TCTrack: Temporal Contexts for Aerial Tracking

Ziang Cao[1], Ziyuan Huang[2], Liang Pan[3], Shiwei Zhang[4], Ziwei Liu[3], Changhong Fu[1,*]

[1]Tongji University [2]National University of Singapore [3]S-Lab, Nanyang Technological University
[4]DAMO Academy, Alibaba Group

caoang233@gmail.com, ziyuan.huang@u.nus.edu, {liang.pan, ziwei.liu}@ntu.edu.sg

zhangjin.zsw@alibaba-inc.com changhongfu@tongji.edu.cn

Abstract

*Temporal contexts among consecutive frames are far from being fully utilized in existing visual trackers. In this work, we present **TCTrack**[1], a comprehensive framework to fully exploit temporal contexts for aerial tracking. The temporal contexts are incorporated at **two levels**: the extraction of **features** and the refinement of **similarity maps**. Specifically, for feature extraction, an online temporally adaptive convolution is proposed to enhance the spatial features using temporal information, which is achieved by dynamically calibrating the convolution weights according to the previous frames. For similarity map refinement, we propose an adaptive temporal transformer, which first effectively encodes temporal knowledge in a memory-efficient way, before the temporal knowledge is decoded for accurate adjustment of the similarity map. TCTrack is effective and efficient: evaluation on four aerial tracking benchmarks shows its impressive performance; real-world UAV tests show its high speed of over 27 FPS on NVIDIA Jetson AGX Xavier.*

1. Introduction

Visual tracking is one of the most fundamental tasks in computer vision. Owing to the superior mobility of unmanned aerial vehicles (UAVs), tracking-based applications are experiencing rapid developments, *e.g.*, motion object analysis [57], geographical survey [61], and visual localization [47]. Nevertheless, aerial tracking still faces two difficulties: 1) aerial conditions inevitably introduce special challenges including motion blur, camera motion, occlusion, *etc*; 2) the limited power of aerial platforms restricts the computational resource, impeding the deployment of time-consuming state-of-the-art methods [6]. Hence, an ideal tracker for aerial tracking must be robust and efficient.

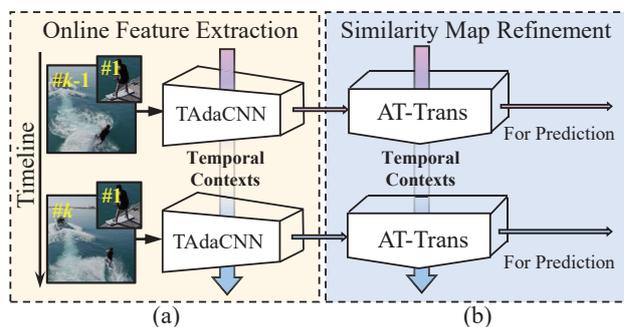Most existing trackers adopt the standard tracking-by-

Figure 1. Overview of our framework namely TCTrack. It exploits temporal information at two levels: (a) *the extraction of features* by the temporally adaptive convolutional neural networks (TAdaCNN) and (b) *the refinement of similarity maps* by the adaptive temporal transformer (AT-Trans).

detection framework and perform detection for each frame independently. Among these trackers, discriminative correlation filter (DCF)-based methods are widely applied on aerial platforms because of their high efficiency and low resource requirements originated from the operations in the Fourier domain [16, 31, 38]. However, these trackers struggle when there are fast motions and severe appearance variations. Recently, the Siamese-based network has emerged as a strong framework for accurate and robust tracking [2, 4, 11, 41, 42]. Its efficiency is also optimized in [7, 21, 22] for the real-time deployment of Siamese-based trackers on aerial platforms.

However, the strong correlations inherently existing among consecutive frames, *i.e.,* the temporal information, are neglected by these frameworks, which makes it difficult for these approaches to perceive the motion information of the target objects. Therefore, those trackers are more likely to fail when the target undergoes severe appearance change caused by different complex conditions such as large motions and occlusions. This has sparked the recent research into how to make use of temporal information for visual tracking. For DCF-based approaches, the variation in the response maps along the temporal dimension is penalized [33, 47], which guides the current response map

by previous ones. In Siamese-based networks, which is the focus of this work, temporal information is introduced in most works through dynamic templates, which integrates historical object appearance in the current template through concatenation [72], weighted sum [74], graph network [24], transformer [68], or memory networks [23, 73]. Despite their success in introducing temporal information into the visual tracking task, most of the explorations are restricted to *only a single stage*, *i.e.,* the template feature, in the whole tracking pipeline.

In this work, we present a comprehensive framework for exploiting temporal contexts in Siamese-based networks, which we call **TCTrack**. As shown in Fig. 1, TCTrack introduces temporal context into the tracking pipeline *at two levels*, *i.e.,* features and similarity maps. At the **feature level**, we propose an online temporally adaptive convolution (TAdaConv), where features are extracted with convolution weights dynamically calibrated by the previous frames. Based on this operation, we transform the standard convolutional networks to temporally adaptive ones (TAdaCNN). Since the calibration in the online TAdaConv is based on the global descriptor of the features in the previous frames, TAdaCNN only introduces a negligible frame rate drop but notably improves the tracking performance. At the **similarity map level**, an adaptive temporal transformer (AT-Trans) is proposed to refine the similarity map according to the temporal information. Specifically, AT-Trans adopts an encoder-decoder structure, where **(i)** the encoder produces the temporal prior knowledge for the current time step, by integrating the previous prior with the current similarity map, and **(ii)** the decoder refines the similarity map based on the produced temporal prior knowledge in an adaptive way. Compared to [23,24,68], AT-Trans is memory efficient and thus edge-platform friendly since we keep updating the temporal prior knowledge at each frame. Overall, our approach provides a holistic temporal encoding framework to handle temporal contexts in Siamese-based aerial tracking.

Extensive evaluations of TCTrack show both the effectiveness and the efficiency of the proposed framework. Competitive accuracy and precision are observed on four standard aerial tracking benchmarks in comparison with 51 state-of-the-art trackers, where TCTrack also has a high frame rate of 125.6 FPS on PC. Real-world deployment on NVIDIA Jetson AGX Xavier shows that TCTrack maintains impressive stability and robustness for aerial tracking, running at a frame rate of over 27 FPS.

2. Related Work

Tracking by detection. After D. S. Bolme *et al.* firstly proposed the MOSSE filter [5], many researches [16,31,38] have been made to boost the tracking performance. However, since they suffer from poor representative feature expression, they are hard to maintain robustness under com-plex aerial tracking conditions. Recently, Siamese-based trackers have stood out attributing to their SOTA accuracy and attractive efficiency [2,3,9,26,41,42,78]. For meeting the aerial tracking requirement, some works propose efficient tracking methods [7,21,22].

Despite achieving SOTA performance, those trackers above disregard the temporal contexts in the tracking scenarios, thereby blocking the performance improvement. Differently, our tracker can effectively model the historical temporal contexts during the tracking for increasing the discriminability and robustness.

Temporal-based tracking methods. Previously, many works are devoted to exploiting the temporal information in tracking scenarios for raising the tracking performance [10, 33, 43, 47]. Recently, many DL-based temporal tracking methods focus on dynamic templates based on transformer integration [68], template memory update [23,27,73], graph network [24], weighted sum [74], and explicit template update [72]. They try to update the template features in an explicit way or implicit way based on the pre-defined parameters. Then, based on the transformed template features, those trackers exploit the discrete temporal information in tracking sequences.

Despite superior tracking performance, they introduce temporal information via only a single level in the whole tracking pipeline, blocking further improvement of tracking performance. To fully exploit the temporal contexts, in this work, we propose a comprehensive framework for exploring the temporal contexts via two levels, *i.e.*, features level and similarity maps level.

Temporal modelling in videos. Modelling the temporal dynamics is essential for a genuine understanding of videos. Hence, it is widely explored in both supervised [20, 35, 48, 49, 63, 70] and self-supervised paradigm [28, 29, 34, 36, 39]. Self-supervised approaches learns temporal modelling by solving various pre-text tasks, such as dense future prediction [28, 29], jigsaw puzzle solving [36, 39], and pseudo motion classification [34], *etc.* Supervised video recognition explores various connections between different frames, such as 3D convolutions [62], temporal convolution [63], and temporal shift [48], *etc.* Closely related to our work is the temporally adaptive convolutions [35], which is applied for temporal modeling in videos. In this work, to adapt to the tracking task, we propose an online CNN which can extract spatial features according to temporal contexts for enriching the temporal information comprehensively.

3. Temporal Contexts for Aerial Tracking

In this section, the detailed structure of our framework is described as shown in Fig. 2. The proposed framework considers temporal contexts from two new perspectives: **(1)** online feature extraction where we incorporate temporal context by TAdaCNN (Sec. 3.1); and **(2)** similarity map refine-
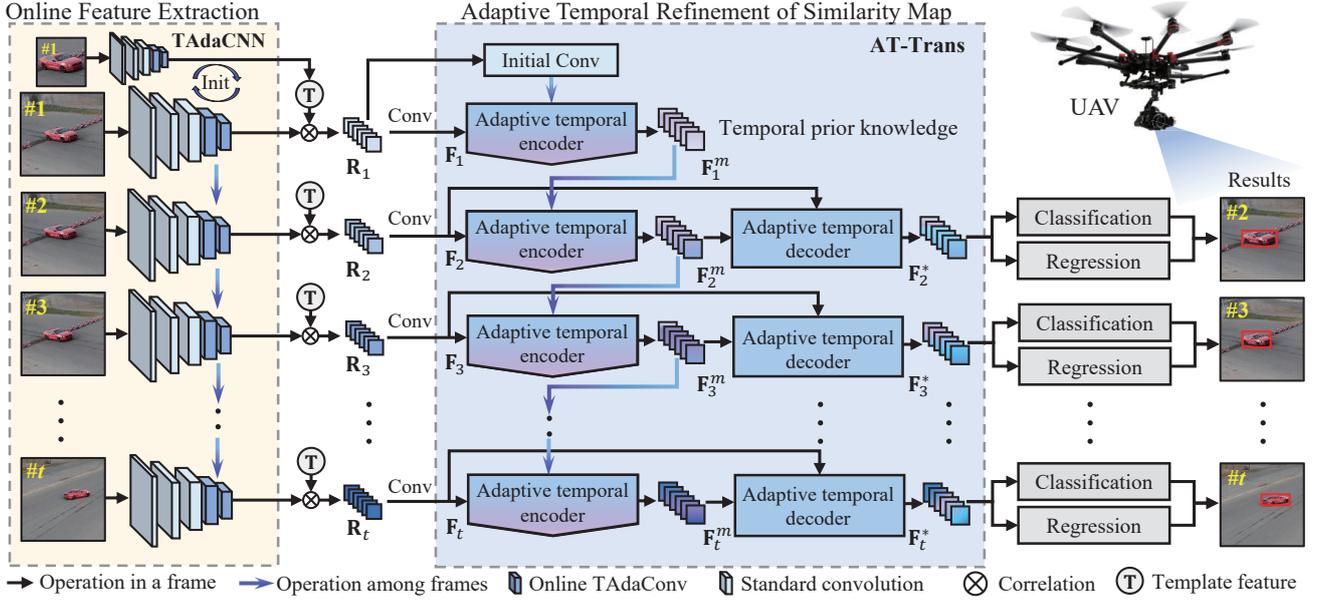
Figure 2. Overview of our framework. It mainly consists of three components, *i.e.*, TAdaCNN for online feature extraction shown in Fig. 3, AT-Trans for similarity map refinement shown in Fig. 4, and classification&regression for final prediction. This figure illustrates the workflow of our TCTrack when tracking sequences are *t* frames. Through temporal contexts before correlation and after, comprehensive temporal knowledge is introduced in our framework. Best view in color.

ment where we use a novel AT-Trans to encode the temporal knowledge and then refine the similarity map according to the temporal prior knowledge (Sec. 3.2).

3.1. Feature extraction with online TAdaConv

As a key component of our framework, an online TAda-Conv is proposed for feature extraction based on [35] to consider temporal contexts whose structures are shown in Fig. 3. Formally, given the input feature to the online TAda-Conv at a certain stage in the network $\tilde{\mathbf{X}}_t$ in the t-th frame, the output of the online TAdaConv $\tilde{\mathbf{X}}_t$ can be obtained as follows:

$$\tilde{\mathbf{X}}_t = \mathbf{W}_t * \mathbf{X}_t + \mathbf{b}_t , \qquad (1)$$

where the operator $*$ denotes the convolution operation and $\mathbf{W}_t, \mathbf{b}_t$ are the temporal weight and bias of our convolution. A standard convolution layer uses learnable parameters for weights and bias, and shares them in the whole tracking sequence. Differently, in our online convolution layer, the parameters are calculated by the learnable parameters ($\mathbf{W}_b$ and $\mathbf{b}_b$) and calibration factors that are varied for each frame, *i.e.,* $\mathbf{W}_t = \mathbf{W}_b \cdot \boldsymbol{\alpha}_t^w$ and $\mathbf{b}_t = \mathbf{b}_b \cdot \boldsymbol{\alpha}_t^b$. Different from the original structure in video understanding, online TAdaConv processes one frame at a time. Hence, it only considers the temporal context in the past just like tracking in the real world. Specifically, we keep a temporal context queue $\hat{\mathbf{X}} \in \mathbb{R}^{L \times C}$ of L frame descriptors $\hat{\mathbf{X}}_t \in \mathbb{R}^C$ including that of the current frame:

$$\hat{\mathbf{X}} = \mathrm{Cat}(\hat{\mathbf{X}}_t, \hat{\mathbf{X}}_{t-1}, ..., \hat{\mathbf{X}}_{t-L+1}) , \qquad (2)$$

where Cat represents the concatenation and the frame descriptor is obtained by a global average pooling (GAP) over the feature of the each coming frame, *i.e.,* $\hat{\mathbf{X}}_t = \mathrm{GAP}(\mathbf{X}_t)$. For the generation of calibration factors $\boldsymbol{\alpha}_t^w$ and $\boldsymbol{\alpha}_t^b$, we perform two convolutions over the temporal context queue $\hat{\mathbf{X}}$ with a kernel size of L, *i.e.,* $\boldsymbol{\alpha}_t^w = \mathcal{F}_{\mathrm{w}}(\hat{\mathbf{X}}) + 1$, $\boldsymbol{\alpha}_t^b = \mathcal{F}_{\mathrm{b}}(\hat{\mathbf{X}}) + 1$, where $\mathcal{F}_i$ denotes the convolution operation. Besides, the weights of $\mathcal{F}$ are initialized to zero so that at the initialization, $\mathbf{W}_t = \mathbf{W}_b$ and $\mathbf{b}_t = \mathbf{b}_b$. For $t \leq L - 1$, where there is not enough previous frames, we fill that with the descriptor of the first frame $\hat{\mathbf{X}}_1$. Given our backbone φ_{tada} that considers the temporal contexts in the feature extraction process, the similarity map $\mathbf{R}_t$ for the t-th frame can be obtained as:

$$\mathbf{R}_t = \varphi_{tada}(\mathbf{Z}) \star \varphi_{tada}(\mathbf{X}_t) , \qquad (3)$$

where $\mathbf{Z}$ denotes the template and $\star$ represents the depth-wise correlation [41]. After that, $\mathbf{F}_t$ can be obtained by a convolution layer, *i.e.*, $\mathbf{F}_t = \mathcal{F}(\mathbf{R}_t)$.

Remark 1: To the best of our knowledge, our online TAdaCNN is the first to integrate temporal contexts in the feature extraction process in the tracking task.

3.2. Similarity Refinement with AT-Trans

Besides considering temporal contexts in the feature extraction process, in this work, we also propose an AT-Trans for refining the similarity map $\mathbf{F}_t$ according to the temporal contexts. Specifically, our AT-Trans has an encoder-decoder structure, where the encoder aims to integrate tem-
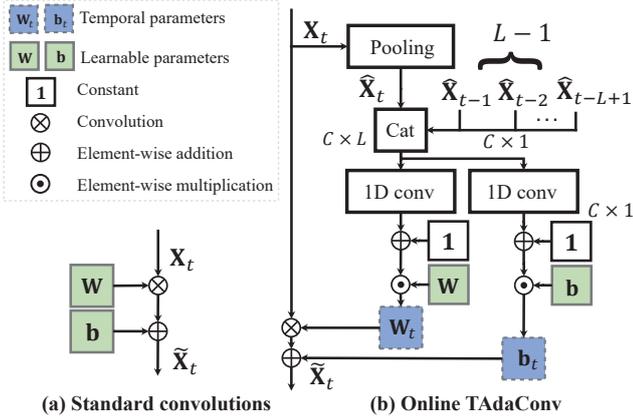
Figure 3. The schema of our online TAdaConv. The temporal calibration factor is generated by the feature sequences (number of its is L). Based on the temporal vectors, the parameters of our online TAdaConv can be adjusted adaptively in every frame.

poral knowledge and the decoder focuses on similarity refinement. In this section, we first revisit the multi-head attention [64] before describing the details of our AT-Trans.

Multi-head attention. As a fundamental component of the transformer, multi-head attention is formulated as follows:

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \Big(\text{Cat}(\mathbf{H}_{att}^1, ..., \mathbf{H}_{att}^N)\Big)\mathbf{W}$$
$$\mathbf{H}_{att}^n = \text{Attention}(\mathbf{Q}\mathbf{W}_q^n, \mathbf{K}\mathbf{W}_k^n, \mathbf{V}\mathbf{W}_v^n) \quad , \quad (4)$$
$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}(\mathbf{Q}\mathbf{K}^{\text{T}}/\sqrt{d})\mathbf{V}$$

where $\sqrt{d}$ is the scaling factor while $\mathbf{W} \in \mathbb{R}^{C_i \times C_i}$, $\mathbf{W}_q^n \in \mathbb{R}^{C_i \times C_h}$, $\mathbf{W}_k^n \in \mathbb{R}^{C_i \times C_h}$, and $\mathbf{W}_v^n \in \mathbb{R}^{C_i \times C_h}$ are learnable weights. In our AT-Trans, we employ multi-head attention with 6 heads, *i.e.*, $N = 6$ and $C_h = C_i/6$.

Compared to CNN, Transformer can more effectively encode the global context information [18, 64]. Hence, to exploit the global temporal contexts more effectively, we propose a transformer-based temporal integration strategy to successively encode global contexts information. Moreover, most existing temporal-based methods generally store the input features for temporal modeling, inevitably introducing sensitive parameters and unnecessary computation. In this work, for eliminating unnecessary operations and sensitive parameters, we adopt an online update strategy for temporal knowledge.

Transformer encoder. The encoder generates temporal prior knowledge by integrating the previous knowledge with current features. Generally, we stack two multi-head attention layers before a temporal information filter is applied. The final temporal prior knowledge for the current step is obtained by further attaching a multi-head attention layer to the filtered information. The structure of the encoder is presented in Fig. 4(a).

Given the previous temporal prior knowledge $\mathbf{F}_{t-1}^m$ and the current similarity map $\mathbf{F}_t$, there are two ways to in-
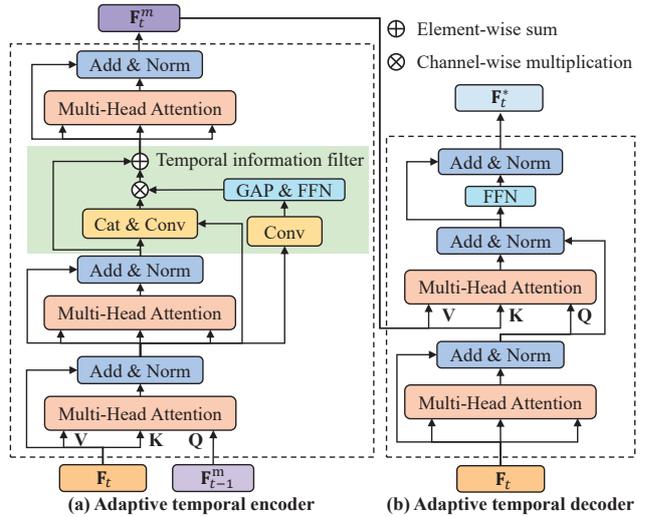

Figure 4. Structure of the adaptive temporal transformer. The left sub-window illustrates the adaptive temporal encoder to model the temporal knowledge. The right sub-window shows the component of the decoder. Best viewed in color.

tegrate their information into the current prior knowledge $\mathbf{F}_t^m$, with respect to the selection of the query, key, and values. One uses $\mathbf{F}_{t-1}^m$ as the query and $\mathbf{F}_t$ as the value and key, while the other uses them in reverse. In our method, we adopt the former, as this essentially puts more emphasis on the current similarity map. This is plausible as closer temporal information is more valuable than the previous one for representing the characteristics of the current object more accurately. Empirical results in Sec. 4.3 also validate the effectiveness of this choice. Hence, we obtain the output of the stacked multi-head attention layer in t-th frame $\mathbf{F}_t^2$ by:

$$\begin{aligned}\mathbf{F}_t^1 &= \text{Norm}(\mathbf{F}_t + \text{MultiHead}(\mathbf{F}_{t-1}^m, \mathbf{F}_t, \mathbf{F}_t)) \\ \mathbf{F}_t^2 &= \text{Norm}(\mathbf{F}_t^1 + \text{MultiHead}(\mathbf{F}_t^1, \mathbf{F}_t^1, \mathbf{F}_t^1))\end{aligned} \quad , \quad (5)$$

where Norm represents the layer normalization.

Since aerial tracking may frequently encounter less useful contexts caused by motion blur or occlusion, some unwanted contexts may be included if we pass along the complete temporal information without any filtering. To eliminate the unwanted information, a neat temporal information filter is generated by attaching a feed-forward network FFN to the global descriptor of $\mathbf{F}_t^1$ obtained by global average pooling GAP, *i.e.*, $\boldsymbol{\alpha} = \text{FFN}(\text{GAP}(\mathcal{F}(\mathbf{F}_t^1)))$. The filtered information $\mathbf{F}_t^f$ is obtained by:

$$\mathbf{F}_t^f = \mathbf{F}_t^2 + \mathcal{F}(\text{Cat}(\mathbf{F}_t^2, \mathbf{F}_t^1)) * \boldsymbol{\alpha} , \quad (6)$$

where $\mathcal{F}$ denotes a convolution layer. With this, the temporal knowledge of t-th frame, $\mathbf{F}_t^m$ can be obtained as follows:

$$\mathbf{F}_t^m = \text{Norm}(\mathbf{F}_t^f + \text{MultiHead}(\mathbf{F}_t^f, \mathbf{F}_t^f, \mathbf{F}_t^f)) . \quad (7)$$

Hence, for each frame, we update the temporal knowledge rather than saving all of them. This makes the memory
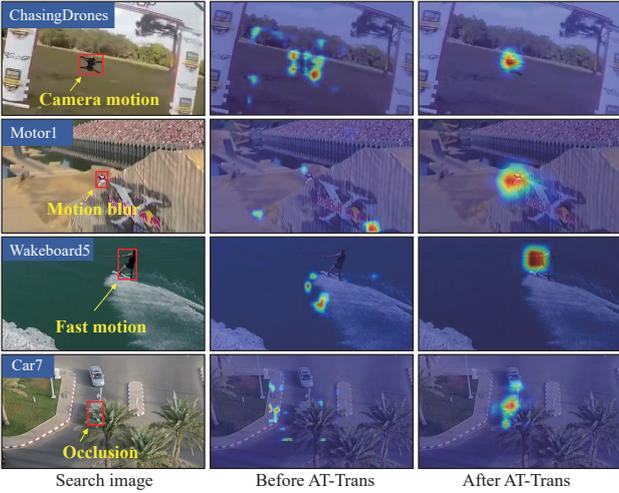
Figure 5. Comparison between similarity maps before refinement (second column) and after (third column) refinement.

occupancy of the temporal prior knowledge fixed during the whole tracking process, which makes TCTrack memory-efficient compared to approaches that require saving all the intermediate temporal information. Overall, owing to this strategy as well as the temporal filter and the multi-head attention, our AT-Trans adaptively encodes the temporal prior in a memory-efficient way.

For the first frame in a tracking sequence, since the characteristics of different targets are distinct, using a unified initialization for the initial temporal prior $\mathbf{F}_0^m$ would be unreasonable. Observing that the similarity map in the first frame essentially represents the semantic features of the target object in an effective way, we set the initial temporal prior by a convolution over the initial similarity map $\mathbf{F}_0$, i.e., $\mathbf{F}_0^m = \mathcal{F}_{init}(\mathbf{R}_1)$. We also empirically show our initialization is better in Sec. 4.3.

Transformer decoder. According to the temporal prior knowledge $\mathbf{F}_t^m$, the decoder aims to refine the similarity map. To better explore the interrelations between temporal knowledge and current spatial features $\mathbf{F}_t$, we adopt two multi-head attention layers with feed-forward before output. Its structure is presented in Fig. 4(b). By generating the attention map, the valid information in the temporal knowledge $\mathbf{F}_t^m$ can be extracted for refining the similarity map $\mathbf{F}_t$ to obtain the final output $\mathbf{F}_t^*$:

$$\mathbf{F}_t^3 = \text{Norm}(\mathbf{F}_t + \text{MultiHead}(\mathbf{F}_t, \mathbf{F}_t, \mathbf{F}_t))$$
$$\mathbf{F}_t^4 = \text{Norm}(\mathbf{F}_t^3 + \text{MultiHead}(\mathbf{F}_t^3, \mathbf{F}_t^m, \mathbf{F}_t^m)) . \quad (8)$$
$$\mathbf{F}_t^* = \text{Norm}(\mathbf{F}_t^4 + \text{FFN}(\mathbf{F}_t^4))$$

Relying on the encoder-decoder structure of AT-Trans, the temporal contexts are effectively exploited to refine the similarity maps for boosting robustness and accuracy. The comparison of similarity maps in Fig. 5 shows the effectiveness of the similarity map refinement, especially where camera motion, severe motion, and occlusion exist.

Table 1. Comparison of inference time and parameters on NVIDIA Jetson AGX Xavier. Here, we use $287 \times 287 \times 3$ as the input image and only evaluate the inference time of the CNN.

Backbone	Inference time	Parameters
AlexNet [40]	3.4ms	2.47M
VGG11 [58]	3.7ms	9.22M
ResNet18 [30]	10.1ms	11.2M
MobileNet_v2 [56]	13.7ms	2.2M
EfficientNet [60]	27.4ms	39.4K
SqueezeNet1_0 [37]	8.8ms	735.42K
ShuffleNet_v2_x0.5 [77]	16.6ms	341.8K

Remark 2: To the best of our knowledge, AT-Trans is the first attempt to use temporal contexts for similarity maps.

4. Experiments

Our framework is evaluated on four public authoritative benchmarks and tested on real-world aerial tracking conditions. In this section, our method is comprehensively evaluated on four well-known aerial tracking benchmarks, i.e., UAV123 [54], UAVTrack112_L [21], UAV123@10fps [54], and DTB70 [45]. 51 existing top trackers are included for a thorough comparison, where their results are obtained by running the official codes with their corresponding hyper-parameters. For a clearer comparison, we divide them into two groups, **(i)** light-weight trackers [1,2,6,7,12,14–17,22, 27,33,38,41,43,44,46,47,51,52,65–67,75,76,80] and **(ii)** deep trackers [4,8,9,11,13,23,25,26,41,50,53,59,68,69, 71,74,78,79].

4.1. Implementation Details

We use AlexNet as the backbone of our tracker, as efficiency is essential for aerial tracking. As shown in Table 1, the comparison in inference time of different popular backbones on the NVIDIA Jetson AGX Xavier platform has shown that AlexNet has the lowest latency, while the recent developments in mobile networks [37,56,77] suffer from high memory access cost (MAC). For initialization, we use ImageNet pre-trained model for AlexNet and use the same initialization for online TAdaConv as in [55]. The AT-Trans in our TCTrack is randomly initialized.

We train our tracker with the videos whose length are 4 from VID [55], Lasot [19], and GOT-10K [32]. We train TCTrack for a total of 100 epochs on two NVIDIA TITAN RTX GPUs. For the first 10 epochs, the parameters of the backbone are frozen, following [41]. The rest of the training process employs a learning rate decreasing from 0.005 to 0.0005 in log space. SGD is employed as the optimizer with a momentum of 0.9, where the mini-batch size is 124 pairs. The input sizes of the template and the search area are 127^2 and 287^2 respectively. The proposed online TAdaConv is used in the replacement of the last two convolutional layers. ***Remark 3:*** For more detailed information about the evaluation criteria and loss function, please refer to the supple-
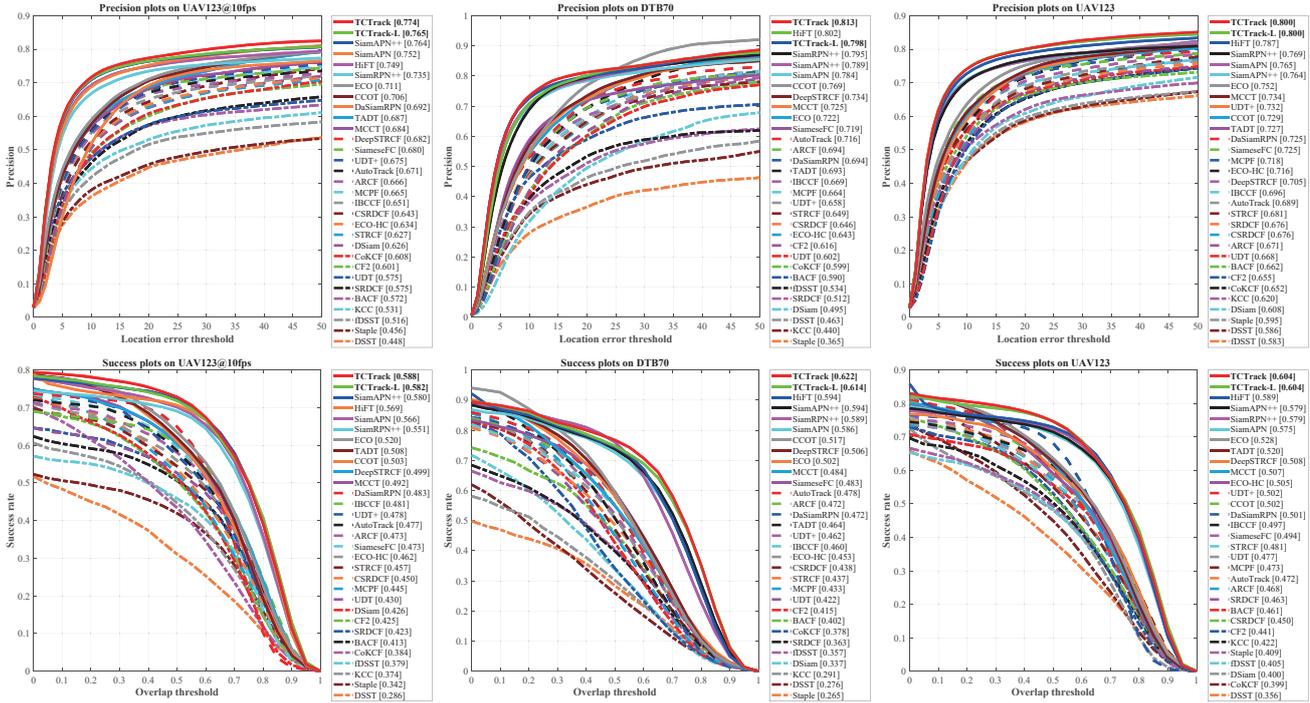
Figure 6. Overall performance of all trackers on three well-known aerial tracking benchmarks. Our tracker achieves superior performance against other SOTA trackers. TCTrack-L represents the tracker with AT-Trans while the TCTrack denotes the full version of our framework.

mentary material.

4.2. Comparison with Light-Weight Trackers

In this subsection, TCTrack is compared with 29 existing efficient trackers on the standard aerial tracking benchmarks. For Siamese-based methods, we evaluate them with the same backbone as ours for a fair comparison.

UAV123. UAV123 [54] is a large-scale aerial tracking benchmark involving 123 challenging sequences with more than 112K frames. Performance evaluation on UAV123 can verify the tracking performance in most commonly aerial tracking conditions. As shown in Fig. 6, our TCTrack outperforms HiFT and SiamRPN++ in AUC (3%) and (4.3%).

DTB70. DTB70 [45] includes 70 severe motion scenarios in various challenging scenes. For evaluating the effectiveness of our method in handling motion, we adopt this benchmark to prove the robustness of TCTrack. Our tracker ranks 1st with an improvement of 5% in AUC against the other best tracker illustrated in Fig. 6.

UAV123@10fps. Adopting an image rate of 10 FPS, the motion and variation are more abrupt and severe in UAV123@10fps [54], thereby significantly raising the difficulty of tracking. From the comparison with our other SOTA trackers, we can clearly see that our tracker maintains superior robustness and exceeds the second-best tracker in terms of success and precision rate.

Attribute-based performance. In aerial tracking conditions, the severe motion of UAVs will increase the difficulty of tracking. To fully analyze the robustness of our tracker in

Table 2. Overall performance on UAVTrack112_L. The best three performances are respectively highlighted with **red**, **green**, and **blue** colors.

Trackers	Succ.	Prec.	Trackers	Succ.	Prec.
AutoTrack [47]	0.405	0.675	C-COT [17]	0.422	0.691
ARCF [33]	0.399	0.640	UDT+ [66]	0.405	0.637
STRCF [43]	0.360	0.609	ECO [12]	0.436	0.684
UDT [66]	0.388	0.620	TADT [46]	0.462	0.712
SRDCF [16]	0.320	0.508	SiameseFC [2]	0.452	0.690
CoKCF [75]	0.283	0.520	DaSiamRPN [80]	0.479	0.729
BACF [38]	0.358	0.593	SiamAPN++ [7]	0.537	**0.735**
DSiam [27]	0.321	0.512	SiamRPN++ [41]	**0.559**	**0.773**
HiFT [6]	**0.551**	0.734	**TCTrack (ours)**	**0.582**	**0.786**

specific challenges such as fast motion, camera motion, occlusion, deformation, *etc*, attribute-based comparisons are conducted. The comparison between other SOTA trackers presented in Fig. 7 proves the robustness of our framework in several challenging conditions. Since our tracker can accumulate the consecutive temporal knowledge from 1st frame to the current frame, our tracker can learn the historical location of the object. Therefore, our tracker achieves superior performance in occlusion and fast-motion scenes. Furthermore, benefiting from our content-adaptive temporal knowledge and online TAdaConv, TCTrack can handle the negative influence introduced by the environment.

UAVTrack112_L. To validate the effectiveness of our framework in long-term tracking performance, we conduct the evaluations on UAVTrack112_L [21], which is the current biggest long-term aerial tracking benchmark including over 60k frames. Table 2 reports the comparison of TCTrack and other SOTA trackers. Thanks to our comprehen-
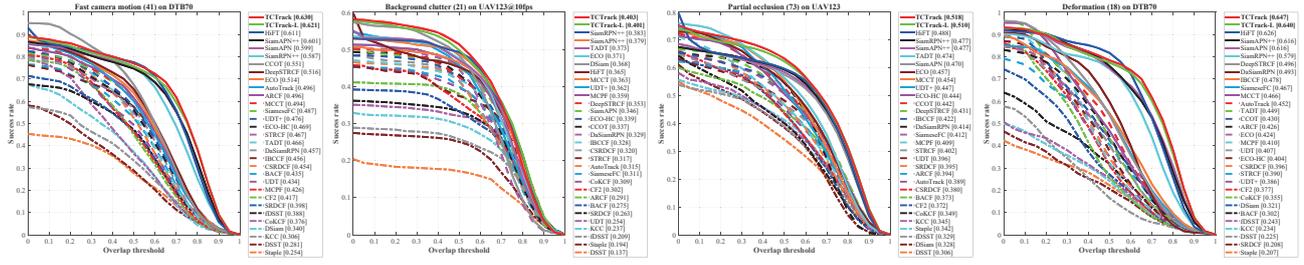
Figure 7. Attribute-based evaluation of all trackers on three well-known aerial tracking benchmarks. Our temporal tracker can maintain promising performance under severe motion, occlusion, and deformation. More results are shown in the supplementary material.

Table 3. Ablation study of different components of adaptive temporal transformer on UAV123 [54]. `TIF` denotes the temporal information filter in the AT-Trans (Fig. 4). `SF`/`MF` refer to single-frame (SF) training, *i.e.,* the standard tracking-by-detection training method and our multi-frame (MF) training method. `CI`/`RI` refer to convolutional initialization and random initialization for temporal prior knowledge. Query denotes which feature map is used as the query in the adaptive temporal encoder in AT-Trans mentioned in Sec. 3.2.

Model	Train Init.	Query	Camera Motion		Fast motion		Partial Occlusion		Overall	
			Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.
Transformer	SF	-	F_{t-1}^m	0.750	0.549	0.712	0.509	0.663	0.458	0.750	0.550
Transformer+TIF	SF	-	F_{t-1}^m	$0.767_{2.3\%\uparrow}$	$0.578_{5.3\%\uparrow}$	$0.720_{1.1\%\uparrow}$	$0.525_{3.1\%\uparrow}$	$0.667_{0.6\%\uparrow}$	$0.474_{3.5\%\uparrow}$	$0.765_{2.0\%\uparrow}$	$0.573_{4.2\%\uparrow}$
Transformer	MF	CI	F_{t-1}^m	$0.749_{2.4\%\downarrow}$	$0.525_{7.6\%\downarrow}$	$0.719_{2.4\%\downarrow}$	$0.500_{7.6\%\downarrow}$	$0.639_{2.4\%\downarrow}$	$0.415_{7.6\%\downarrow}$	$0.732_{2.4\%\downarrow}$	$0.508_{7.6\%\downarrow}$
Transformer+TIF	MF	RI	F_{t-1}^m	$0.779_{3.9\%\uparrow}$	$0.592_{7.8\%\uparrow}$	$0.766_{7.6\%\uparrow}$	$0.566_{11.2\%\uparrow}$	$0.670_{1.1\%\uparrow}$	$0.483_{5.5\%\uparrow}$	$0.772_{2.9\%\uparrow}$	$0.586_{6.6\%\uparrow}$
Transformer+TIF	MF	CI	F_t	$0.785_{4.7\%\uparrow}$	$0.587_{6.9\%\uparrow}$	$0.726_{2.0\%\uparrow}$	$0.528_{3.7\%\uparrow}$	$0.676_{2.0\%\uparrow}$	$0.480_{4.8\%\uparrow}$	$0.771_{2.8\%\uparrow}$	$0.580_{5.5\%\uparrow}$
Transformer+TIF	**MF**	**CI**	$\mathbf{F_{t-1}^m}$	$\mathbf{0.810_{8.0\%\uparrow}}$	$\mathbf{0.615_{12.0\%\uparrow}}$	$\mathbf{0.793_{11.3\%\uparrow}}$	$\mathbf{0.586_{15.1\%\uparrow}}$	$\mathbf{0.710_{7.1\%\uparrow}}$	$\mathbf{0.510_{11.4\%\uparrow}}$	$\mathbf{0.800_{6.7\%\uparrow}}$	$\mathbf{0.604_{9.8\%\uparrow}}$

Table 4. Different sequence lengths for the online TAdaConv on UAV123 [54].

Different Variations	Overall Precision	Overall Success
Transformer	0.750	0.550
Transformer+TAdaConv (L=1)	$0.749_{0.1\%\downarrow}$	$0.561_{2.0\%\uparrow}$
Transformer+TAdaConv (L=2)	$0.774_{3.2\%\uparrow}$	$0.573_{4.2\%\uparrow}$
Transformer+TAdaConv (L=3)	$\mathbf{0.776_{3.5\%\uparrow}}$	$\mathbf{0.580_{5.5\%\uparrow}}$

sive framework that fully exploits temporal contexts, TC-Track achieves superior performance against other trackers in terms of precision (0.786) and success rate (0.582).

4.3. Ablation Study

To verify the effectiveness of our framework, comprehensive ablation studies are presented in this subsection.

Clarification of symbol. In Table 3, we denote our proposed transformer architecture without temporal information filter as `Transformer`. We analyze the influence caused by different models, training methods, initializations, and query selections. Furthermore, for ensuring the correctness of our experiments, all tracker adopts the same process (including training, parameter settings, *etc.*) except for the studied module.

Analysis on AT-Trans. I) Adding the consecutive temporal knowledge without filtering out the invalid information (third line) will confuse the tracker. Therefore, the tracking performance is impeded significantly. By adding our information filter in the tracking-by-detection framework, our module can also raise the performance by adaptively selecting valid contexts (second line). **II)** As we discussed before, using the unique information of the tracking object in the first frame to initiate the temporal knowledge is more

appropriate than random initiation, especially in occlusion conditions (raising about 6%). **III)** We also analyze the effect caused by the different queries. The results prove that refinement based on the current similarity map is more effective and suitable for raising performance, especially in motion scenarios (improved over 10%).

Compared with `Transformer`, there is a significant improvement brought by our temporal knowledge encoded by AT-Trans (**9.8%** in overall AUC and **6.7%** in overall precision). Specifically, our tracker yields the best performance with an improvement of about **12.0%** and **15.1%** in handling the motion scenes. In the occlusion conditions, owing to the consecutive temporal contexts, our tracker can relocate the object via the previous information, thereby boosting the success rate by **11.4%**.

Studies about the length of temporal sequences in TAda-Conv. As shown in Table 4, when the image range of TAdaConv is increasing, the performance is raising. To introduce the temporal contexts effectively and efficiently, in this work, we adopt 3 as the length of sequences, *i.e.*, L=3.

4.4. Comparison with Deep Trackers

Our approach aims to introduce temporal information to raise the robustness and handle the challenges in aerial tracking. Therefore, to comprehensively illustrate our efficiency and performance against other SOTA trackers with deeper backbones, further comparisons are constructed including over 20 trackers on NVIDIA TITAN RTX. As illustrated in Fig. 8, although adopting the lightweight CNN as our backbone, TCTrack achieves competitive performance compared with the best tracker while running **2.49** times faster than the best tracker (TransT). Attribute to
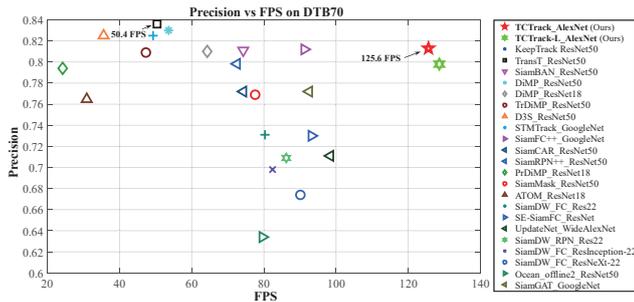
Figure 8. Comparisons to trackers with deeper backbones on DTB70. Our tracker achieves competitive performance compared with other deeper trackers while possessing superior efficiency.

our content-adaptive and memory-efficient structure, our framework with temporal contexts can fill the performance margin caused by deeper backbones while maintaining the promising efficiency in aerial tracking conditions.

5. Real-world Tests

In this section, we implement our tracker on UAV to validate its practicability in real-world conditions. Specifically, NVIDIA Jetson AGX Xavier and Pixhawk[2] are adopted as the aerial onboard computer and flight controller. During the real-world UAV tests, RAM usage and GPU VRAM usage are 15.29% and 3%, respectively. Additionally, the utilization of GPU and CPU is 46% and 12.43% on average. The center location error (CLE) is adopted to evaluate the tracking performance (20 is the success threshold).

The special challenges in the real-world tests involve different illumination, scale variation, occlusion, motion blur, and low-resolution scenes. The visualization of our tracking recording of practical UAV is shown in Fig. 9. When facing partial occlusion and low illumination (the first row), our tracker can maintain impressive stability and robustness via exploiting the consecutive temporal knowledge. Meantime, our tracker also achieves satisfying accuracy when facing motion blur and the occluded object (the second row). Additionally, the visualization of the third row strongly presents the powerful ability of our tracker under camera motion conditions. Finally, our tracker remains at a speed of over 27 FPS during the tests without the acceleration of TensorRT[3]. The real-world tests on our practical UAV strongly demonstrate the practicability and feasible deployment ability of our framework. Furthermore, our tracker presents stable and promising tracking performance in complex aerial tracking conditions.

6. Conclusion and Discussion

In this work, we propose a comprehensive framework for introducing temporal contexts into aerial tracking which
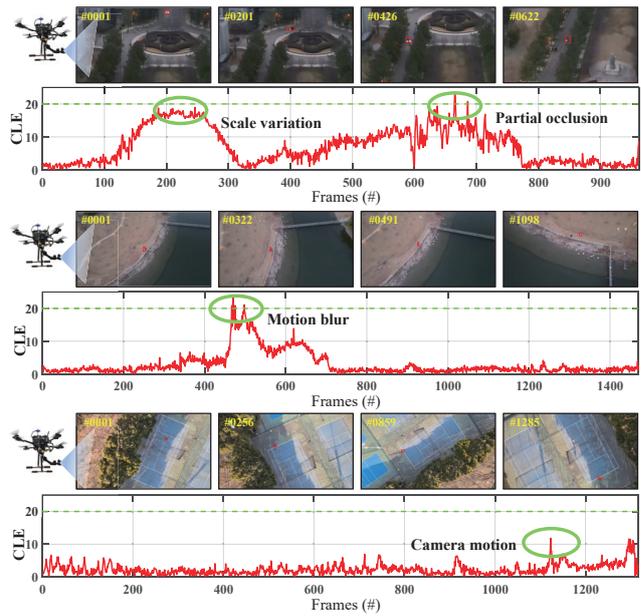
Figure 9. Recording of real-world tests on the embedded platform. The tracking targets are marked with red while the CLE represents the center location error. To avoid unpredictable disclosure of personally identifiable information, images are processed merely.

consists of two perspectives, *e.g.*, feature extraction and similarity refinement. Specifically, in this work, AT-Trans and online TAdaCNN are the first attempts for exhaustively exploring temporal contexts. Besides, attributing to our online updating strategy, unnecessary operations and memory loading are avoided. Extensive experiments on four benchmarks and real-world tests on our UAV demonstrate the effectiveness and efficiency of our framework. We hope that our framework can inspire further research in aerial and even general tracking with temporal contexts.

Potential limitations. Hindered by the short-term training method, the potential of our framework in very long-term temporal modeling and long-time occlusion is not fully explored. Moreover, the TensorRT and ONNX versions will be developed in our future works.

Negative impacts. Although TCTrack aims to explore temporal contexts comprehensively for aerial tracking, impressive efficiency and effectiveness make it easy to be deployed on UAVs for unauthorized surveillance.

References

[1] Luca Bertinetto, Jack Valmadre, Stuart Golodetz, Ondrej Miksik, and Philip HS Torr. Staple: Complementary Learners for Real-Time Tracking. In *CVPR*, pages 1401–1409, 2016. 5

[2] Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. Fully-Convolutional Siamese Networks for Object Tracking. In *ECCV*, pages 850–865, 2016. 1, 2, 5, 6

[3] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Know Your Surroundings: Exploiting Scene Information for Object Tracking. In *ECCV*, pages 205–221, 2020. 2

[4] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning Discriminative Model Prediction for Tracking. In *ICCV*, pages 6181–6190, 2019. 1, 5

[5] David S Bolme, J Ross Beveridge, Bruce A Draper, and Yui Man Lui. Visual Object Tracking Using Adaptive Correlation Filters. In *CVPR*, pages 2544–2550, 2010. 2

[6] Ziang Cao, Changhong Fu, Junjie Ye, Bowen Li, and Yiming Li. HiFT: Hierarchical Feature Transformer for Aerial Tracking. In *ICCV*, pages 15437–15446, 2021. 1, 5, 6

[7] Ziang Cao, Changhong Fu, Junjie Ye, Bowen Li, and Yiming Li. SiamAPN++: Siamese Attentional Aggregation Network for Real-Time UAV Tracking. In *IROS*, pages 1–7, 2021. 1, 2, 5, 6

[8] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer Tracking. In *CVPR*, pages 8126–8135, 2021. 5

[9] Zedu Chen, Bineng Zhong, Guorong Li, Shengping Zhang, and Rongrong Ji. Siamese Box Adaptive Network for Visual Tracking. In *CVPR*, pages 6668–6677, 2020. 2, 5

[10] Kenan Dai, Dong Wang, Huchuan Lu, Chong Sun, and Jianhua Li. Visual Tracking via Adaptive Spatially-Regularized Correlation Filters. In *CVPR*, pages 4670–4679, 2019. 2

[11] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. ATOM: Accurate Tracking by Overlap Maximization. In *CVPR*, pages 4655–4664, 2019. 1, 5

[12] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. ECO: Efficient Convolution Operators for Tracking. In *CVPR*, pages 6931–6939, 2017. 5, 6

[13] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic Regression for Visual Tracking. In *CVPR*, pages 7181–7190, 2020. 5

[14] Martin Danelljan, Gustav Häger, Fahad Khan, and Michael Felsberg. Accurate Scale Estimation for Robust Visual Tracking. In *BMVC*, 2014. 5

[15] Martin Danelljan, Gustav Häger, Fahad Shahbaz Khan, and Michael Felsberg. Discriminative Scale Space Tracking. *PAMI*, 39(8):1561–1575, 2016. 5

[16] Martin Danelljan, Gustav Häger, Fahad Shahbaz Khan, and Michael Felsberg. Learning Spatially Regularized Correlation Filters for Visual Tracking. In *ICCV*, pages 4310–4318, 2015. 1, 2, 5, 6

[17] Martin Danelljan, Andreas Robinson, Fahad Shahbaz Khan, and Michael Felsberg. Beyond Correlation Filters: Learning Continuous Convolution Operators for Visual Tracking. In *ECCV*, pages 472–488, 2016. 5, 6

[18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*, 2021. 4

[19] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. Lasot: A High-Quality Benchmark for Large-Scale Single Object Tracking. In *CVPR*, pages 5374–5383, 2019. 5

[20] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast Networks for Video Recognition. In *ICCV*, pages 6202–6211, 2019. 2

[21] Changhong Fu, Ziang Cao, Yiming Li, Junjie Ye, and Chen Feng. Onboard Real-Time Aerial Tracking With Efficient Siamese Anchor Proposal Network. *TGRS*, pages 1–13, 2021. 1, 2, 5, 6

[22] Changhong Fu, Ziang Cao, Yiming Li, Junjie Ye, and Chen Feng. Siamese Anchor Proposal Network for High-Speed Aerial Tracking. In *ICRA*, pages 1–7, 2021. 1, 2, 5

[23] Zhihong Fu, Qingjie Liu, Zehua Fu, and Yunhong Wang. STMTrack: Template-free Visual Tracking with Space-time Memory Networks. In *CVPR*, pages 13774–13783, 2021. 2, 5

[24] Junyu Gao, Tianzhu Zhang, and Changsheng Xu. Graph Convolutional Tracking. In *CVPR*, pages 4649–4659, 2019. 2

[25] Dongyan Guo, Yanyan Shao, Ying Cui, Zhenhua Wang, Liyan Zhang, and Chunhua Shen. Graph Attention Tracking. In *CVPR*, pages 1–10, 2021. 5

[26] Dongyan Guo, Jun Wang, Ying Cui, Zhenhua Wang, and Shengyong Chen. SiamCAR: Siamese Fully Convolutional Classification and Regression for Visual Tracking. In *CVPR*, pages 6268–6276, 2020. 2, 5

[27] Qing Guo, Wei Feng, Ce Zhou, Rui Huang, Liang Wan, and Song Wang. Learning Dynamic Siamese Network for Visual Object Tracking. In *ICCV*, pages 1781–1789, 2017. 2, 5, 6

[28] Tengda Han, Weidi Xie, and Andrew Zisserman. Video Representation Learning by Dense Predictive Coding. In *CVPRW*, pages 1–10, 2019. 2

[29] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-Augmented Dense Predictive Coding for Video Representation Learning. In *ECCV*, pages 312–329, 2020. 2

[30] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, pages 770–778, 2016. 5

[31] João F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-Speed Tracking with Kernelized Correlation Filters. *PAMI*, pages 583–596, 2015. 1, 2

[32] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A Large High-Diversity Benchmark for Generic Object Tracking in The Wild. *TPAMI*, 43(5):1562–1577, 2019. 5

[33] Ziyuan Huang, Changhong Fu, Yiming Li, Fuling Lin, and Peng Lu. Learning Aberrance Repressed Correlation Filters for Real-Time UAV Tracking. In *ICCV*, pages 2891–2900, Nov. 2019. 1, 2, 5, 6

[34] Ziyuan Huang, Shiwei Zhang, Jianwen Jiang, Mingqian Tang, Rong Jin, and Marcelo H Ang. Self-Supervised Motion Learning from Static Images. In *CVPR*, pages 1276–1285, 2021. 2

[35] Ziyuan Huang, Shiwei Zhang, Liang Pan, Zhiwu Qing, Mingqian Tang, Ziwei Liu, and Marcelo H Ang Jr. TAda! Temporally-Adaptive Convolutions for Video Understanding. In *ICLR*, 2022. 2, 3

[36] Yuqi Huo, Mingyu Ding, Haoyu Lu, Ziyuan Huang, Mingqian Tang, Zhiwu Lu, and Tao Xiang. Self-Supervised Video Representation Learning with Constrained Spatiotemporal Jigsaw. In *IJCAI*, pages 751–757, 2021. 2

[37] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. SqueezeNet: AlexNet-Level Accuracy with 50x Fewer Parameters and¡ 0.5 MB Model Size. *arXiv preprint arXiv:1602.07360*, 2016. 5

[38] Hamed Kiani Galoogahi, Ashton Fagg, and Simon Lucey. Learning Background-Aware Correlation Filters for Visual Tracking. In *ICCV*, pages 1135–1143, 2017. 1, 2, 5, 6

[39] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-Supervised Video Representation Learning with Space-Time Cubic Puzzles. In *AAAI*, volume 33, pages 8545–8552, 2019. 2

[40] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet Classification with Deep Convolutional Neural Networks. In *NIPS*, pages 1097–1105, 2012. 5

[41] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. SiamRPN++: Evolution of Siamese Visual Tracking With Very Deep Networks. In *CVPR*, pages 4277–4286, 2019. 1, 2, 3, 5, 6

[42] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High Performance Visual Tracking with Siamese Region Proposal Network. In *CVPR*, pages 8971–8980, 2018. 1, 2

[43] Feng Li, Cheng Tian, Wangmeng Zuo, Lei Zhang, and Ming-Hsuan Yang. Learning Spatial-Temporal Regularized Correlation Filters for Visual Tracking. In *CVPR*, pages 4904–4913, 2018. 2, 5, 6

[44] 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 *ICCVW*, pages 2001–2009, 2017. 5

[45] Siyi Li and Dit-Yan Yeung. Visual Object Tracking for Unmanned Aerial Vehicles: A Benchmark and New Motion Models. In *AAAI*, pages 1–7, 2017. 5, 6

[46] Xin Li, Chao Ma, Baoyuan Wu, Zhenyu He, and Ming-Hsuan Yang. Target-Aware Deep Tracking. In *CVPR*, pages 1369–1378, 2019. 5, 6

[47] 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 *CVPR*, pages 11920–11929, Jun. 2020. 1, 2, 5, 6

[48] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal Shift Module for Efficient Video Understanding. In *ICCV*, pages 7083–7093, 2019. 2

[49] Zhaoyang Liu, Limin Wang, Wayne Wu, Chen Qian, and Tong Lu. TAM: Temporal Adaptive Module for Video Recognition. In *ICCV*, pages 13708–13718, 2021. 2

[50] Alan Lukezic, Jiri Matas, and Matej Kristan. D3S-A Discriminative Single Shot Segmentation Tracker. In *CVPR*, pages 7133–7142, 2020. 5

[51] Alan Lukezic, Tomas Vojir, Luka Cehovin Zajc, Jiri Matas, and Matej Kristan. Discriminative Correlation Filter with Channel and Spatial Reliability. In *CVPR*, pages 6309–6318, 2017. 5

[52] Chao Ma, Jia-Bin Huang, Xiaokang Yang, and Ming-Hsuan Yang. Hierarchical Convolutional Features for Visual Tracking. In *ICCV*, pages 3074–3082, 2015. 5

[53] Christoph Mayer, Martin Danelljan, Danda Pani Paudel, and Luc Van Gool. Learning Target Candidate Association to Keep Track of What Not to Track. In *ICCV*, pages 13444–13454, 2021. 5

[54] Matthias Mueller, Neil Smith, and Bernard Ghanem. A Benchmark and Simulator for UAV Tracking. In *ECCV*, pages 445–461, 2016. 5, 6, 7

[55] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet Large Scale Visual Recognition Challenge. *International journal of computer vision*, 115(3):211–252, 2015. 5

[56] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *CVPR*, pages 4510–4520, 2018. 5

[57] Jia Shao, Bo Du, Chen Wu, and Lefei Zhang. Tracking Objects from Satellite Videos: A Velocity Feature Based Correlation Filter. *TGRS*, 57(10):7860–7871, 2019. 1

[58] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[59] Ivan Sosnovik, Artem Moskalev, and Arnold WM Smeulders. Scale Equivariance Improves Siamese Tracking. In *WACV*, pages 2765–2774, 2021. 5

[60] Mingxing Tan and Quoc Le. Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks. In *ICML*, pages 6105–6114, 2019. 5

[61] Mani Thomas, Chandra Kambhamettu, and Cathleen A Geiger. Motion Tracking of Discontinuous Sea Ice. *TGRS*, 49(12):5064–5079, 2011. 1

[62] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning Spatiotemporal Features with 3D Convolutional Networks. In *ICCV*, pages 4489–4497, 2015. 2

[63] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A Closer Look at Spatiotemporal Convolutions for Action Recognition. In *CVPR*, pages 6450–6459, 2018. 2

[64] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 6000–6010, 2017. 4

[65] Chen Wang, Le Zhang, Lihua Xie, and Junsong Yuan. Kernel Cross-Correlator. In *AAAI*, volume 32, 2018. 5

[66] Ning Wang, Yibing Song, Chao Ma, Wengang Zhou, Wei Liu, and Houqiang Li. Unsupervised Deep Tracking. In *CVPR*, pages 1308–1317, 2019. 5, 6

[67] Ning Wang, Wengang Zhou, Qi Tian, Richang Hong, Meng Wang, and Houqiang Li. Multi-Cue Correlation Filters for Robust Visual Tracking. In *CVPR*, pages 4844–4853, 2018. 5

[68] Ning Wang, Wengang Zhou, Jie Wang, and Houqiang Li. Transformer Meets Tracker: Exploiting Temporal Context for Robust Visual Tracking. In *CVPR*, pages 1571–1580, 2021. 2, 5

[69] Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip HS Torr. Fast Online Object Tracking and Segmentation: A Unifying Approach. In *CVPR*, pages 1328–1338, 2019. 5

[70] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803, 2018. 2

[71] Yinda Xu, Zeyu Wang, Zuoxin Li, Ye Yuan, and Gang Yu. SiamFC++: Towards Robust and Accurate Visual Tracking with Target Estimation Guidelines. In *AAAI*, volume 34, pages 12549–12556, 2020. 5

[72] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning Spatio-Temporal Transformer for Visual Tracking. In *CVPR*, pages 1–10, 2021. 2

[73] Tianyu Yang and Antoni B Chan. Learning Dynamic Memory Networks for Object Tracking. In *ECCV*, pages 152–167, 2018. 2

[74] Lichao Zhang, Abel Gonzalez-Garcia, Joost van de Weijer, Martin Danelljan, and Fahad Shahbaz Khan. Learning the Model Update for Siamese Trackers. In *ICCV*, pages 4010–4019, 2019. 2, 5

[75] Le Zhang and Ponnuthurai Nagaratnam Suganthan. Robust Visual Tracking via Co-Trained Kernelized Correlation Filters. *PR*, 69:82–93, 2017. 5, 6

[76] Tianzhu Zhang, Changsheng Xu, and Ming-Hsuan Yang. Multi-Task Correlation Particle Filter for Robust Object Tracking. In *CVPR*, pages 4335–4343, 2017. 5

[77] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. In *CVPR*, pages 6848–6856, 2018. 5

[78] Zhipeng Zhang and Houwen Peng. Deeper and Wider Siamese Networks for Real-time Visual Tracking. In *CVPR*, pages 4591–4600, 2019. 2, 5

[79] Zhipeng Zhang, Houwen Peng, Jianlong Fu, Bing Li, and Weiming Hu. Ocean: Object-Aware Anchor-Free Tracking. In *ECCV*, pages 771–787, 2020. 5

[80] Zheng Zhu, Qiang Wang, Bo Li, Wei Wu, Junjie Yan, and Weiming Hu. Distractor-Aware Siamese Networks for Visual Object Tracking. In *ECCV*, pages 101–117, 2018. 5, 6