that minimize the tracking drift. The IGSSTRCF also includes an adaptive channel importance estimation strategy that assigns an importance weight to each feature channel during training. The proposed formulation is efficiently solved via the alternating direction method of multipliers. A comparative analysis is shown on TC128, UAV123, VOT-2017, and VOT-2019 datasets; and we present an ablation study to demonstrate the contribution of each component of the IGSSTRCF. It is observed that we outperform several state-of-the-art trackers and each component of the proposed IGSSTRCF contributes positively towards tracker performance.

1. Introduction

Visual object tracking is a widely scrutinized research problem in the field of computer vision and video analytics. Recent Correlation Filter (CF) based trackers use deep features extracted using a pre-trained CNN. These trackers model the target appearance with generalized and discriminative deep features, whilst achieving real time speed due to fast computations in the Fourier domain [28]. However, the computational efficiency of these CF trackers comes with a trade off. Negative training samples are generated by circularly shifting the base patch in the frequency domain [28]. Such negative samples do not represent the actual background and are impacted by boundary effects. Training with these artificial negative samples can result in an over-fitted CF that struggles to adapt to rapid visual deformations of the target, leading to tracker drift [11]. To mitigate these issues, CF trackers with spatial and temporal constraints have been proposed [6, 11, 23, 30]. However, these trackers either do not model or weakly incorporate the spatial and temporal variations between consecutive frames. In addition, the multi-channel CNN feature encodes a different attribute of the target in each channel. Therefore, the importance of each channel may change from one tracking step to the next. Some channels may offer more informative features for tracking, while others with less useful information may degrade the tracking and eventually lead to tracker drift [11]. To address this issue of channel importance, feature selection [49], adaptive importance maps [29] and reliability learning [45] methods have been proposed. However, these tracker formulations do not offer efficient spatial and temporal regularizations, reducing awareness of previous and spatially adjacent observations.

The challenges with the spatial regularization based CF trackers are that they either have fixed spatial weights [11], or learn spatial weights that are similar to some reference weights [6]. Likewise, the temporal regularization based CF tracker imposes a constraint such that the current learned filter is similar to the previous filter [30]. However, due to continuous temporal and spatial variations in a tracking sequence, the filter and spatial weights in a tracking step will not be identical to their reference counterparts.

The motivation for our proposed approach stems from the proposition that the above variations can be explicitly modelled using a constrained optimization framework. To model the spatial and temporal variations, we propose an Importance Guided Sparse Spatio-Temporal Regularization based CF (IGSSTRCF) tracker with following advances:

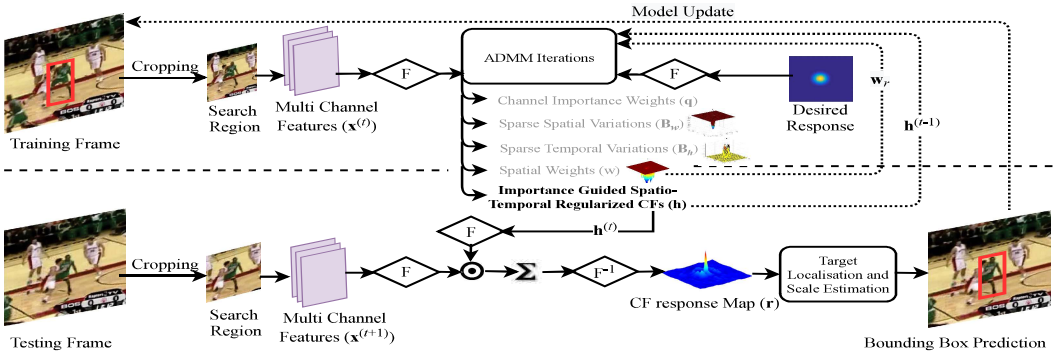

Figure 1: A Block diagram for the proposed IGSSTRCF tracker. During training, $\mathbf{q}, \mathbf{B}_w, \mathbf{B}_h, \mathbf{w}$ and $\mathbf{h}$ are learned via ADMM iterations. During testing, we extract an ensemble of deep and hand-crafted features from the search area. The target is localized using a response map obtained by the dot product of the Fourier transformed features and filters. For target scale estimation, we follow [6]. F and F^{-1} denotes Fourier and inverse Fourier transform operations respectively

1. We introduce a Sparse Spatial Regularization (SSR) component that learns the spatial weights with the help of reference weights, and simultaneously models the sparse difference between the reference weights and the learned spatial weights. The filter coefficients belonging to the background region are assigned higher penalty weights. This suppresses the effect of unfavorable background information and boundary effects in the learned filter.
2. We introduce a Sparse Temporal Regularization (STR) component that learns a correlation filter by modelling the sparse difference between the previous and the current filter. As a result, the filter sparsely adapts to appearance changes, preventing drift.
3. We introduce a Channel Importance (CI) term that assigns higher weights to the feature channels that encode useful target information, and lowers weights to the less informative channels. As a result, less informative channels that may adversely effect training are suppressed.

Figure 1 shows a block diagram that describes the process flow of the proposed IGSSTRCF tracker. We evaluate the tracker on the benchmark datasets: TC128 [36], UAV123 [40], VOT-2019 [27] and VOT-2017 [26]. A comparative analysis shows that the proposed formulation results in a significant improvement over the baselines [6, 30] and other recent trackers. An ablation study is also presented that demonstrates the importance of each regularization term for tracker performance.

2. Related Work

In recent years, an increased demand for computational efficiency has made Correlation Filters (CFs) a frequently used formulation for visual object tracking [8, 11, 14, 18, 20, 23, 47, 50]. MOSSE [4] is one of the earliest trackers that learns a CF in the frequency domain. It uses a single-channel gray-scale image to train the CF and offers impressive tracking speed. The limitations of MOSSE [4] have been addressed by many tracking algorithms proposed afterwards. To list a few, a multi-channel version of MOSSE [4] is proposed in [24]. Henriques *et al.* proposed kernelized CFs [20], and the use of high dimensional features is proposed in [19].

Although computationally efficient, CFs learned in the frequency domain are impacted by boundary effects that plague the circularly shifted training patches. This leads to sub-optimal training. Also, the learning is done solely using the shifted patches of the target, and background information is completely overlooked in the learning process. This results in an over-fitted CF that is prone to poor discrimination when encountering background clutter and occlusion [11]. Several methods have been proposed to alleviate the aforementioned issues. Danelljan *et al.* [11] introduces a weighted regularization constraint in the CF formulation to penalize filter coefficients near the boundary region. BACF [23] proposes generating real world positive and negative training samples by directly multiplying the filter with a binary matrix. This improves the discriminative ability of the filters. The above approaches have been used as a baseline for many subsequent CF based trackers [8, 13, 15, 30, 45].

SITUP [39] introduces an exhaustive scale searching generic framework that can be easily employed in other CF based trackers for efficient scale estimation. [17] proposes to avoid ad hoc linear interpolation and learns increments of the CFs by using a smooth incremental learning framework. A structural spatio-temporal model for tracking is introduced in [54] that uses an adaptive generative learning method for extracting complementary features that can represent the temporal appearance changes of target and impose adaptive spatial regularization. A spatio-temporally regularized CF tracker is introduced in [32] that simultaneously exploits the local and global information in the re-

sponse maps and automatically tunes the hyper-parameters.

Many other works propose different spatial constraints and use spatially larger training samples compared to the trained filter [10, 11, 12, 25, 48]. These approaches suppress the background information during training [11] and have demonstrated a significant reduction of boundary effects [25]. Further advancements over SRDCF [11] are made in STRCF [30] and ASRCF [6]. Li *et al.* [30] employs spatio-temporal constraints that utilize CFs learned in the previous frame to learn the CFs in the current frame. Dai *et al.* in [6] introduces an object aware spatial regularization that attempts to learn spatial weights that are similar to the reference spatial weights. The regularization terms in [30] and [6] make use of a reference to learn the CFs and spatial weights. However, the target appearance varies with every frame. Therefore, the spatial weights or CFs learned in consecutive frames should be constrained to be similar while still adapting to variations.

Besides the above methods, many CF based trackers focus on modeling channel importance as each feature channel can make a dynamic contribution during each tracking step. The benefits of selecting the optimal channels during tracking with multi-channel features are investigated in [16]. Lu *et al.* [37] makes use of channel regularization and learns a weight for each feature channel in order to suppress redundant information. Li *et al.* [29] introduced a feature integration method for correlation filters, where the filters and importance maps are jointly learned in each frame. Sun *et al.* [45] propose a CF-based optimization method that jointly models discrimination and reliability information. Zhou *et al.* [53] propose to learn a discriminative and robust dictionary that preserves the locality and similarity of the input to achieve more accurate visual tracking. However, these channel importance based CF trackers do not employ spatial [10, 11, 12, 25] or temporal [6, 30] regularization.

To combat the shortcomings of the above spatio-temporal regularization based [6, 30] and channel importance based [29, 45, 53] CF trackers, we propose an Importance Guided Sparse Spatio-Temporal Regularization based CF (IGSSTRCF) tracker. The proposed tracker includes an object aware sparse spatial regularization and a sparse temporal regularization. The proposed tracker also incorporates an adaptive channel importance estimation mechanism that assigns importance weights to each feature channel. To obtain a local optimal solution for the complete formulation, Alternating Direction Method of Multipliers (ADMM) [5] is used. The proposed regularizations result in learning more discriminative filter coefficients compared to the baseline trackers [6, 30]. The baseline trackers are presented briefly below.

2.1. ASRCF

The ASRCF [6] formulation can be given by,

$$E(\mathbf{H},\mathbf{w}) = \frac{1}{2}\left\|\mathbf{y} - \sum_{k=1}^{K}\mathbf{x}_k * (\mathbf{P}^T\mathbf{h}_k)\right\|_2^2 + \frac{\lambda_1}{2}\sum_{k=1}^{K}\|\mathbf{w}\odot\mathbf{h}_k\|_2^2 + \frac{\lambda_2}{2}\left\|\mathbf{w}-\mathbf{w}_r\right\|_2^2, \quad (1)$$

where K is the total number of feature channels, and $\mathbf{y} \in \mathbb{R}^{T\times 1}$ is the desired Gaussian shaped correlation filter response. $\mathbf{x}_k \in \mathbb{R}^{T\times 1}$ is the vectorized feature and $\mathbf{h}_k \in \mathbb{R}^{T\times 1}$ is the vectorized filter for the k^{th} channel. λ_1 and λ_2 are the regularization parameters and $*$ is the spatial correlation operator. $\frac{\lambda_1}{2}\sum_{k=1}^{K}\|\mathbf{w}\odot\mathbf{h}_k\|_2^2$ is the spatial regularizer. $\mathbf{w}_r$ is the reference spatial weight and $\mathbf{w}$ is spatial weight to be learned. $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_K]$ is the matrix of filters from all K channels, $\mathbf{P} \in \mathbb{R}^{T\times T}$ represents a binary matrix that applies the correlation operation to the true foreground and background samples directly.

2.2. STRCF

The STRCF [30] formulation can be given by,

$$E(\mathbf{H},\mathbf{w}) = \frac{1}{2}\left\|\mathbf{y} - \sum_{k=1}^{K}\mathbf{x}_k * \mathbf{h}_k\right\|_2^2 + \frac{1}{2}\sum_{k=1}^{K}\|\mathbf{w}\odot\mathbf{h}_k\|_2^2 + \frac{\theta}{2}\left\|\mathbf{h}^{(t)} - \mathbf{h}^{(t-1)}\right\|_2^2, \quad (2)$$

where θ is the regularization parameter, $\mathbf{w}$ are the spatial weights, $\frac{\theta}{2}\left\|\mathbf{h}^{(t)} - \mathbf{h}^{(t-1)}\right\|_2^2$ is the temporal regularization term, $\frac{1}{2}\sum_{k=1}^{K}\|\mathbf{w}\odot\mathbf{h}_k\|_2^2$ is the spatial regularizer. $\mathbf{h}^{(t)}$ and $\mathbf{h}^{(t-1)}$ are the CFs used in the t^{th} and $(t-1)^{th}$ frames respectively.

3. Proposed Approach

Motivated by the above discussions, we propose an Importance Guided Sparse Spatio-Temporal Regularization based Correlation Filter (IGSSTRCF) tracker, formulated as follows,

$$E(\mathbf{h},\mathbf{q},\mathbf{w},\mathbf{B}_w,\mathbf{B}_h) = \frac{1}{2}\left\|\mathbf{y} - \sum_{k=1}^{K}\overbrace{q_k}^{\text{CI}}(\mathbf{x}_k * (\mathbf{P}^T\mathbf{h}_k))\right\|_2^2 + \frac{\lambda_1}{2}\sum_{k=1}^{K}\|\mathbf{w}\odot\mathbf{h}_k\|_2^2 + \underbrace{\frac{\lambda_2}{2}\left\|\mathbf{w}-\mathbf{w}_r-\mathbf{B}_w\right\|_2^2 + \zeta\|\mathbf{B}_w\|_1}_{\text{Sparse Spatial Regularization (SSR)}} + \underbrace{\frac{\theta}{2}\left\|\mathbf{h}^{(t)} - \mathbf{h}^{(t-1)} - \mathbf{B}_h\right\|_2^2 + \eta\|\mathbf{B}_h\|_1}_{\text{Sparse Temporal Regularization (STR)}} + \underbrace{\frac{\beta}{2}\|\mathbf{q}\|_2^2}_{\text{Channel Importance (CI)}}, \quad (3)$$

where q_k is a scalar weight for response channel k, $\mathbf{q} = \{q_1, q_2, ..., q_K\}$, and $\frac{\beta}{2}\|\mathbf{q}\|_2^2$ is a regularization term for the channel weights. $\frac{\lambda_2}{2}\|\mathbf{w}-\mathbf{w}_r-\mathbf{B}_w\|_2^2$ is the spatial regularization component and $\mathbf{B}_w$ is a sparse vector that

learns the spatial changes between the current and reference spatial weights. $\frac{\theta}{2}\left\|\mathbf{h}^{(t)}-\mathbf{h}^{(t-1)}-\mathbf{B}_h\right\|_2^2$ is the temporal regularization term and $\mathbf{B}_h$ is a sparse vector that learns the temporal changes between the current and past filter. $\lambda_1, \lambda_2, \theta, \zeta, \eta$ and β are the regularization parameters. Using Parseval's theorem to express (3) in frequency domain, the equality constrained optimization form is given by,

$$E(\hat{\mathbf{G}}, \mathbf{H}, \mathbf{q}, \mathbf{w}, \mathbf{B}_w, \mathbf{B}_h) = \frac{1}{2}\left\|\hat{\mathbf{y}} - \sum_{k=1}^{K} \hat{\mathbf{x}}_k \odot \hat{\mathbf{g}}_k\right\|_2^2 + \frac{\lambda_1}{2}\sum_{k=1}^{K}\|\mathbf{w}\odot\mathbf{h}_k\|_2^2 + \frac{\lambda_2}{2}\left\|\mathbf{w}-\mathbf{w}_r-\mathbf{B}_w\right\|_2^2 + \zeta\|\mathbf{B}_w\|_1 + \frac{\theta}{2}\sum_{k=1}^{K}\left\|\mathbf{h}_k^{(t)}-\mathbf{h}_k^{(t-1)}-\mathbf{B}_h\right\|_2^2 + \eta\|\mathbf{B}_h\|_1 + \frac{\beta}{2}\|\mathbf{q}\|_2^2,$$
$$s.t.,\ \hat{\mathbf{g}}_k = \sqrt{T}\mathbf{F}\mathbf{P}^T\mathbf{h}_k q_k, \tag{4}$$

where $\hat{}$ denotes the Discrete Fourier Transform (DFT) of a signal, such that $\hat{\mathbf{a}} = \sqrt{T}\mathbf{F}\mathbf{a}$, $\mathbf{a} \in \mathbb{R}^{T\times 1}$, $\mathbf{F}$ is a $T \times T$ orthonormal matrix of complex basis vectors that transforms any T dimensional vectorized signal into the Fourier domain and $\hat{\mathbf{G}} = [\hat{\mathbf{g}}_1, \hat{\mathbf{g}}_2, ..., \hat{\mathbf{g}}_K]$ is an auxiliary variable matrix. The local optimal solution to the model in (4) can be obtained using ADMM [5]. The augmented Lagrangian form of (4) is given by,

$$E(\hat{\mathbf{G}}, \mathbf{H}, \mathbf{q}, \mathbf{w}, \mathbf{B}_w, \mathbf{B}_h) = \frac{1}{2}\left\|\hat{\mathbf{y}} - \sum_{k=1}^{K} \hat{\mathbf{x}}_k \odot \hat{\mathbf{g}}_k\right\|_2^2 + \frac{\lambda_1}{2}\sum_{k=1}^{K}\|\mathbf{w}\odot\mathbf{h}_k\|_2^2 + \frac{\lambda_2}{2}\left\|\mathbf{w}-\mathbf{w}_r-\mathbf{B}_w\right\|_2^2 + \zeta\|\mathbf{B}_w\|_1 + \frac{\theta}{2}\left\|\mathbf{h}^{(t)}-\mathbf{h}^{(t-1)}-\mathbf{B}_h\right\|_2^2 + \eta\|\mathbf{B}_h\|_1 + \frac{\mu}{2}\sum_{k=1}^{K}\left\|\hat{\mathbf{g}_k} - \sqrt{T}\mathbf{F}\mathbf{P}^T\mathbf{h}_k q_k + \frac{\hat{\mathbf{s}}_k}{\mu}\right\|_2^2 + \frac{\beta}{2}\|\mathbf{q}\|_2^2, \tag{5}$$

where μ is the penalty factor and $\hat{\mathbf{S}} = [\hat{\mathbf{s}}_1, \hat{\mathbf{s}}_2, ..., \hat{\mathbf{s}}_K] \in \mathbb{R}^{T\times K}$ is the Fourier transform of the Lagrange multiplier. The above problem can be solved by using ADMM for the following sub-problems:

3.1. Solving for H

Given $\hat{\mathbf{G}}, \mathbf{q}, \mathbf{w}, \mathbf{B}_w, \mathbf{B}_h$ in (5), the optimal solution for $\mathbf{H}^*$ can be obtained by,

$$\mathbf{h}_k^* = \underset{\mathbf{h}_k}{\operatorname{argmin}}\ \frac{\lambda_1}{2}\sum_{k=1}^{K}\|\mathbf{w}\odot\mathbf{h}_k\|_2^2 + \frac{\theta}{2}\sum_{k=1}^{K}\left\|\mathbf{h}_k^{(t)}-\mathbf{h}_k^{(t-1)}-\mathbf{B}_h\right\|_2^2 + \frac{\mu}{2}\sum_{k=1}^{K}\left\|\hat{\mathbf{g}_k} - \sqrt{T}\mathbf{F}\mathbf{P}^T\mathbf{h}_k q_k + \frac{\hat{\mathbf{s}}_k}{\mu}\right\|_2^2. \tag{6}$$

Solving (6), we get,

$$\mathbf{h}_k^* = (\lambda_1\mathbf{W}\mathbf{W}^T + \mu T q_k^2\mathbf{I} + \theta\mathbf{I})^{-1}(T q_k\mathbf{P}(\mu\mathbf{g}_k + \mathbf{s}_k) + \theta\mathbf{h}_k^{(t-1)} + \theta\mathbf{B}_h), \tag{7}$$

where $\mathbf{W} = diag(\mathbf{w}) \in \mathbb{R}^{T\times T}$ and the inverse term can be conveniently obtained by computing the reciprocal of each element. $\mathbf{H}^*$ can be obtained using $\mathbf{H}^* = [\mathbf{h}_1^*, \mathbf{h}_2^*, ..., \mathbf{h}_K^*]$.

3.2. Solving for $\hat{\mathbf{G}}$

Fixing $\mathbf{H}, \mathbf{q}, \mathbf{w}, \mathbf{B}_w, \mathbf{B}_h$ in (5), the optimal $\hat{\mathbf{G}}^*$ can be obtained by solving,

$$\hat{\mathbf{G}}^* = \underset{\hat{\mathbf{G}}}{\operatorname{argmin}}\ \frac{1}{2}\left\|\hat{\mathbf{y}} - \sum_{k=1}^{K} \hat{\mathbf{x}}_k \odot \hat{\mathbf{g}}_k\right\|_2^2 + \frac{\mu}{2}\sum_{k=1}^{K}\left\|\hat{\mathbf{g}_k} - \sqrt{T}\mathbf{F}\mathbf{P}^T\mathbf{h}_k q_k + \frac{\hat{\mathbf{s}}_k}{\mu}\right\|_2^2. \tag{8}$$

However, due to high computational complexity, it is difficult to optimize (8) [6]. Therefore, we proceed pixel-wise for all channels. The reformulated optimization problem in (8) is given by,

$$\mathcal{V}_j^*(\hat{\mathbf{G}}) = \underset{\mathcal{V}_j(\mathbf{G})}{\operatorname{argmin}}\ \frac{1}{2}\left\|\hat{\mathbf{y}}_j - \mathcal{V}_j(\hat{\mathbf{X}})^T\mathcal{V}_j(\hat{\mathbf{G}})\right\|_2^2 + \frac{\mu}{2}\left\|\mathcal{V}_j(\hat{\mathbf{G}}) + \mathcal{V}_j(\hat{\mathbf{M}})\right\|_2^2, \tag{9}$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_K]$ and $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, ..., \hat{\mathbf{x}}_K]$. $\mathcal{V}_j(\hat{\mathbf{X}}) = [\hat{\mathbf{x}}_{1j}, \hat{\mathbf{x}}_{2j}..., \hat{\mathbf{x}}_{Kj}]^T$ is a $K \times 1$ vector, picking the j^{th} element from each channel of $\hat{\mathbf{X}}$, i.e., $\mathcal{V}_1(\hat{\mathbf{X}}) = [\hat{\mathbf{x}}_{11}, \hat{\mathbf{x}}_{21}, ..., \hat{\mathbf{x}}_{K1}]^T$ and $\mathcal{V}_j(\hat{\mathbf{G}}) = [\hat{\mathbf{g}}_{1j}, \hat{\mathbf{g}}_{2j}, ..., \hat{\mathbf{g}}_{Kj}]^T$. Similarly, we form, $\mathcal{V}_j(\hat{\mathbf{M}}) = \mathcal{V}_j\left(\frac{\hat{\mathbf{S}}}{\mu}\right) - \mathcal{V}_j(\sqrt{T}\mathbf{F}\mathbf{P}^T\mathbf{H}\mathbf{q})$, where $\mathcal{V}_j\left(\frac{\hat{\mathbf{S}}}{\mu}\right) = [\frac{\hat{\mathbf{s}}_{1j}}{\mu}, \frac{\hat{\mathbf{s}}_{2j}}{\mu}, ..., \frac{\hat{\mathbf{s}}_{kj}}{\mu}]^T$. Solving (9), we get,

$$\mathcal{V}_j^*(\hat{\mathbf{G}}) = (\mu\mathbf{I} + \mathcal{V}_j(\hat{\mathbf{X}})\mathcal{V}_j(\hat{\mathbf{X}})^T)^{-1}(\hat{\mathbf{y}}_j\mathcal{V}_j(\hat{\mathbf{X}}) - \mu\mathcal{V}_j\left(\frac{\hat{\mathbf{S}}}{\mu}\right) + \mu\mathcal{V}_j(\sqrt{T}\mathbf{F}\mathbf{P}^T\mathbf{H}\mathbf{q})). \tag{10}$$

Equation (10) can be efficiently computed using the Sherman-Morrison formula [6] as follows.

$$\mathcal{V}_j^*(\hat{\mathbf{G}}) = \frac{1}{\mu}\left(\mathbf{I} - \frac{\mathcal{V}_j(\hat{\mathbf{X}})\mathcal{V}_j(\hat{\mathbf{X}})^T}{\mu + \mathcal{V}_j(\hat{\mathbf{X}})^T\mathcal{V}_j(\hat{\mathbf{X}})}\right)(\hat{\mathbf{y}}_j\mathcal{V}_j(\hat{\mathbf{X}}) - \mu\mathcal{V}_j\left(\frac{\hat{\mathbf{S}}}{\mu}\right) + \mu\mathcal{V}_j(\sqrt{T}\mathbf{F}\mathbf{P}^T\mathbf{H}\mathbf{q})). \tag{11}$$

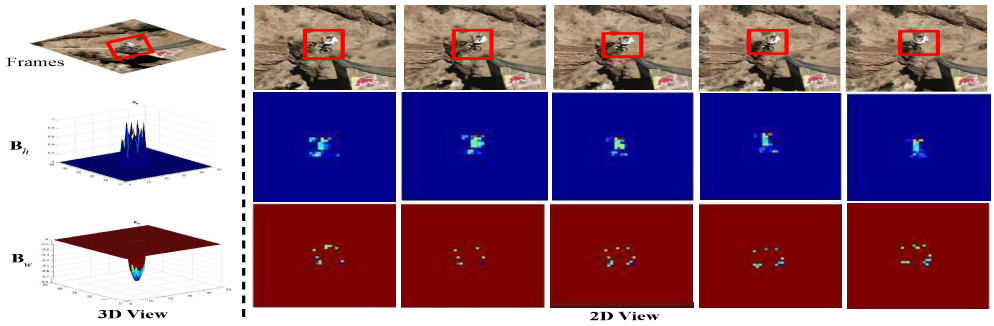

Figure 2: Pictorial representation of $\mathbf{B}_w$ and $\mathbf{B}_h$ learned for consecutive ADMM updates from the sequence *Mountainbike* (Frame# 46, 48, 50, 52 and 54) of the TC128 dataset [36]. $\mathbf{B}_h$ is normalized between [0 1] for display purpose

3.3. Solving for q

If $\hat{\mathbf{G}}, \mathbf{H}, \mathbf{w}, \mathbf{B}_w, \mathbf{B}_h$ are fixed in (5), q_k can be computed as follows,

$$q_k^* = \underset{q_k}{\operatorname{argmin}} \frac{\mu}{2} \sum_{k=1}^{K} \left\| \hat{\mathbf{g}_k} - \sqrt{T}\mathbf{F}\mathbf{P}^T\mathbf{h}_k q_k + \frac{\hat{\mathbf{s}}_k}{\mu} \right\|_2^2 + \frac{\beta}{2}\|\mathbf{q}\|_2^2. \tag{12}$$

Solving (12), we get,

$$q_k^* = \frac{\mu\sqrt{T}{\mathbf{h}_k}^T\mathbf{P}\mathbf{g}_k + T{\mathbf{h}_k}^T\mathbf{P}\mathbf{s_k}}{\mu\sqrt{T}{\mathbf{h}_k}^T\mathbf{P}\mathbf{P}^T\mathbf{h}_k + \beta}. \tag{13}$$

3.4. Solving for w

Fixing $\hat{\mathbf{G}}, \mathbf{H}, \mathbf{q}, \mathbf{B}_w$ and $\mathbf{B}_h$ in (5), the closed-form solution for $\mathbf{w}$ is given by,

$$\mathbf{w}^* = \frac{\lambda_1}{2} \sum_{k=1}^{K} \|\mathbf{N}_k\mathbf{w}\|_2^2 + \frac{\lambda_2}{2} \left\|\mathbf{w} - \mathbf{w}_r - \mathbf{B}_w\right\|_2^2, \tag{14}$$

$$= \left(\lambda_1 \sum_{k=1}^{K} \mathbf{N}_k^T\mathbf{N}_k + \lambda_2\mathbf{I}\right)^{-1} \lambda_2(\mathbf{w}_r + \mathbf{B}_w), \tag{15}$$

$$= \frac{\lambda_2(\mathbf{w}_r + \mathbf{B}_w)}{\lambda_1 \sum_{k=1}^{K} \mathbf{h}_k \bigodot \mathbf{h}_k + \lambda_2}, \tag{16}$$

where $\mathbf{N}_k = \text{diag}(\mathbf{h}_k) \in \mathbb{R}^{T\times T}$.

3.5. Solving for $\mathbf{B}_w$

In ASRCF [6], the authors attempt to learn the spatial weights $\mathbf{w}$ by incorporating the term $\|\mathbf{w} - \mathbf{w}_r\|_2^2$ in the CF formulation. This term attempts to make $\mathbf{w}$ similar to a reference weight $\mathbf{w}_r$. However, due to constant changes in the target appearance and background, $\mathbf{w}$ will not be exactly similar to $\mathbf{w}_r$. To capture the minor variation between $\mathbf{w}_r$ and $\mathbf{w}$, we propose to learn a sparse difference vector $\mathbf{B}_w$. Given $\hat{\mathbf{G}}, \mathbf{H}, \mathbf{q}, \mathbf{w}$ and $\mathbf{B}_h$, the solution for $\mathbf{B}_w$ can be obtained using,

$$\mathbf{B}_w^* = \underset{\mathbf{B}_w}{\operatorname{argmin}} \frac{\lambda_2}{2} \left\|\mathbf{w} - \mathbf{w}_r - \mathbf{B}_w\right\|_2^2 + \zeta\|\mathbf{B}_w\|_1. \tag{17}$$

The solution for (17) can obtained using the Iterative Soft Thresholding algorithm (IST) [2] by

$$\mathbf{B}_w^* = \mathcal{S}_{\frac{\zeta}{\lambda_2}}(\mathbf{w} - \mathbf{w}_r). \tag{18}$$

Here, $\mathcal{S}_\alpha(\mathbf{z}_i) = sign(\mathbf{z}_i)\max(0, |\mathbf{z}_i| - \alpha)$, is the soft-thresholding operator for a vector $\mathbf{z}$. Figure 2 shows a pictorial representation of $\mathbf{B}_w$ learned for consecutive ADMM updates from the sequence *Mountainbike* in the TC128 dataset [36]. It can be seen that $\mathbf{B}_w$ has low penalty weights for the pixels corresponding to the target. Thus, when $\mathbf{B}_w$ is added to the reference $\mathbf{w}_r$, the resultant $\mathbf{w}$ has high penalty weights near the boundary region and low penalty weights near the target.

3.6. Solving for $\mathbf{B}_h$

In STRCF [30], the temporal regularization term, $\left\|\mathbf{h}_k^{(t)} - \mathbf{h}_k^{(t-1)}\right\|$, is used to learn a filter $\mathbf{h}_k^{(t)}$ similar to $\mathbf{h}_k^{(t-1)}$, where t is the frame index and k represents the feature channel index. However, since the target appearance changes every frame, the filter $\mathbf{h}_k^{(t)}$ will be similar to $\mathbf{h}_k^{(t-1)}$, but should also adapt to the variations in the object appearance between consecutive frames. To capture the variation between $\mathbf{h}_k^{(t)}$ and $\mathbf{h}_k^{(t-1)}$, we learn a sparse difference vector $\mathbf{B}_h$. Given $\hat{\mathbf{G}}, \mathbf{H}, \mathbf{q}, \mathbf{w}$ and $\mathbf{B}_w$, the solution for $\mathbf{B}_h$ can be obtained using,

$$\mathbf{B}_h^* = \underset{\mathbf{B}_h}{\operatorname{argmin}} \frac{\theta}{2} \left\|\mathbf{h}_k^{(t)} - \mathbf{h}_k^{(t-1)} - \mathbf{B}_h\right\|_2^2 + \eta\|\mathbf{B}_h\|_1. \tag{19}$$

The solution for (19) can obtained using IST [2] by

$$\mathbf{B}_h^* = \mathcal{S}_{\frac{\eta}{\theta}}(\mathbf{h}_k^{(t)} - \mathbf{h}_k^{(t-1)}). \tag{20}$$

Here, $\mathcal{S}$ is the soft-thresholding operator. Figure 2 shows a pictorial representation of $\mathbf{B}_h$ learned for consecutive

ADMM updates. It can be seen that $\mathbf{B}_h$ is non-zero for the pixels corresponding to the target and zero for the background. Thus, when $\mathbf{B}_h$ is added to $\mathbf{h}_{(t-1)}$, the resultant $\mathbf{h}_{(t)}$ contains refined filter coefficients in the target region.

3.7. Lagrangian Multiplier Update

The Lagrangian multipliers are updated using,

$$\mu^{(t+1)} = min(\mu_{max}, \nu\mu^{(t)}), \tag{21}$$

$$\hat{\mathbf{S}}^{(t+1)} = \hat{\mathbf{S}}^{(t)} + \mu^{(t+1)}(\hat{\mathbf{G}}^{(t+1)} - \hat{\mathbf{H}}^{(t+1)}), \tag{22}$$

$$\lambda_2^{(t+1)} = \rho\lambda_2^{(t)}, \tag{23}$$

$$\theta^{(t+1)} = \rho\theta^{(t)}, \tag{24}$$

where $\rho > 1$, $\hat{\mathbf{H}}^{(t+1)}$ and $\hat{\mathbf{G}}^{(t+1)}$ are the current solutions to $\hat{\mathbf{H}}$ and $\hat{\mathbf{G}}$ respectively, and $\hat{\mathbf{S}}^{(t)}$ is the Fourier transform of the Lagrangian variable in the previous state. Thus, the optimal filter $\mathbf{H}^*$, feature channel weight q_k^*, spatial weight $\mathbf{w}^*$ and the sparse difference components, $\mathbf{B}_w^*$ and $\mathbf{B}_h^*$, can be obtained by iteratively solving for $\mathbf{H}$, $\mathbf{G}$, q_k, $\mathbf{w}$, $\mathbf{B}_w$ and $\mathbf{B}_h$ followed by the Lagrangian update, until convergence.

3.8. Target Localization

The target location is determined using,

$$\hat{\mathbf{r}} = \sum_{k=1}^{K} \hat{\mathbf{x}}_\mathbf{k} \odot \hat{\mathbf{g}}_\mathbf{k}, \tag{25}$$

where $\hat{\mathbf{r}}$ is the response map in the Fourier domain. The location at which $\hat{\mathbf{r}}$ shows the maximum value is used to estimate the target location. For target scale estimation, we follow the same strategy as [6].

3.9. Model Update

In order to adjust to target appearance variations, we use an online adaptive template scheme [3, 4, 52] to update the template model,

$$\hat{\mathbf{X}}_{model}^{(t)} = (1-\omega)\hat{\mathbf{X}}_{model}^{(t-1)} + \omega\hat{\mathbf{X}}^{(t)}, \tag{26}$$

where ω is the online learning rate, $\hat{\mathbf{X}}^{(t)}$ is the current observation, $\hat{\mathbf{X}}_{model}^{(t-1)}$ is the old template model and $\hat{\mathbf{X}}_{model}^{(t)}$ is the updated template model. To introduce a reasonable prior for adaptive spatial regularization, the reference spatial weights are updated using $\mathbf{w}_r \leftarrow \mathbf{w}^*$. In the first frame, $\mathbf{w}_r$ is initialized with a negative Gaussian shape [6, 11]. The above update schemes ensure our model is adaptive to target appearance variations during tracking.

4. Experiments

This section provides implementation details and presents the performance analysis of the proposed tracker on four benchmark datasets: TC128 [36], UAV123 [40], VOT-2019 [27] and VOT-2017 [26], in comparison to state-of-the-art trackers. The section ends with an extensive ablation study examining the contribution of each regularization component of the proposed IGSSTRCF tracker.

4.1. Implementation Details

All proposed formulations are implemented using MATLAB2019a with the MatConvNet toolbox. We use an ensemble of features extracted from *Norm1* of VGG-M, *Conv4-3* of VGG-16 [44] and HOG features to represent and localize the target. In Equation (3), the parameters θ and λ_2 are 1.2, λ_1 and β are 0.01, and η and ζ are 0.001. ρ in (23) is 1.5 and the learning rate, ω, in (26) is 0.0186. ν in (21) is 10 and μ_{max} is 10^4. The initial value of the ADMM penality factor, μ, is set to 1. The ADMM is updated every 2 frames. The value of each parameter is selected empirically.

4.2. Performance Analysis

We present an extensive evaluation of IGSSTRCF on four challenging tracking benchmarks with a tracking speed of 8 fps. For TC128 [36] and UAV123 [40] datasets, we report a one-pass evaluation with distance precision and overlap success plots. For the VOT datasets [26, 27], we use the benchmark protocol to evaluate the tracker in terms of Expected Average Overlap (EAO), Accuracy (A) and Robustness (R) for the baseline experiments, and overlap Area-Under-the-Curve (AUC) for the unsupervised experiments [26]. For VOT datasets, the expected overlap curves, scores, unsupervised overlap AUC, and A-R analysis for individual challenges are included in the supplementary material.

4.2.1 Evaluation on TC128 Dataset

The proposed tracker is evaluated on TC128 [36]. Figure 3 (a) and (b) shows the success and precision plots comparing the proposed IGSSTRCF tracker with recent trackers: GFSDCF [49], ECO [7], ASRCF [6], IBCCF [31], AutoTrack [32], CCOT [13], LDES [35], ARCF [21], STRCF [30], BACF [23], KAOT [33], CF2 [38], HDT [41], DRCF [15], SITUP [39], SAMF [34], MEEM [51], EnKCF [46], DSST [9], KCF [20], ASLA [22], L1APG [1], DFT [43] and IVT [42]. The proposed IGSSTRCF outperforms all the compared trackers in terms of overlap success and distance precision, except for GFSDCF [49] and ECO [7].

4.2.2 Evaluation on UAV123 Dataset

The proposed tracker is evaluated on UAV123 dataset [40]. Figure 3 (c) and (d) shows the success and precision plots comparing the proposed IGSSTRCF tracker with recent trackers: GFSDCF [49], ECO [7], CCOT [13], DRCF [15], KAOT [33], AutoTrack [32], LDES [35], STRCF [30] ARCF [21], ASRCF [6], SITUP [39] and EnKCF [46]. The

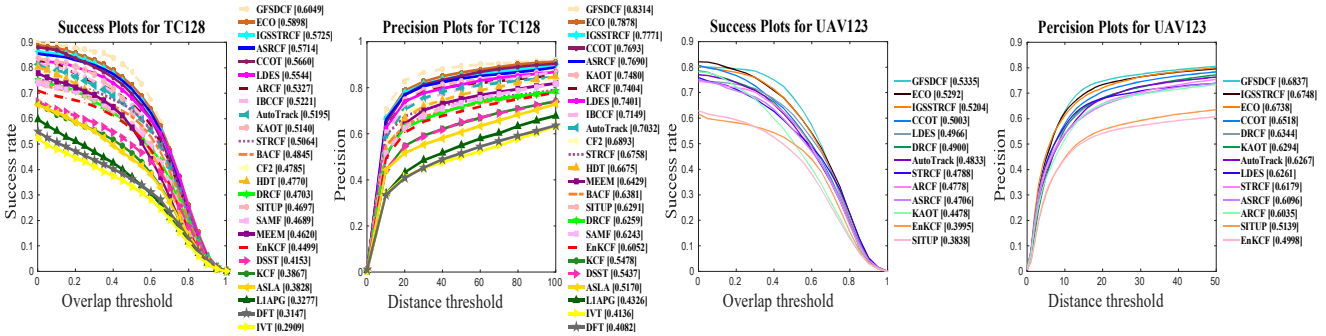

Figure 3: Success and Precision plots for TC128 [36] ((a) and (b)) and UAV123 dataset [40] ((c) and (d)), with trackers arranged in descending order of their performance. The legend of the precision plots contains the scores at a threshold of 20 pixels and the legend of the success plots contains Area-Under-the-Curve scores for each tracker

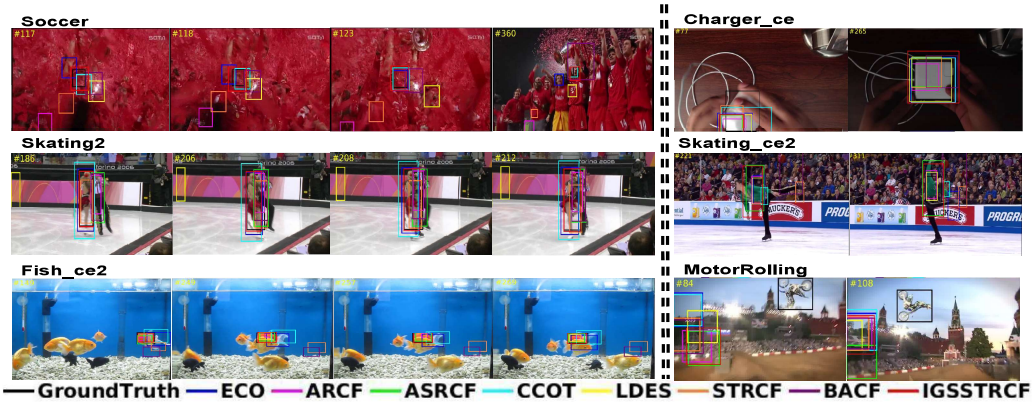

Figure 4: Intermediate frames showing examples of successfully tracked frames (left) and failure cases (right) in different sequences from the TC128 dataset [36]

proposed IGSSTRCF tracker is second best in terms of precision and third best in terms of success.

4.2.3 Evaluation on VOT-2019 Dataset

The proposed IGSSTRCF tracker is evaluated using the VOT toolkit on VOT-2019 [27]. A comparison is shown with the recent state of the art trackers: ASRCF [6], STRCF [30], LDES [35], ARCF [21], BACF [23], CISRDCF [27], ANT [27], LGT [27], FoT [27], MIL [27], KCF [27], Struck [27], IVT [27] and, L1APG [27]. Table 1 shows the accuracy, robustness and EAO for the baseline experiments [26]. It is observed that IGSSTRCF performs best in terms of robustness, and second best in terms of EAO and accuracy. For the unsupervised experiments [26], Table 1 shows the overlap AUC. It is observed that IGSSTRCF performs best in the overlap criterion.

4.2.4 Evaluation on VOT-2017 Dataset

The proposed tracker is evaluated using the VOT toolkit on VOT-2017 [26]. A comparison is shown with recent trackers: ASRCF [6], STRCF [30], LDES [35], ARCF [21], BACF [23], ECO [8], CCOT [13], SRDCF [11], ANT [26], BST [26], CGS [26], ATLAS [26], and GMD [26]. Table 2 shows the accuracy, robustness and EAO for baseline experiments [26]. It is observed that IGSSTRCF is third best in terms of robustness and EAO. For the unsupervised experiments, Table 2 shows the overlap AUC. It is observed that the proposed tracker is third best in the overlap criterion.

4.2.5 Qualitative Evaluation

To demonstrate the performance qualitatively, we present examples of success and failure for some tracking sequences of TC128 dataset [36]. The proposed IGSSTRCF tracker is compared with ECO [8], ARCF [21], ASRCF [6], CCOT [13], LDES [35], STRCF [30] and BACF [23]. Figure 4 (left) shows frames from the sequences *soccer*, *Skating2*, and *Fish_ce2*, where the IGSSTRCF tracks successfully during background clutter, similar object presence, and multi-object presence, while most other trackers fail. Figure 4 (right) shows frames from the sequences *Charger_ce*, *skating_ce2*, and *MotorRolling*, where most of the trackers, including IGSSTRCF, fail during scale change,

Table 1: VOT toolkit report for VOT-2019 showing Accuracy (A), Robustness (R) and Expected Average Overlap (EAO) for the baseline experiment and Overlap AUC for the unsupervised experiment. The top three trackers are shown in **red**, **blue** and **green**

	Baseline			Unsupervised
	EAO	A	R	AUC
IGSSTRCF	**0.1559**	**0.4730**	**39.0094**	**0.3286**
ASRCF [6]	0.1451	0.4652	**44.5818**	**0.3230**
STRCF [30]	0.1140	0.4520	70.02	**0.2980**
LDES [35]	**0.1747**	**0.4882**	50.2721	0.2940
ARCF [21]	0.1351	**0.4669**	52.7181	0.2690
BACF [23]	0.1162	0.4476	65.7094	0.1959
CISRDCF [27]	**0.1533**	0.4147	**48.9861**	0.2417
ANT [27]	0.1509	0.4518	53.0936	0.2390
LGT [27]	0.1308	0.3960	54.8683	0.2062
FoT [27]	0.1290	0.3621	70.4328	0.1354
MIL [27]	0.1179	0.3847	73.6540	0.1664
KCF [27]	0.1103	0.4348	73.0953	0.2059
Struck [27]	0.0944	0.4103	96.3228	0.1743
IVT [27]	0.0869	0.3811	117.7786	0.1095
L1APG [27]	0.0774	0.3901	147.7737	0.1224

object deformation, and in-plane-rotation. Further examples using sequences from the VOT-2019 [27] and TC128 dataset [36] are included in the supplementary material.

4.2.6 Ablation Study

To demonstrate the contribution of each regularization component of the proposed IGSSTRCF, we remove the regularization terms from the IGSSTRCF formulation one at a time, and evaluate the performance. Table 3 shows a comparison of the complete IGSSTRCF formulation with formulations without the regularizations. In Table 3, column 1, "IGSSTRCF - *regularization term*" denotes the IGSSTRCF tracker without the stated *regularization term*. The comparison is shown on VOT-2019 [27] and VOT-2017 [26] in terms of overlap score for baseline experiments, and on TC128 [36] in terms of success plot AUC and precision score at a threshold of 20 pixels. It is observed that the proposed IGSSTRCF works best with the SSR, STR and CI terms all included. An additional ablation study that demonstrates the impact of SSR, STR and CI terms on the baselines [6, 23, 30] is included in the supplementary material.

5. Conclusions

In this work, we propose a novel importance guided sparse spatio-temporal regularization based CF tracker. The sparse spatial regularization learns the spatial weights by modelling the sparse difference between the current spatial weights and the reference spatial weights. The learned spatial weights are used to penalize the filter coefficients near the boundary region to prevent boundary effects. The sparse temporal regularization models the sparse difference between the current filter and the past reference filter. The sparse difference term helps filter out irrelevant information from the previous filter, while learning the current filter. This prevents irrelevant appearance information from persisting through further tracking steps. We also propose to learn the adaptive importance weights for each feature channel during training. This helps in suppressing the contribution of adverse feature channels and enhancing the contribution of useful feature channels. As a result of the proposed regularizations, a more discriminative and robust CF is trained, that achieves efficient object tracking during multiple challenges. We use ADMM to efficiently obtain an optimal solution for the proposed formulation. The positive effects of the proposed formulation are demonstrated on the TC128 [36], UAV123 [40], VOT-2019 [27] and VOT-2017 [26] datasets. A comparative analysis with recent top ranked trackers reveals that the proposed approach outperforms many state-of-the-art trackers.

Table 2: VOT toolkit report for VOT-2017 showing Accuracy (A), Robustness (R) and Expected Average Overlap (EAO) for the baseline experiment and Overlap AUC for the unsupervised experiment. The top three trackers are shown in **red**, **blue** and **green**

	Baseline			Unsupervised
	A	R	EAO	AUC
IGSSTRCF	0.4761	**26.2841**	**0.2040**	**0.3539**
ASRCF [6]	0.4654	30.9708	0.1851	0.3411
STRCF [30]	0.4510	61.3300	0.1180	0.3000
LDES [35]	**0.4929**	39.6484	0.1875	0.3237
ARCF [21]	0.4615	41.4174	0.1547	0.2824
BACF [23]	0.4476	55.7769	0.1235	0.2083
ECO [7]	0.4762	**17.6628**	**0.2809**	**0.4025**
CCOT [13]	**0.4851**	**20.4138**	**0.2674**	**0.3909**
SRDCF [11]	0.4767	64.1136	0.1179	0.2445
ANT [26]	0.4540	40.1593	0.1676	0.2770
BST [26]	0.2627	55.5033	0.1150	0.1458
CGS [26]	**0.4979**	53.3758	0.1406	0.3386
ATLAS [26]	0.4835	37.4268	0.1969	0.3431
GMD [26]	0.4422	54.7325	0.1295	0.2492

Table 3: Ablation analysis showing contribution of each regularization component of the proposed IGSSTRCF tracker in terms of Overlap Score (OP), Success (S) and Precision (P). The best performance is shown in **red**

	VOT-2019	VOT-2017	TC128	
	OS	OS	S	P
IGSSTRCF	**0.155**	**0.204**	**57.25**	**77.71**
IGSSTRCF - SSR	0.152	0.195	56.84	77.35
IGSSTRCF - CI	0.127	0.150	55.85	76.04
IGSSTRCF - STR	0.151	0.178	55.78	76.78
IGSSTRCF - SSR - CI	0.152	0.196	56.35	76.51
IGSSTRCF - STR - CI	0.153	0.193	56.68	77.30
IGSSTRCF - STR - SSR	0.149	0.171	54.25	72.74

Acknowledgement

This research is supported by the Australian Research Council (ARC) Linkage Grant [Grant Number LP140100221] and Early Career Research Award, Department of Science and Technology, Government of India [Grant Number ECR/2018/002449].

References

[1] Chenglong Bao, Yi Wu, Haibin Ling, and Hui Ji. Real time robust l1 tracker using accelerated proximal gradient approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1830–1837, 2012.

[2] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[3] Luca Bertinetto, Jack Valmadre, Stuart Golodetz, Ondrej Miksik, and Philip HS Torr. Staple: Complementary learners for real-time tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1401–1409, 2016.

[4] David S Bolme, J Ross Beveridge, Bruce A Draper, and Yui Man Lui. Visual object tracking using adaptive correlation filters. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2544–2550, 2010.

[5] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[6] Kenan Dai, Dong Wang, Huchuan Lu, Chong Sun, and Jianhua Li. Visual tracking via adaptive spatially-regularized correlation filters. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4670–4679, 2019.

[7] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Eco: Efficient convolution operators for tracking. *arXiv preprint arXiv:1611.09224*, 2016.

[8] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Eco: Efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6638–6646, 2017.

[9] Martin Danelljan, Gustav Häger, Fahad Khan, and Michael Felsberg. Accurate scale estimation for robust visual tracking. In *British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press, 2014.

[10] Martin Danelljan, Gustav Hager, Fahad Shahbaz Khan, and Michael Felsberg. Convolutional features for correlation filter based visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 58–66, 2015.

[11] Martin Danelljan, Gustav Hager, Fahad Shahbaz Khan, and Michael Felsberg. Learning spatially regularized correlation filters for visual tracking. In *Proceedings of the IEEE international conference on computer vision*, pages 4310–4318, 2015.

[12] Martin Danelljan, Gustav Hager, Fahad Shahbaz Khan, and Michael Felsberg. Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1430–1438, 2016.

[13] Martin Danelljan, Andreas Robinson, Fahad Shahbaz Khan, and Michael Felsberg. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In *European Conference on Computer Vision*, pages 472–488. Springer, 2016.

[14] Martin Danelljan, Fahad Shahbaz Khan, Michael Felsberg, and Joost Van de Weijer. Adaptive color attributes for real-time visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1090–1097, 2014.

[15] Changhong Fu, Juntao Xu, Fuling Lin, Fuyu Guo, Tingcong Liu, and Zhijun Zhang. Object saliency-aware dual regularized correlation filter for real-time aerial tracking. *IEEE Transactions on Geoscience and Remote Sensing*, 2020.

[16] Shiming Ge, Zhao Luo, Chunhui Zhang, Yingying Hua, and Dacheng Tao. Distilling channels for efficient deep tracking. *IEEE Transactions on Image Processing*, 2019.

[17] Jie Guo, Long Zhuang, and Ping Zheng. Smooth incremental learning of correlation filters for visual tracking. *IEEE Signal Processing Letters*, 27:336–340, 2020.

[18] Zhenjun Han, Pan Wang, and Qixiang Ye. Adaptive discriminative deep correlation filter for visual object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 2018.

[19] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In *European conference on computer vision*, pages 702–715. Springer, 2012.

[20] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE transactions on pattern analysis and machine intelligence*, 37(3):583–596, 2014.

[21] Ziyuan Huang, Changhong Fu, Yiming Li, Fuling Lin, and Peng Lu. Learning aberrance repressed correlation filters for real-time uav tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2891–2900, 2019.

[22] Xu Jia, Huchuan Lu, and Ming-Hsuan Yang. Visual tracking via adaptive structural local sparse appearance model. In *IEEE Conference on computer vision and pattern recognition*, pages 1822–1829, 2012.

[23] Hamed Kiani Galoogahi, Ashton Fagg, and Simon Lucey. Learning background-aware correlation filters for visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1135–1143, 2017.

[24] Hamed Kiani Galoogahi, Terence Sim, and Simon Lucey. Multi-channel correlation filters. In *Proceedings of the IEEE international conference on computer vision*, pages 3072–3079, 2013.

[25] Hamed Kiani Galoogahi, Terence Sim, and Simon Lucey. Correlation filters with limited boundaries. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4630–4638, 2015.

[26] Matej Kristan, Ales Leonardis, Jiri Matas, Michael Felsberg, et al. The visual object tracking vot2017 challenge results. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1949–1972, 2017.

[27] Matej Kristan, Jiri Matas, Ales Leonardis, Michael Felsberg, et al. The seventh visual object tracking vot2019 challenge

results. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[28] BVK Vijaya Kumar, Abhijit Mahalanobis, and Richard D Juday. *Correlation pattern recognition*. Cambridge University Press, 2005.

[29] Aishi Li, Ming Yang, and Wanqi Yang. Feature integration with adaptive importance maps for visual tracking. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 779–785. AAAI Press, 2018.

[30] Feng Li, Cheng Tian, Wangmeng Zuo, Lei Zhang, and Ming-Hsuan Yang. Learning spatial-temporal regularized correlation filters for visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4904–4913, 2018.

[31] Feng Li, Yingjie Yao, Peihua Li, David Zhang, Wangmeng Zuo, and Ming-Hsuan Yang. Integrating boundary and center correlation filters for visual tracking with aspect ratio variation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2001–2009, 2017.

[32] Yiming Li, Changhong Fu, Fangqiang Ding, Ziyuan Huang, and Geng Lu. Autotrack: Towards high-performance visual tracking for uav with automatic spatio-temporal regularization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[33] Yiming Li, Changhong Fu, Ziyuan Huang, Yinqiang Zhang, and Jia Pan. Keyfilter-aware real-time uav object tracking. *arXiv preprint arXiv:2003.05218*, 2020.

[34] Yang Li and Jianke Zhu. A scale adaptive kernel correlation filter tracker with feature integration. In *European conference on computer vision*, pages 254–265. Springer, 2014.

[35] Yang Li, Jianke Zhu, Steven CH Hoi, Wenjie Song, Zhefeng Wang, and Hantang Liu. Robust estimation of similarity transformation for visual object tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8666–8673, 2019.

[36] Pengpeng Liang, Erik Blasch, and Haibin Ling. Encoding color information for visual tracking: Algorithms and benchmark. *IEEE Transactions on Image Processing*, 24(12):5630–5644, 2015.

[37] Xiankai Lu, Chao Ma, Bingbing Ni, and Xiaokang Yang. Adaptive region proposal with channel regularization for robust object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2019.

[38] Chao Ma, Jia-Bin Huang, Xiaokang Yang, and Ming-Hsuan Yang. Hierarchical convolutional features for visual tracking. In *Proceedings of the IEEE international conference on computer vision*, pages 3074–3082, 2015.

[39] Haoyi Ma, Scott T Acton, and Zongli Lin. Situp: Scale invariant tracking using average peak-to-correlation energy. *IEEE Transactions on Image Processing*, 29:3546–3557, 2020.

[40] Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *European Conference on Computer Vision*, pages 445–461. Springer, 2016.

[41] Yuankai Qi, Shengping Zhang, Lei Qin, Hongxun Yao, Qingming Huang, Jongwoo Lim, and Ming-Hsuan Yang. Hedged deep tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4303–4311, 2016.

[42] David A Ross, Jongwoo Lim, Ruei-Sung Lin, and Ming-Hsuan Yang. Incremental learning for robust visual tracking. *International journal of computer vision*, 77(1-3):125–141, 2008.

[43] Laura Sevilla-Lara and Erik Learned-Miller. Distribution fields for tracking. In *IEEE Conference on computer vision and pattern recognition*, pages 1910–1917, 2012.

[44] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[45] Chong Sun, Dong Wang, Huchuan Lu, and Ming-Hsuan Yang. Correlation tracking via joint discrimination and reliability learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 489–497, 2018.

[46] Burak Uzkent and YoungWoo Seo. Enkcf: Ensemble of kernelized correlation filters for high-speed object tracking. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1133–1141, 2018.

[47] Shuo Wang, Dong Wang, and Huchuan Lu. Tracking with static and dynamic structured correlation filters. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):2861–2869, 2017.

[48] Wuwei Wang, Ke Zhang, Meibo Lv, and Jingyu Wang. Hierarchical spatiotemporal context-aware correlation filters for visual tracking. *IEEE Transactions on Cybernetics*, 2020.

[49] Tianyang Xu, Zhen-Hua Feng, Xiao-Jun Wu, and Josef Kittler. Joint group feature selection and discriminative filter learning for robust visual object tracking. *arXiv preprint arXiv:1907.13242*, 2019.

[50] Tianyang Xu, Zhen-Hua Feng, Xiao-Jun Wu, and Josef Kittler. Learning low-rank and sparse discriminative correlation filters for coarse-to-fine visual object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[51] Jianming Zhang, Shugao Ma, and Stan Sclaroff. Meem: robust tracking via multiple experts using entropy minimization. In *European Conference on Computer Vision*, pages 188–203. Springer, 2014.

[52] Le Zhang and Ponnuthurai Nagaratnam Suganthan. Robust visual tracking via co-trained kernelized correlation filters. *Pattern Recognition*, 69:82 – 93, 2017.

[53] Tao Zhou, Harish Bhaskar, Fanghui Liu, and Jie Yang. Graph regularized and locality-constrained coding for robust visual tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(10):2153–2164, 2016.

[54] Xue-Feng Zhu, Xiao-Jun Wu, Tianyang Xu, Zhen-Hua Feng, and Josef Kittler. Complementary discriminative correlation filters based on collaborative representation for visual object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.