CHINESE OPTICS LETTERS

Vol. 20, No. 8 | August 2022

FAANet: feature-aligned attention network for real-time multiple object tracking in UAV videos

Zhenqi Liang (梁振起)[1], Jingshi Wang (王景石)[1,2], Gang Xiao (肖 刚)[1*], and Liu Zeng (曾 柳)[1]

[1] School of Aeronautics and Astronautics, Shanghai Jiao Tong University, Shanghai 200240, China

[2] Jiangsu Automation Research Institute, Lianyungang 222061, China

*Corresponding author: xiaogang@sjtu.edu.cn

Received February 5, 2022 | Accepted April 28, 2022 | Posted Online May 26, 2022

Multiple object tracking (MOT) in unmanned aerial vehicle (UAV) videos has attracted attention. Because of the observation perspectives of UAV, the object scale changes dramatically and is relatively small. Besides, most MOT algorithms in UAV videos cannot achieve real-time due to the tracking-by-detection paradigm. We propose a feature-aligned attention network (FAANet). It mainly consists of a channel and spatial attention module and a feature-aligned aggregation module. We also improve the real-time performance using the joint-detection-embedding paradigm and structural re-parameterization technique. We validate the effectiveness with extensive experiments on UAV detection and tracking benchmark, achieving new state-of-the-art 44.0 MOTA, 64.6 IDF1 with 38.24 frames per second running speed on a single 1080Ti graphics processing unit.

Keywords: multiple object tracking; unmanned aerial vehicle; feature alignment; deep learning.

DOI: 10.3788/COL202220.081101

1. Introduction

Object detection and tracking have been crucial concerns in the 2D[1,2] and 3D[3,4] vision community. Multiple object tracking (MOT) aims to analyze videos to identify and track objects. Because of the flexibility of unmanned aerial vehicles (UAVs) equipped with cameras, MOT in UAV videos has become a new research hotspot and trend in recent years.

Online MOT algorithms can be reduced to the two-step and one-shot approaches. The two-step approach[1,2] mainly follows the tracking-by-detection (TBD) paradigm[5]. It formulates the MOT task as two steps of object detection and data association. The one-shot approach[6,7] joints detection and embedding learning, and does not separate the re-identification (Re-ID) model as usual in TBD. As of now, almost all UAV-based MOT algorithms adopt the TBD paradigm. Different from the common observation perspectives in MOT, such as fixed security cameras and moving cars, MOT in UAV videos must pay more attention to the following challenges. (1) Scale changes and small objects: the flight altitude of the UAV changes with time, and the object scale varies greatly at different altitudes. Because of the wide field of view and high altitude, the objects are usually small. (2) Real-time performance: MOT in UAV videos needs to locate fast moving ground objects, so algorithms should be fast enough to be used in industrial applications.

Previous efforts have been made to solve these problems. IPGAT[8] predicts complex motions using a conditional generative adversarial networks model. However, it neglects to utilize multi-scale appearance information of an object. M-CMSN-M[9] unifies single object tracking and MOT for multi-task learning using a Siamese network. Nevertheless, the speed of M-CMSN-M[9] is extremely slow due to the numerous objects in UAV videos.

In this Letter, we are committed to coping with both aforementioned problems simultaneously. To address the problem of scale changes, we design a feature-aligned attention network (FAANet), which is mainly composed of two modules: the channel and spatial attention (CSA) module and feature-aligned aggregation (FAA) module. The CSA module adaptively enhances multi-scale features, and the FAA module successively generates alignment bias of two different resolution features. FAANet integrates multi-scale features to improve the robustness of object scale changes. To address the problem of real-time, we adopt the joint-detection-embedding (JDE) paradigm[6] and adopt the structural re-parameterization technique[10] to increase the network inference speed.

The major contributions of this paper are summarized as follows. (1) We propose an FAANet to enhance and aggregate multi-scale features so as to cope with drastic scale changes in UAV videos. (2) We introduce the JDE paradigm to MOT in UAV videos and use the structural re-parameterization technique to increase the network inference speed. (3) Extensive experiments are conducted on UAV detection and tracking

(UAVDT)[9] benchmarks to verify effectiveness of the proposed method.

2. Methods

In this section, we first present the architecture of the proposed FAANet and then explain the details of the CSA module, FAA module, and online inference.

2.1. Overview

As shown in Fig. 1, the framework of proposed FAANet contains four components: feature extractor backbone, feature fusion neck, detection and Re-ID prediction heads, and online tracking association. We adopt RepVGG[10] as our backbone to extract multi-scale features with minor modifications to their channels and layers. Let the shape of the input image be $3 \times H_{\text{input}} \times H_{\text{input}}$; then the shape of multi-scale output features is, respectively, $C_i \times H_i \times W_i$, where $C_i = 64 \times 2^{i-1}$, $H_i = H_{\text{input}}/2^{i+1}$, $W_i = W_{\text{input}}/2^{i+1}$, and $i = 1, 2, 3, 4$. We adopt the same detection and Re-ID heads as FairMOT[7], including heat map, box size, center offset, and Re-ID embeddings.

2.2. Channel and spatial attention module

In general, the input and output dimensions of the channel attention (CA) or spatial attention (SA) are the same[11,12]. In order to improve real-time performance and reduce the risk of overfitting, we propose a method combining feature dimension reduction, channel attention, and spatial attention, as shown in Fig. 2. We perform channel attention first, and then spatial attention. We refer to the effective channel attention mechanism[11] and modify it by 2D 1×1 convolution and one-dimensional convolution with variant kernel and stride.

Specifically, let the one output of the backbone be $F \in R^{C \times H \times W}$, where C, H, and W are channel dimension, height, and width. Accordingly, the weights of channels can be computed as

$$w_{\text{channel}} = \sigma(\text{Conv1d}(g(F))), \tag{1}$$

where $g(F) = \frac{1}{WH}\sum_{i,j=1}^{W,H} F_{i,j}$ is the channel-wise global average pooling (GAP), and σ is the sigmoid nonlinear activation function. We set the Conv1d with a kernel as 3, 7, 11, 15 and stride as 1, 2, 4, 8, respectively, in four multi-scale features. In this way, all channel dimensions of multi-scale features can be reduced to 64. Subsequently, the output of channel attention can be computed as

$$F_{\text{channel}} = w_{\text{channel}} \odot \text{Conv2d}(F) + \text{Conv2d}(F), \tag{2}$$

where $\odot$ denotes element-wise product. The output dimension of Conv2d(1×1) is 64. Specially, for $i = 1$, $C = 64$, we replace Conv2d(1×1) with an identity function.

In contrast to channel attention, which focuses on channel dimension, spatial attention mainly focuses on the height and width dimension. Inspired by polarized self-attention[12], we

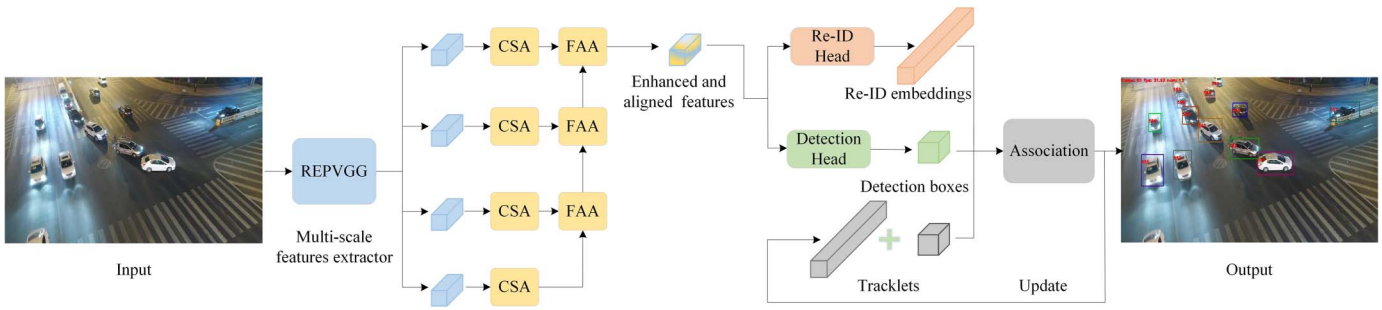

Fig. 1. Architecture of our tracker FAANet tracking framework. This framework contains four components: backbone (RepVGG), neck (CSA + FAA), head (Re-ID + detection), and association.

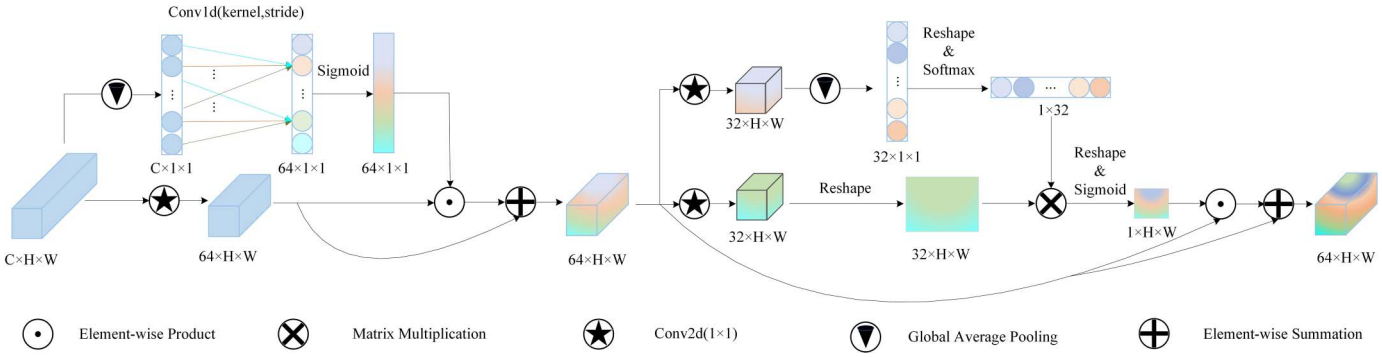

Fig. 2. Architecture of CSA module.

also adopt the polarized method to perform spatial attention. Specifically, let the output of channel attention be $F_{\text{channel}} \in R^{64\times H\times W}$, so the spatial weights can be computed by the following formulas:

$$F_1 = \text{soft max}(f_1(g(\text{Conv2d}(F_{\text{channel}})))), \tag{3}$$

$$F_2 = f_2(\text{Conv2d}(F_{\text{channel}})), \tag{4}$$

$$w_{\text{spatial}} = \sigma(f_3(F_1 \otimes F_2)), \tag{5}$$

where f_1, f_2, and f_3 denote different reshape operation, and softmax(·) denotes the softmax function. After calculating the spatial weights, the final output of the CSA module can be computed as

$$F_{\text{spatial}} = w_{\text{spatial}} \odot F_{\text{channel}}, \tag{6}$$

$$F_{\text{output}} = F_{\text{channel}} + F_{\text{spatial}}. \tag{7}$$

Because the channel attention adopts one-dimensional convolution instead of the full connection as usual, the number of parameters is greatly reduced, and the inference speed is improved. The number of channels is reduced to 64 dimensions through the proposed CSA module, which also reduces the number of parameters for real-time performance and provides a channel consistent input to the FAA module.

2.3. Feature-aligned aggregation module

Traditional multi-scale feature aggregation usually adopts the method of bi-linear interpolation up-sampling and element-size summation. However, because of the feature misalignment, feature degradation will occur, which leads to the degradation of the ability to locate objects at different scales. After our meticulous research, we refer to the feature-aligned mechanism in AlignSeg[13] and apply it to the UAV-based MOT domain.

The FAA module is shown in Fig. 3. Specifically, let the corresponding two outputs of the CSA module be $F_{\text{high}} \in R^{64\times H\times W}$ and $F_{\text{low}} \in R^{64\times (H/2)\times (W/2)}$, where F_{high} and F_{low}, respectively, denote the feature map with high and low resolution. The network will learn to generate the offsets of two feature maps for alignment. Let the corresponding two offsets be $\Delta_{\text{high}} \in R^{64\times H\times W}$ and $\Delta_{\text{low}} \in R^{64\times (H/2)\times (W/2)}$; then they can be computed by the following formulas:

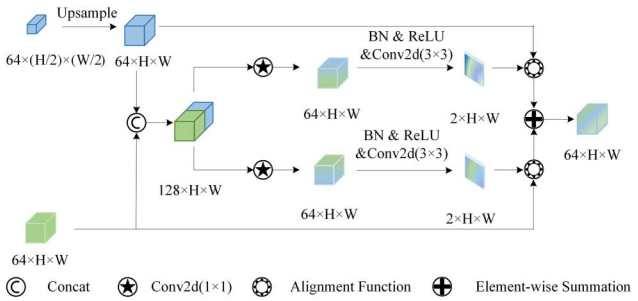

Fig. 3. Architecture of FAA module.

$$F_{\text{concat}} = \text{Concat}(\text{upsample}(F_{\text{low}}), F_{\text{high}}), \tag{8}$$

$$\Delta_{\text{low}} = \text{Conv2d}_{3\times 3}(\text{ReLU}(\text{BN}(\text{Conv2d}_{1\times 1}(F_{\text{concat}})))), \tag{9}$$

$$\Delta_{\text{high}} = \text{Conv2d}_{3\times 3}(\text{ReLU}(\text{BN}(\text{Conv2d}_{1\times 1}(F_{\text{concat}})))), \tag{10}$$

where Concat(F,F) denotes the concatenation operation of two feature maps along the channel dimension, upsample(F) denotes the bilateral interpolation function, $\text{Conv2d}_{3\times 3}$ and $\text{Conv2d}_{1\times 1}$ denote the convolution layer with 3×3 and 1×1 kernels, and ReLU and BN, respectively, denote the rectified linear unit activation function and batch normalization layer. Let the output of the FAA module be F_{output}; then F_{output} can be computed as

$$F_{\text{output}} = f(\text{upsample}(F_{\text{low}}), \Delta_{\text{low}}) + f(F_{\text{high}}, \Delta_{\text{high}}), \tag{11}$$

where $f(F,\Delta)$ is defined in AlignSeg[13]. It can be described like this: suppose $A_{h,w}$ is the output of the alignment function $f(F,\Delta)$ in the spatial coordinates for position (h,w); then $A_{h,w}$ can be computed as

$$A_{h,w} = \sum_{h'=1}^{H}\sum_{w'=1}^{W} F_{h',w'} \cdot \max(0, 1 - |h + \Delta_{1\,hw} - h'|) \cdot \max(0, 1 - |w + \Delta_{2\,hw} - w'|), \tag{12}$$

where $\Delta_{1\,hw}$ and $\Delta_{2\,hw}$ indicate the learned 2D transformation offsets for position (h,w).

2.4. Online inference

Herein, the two important components of online inference are data association and structural re-parameterization, which we will illustrate further.

We follow the standard online tracking algorithm to associate boxes. As shown in Fig. 4, we first initialize a few tracklets based on the estimated boxes in the first frame and use a Kalman filter to predict the locations of the tracklets in the next frame. We perform two Hungarian matchings between detections and tracklets sequentially. The first matching considers the appearance information (Re-ID embedding) measured by cosine distance and the motion information measured by Mahalanobis distance. The second matching only considers intersection over union (IOU) distance, which is simple but useful. Finally, we

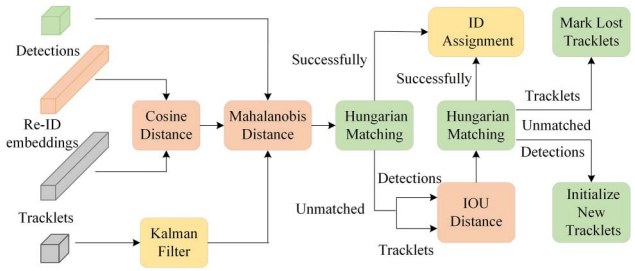

Fig. 4. Procedure of association between detections and tracklets.

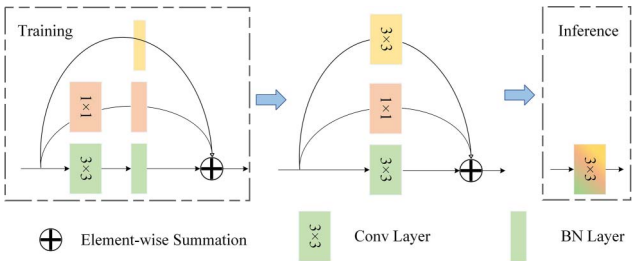

Fig. 5. Structural re-parameterization of a RepVGG block.

initialize new tracklets that meet confidence thresholds and mark the lost tracklets.

Structural re-parameterization is used in RepVGG[10] to decouple a multi-branch topology with a plain architecture. Herein, we also utilize structural re-parameterization to increase the backbone network inference speed. As shown in Fig. 5, during the training phase, multi-branch topology is adopted to avoid the problem like gradient vanishing. After training, we perform the transformation with simple algebra in RepVGG[10] and save the plain backbone architecture for online inference.

3. Experiment

3.1. Datasets and metrics

We evaluate our method on the dataset UAVDT[9]. The dataset offers 50 sequences recorded from a UAV (60% for training and 40% for testing). We compare our method with various classic and recent algorithms on the testing sequences. We use multiple metrics, including MOT accuracy (MOTA), identification F1 score (IDF1), MOT precision (MOTP), mostly tracked targets (MT), mostly lost targets (ML), false positive (FP), false negative (FN), identification switches (IDS), and fragmented (FM) to evaluate different aspects of the tracking performance. MOTA is computed based on FP, FN, and IDS. Considering the amount of FP and FN is larger than that of IDS, MOTA focuses more on the detection performance. IDF1 focuses more on the association performance.

3.2. Implementation details

We choose RepVGG[10] as our backbone. Its weights are initialized by the ImageNet-pretrained model. Specifically, we choose a slightly modified RepVGG-B0 as our default backbone, which has [2,4,6,16] blocks and [64,128,256,512] output channels. Our model is trained and tested on a NVIDIA GEFORCE RTX 1080Ti graphics processing unit. We adopt standard data augmentation techniques including scaling, rotation and color jittering. The input image is resized to 1024×544, and the feature map resolution is 256×136. We choose Adam as our optimizer. We set the starting learning rate as 8×10^{-5} and batch size as 10. The learning rate will decrease by a factor 10 per 15 epochs. The training step takes about 18 h with 35 epochs in total.

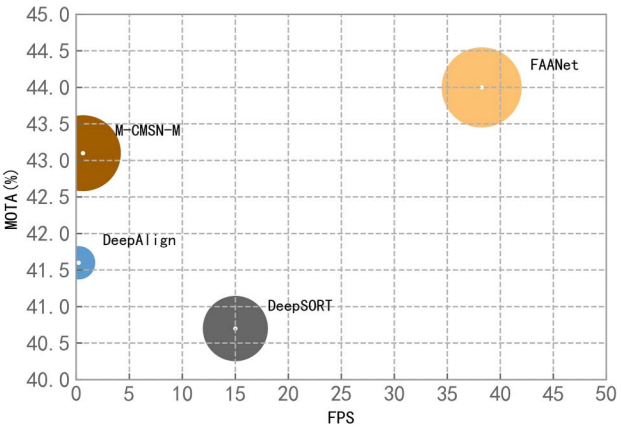

Fig. 6. MOTA-IDF1-FPS comparison with other UAV-based MOT trackers on the UAVDT test dataset. The horizontal axis is FPS, the vertical axis is MOTA, and the radius of the circle is IDF1.

3.3. Experiment analysis

We evaluate our FAANet together with various classic and recent algorithms including CEM[14], CMOT[15], SORT[1], DeepSORT[2], GOG[16], IOUT[17], MDP[18], SMOT[19], DeepAlign[20], SBMA[21], IPGAT[8], M-CMSN-M[9], and Quadruplet[22]. All the results of the comparison algorithms are obtained from recent published papers and the UAVDT benchmark[9].

Since most of the current UAV-based MOT algorithms follow the TBD paradigm, many algorithms do not publish their own speed or only publish the speed of the association phase. That leads to difficulty and ambiguity in speed comparison of algorithms. As much as we can, we collect the currently publicly available algorithm speed and performance, which are shown in Fig. 6. Note that the three algorithms in comparison only calculate the time consumption in the association phase. It can be observed that the speed of our FAANet is 60 times higher than that of the current state-of-the-art M-CMSN-M[9], while MOTA and IDF1 are slightly ahead. Detailed comparison between algorithms is shown in Table 1. We list the seven kinds of the most excellent trackers in recent years. We have the best performance on the 7 out of 10 metrics.

The comparison based on scene attributes is shown in Fig. 7. Our algorithm performs better than other algorithms in the high-alt, bird-view, and fog scenes. It demonstrates the effectiveness of the proposed module.

3.4. Ablation experiments

To validate the effectiveness of CA, SA, and FAA modules, we introduce a baseline RepVGG-B0 with a re-parameterization technique. The baseline reduces the feature dimension by 1×1 convolution and performs multi-scale fusion by bi-linear interpolation up-sampling and element-size summation. Except for the above, all the other settings are consistent, such as training hyperparameters and association details in tracking. As shown in Table 2, the contributions of CA and SA are similar,

Table 1. Results of a Quantitative Comparison among Classic MOT Methods and Recent UAV-Based Methods on the UAVDT Test Dataset[a].

MOT Methods	Year	Framework	MOTA ↑	IDF1 ↑	MOTP ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓	FM ↓	FPS ↑
SORT[1]	2016	Faster RCNN	39.0	43.7	74.3	33.9	28.0	33,037	172,628	2350	5787	Nan
DeepSORT[2]	2017	Faster RCNN	40.7	58.2	73.2	41.7	23.7	44,868	155,290	2061	6432	15.01
DeepAlign[20]	2018	Faster RCNN	41.6	49.0	73.3	43.7	24.3	45,420	152,224	1546	**3733**	0.23
SBMA[21]	2019	LSTM	38.6	48.5	72.1	38.9	24.4	44,724	160,950	3489	11,796	Nan
IPGAT[8]	2020	LSTM + CGAN	39.0	49.4	72.2	37.4	25.2	42,135	163,837	2091	10,057	Nan
M-CMSN-M[9]	2020	Faster RCNN	43.1	62.6	73.5	45.3	22.7	45,900	147,638	**390**	4259	0.64
Quadruplet[22]	2021	Faster RCNN	40.3	55.0	74.0	Nan	Nan	**30,065**	150,837	1091	3057	Nan
FAANet	Nan	RepVGG + JDE	**44.0**	**64.6**	**77.9**	**47.9**	**22.6**	57,146	**133,496**	403	7202	**38.24**

[a]The best performers are highlighted in bold.

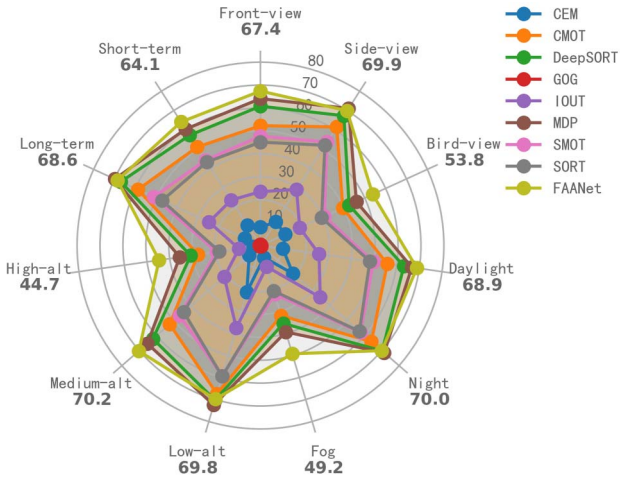

Fig. 7. IDF1 comparison with other UAV-based MOT trackers on the UAVDT test dataset based on scene attributes. The IDF1 of FAANet is marked outside the circle.

Table 2. Evaluation of the Critical Factors in FAANet[a].

RepVGG-B0	CA	SA	FAA	MOTA ↑	IDF1 ↑	FPS ↑
√				38.2	56.8	**45.70**
√	√			39.7	59.2	43.52
√		√		39.3	59.4	43.41
√	√	√		40.4	60.2	41.35
√			√	42.1	63.7	40.54
√	√	√	√	**44.0**	**64.6**	38.24

[a]The best performers are highlighted in bold.

Table 3. The Improvement of Re-parameterization Technique.

Rep	Params (10^6)	FLOPs (10^9)	MOTA ↑	IDF1 ↑	FPS ↑
	15.9	62.3	44.0	64.6	30.32
√	14.4	58.3	44.0	64.6	38.24

but their combination can lead to better performance. Compared with CSA, FAA improves more IDF1, which may be due to the feature-aligned multi-scale aggregation boosting robustness to small objects and scale changes.

As shown in Table 3, we illustrate the speed improvement of the re-parameterization technique. It decreases the number of model parameters from 15.9×10^6 to 14.4×10^6 and the amount of floating-point operations (FLOPs) from 62.3×10^9 to 58.3×10^9. Generally, it increases frames per second (FPS) from 30.32 to 38.24. This is a 26% speed improvement without degeneration of any accuracy performance.

3.5. Visualization results

Figure 8 visualizes several typical scenes tracking comparison results between DeepSORT[2] and FAANet on the UAVDT[9] test dataset. From the results of M0403, M1301, and M1004, we can see that FAANet can better track small objects at the end of the road, which is caused by the wide field of vision. From the results of M0701, we can see that FAANet performs better in scenes of scale changes, which is caused by flight altitude.

The vertical numbers denote the number of objects tracked in the three frames. On average, FAANet can track 29% more objects than the classical DeepSORT[2] method in the four typical scene examples. This is mainly attributed to the multi-scale feature enhancement and fusion of the CSA and FAA modules.

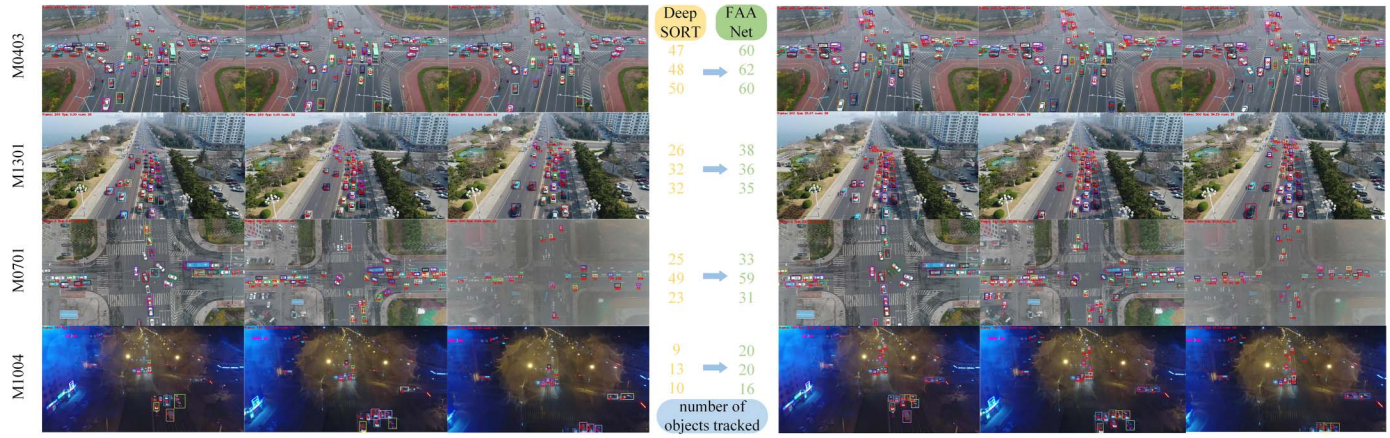

Fig. 8. Examples and comparison of tracking results between DeepSORT and FAANet on the UAVDT test dataset.

4. Conclusions

In this Letter, we propose an FAANet for MOT in UAV videos. Experimental results demonstrate that our methods can better cope with the problem of scale changes and small object with real-time speed. We hope that our method is attractive for application to industry due to its high accuracy and fast speed.

Acknowledgement

This work was supported by National Program on Key Basic Research Project (No. 2014CB744903), National Natural Science Foundation of China (Nos. 61673270 and 61973212), and Key Technology Research Program of Sichuan Provincial Department of Science and Technology (No. 2020YFSY0027).

References

1. A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and real-time tracking," in *IEEE International Conference on Image Processing* (2016), p. 3464.
2. N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *IEEE International Conference on Image Processing* (2017), p. 3645.
3. Q. Qian, Y. Hu, N. Zhao, M. Li, F. Shao, and X. Zhang, "Object tracking method based on joint global and local feature descriptor of 3D LIDAR point cloud," Chin. Opt. Lett. **18**, 061001 (2020).
4. J. Dai, L. Huang, K. Guo, L. Ling, and H. Huang, "Reflectance transformation imaging of 3D detection for subtle traces," Chin. Opt. Lett. **19**, 031101 (2021).
5. D. Ramanan and D. A. Forsyth, "Finding and tracking people from the bottom up," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2003), p. 1.
6. Z. Wang, L. Zheng, Y. Liu, Y. Li, and S. Wang, "Towards real-time multi-object tracking," in *European Conference on Computer Vision* (2020), p. 107.
7. Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "FairMOT: on the fairness of detection and re-identification in multiple object tracking," Int. J. Comput. Vis. **129**, 3069 (2021).
8. H. Yu, G. Li, L. Su, B. Zhong, and Q. Huang, "Conditional GAN based individual and global motion fusion for multiple object tracking in UAV videos," Pattern Recognit. Lett. **131**, 219 (2020).
9. H. Yu, G. Li, W. Zhang, Q. Huang, D. Du, Q. Tian, and N. Sebe, "The unmanned aerial vehicle benchmark: object detection, tracking and baseline," Int. J. Comput. Vis. **128**, 1141 (2020).
10. X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, and J. Sun, "RepVGG: making VGG-style ConvNets great again," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2021), p. 13728.
11. Q. Wang, B. Wu, P. Zhu, P. Li, and Q. Hu, "ECA-Net: efficient channel attention for deep convolutional neural networks," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2020), p. 11531.
12. H. Liu, F. Liu, X. Fan, and D. Huang, "Polarized self-attention: towards high-quality pixel-wise regression," arXiv:210700782 (2021).
13. Z. Huang, Y. Wei, X. Wang, H. Shi, and T. S. Huang, "AlignSeg: feature-aligned segmentation networks," IEEE Trans. Pattern Anal. Mach. Intell. **44**, 550 (2021).
14. A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," IEEE Trans. Pattern Anal. Mach. Intell. **36**, 58 (2013).
15. S. H. Bae and K. J. Yoon, "Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2014), p. 1218.
16. H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2011), p. 1201.
17. E. Bochinski, V. Eiselein, and T. Sikora, "High-speed tracking-by-detection without using image information," in *IEEE International Conference on Advanced Video and Signal Based Surveillance* (2017), p. 1.
18. Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: online multi-object tracking by decision making," in *IEEE International Conference on Computer Vision* (2015), p. 4705.
19. C. Dicle, O. I. Camps, and M. Sznaier, "The way they move: tracking multiple targets with similar appearance," in *IEEE International Conference on Computer Vision* (2013), p. 2304.
20. Q. Zhou, B. Zhong, Y. Zhang, J. Li, and Y. Fu, "Deep alignment network based multi-person tracking with occlusion and motion reasoning," IEEE Trans. Multimed. **21**, 1183 (2018).
21. H. Yu, G. Li, W. Zhang, H. Yao, and Q. Huang, "Self-balance motion and appearance model for multi-object tracking in UAV," in *Proceedings of the ACM Multimedia Asia* (2019), p. 1.
22. H. U. Dike and Y. Zhou, "A robust quadruplet and faster region-based CNN for UAV video-based multiple object tracking in crowded environment," Electronics **10**, 795 (2021).