2023 IEEE International Conference on Multimedia and Expo (ICME) | 978-1-6654-6891-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICME55011.2023.00234

Towards Discriminative Representations with Contrastive Instances for Real-Time UAV Tracking

1st Dan Zeng
Southern University of Science and Technology
Shenzhen, China
zengd@sustech.edu.cn

2nd Mingliang Zou
Guilin University of Technology
Guilin,China
393038726@qq.com

3rd Xucheng Wang
Guilin University of Technology
Guilin, China
xcwang@glut.edu.cn

4th Shuiwang Li✉
Guilin University of Technology
Guilin, China
lishuiwang0721@163.com

***Abstract*—Maintaining high efficiency and high precision are two fundamental challenges in UAV tracking due to the constraints of computing resources, battery capacity, and UAV maximum load. Discriminative correlation filters (DCF)-based trackers can yield high efficiency on a single CPU but with inferior precision. Lightweight Deep learning (DL)-based trackers can achieve a good balance between efficiency and precision but performance gains are limited by the compression rate. High compression rate often leads to poor discriminative representations. To this end, this paper aims to enhance the discriminative power of feature representations from a new feature-learning perspective. Specifically, we attempt to learn more disciminative representations with contrastive instances for UAV tracking in a simple yet effective manner, which not only requires no manual annotations but also allows for developing and deploying a lightweight model. We are the first to explore contrastive learning for UAV tracking. Extensive experiments on four UAV benchmarks, including UAV123@10fps, DTB70, UAVDT and VisDrone2018, show that the proposed DRCI tracker significantly outperforms state-of-the-art UAV tracking methods.**

***Index Terms*—UAV tracking, Discriminative representation, Contrastive learning, Contrastive Instances**

I. Introduction

UAV tracking aims to infer and predict the location and scale of arbitrary objects in consecutive aerial image frames and has a broad range of potential applications in navigation, agriculture, transportation, disaster response, and public safety [1]–[5]. Compared with general object tracking, UAV tracking is challenging due to motion blur, severe occlusion, extreme viewing angle, and scale changes, making it difficult to achieve high precision. In addition, limited computing resources, low power requirements, battery capacity limitations, and the maximum load of UAVs also pose a considerable challenge to tracking efficiency [3], [4], [6].

Maintaining high efficiency and high precision are two fundamental challenges in UAV tracking. Discriminative correlation filters (DCF)-based trackers dominate in this field because of their high efficiency on a single CPU. However, their precisions are not comparable to most cutting-edge deep learning (DL)-based trackers [1], [7]–[9]. DL-based trackers are well known for their high precision, but they usually rely on complex architecture, leading to low efficiency. To combat efficiency drop, some lightweight DL-based trackers have recently been proposed for UAV tracking [3], [4], [10], [11], which mainly utilize model compressing techniques such as filter pruning to boost efficiency while maintaining high precision. Unfortunately, the filter pruning methods utilized by these works such as rank-based filter pruning [3] and Fisher pruning [4], though simple, the achieved tracking precision and efficiency are very limited and far from satisfactory. The performance limitation is because the high compression rates of these methods are prone to produce inferior discriminative representations. To this end, in this paper, we explore dealing with low performance in UAV tracking from a new feature-learning perspective to enhance the discriminative power of feature representations.

Contrastive learning is a discriminative approach that aims to learn an embedding space where similar sample pairs (aka positive pairs) stay close to each other and dissimilar ones (aka negative pairs) are far apart. It has been successfully used in many vision tasks such as image classification [12], image-to-image translation [13], text-to-image generation [14], and natural language understanding [15]. It is worth nothing that contrastive learning has also been applied to single object tracking [16], [17] and multiple object tracking [18], [19]. However, these methods usually require collecting additional annotations for positive pairs which is expensive and time-consuming [17]. Or contrastive learning of these methods is intertwined with heavy and complicated tracking frameworks [16], [18], [19], making it impossible to transfer the learning mechanism to UAV tracking. In this paper, we attempt to utilize contrastive learning for UAV tracking in a simple yet effective manner, which not only requires no manual annotations but also allows for developing and deploying a lightweight model.

Specifically, we use intra- and inter-video templates of targets as our contrastive instances to facilitate discrimina-

1945-788X/23/$31.00 ©2023 IEEE
DOI 10.1109/ICME55011.2023.00234

tive representation learning for UAV tracking. Unlike classic contrastive learning [12] where positive pairs are constructed from image augmentation, we construct positive pairs from a video. To avoid selecting hard positive samples (e.g., occluded target), we empirically randomly select 2 frames from the video to construct positive sample pairs as we observe most of the positive sample pairs are of good quality. As a result, the proposed tracker learns discriminative representations with contrastive instances (DRCI), which achieves state-of-the-art efficiency and precision compared with existing CPU-based and lightweight DL-based trackers in UAV tracking. In the inference stage, there is no additional computation burden when applying our DRCI.

To sum up, this paper makes the following contributions:

- We make the first attempt to explore contrastive learning for UAV tracking, a new feature-learning perspective to obtain lightweight DL-based trackers with better tracking precision and efficiency.
- We propose the DRCI tracker that learns discriminative representations with contrastive instances, achieving a remarkable balance between tracking efficiency and precision.
- We demonstrate the proposed method on four public UAV benchmarks. Experimental results show that the proposed DRCI tracker achieves state-of-the-art performance.

II. Related Work

A. UAV Tracking Methods

Modern trackers can be roughly divided into two categories: DCF-based trackers and DL-based trackers. The former dominates in UAV tracking with its more favorable efficiency. DCF-based trackers start with a minimum output sum of squared error (MOSSE) filter. Since then, DCF-based trackers have made great progress in many variants [6], including state-of-the-art UAV tracking methods [1], [6], [7], [20]–[22]. Despite their relatively higher efficiency, they are difficult to maintain robustness under challenging conditions due to the poor representation ability of handcrafted features.

Thanks to the powerful feature representation ability, deep learning has proven to be very successful in visual tracking in recent years. To substantially improve tracking precision and robustness, some DL-based trackers have recently been developed for UAV tracking. For instance, Cao et al. [2] proposed a hierarchical feature transformer to enable interactive fusion of spatial (shallow layers) and semantics cues (deep layers) for UAV tracking. Fu et al. [23] proposed a two-stage Siamese network-based method in which high-quality anchor proposals are generated in stage 1 and then refined in stage 2. Cao et al. [24] proposed a comprehensive framework to fully exploit temporal contexts with an adapative temporal transformer for aerial tracking. However, the efficiency of these methods is still much lower than most DCF-based trackers. To further improve the efficiency of DL-based trackers for UAV tracking, model compression techniques have been recently utilized to reduce model size [3], [4]. Unfortunately, the model compression methods used by these works, although simple, still cannot achieve satisfying tracking precision at higher compression rates. In contrast, in this paper, we explore dealing with low performance in UAV tracking from a new feature-learning perspective (i.e., contrastive learning) to enhance the discriminative power of feature representations.

B. Contrastive Learning

Contrastive learning aims at learning representations by contrasting between similar and dissimilar samples. Specifically, it attempts to bring similar samples closer together in the representation space while pushing dissimilar ones apart. It has received a great deal of attention because of its outstanding performance in the field of self-supervised learning [12]–[15]. Although contrastive learning has been deployed in many fields, until recently it was applied to multiple object tracking [18], [19] and single object tracking [16], [17]. For instance, Pang et al. [18] presented a quai-dense similarity learning that densely samples hundreds of region proposals on a pair of images for contrastive learning to exploit most informative regions on images. Yu et al. [19] proposed a trajectory-level contrastive loss to exploit the inter-frame information contained in the entire trajectory of a certain target. Wu et al. [16] proposed a progressive unsupervised learning (PUL) framework, which is the first discrimination model that learn to effectively distinguish objects from backgrounds in a contrastive learning manner. Pi et al. [17] developed instance-aware and category-aware modules to exploit different semantic levels with contrastive learning to produce robust feature embeddings. However, these methods usually require collecting additional annotations for positive pairs which is expensive and time-consuming [17]. Or contrastive learning of these methods is intertwined with heavy and complicated tracking frameworks [16], [18], [19], making it impossible to transfer the learning mechanism to UAV tracking. In this paper, we attempt to leverage contrastive learning in a simple yet effective manner to achieve more discriminative feature representations to improve both precision and efficiency of lightweight DL-based trackers for UAV tracking.

III. Learning Discriminative Representation with Contrastive Instances

A. DRCI Overview

As illustrated in Fig. 1, the proposed DRCI consists of a backbone, a neck, a head network and a discriminative representation learning (DRL) module. Specifically, the backbone network $\phi(\cdot)$ is a Siamese network, shared by the template branch and the search branch, which take template image Z and search image X as input, respectively. The neck contains four convolutional layers to adjust feature sizes. The head consists of two dense head branches followed by three convolutional layers to produce outputs for classification, quality assessment, and regression tasks. Backbone features from two branches are adjusted at the neck and then coupled with cross-

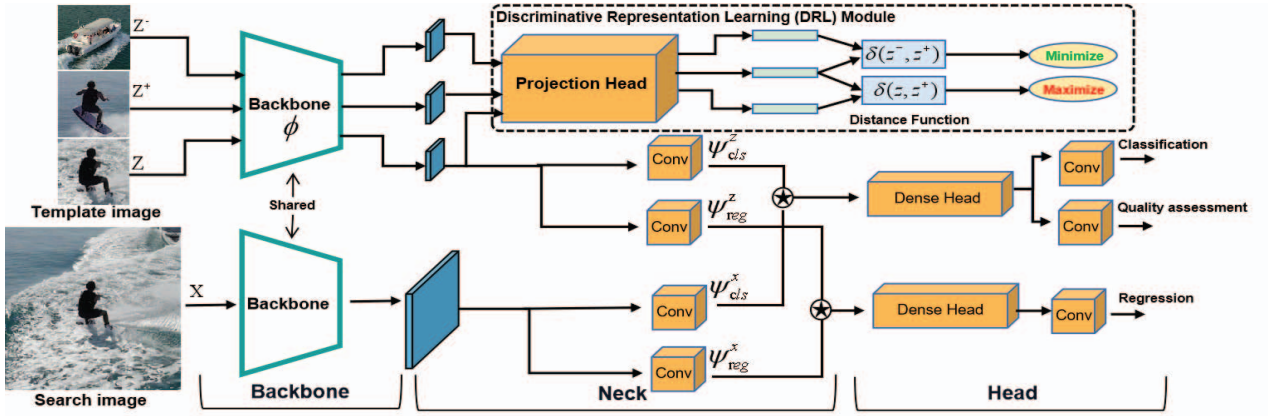

Fig. 1. An illustration of the proposed DRCI method. Note that $\psi_{cls}^{\cdot}$ and $\psi_{reg}^{\cdot}$ denote the task-specific convolutional layers for classification and regression, respectively. The template Z is taken as an anchor in our contrastive learning while Z^+ and Z^- are positive and negative samples, respectively.

correlation before they are finally fed into the classification and regression heads. The coupling features are formulated by:

$$f_l(Z,X) = E_2(\psi_l^z(\phi(Z))) \star E_2(\psi_l^x(\phi(X))), l \in \{cls, reg\}, \quad (1)$$

where $\star$ denotes the cross-correlation operation, E_2 represents the encoder for identity-related feature embedding. $\psi_{cls}^x(\cdot)$ and $\psi_{reg}^x(\cdot)$ denote the task-specific layer for classification and regression, respectively, with the same output size. $\psi_{cls}^z(\cdot)$ and $\psi_{reg}^z(\cdot)$ have the similar meaning. In the training stage, we use a DRL module to enhance the discriminative power of feature representations for UAV tracking. In the inference stage, the DRL module is removed, so there is no additional computation burden when applying our DRCI.

B. Discriminative Representation Learning (DRL)

The DRL module utilizes a project head $Proj(\cdot)$ to project the backbone features into an embedding space that the similarity of the backbone features, hopefully, can be well evaluated by a relatively simple distance function. For simplicty, we instantiate the projection head as fully connected layer followed by a ReLU activation, similar to SimCLR [12]. A more refined design of the projection head could lead to further performance improvements, which we leave for future research. To obtain instance samples for contrastive learning, we first randomly sample a minibatch of N frame pairs from N different sequences. We then crop the target templates from each frame, yielding N positive pairs and $(C_N^2 - N)$ negative constrative pairs. Denote these contrastive template samples as $\{Z_i\}_{i=1}^{2N}$, let $I \equiv \{1, ..., 2N\}$ and $j(i)$ be the index of the other sample originating from the same target, i.e., Z_i and $Z_{j(i)}$ make a positive pair, denoted by $Z_i \leftrightarrow Z_{j(i)}$. We adopt the supervised contrastive loss proposed in [25] for our discriminative representation learning, except that the negative sample pairs are pseudo or not ground truth, which takes the following form,

$$L_{DRL} = \sum_{i \in I} \frac{1}{|P(i)|} \sum_{p \in P(i)} log \frac{exp(z_i \cdot z_p/\tau)}{\sum_{a \in A(i)} exp(z_i \cdot z_a/\tau)}, \quad (2)$$

where $z_i = Proj(\phi(Z_i))$, $\cdot$ denotes the inner product, $\tau \in \mathbb{R}^+$ is a scalar temperature parameter, $A(i) = I \setminus \{i\}$, $P(i) = \{p \in A(i) : Z_p \leftrightarrow Z_i\}$ is the set of indices of all positive samples in the minibatch of i except itself, and $|P(i)|$ denotes the cardinality of $P(i)$. The DRL loss tries to increase the similarity between feature representations of the targets in the same sequence, while suppressing that of different sequences.

C. Classification, Regression and Quality Assessment Losses

The classification branch predicts the category for each location and the regression branch calculates the target bounding box for that location. The outputs of two branches are represented as $O_{h \times w \times 2}^{cls}$ and $O_{h \times w \times 4}^{reg}$, respectively, and w and h denote the width and height. Specifically, $O_{h \times w \times 2}^{cls}(i,j,:)$ is a 2D vector, representing the foreground and background scores at position (i,j). $O_{h \times w \times 4}^{reg}(i,j,:)$ is a 4D vector, representing the distances from the corresponding position to the four sides of the bounding box. At the same time, the quality assessment branch, with output being $O_{h \times w \times 1}^{qs}$, is in parallel with the classification branch to assess classification quality, which is finally used to reweight the classification score. Following P-SiamFC++ [3], the losses for learning these tasks is as follows:

$$L_{CRQ} = \frac{1}{N_{pos}} \sum_z (L_{cls}(p_z, p_z^*) + \lambda_1 I_{\{p_z^* > 0\}} L_{reg}(t_z, t_z^*) + \lambda_2 I_{\{p_z^* > 0\}} L_{qs}(q_z, q_z^*)) \quad (3)$$

where L_{cls}, L_{reg} and L_{qs} denote the focal loss, the IoU loss and the binary cross entropy loss for classification, regression and quality assessment, respectively. z represents a coordinate on a feature map, p_z is a prediction while p_z^* is the corresponding target label, $I_{\{\cdot\}}$ is the indicator function, $N_{pos} = \sum_z I_{\{p_z^* > 0\}}$. λ_1 and λ_2 are weight terms to balance the losses. Note that p_z^* is assigned 1 if z is considered a positive sample, otherwise 0 if it is considered a negative sample.

Taken together, the overall loss for training our DRCI is:

$$L = L_{CRQ} + \rho L_{DRL}, \quad (4)$$

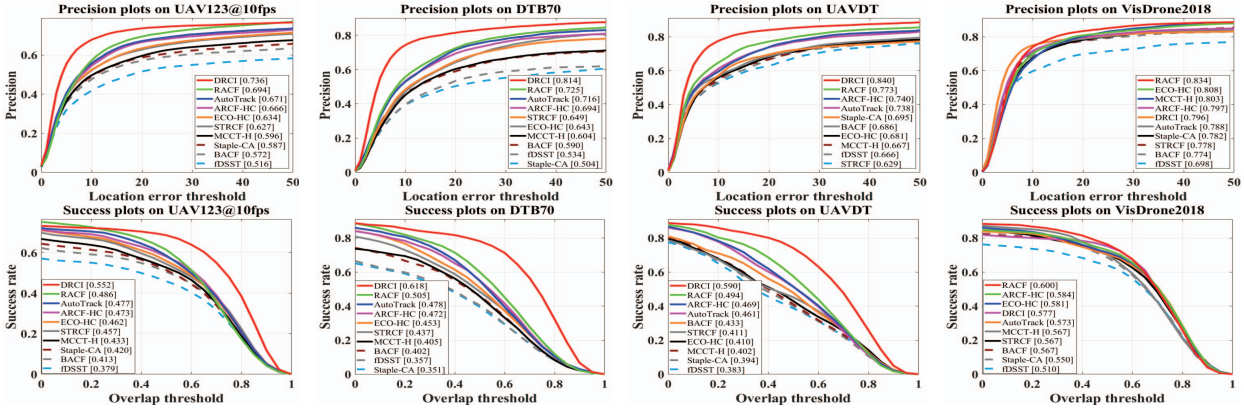

Fig. 2. Overall performance of hand-crafted based trackers on datasets, from left to right, UAV123@10fps, DTB70, UAVDT and VisDrone2018. Precision and success rate for one-pass evaluation (OPE) are used for evaluation. The precision at 20 pixels and area under curve (AUC) are used for ranking, marked in the precision plots and success plots, respectively.

TABLE I

AVERAGE PRECISION AND SPEED (FPS) COMPARISION BETWEEN DRCI AND HAND-CRAFTED BASED TRACKERS ON UAV123@10FPS, DTB70, UAVDT AND VISDRONE2018. ALL THE REPORTED FPSS ARE EVALUATED ON A SINGLE CPU. RED, BLUE AND GREEN RESPECTIVELY INDICATE THE FIRST, SECOND AND THIRD PLACE.

	KCF [26]	fDSST [27]	BACF [28]	ECO-HC [29]	STRCF [30]	ARCF-HC [7]	AutoTrack [1]	RACF [6]	**DRCI (Ours)**
Precision	53.3	60.4	64.2	68.8	67.1	71.9	**72.3**	**75.7**	**79.7**
FPS (CPU)	**622.5**	**193.4**	54.2	**84.5**	28.4	34.2	58.7	35.7	58.9

where ρ is a constant coefficient to balance L_{CRQ} and L_{DRL}.

IV. EXPERIMENTS

We conduct our experiments on four challenging UAV benchmarks, i.e., UAV123@10fps [36], DTB70 [37], UAVDT [31] and VisDrone2018 [38]. All evaluation experiments are conducted on a PC equipped with i9-10850K processor (3.6GHz), 16GB RAM and an NVIDIA TitanX GPU. The backbone, neck, and head architectures are inherited from F-SiamfC++ but with block-wise pruning ratios of 0.7, 0.5 and 0.3, respectively. The temperature parameter τ is set to 0.5. The default setting of ρ is 0.1 and other parameters such as λ_1 and λ_2 for training and inference follow P-SiamFC++. Code wiil be available on: https://github.com/DRCI2022.

A. Comparison with CPU-based Trackers

Eight state-of-the-art trackers based on hand-crafted features for comparison are: KCF [26], fDSST [27], BACF [28], ECO-HC [29], STRCF [30], ARCF-HC [7], AutoTrack [1], RACF [6].

The overall performance of DRCI with the competing trackers on the four benchmarks is shown in Fig. 2. It can be seen that DRCI outperforms all other trackers on all benchmarks except for the VisDrone2018. Specifically, on UAV123@10fps, DTB70 and UAVDT, DRCI significantly outperforms the second tracker RACF in terms of precision and AUC, with gains of (4.2%, 6.6%), (8.9%, 11.3%) and (6.7%, 9.6%), respectively. On VisDrone2018, our DRCI is inferior to the first tracker RACF in precision and AUC, the gaps are 3.8% and 2.3%, respectively. The reason is that the parameters of RACF is dataset specific, while our DRCI is not. DRCI is also slightly better than ECO-HC, MCCT-H, and ARCF-HC in precision with a max gap being 1.1%, and surpassed by ARCF-HC and ECO-HC in AUC with a max gap being 0.7%. In terms of speed, we use the average FPS over the aforementioned four benchmarks on CPU as a tracking metric. Table I illustrates the average precision and FPS produced by different trackers. It can be seen that DRCI outperforms all competing trackers in precision, and is the best real-time tracker (speed of >30FPS) on CPU. Specifically, DRCI achieves 79.7% in precision at a speed of 58.9 FPS.

B. Comparison with DL-based Trackers

The proposed DCRI is also compared with eight state-of-the-art DL-based trackers on the UAVDT dataset [38], including SiamGAT [32], HiFT [2], AutoMatch [33], SLT-SiamRPN++ [34], SparseTT [35], TCTrack [24], F-SiamFC++ [4], P-SiamFC++ [3].

The FPSs and the precisions on UAVDT are shown in Table II. As can be seen, the precision and the GPU speed of our DRCI outperform that of the competing DL-based trackers, surpassing the second tracker SparseTT [35] by 1.2% in precision, and its GPU speed is more than 6 times faster than the second tracker SparseTT [35]. This not only verifies that the proposed method can obtain a lightweight DL-based tracker with better tracking precision and efficiency, but also supports our solution to address the low performance in UAV tracking from a new feature-learning perspective, which indeed enhances the discriminative power of feature representations.

TABLE II

PRECISION AND SPEED (FPS) COMPARISON BETWEEN DRCI AND DEEP-BASED TRACKERS ON UAVDT [31]. ALL THE REPORTED FPSS ARE EVALUATED ON A SINGLE GPU. RED, BLUE AND GREEN INDICATE THE FIRST, SECOND AND THIRD PLACE.

	SiamGAT [32]	HiFT [2]	AutoMatch [33]	TCTrack [24]	F-SiamFC++ [4]	P-SiamFC++ [3]	SLT-TransT [34]	SparseTT [35]	**DRCI (Ours)**
Precision	76.4	65.2	73.8	69.6	79.4	80.7	82.9	82.8	**84.0**
FPS (GPU)	71.0	137.3	43.1	125.7	266.2	258.8	29.9	45.1	**298.3**

TABLE III

COMPARISON OF MODEL SIZE (PARAMETERS), PRECISION AND TRACKING SPEED BETWEEN THE PROPOSED DRCI AND THE BASELINE METHOD P-SIAMFC++ ON FOUR UAV BENCHMARKS. PRC IS SHORT FOR PRECISION. NOTE THAT ONLY THE PRECISION ON CPU IS SHOWN HERE SINCE THE DIFFERENCE OF PRECISION ON CPU AND GPU IS VERY SMALL.

Methods	Parameters	UAV123@10fps			DTB70			UAVDT			VisDrone2018			Avg.		
		PRC	FPS		PRC	FPS		PRC	FPS		PRC	FPS		PRC	FPS	
			CPU	GPU		CPU	GPU		CPU	GPU		CPU	GPU		CPU	GPU
P-SiamF++ [3]	7.49M	73.1	45.1	236.4	80.3	45.6	238.2	80.7	48.8	258.8	**80.9**	45.0	230.5	78.8	46.1	241.0
DRCI (Ours)	**5.05M**	**73.6**	**59.2**	**300.7**	**81.4**	**60.1**	**297.7**	**84.0**	**59.4**	**298.3**	79.6	**57.0**	**284.6**	**79.7**	**58.9**	**295.3**

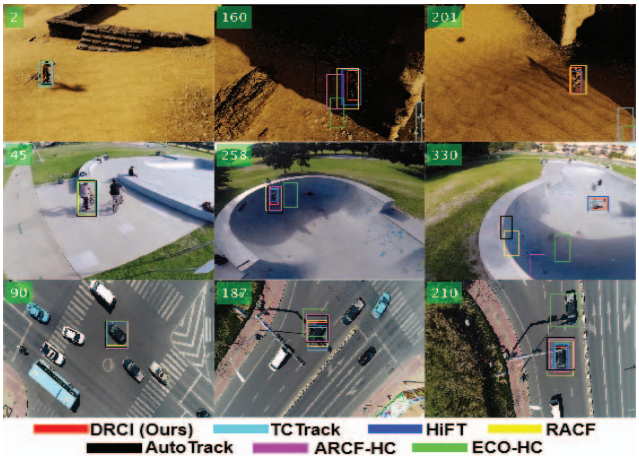

Fig. 3. Qualitative evaluation on 3 sequences from, respectively, UAV123@10fps, DTB70 and UAVDT (i.e. person1_s, BMX4 and S0309). The results of different methods are represented by different colors.

C. Qualitative Comparison with SOTA Trackers

We show some qualitative tracking results of our method and six state-of-the-art trackers in Fig. 3. As can be seen, only our tracker DRCI successfully track the targets in all three challenging examples, where the objects are experiencing illumination change (i.e., persion1_s and BMX4) or pose variations (i.e., BMX4 and S0309). Our method performs much better and is more visually pleasing in these cases, further supporting the effectiveness of the proposed method of learning discriminative representation using contrastive instances for UAV tracking.

D. Ablation Study

Effect of Discriminative Representation Learning (DRL): We compare the proposed DRCI with the baseline P-SiamFC++ on all four UAV benchmarks in terms of model size, precision and tracking speed to understand its effectiveness. Their comparisons are shown in Table III. As can be seen, the model size of DRCI is reduced to 67.4% (≈5.05/7.49) of the original. Both CPU and GPU speed have been increased. Specifically, on average, the CPU speed increased from 46.1 FPS to 58.9 FPS while the GPU speed increased from 241.0 FPS to 295.3 FPS. Although DRCI is slightly inferior to the baseline on VisDrone2018 in precision by 1.3%, the improvement on DTB70 and UAVDT is significant, specifically, with gains of 1.1% and 3.3%, respectively. These results justify that the effectiveness of using DRL (a new feature-learning perspective) to assist UAV tracking by improving both efficiency and precision.

TABLE IV

ILLUSTRATION OF HOW THE PRECISION OF DRCI ON THE FOUR BENCHMARKS VARIES WITH THE WEIGHT (I.E., ρ.) OF THE LOSS OF DISCRIMINATIVE REPRESENTATION LEARNING.

ρ	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
DTB70	80.5	81.5	80.1	78.9	79.0	80.4	78.6	78.1	78.9	77.9
UAVDT	76.2	84.0	82.7	81.9	78.9	80.8	81.8	78.9	76.5	79.5
UAV123@10fps	72.8	72.1	69.9	70.0	69.4	70.8	71.2	70.7	69.3	69.5
VisDrone2018	72.5	79.6	76.9	77.4	76.4	76.0	76.0	74.5	77.5	74.5

Impact of loss L_{DRL}: To see how the DRL loss affects the precision of DRCI, we train DRCI with different DRL loss weights and evaluate on four benchmarks. The weight ρ (refer to Eq. 4) ranges from 0.0 to 1.0 in step of 0.1. Table IV shows the precision of DRCI with different ρ on four benchmarks. Note that $\rho = 0.0$ represents the baseline tracker P-SiamFC++. It can be seen that when ρ is 0.1, DRCI achieves the best precision on four benchmarks except UAV123@10fps. Remarkably, significant improvements can be seen on UAVDT and VisDrone2018 with $\rho > 0.0$, namely imposing the proposed DRL loss, although the precision fluctuates on DTB70 and UAV123@10fps. Overall, the best precisions occur when ρ is about 0.1. This result suggests that appropriately imposing

the proposed DRL loss can help improve the precision of the baseline tracker, justifying the effectiveness of the proposed DRCL.

V. Conclusion

In this work, we are the first to explore learning discriminative representation with contrastive instances for UAV tracking, which not only requires no manual annotations but also allows for developing and deploying a lightweight model. The proposed DRCI is able to learn more effective and more compact representations, and demonstrates state-of-the-art performance on four UAV benchmarks in terms of efficiency as well as tracking precision. We believe our work will draw more attention to developing more effective and more efficient lightweighted DL-based trackers for UAV tracking.

Acknowledgment

Thanks to the supports by Guangxi Key Laboratory of Embedded Technology and Intelligent System, Research Institute of Trustworthy Autonomous Systems, the National Natural Science Foundation of China (No. 62176170, 62066042, 61971005), the Science and Technology Department of Tibet (No. XZ202102YD0018C), the Sichuan Province Key Research and Development Project (No. 2020YJ0282), and the Guangxi Science and Technology Base and Talent Special Project (No. 2021AC9330).

References

[1] Li Y. and et al., "Autotrack: Towards high-performance visual tracking for uav with automatic spatio-temporal regularization," in *CVPR,2020*, pp. 11923–11932.
[2] Cao Z. and et al., "Hift: Hierarchical feature transformer for aerial tracking," in *ICCV*, 2021, pp. 15457–15466.
[3] Wang X. and et al., "Rank-based filter pruning for real-time uav tracking," *ICME*, pp. 01–06, 2022.
[4] Wu W. and et al., "Fisher pruning for real-time uav tracking," *IJCNN*, pp. 1–7, 2022.
[5] Wang X. and et al., "Exploiting rank-based filter pruning for real-time uav tracking," *SSRN Electronic Journal*, 01 2022.
[6] Li S. and et al., "Learning residue-aware correlation filters and refining scale for real-time uav tracking," *PR*, vol. 127, pp. 108614, 2022.
[7] Huang Z. and et al., "Learning aberrance repressed correlation filters for real-time uav tracking," in *ICCV,2019*, pp. 2891–2900.
[8] Zhewen Zhang and et al., "Tracking small and fast moving objects: A benchmark," in *Asian Conference on Computer Vision*, 2022.
[9] Zhewen Zhang and et al., "Tsfmo: A benchmark for tracking small and fast moving objects," *SSRN Electronic Journal*, 2023.
[10] Liu M. and et al., "Global filter pruning with self-attention for real-time uav tracking," in *BMVC*, 2022.
[11] Zhong P. and et al., "Efficiency and precision trade-offs in uav tracking with filter pruning and dynamic channel weighting," in *FSDM*, 2022.
[12] Chen T. and et al., "A simple framework for contrastive learning of visual representations," in *ICML*, 2020, pp. 1597–1607.
[13] Park T. and et al., "Contrastive learning for unpaired image-to-image translation," in *ECCV*, 2020.
[14] H. Z and et al., "Cross-modal contrastive learning for text-to-image generation," *CVPR*, pp. 833–842, 2021.
[15] Li S. and et al., "Pair-level supervised contrastive learning for natural language inference," *ICASSP*, pp. 8237–8241, 2022.
[16] Wu Wan J. and Chan A.B., "Progressive unsupervised learning for visual object tracking," *CVPR*, pp. 2992–3001, 2021.
[17] Pi Z. and et al., "Hierarchical feature embedding for visual tracking," in *ECCV*. Springer, 2022, pp. 428–445.
[18] Pang J. and et al., "Quasi-dense similarity learning for multiple object tracking," *CVPR*, pp. 164–173, 2021.
[19] Yu E. and et al., "Multi-view trajectory contrastive learning for online multi-object tracking," *CVPR*, pp. 8824–8833, 2022.
[20] Li S. and et al., "Learning residue-aware correlation filters and refining scale estimates with the grabcut for real-time uav tracking," *3DV*, pp. 1238–1248, 2021.
[21] Li S. and et al., "Asymmetric discriminative correlation filters for visual tracking," *FITEE*, vol. 21, no. 10, pp. 1467–1484, 2020.
[22] Shuiwang Li and et al., "Equivalence of correlation filter and convolution filter in visual tracking," *ArXiv*, vol. abs/2105.00158, 2021.
[23] Fu C. and et al., "Siamese anchor proposal network for high-speed aerial tracking," *ICRA*, pp. 510–516, 2021.
[24] Cao Z. and et al., "Tctrack: Temporal contexts for aerial tracking," *CVPR*, pp. 14778–14788, 2022.
[25] Prannay K. and et al., "Supervised contrastive learning," *NIPS*, vol. 33, pp. 18661–18673, 2020.
[26] Joao F. and et al., "High-speed tracking with kernelized correlation filters," *TPAMI,2015*, vol. 37, pp. 583–596.
[27] Danelljan M. and et al., "Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking," in *CVPR,2016*, pp. 1430–1438.
[28] Hamed G. and et al., "Learning background-aware correlation filters for visual tracking," in *ICCV*, 2017.
[29] Danelljan M. and et al., "Eco: Efficient convolution operators for tracking," in *CVPR,2017*, pp. 6931–6939.
[30] Li F. and et al., "Learning spatial-temporal regularized correlation filters for visual tracking," in *CVPR,2018*, pp. 4904–4913.
[31] Du D. and et al., "The unmanned aerial vehicle benchmark: Object detection and tracking," in *ECCV,2018*, pp. 375–391.
[32] Guo D. and et al., "Graph attention tracking," in *CVPR,2021*, pp. 9543–9552.
[33] Zhang Z. and et al., "Learn to match: Automatic matching network design for visual tracking," in *ICCV,2021*, pp. 13339–13348.
[34] Kim M. and et al., "Towards sequence-level training for visual tracking," *ArXiv*, vol. abs/2208.05810, 2022.
[35] Fu Z. and et al., "Sparsett: Visual tracking with sparse transformers," *ArXiv*, vol. abs/2205.03776, 2022.
[36] Matthias M. and et al., "A benchmark and simulator for uav tracking," *FJMS,2016*, vol. 2, no. 2, pp. 445–461.
[37] Li S. and et al., "Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models.," in *AAAI,2017*, pp. 4140–4146.
[38] Wen L. and et al., "Visdrone-sot2018: The vision meets drone single-object tracking challenge results," in *ECCV*, 2018, pp. 469–495.