VisDrone-VDT2018: The Vision Meets Drone Video Detection and Tracking Challenge Results

Pengfei Zhu[1(✉)], Longyin Wen[2], Dawei Du[3], Xiao Bian[4], Haibin Ling[5], Qinghua Hu[1], Haotian Wu[1], Qinqin Nie[1], Hao Cheng[1], Chenfeng Liu[1], Xiaoyu Liu[1], Wenya Ma[1], Lianjie Wang[1], Arne Schumann[9], Dan Wang[11], Diego Ortego[16], Elena Luna[16], Emmanouil Michail[6], Erik Bochinski[17], Feng Ni[7], Filiz Bunyak[14], Gege Zhang[11], Guna Seetharaman[15], Guorong Li[13], Hongyang Yu[12], Ioannis Kompatsiaris[6], Jianfei Zhao[8], Jie Gao[11], José M. Martínez[16], Juan C. San Miguel[16], Kannappan Palaniappan[14], Konstantinos Avgerinakis[6], Lars Sommer[9,10], Martin Lauer[10], Mengkun Liu[11], Noor M.Al-Shakarji[14], Oliver Acatay[9], Panagiotis Giannakeris[6], Qijie Zhao[7], Qinghua Ma[11], Qingming Huang[13], Stefanos Vrochidis[6], Thomas Sikora[17], Tobias Senst[17], Wei Song[11], Wei Tian[10], Wenhua Zhang[11], Yanyun Zhao[8], Yidong Bai[11], Yinan Wu[11], Yongtao Wang[7], Yuxuan Li[11], Zhaoliang Pi[11], and Zhiming Ma[10]

[1] Tianjin University, Tianjin, China
zhupengfei@tju.edu.cn
[2] JD Finance, Mountain View, CA, USA
[3] University at Albany, SUNY, Albany, NY, USA
[4] GE Global Research, Niskayuna, NY, USA
[5] Temple University, Philadelphia, PA, USA
[6] Centre for Research and Technology Hellas, Thessaloniki, Greece
[7] Peking University, Beijing, China
[8] Beijing University of Posts and Telecommunications, Beijing, China
[9] Fraunhofer IOSB, Karlsruhe, Germany
[10] Karlsruhe Institute of Technology, Karlsruhe, Germany
[11] Xidian University, Xi'an, China
[12] Harbin Institute of Technology, Harbin, China
[13] University of Chinese Academy of Sciences, Beijing, China
[14] University of Missouri-Columbia, Columbia, USA
[15] U.S. Naval Research Laboratory, Washington, D.C., USA
[16] Universidad Autónoma de Madrid, Madrid, Spain
[17] Technische Universität Berlin, Berlin, Germany

Abstract. Drones equipped with cameras have been fast deployed to a wide range of applications, such as agriculture, aerial photography, fast delivery, and surveillance. As the core steps in those applications, video object detection and tracking attracts much research effort in recent years. However, the current video object detection and tracking algorithms are not usually optimal for dealing with video sequences captured by drones, due to various challenges, such as viewpoint change and scales.

L. Leal-Taixé and S. Roth (Eds.): ECCV 2018 Workshops, LNCS 11133, pp. 496–518, 2019.
https://doi.org/10.1007/978-3-030-11021-5_29

To promote and track the development of the detection and tracking algorithms with drones, we organized the Vision Meets Drone Video Detection and Tracking (VisDrone-VDT2018) challenge, which is a subtrack of the Vision Meets Drone 2018 challenge workshop in conjunctioStefanoshe 15th European Conference on Computer Vision (ECCV 2018). Specifically, this workshop challenge consists of two tasks, (1) video object detection, and (2) multi-object tracking. We present a large-scale video object detection and tracking dataset, which consists of 79 video clips with about 1.5 million annotated bounding boxes in 33,366 frames. We also provide rich annotations, including object categories, occlusion, and truncation ratios for better data usage. Being the largest such dataset ever published, the challenge enables extensive evaluation, investigation and tracking the progress of object detection and tracking algorithms on the drone platform. We present the evaluation protocol of the VisDrone-VDT2018 challenge and the results of the algorithms on the benchmark dataset, which are publicly available on the challenge website: http://www.aiskyeye.com/. We hope the challenge largely boost the research and development in related fields.

Keywords: Drone · Benchmark
Object detection in videos · Multi-object tracking

1 Introduction

Developing autonomous drone systems that are helpful for humans in everyday tasks, *e.g.*, agriculture, aerial photography, fast delivery, and surveillance, is one of the grand challenges in computer science. An example is autonomous drone systems that can help farmers to spray pesticide regularly. Consequently, automatic understanding of visual data collected from these platforms become highly demanding, which brings computer vision to drones more and more closely. Video object detection and tracking are the critical steps in those applications, which attract much research in recent years.

Several benchmark datasets have been proposed in video object detection and tracking, such as ImageNet-VID [43] and UA-DETRAC [30,51] for object detection in videos, and KITTI [16] and MOTChallenge [25] for multi-object tracking, to promote the developments in related fields. The challenges in those datasets are quite different from that on drones for the video object detection and tracking algorithms, such as large viewpoint change and scales. Thus, these algorithms in video object detection and tracking are not usually optimal for dealing with video sequences generated by drones. As pointed out in recent studies (*e.g.*, [20,34]), autonomous video object detection and tracking is seriously limited by the lack of public large-scale benchmarks or datasets. Some recent preliminary efforts [20,34,42] have been devoted to construct datasets captured using a drone platform, which are still limited in size and scenarios covered, due to the difficulties in data collection and annotation. Thus, a more general and comprehensive benchmark is desired to further boost research on computer

vision problems with drone platform. Moreover, thorough evaluations of existing or newly developed algorithms remains an open problem.

To this end, we organized a challenge workshop, "Vision Meets Drone Video Object Detection and Tracking" (VisDrone-VDT2018), which is a part of the "Vision Meets Drone: A Challenge" (VisDrone2018) on September 8, 2018, in conjunction with the 15th European Conference on Computer Vision (ECCV 2018) in Munich, Germany. This challenge focuses on two tasks, *i.e.*, (1) video object detection and (2) multi-object tracking, which are described as follows.

- **Video object detection** aims to detect objects of a predefined set of object categories (*e.g.*, pedestrian, car, and van) from videos taken from drones.
- **Multi-object tracking** aims to recover the object trajectories in video sequences.

We collected a large-scale video object detection and tracking dataset with several drone models, *e.g.*, DJI Mavic, Phantom series 3, and 3A, in various scenarios, which are taken at different locations, but share similar environments and attributes.

We invite researchers to submit the results of algorithms on the proposed VisDrone-VDT2018 dataset, and share their research at the workshop. We also present the evaluation protocol of the VisDrone-VDT2018 challenge, and the results of a comparison of the submitted algorithms on the benchmark dataset, on the challenge website: www.aiskyeye.com/. The authors of the submitted algorithms have an opportunity to publish the source code at our website, which will be helpful to track and boost research on video object detection and tracking with drones.

2 Related Work

2.1 Existing Datasets and Benchmarks

The ILSVRC 2015 challenge [43] opens the "object detection in video" track, which contains a total of 3,862 snippets for training, 555 snippets for validation, and 937 snippets for testing. YouTube-Object dataset [37] is another large-scale dataset for video object detection, which consists of 155 videos with over 720,152 frames for 10 classes of moving objects. However, only 1,258 frames are annotated with a bounding-box around an object instance. Based on this dataset, Kalogeiton *et al.* [23] further provide the annotations of instance segmentation[1] for the YouTube-Object dataset.

Multi-object tracking is a hot topic in computer vision with many applications, such as surveillance, sport video analysis, and behavior analysis. Several datasets are presented to promote the developments in this field. The MOTChallenge team[2] release a series of datasets, *i.e.*, MOT15 [25], MOT16 [31], and MOT17 [1], for multi-pedestrian tracking evaluation. Wen *et al.* [51] collect

[1] http://calvin.inf.ed.ac.uk/datasets/youtube-objects-dataset/.
[2] https://motchallenge.net/.

the UA-DETRAC dataset for multi-vehicle detection and tracking evaluation, which contains 100 challenging videos captured from real-world traffic scenes (over 140,000 frames with rich annotations, including illumination, vehicle type, occlusion, truncation ratio, and vehicle bounding boxes). Recently, Du *et al.* [12] construct a UAV dataset with approximate 80,000 fully annotated video frames as well as 14 different kinds of attributes (*e.g.*, weather condition, flying altitude, vehicle category, and occlusion) for object detection, single-object tracking, and multi-object tracking evaluation. We summarize the related datasets in Table 1.

2.2 Brief Review of Video Object Detection Methods

Object detection has achieved significant improvements in recent years, with the arriving of convolutional neural networks (CNNs), such as R-CNN [17], Faster-RCNN [40], YOLO [38], SSD [29], and RefineDet [57]. However, the aforementioned methods focus on detecting objects in still images. The object detection accuracy in videos suffers from appearance deterioration that are seldom observed in still images, such as motion blur, video defocus, etc. To that end, some previous methods are designed to detect specific classes of objects from videos, such as pedestrians [49] and cars [26]. Kang *et al.* [24] develop a multi-stage framework based on deep CNN detection and tracking for object detection in videos in [43], which uses a tubelet proposal module to combine object detection and tracking for tubelet object proposal, and a tubelet classification and re-scoring module to incorporate temporal consistency. The Seq-NMS method [18] uses high-scoring object detections from nearby frames to boost scores of weaker detections within the same clip to improve the video detection accuracy. Zhu [59] design an end-to-end learning framework for video object detection based on flow-guided feature aggregation and temporal coherence. Galteri *et al.* [14] connect detectors and object proposal generating functions to exploit the ordered and continuous nature of video sequences in a closed-loop. Bertasius *et al.* [5] propose to learn the spatially sample features from adjacent frames, which is robust to occlusion or motion blur in individual frames.

2.3 Brief Review of Multi-object Tracking Methods

Multi-object tracking aims to recover the target trajectories in video sequences. Most of the previous methods formulate the tracking problem as a data association problem [11,32,36,56]. Some methods [3,9,45,55] attempt to learn the affinity in association for better performance. In addition, Sadeghian *et al.* [44] design a Recurrent Neural Network (RNN) structure, which jointly integrates multiple cues based on the appearance, motion, and interactions of objects over a temporal window. Wen *et al.* [52] formulate the multi-object tracking task as dense structure exploiting on a hypergraph, whose nodes are detections and hyperedges describe the corresponding high-order relations. Tang *et al.* [46] use a graph-based formulation that links and clusters person hypotheses over time by solving an instance of a minimum cost lifted multicut problem for multiple

Table 1. Comparison of current state-of-the-art benchmarks and datasets. Note that, the resolution indicates the maximum resolution of the videos/images included in the dataset.

Video object detection	Scen.	#Frms	Cat.	Avg. #Labels/cat.	Res.	Occ.	Year
ImageNet VID [43]	life	2017.6k	30	66.8k	1280 × 1080	√	2015
UA-DETRAC [51]	surv.	140.1k	4	302.5k	960 × 540	√	2015
MOT17Det [1]	life	11.2k	1	392.8k	1920 × 1080	√	2017
Okutama-Action [4]	drone	77.4k	1	422.1k	3840 × 2160		2017
VisDrone-VDT2018	drone	33.4k	10	149.9k	3840 × 2160	√	2018
Multi-object tracking	Scen.	#Frms	Cat.	Avg. #Labels/cat.	Res.	Occ.	Year
KITTI [16]	driving	19.1k	5	19.0k	1392 × 512	√	2013
MOT2015 [25]	surveillance	11.3k	1	101.3k	1920 × 1080		2015
UA-DETRAC [51]	surveillance	140.1k	4	302.5k	960 × 540	√	2015
DukeMTMC [41]	surveillance	2852.2k	1	4077.1k	1920 × 1080		2016
Campus [42]	drone	929.5k	6	1769.4k	1417 × 2019		2016
MOT17 [1]	surveillance	11.2k	1	392.8k	1920 × 1080		2017
VisDrone-VDT2018	drone	33.4k	10	149.9k	3840 × 2160	√	2018

object tracking. Feichtenhofer *et al.* [13] set up a CNN architecture for simultaneous detection and tracking, using a multi-task objective for frame-based object detection and across-frame track regression.

3 The VisDrone-VDT2018 Challenge

As described above, the VisDrone-VDT2018 challenge focuses on two tasks in computer vision, *i.e.*, (1) video object detection, and (2) multi-object tracking, which use the same video data. We release a large-scale video object detection and tracking dataset, including 79 video clips with approximate 1.5 million annotated bounding boxes in 33,366 frames. Some other useful annotations, such as object category, occlusion, and truncation ratios, are also provided for better data usage. Participants are expected to submit a single set of results per algorithm in the VisDrone-VDT2018 dataset. We also allow the participants to submit the results of multiple different algorithms. However, changes in the parameters of the algorithms are not considered as the different algorithms. Notably, the participants are allowed to use additional training data to optimize their models. The use of external data should be explained in submission.

3.1 Dataset

The VisDrone-VDT2018 dataset consists of 79 challenging sequences with a total of 33,366 frames, which is divided into three non-overlapping subsets, *i.e.*, `training` set (56 video clips with 24,198 frames), `validation` set (7 video clips with 2,846 frames), and `testing` set (16 video clips with 6,322 frames). These video sequences are captured from different cities under various weather and lighting conditions. The manually generated annotations for the `training` and `validation` subsets are made available to users, but the annotations of the

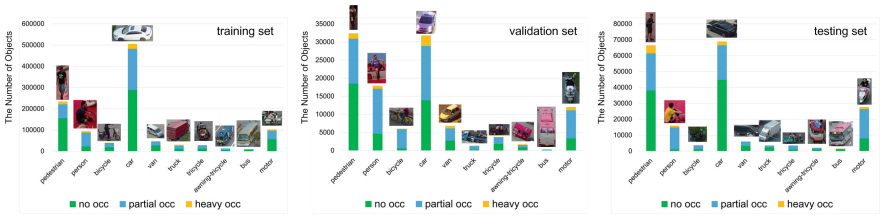

Fig. 1. The number of objects with different occlusion degrees of each object category in the `training`, `validation` and `testing` subsets for the video object detection and multi-object tracking tasks.

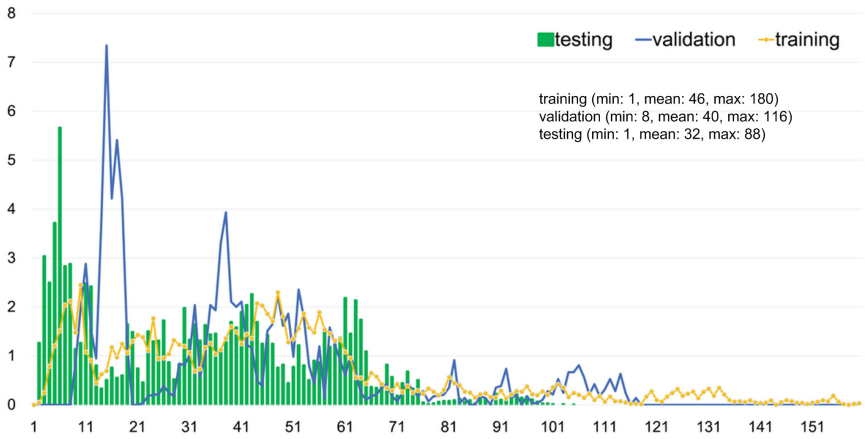

Fig. 2. The number of objects per frame *vs.* percentage of video frames in the `training`, `validation` and `testing` subsets for the video object detection and multi-object tracking tasks. The maximal, mean and minimal numbers of objects per image in the three subsets are presented in the legend.

`testing` set are reserved to avoid (over)fitting of algorithms. The video sequences of the three subsets are captured at different locations, but share similar environments and attributes. We focus on five object categories in this challenge, *i.e.*, *pedestrian*[3], *car*, *van*, *bus*, and *truck*, and carefully annotate more than 1 million bounding boxes of object instances in the video sequences. Some annotated example frames are shown in Fig. 3. We present the number of objects with different occlusion degrees of each object category in the `training`, `validation`, and `testing` subsets in Fig. 1, and plot the number of objects per frame *vs.* percentage of video frames in the `training`, `validation`, and `testing` subsets to show the distributions of the number of objects in each video frame in Fig. 2.

[3] If a human maintains standing pose or walking, we classify it as a *pedestrian*; otherwise, it is classified as a *person*.

In addition, we also provide the occlusion and truncation ratios annotations for better usage. Specifically, we annotate the occlusion relationships between objects, and use the fraction of pixels being occluded to define the occlusion ratio. Three degrees of occlusions of objects are provided, *i.e.*, no occlusion (occlusion ratio 0%), partial occlusion (occlusion ratio 1%∼), and heavy occlusion (occlusion ratio >50%). We also provide the truncation ratio of objects, which is used to indicate the degree of object parts that appear outside a frame. If an object is not fully captured within a frame image, we label the bounding box inside the frame boundary and estimate the truncation ratio based on the region outside the image. It is worth mentioning that a target trajectory is regarded as ending if its truncation ratio starts to be larger than 50%.

3.2 Video Object Detection

Video object detection aims to locate object instances from a predefined set of five object categories in the videos. For the video object detection task, we require the participating algorithms to predict the bounding boxes of each predefined object class in each video frame.

Evaluation Protocol. For the video object detection task, we require each algorithm to produce the bounding boxes of objects in each video frame of each video clip. Motivated by the evaluation protocols in MS COCO [28] and the ILSVRC 2015 challenge [43], we use the $AP^{IoU=0.5:0.05:0.95}$, $AP^{IoU=0.5}$, $AP^{IoU=0.75}$, $AR^{max=1}$, $AR^{max=10}$, $AR^{max=100}$, and $AR^{max=500}$ metrics to evaluate the results of the video detection algorithms. Specifically, $AP^{IoU=0.5:0.05:0.95}$ is computed by averaging over all 10 intersection over union (IoU) thresholds (*i.e.*, in the range $[0.50 : 0.95]$ with the uniform step size 0.05) of all object categories, which is used as the primary metric for ranking. $AP^{IoU=0.50}$ and $AP^{IoU=0.75}$ are computed at the single IoU thresholds 0.5 and 0.75 over all object categories, respectively. The $AR^{max=1}$, $AR^{max=10}$, $AR^{max=100}$ and $AR^{max=500}$ scores are the maximum recalls with 1, 10, 100 and 500 detections per frame, averaged over all categories and IoU thresholds. Please refer to [28] for more details.

Detectors Submitted. We have received 6 entries in the VisDrone-VDT2018 challenge. Four submitted detectors are derived directly from the image object detectors, including CERTH-ODV (A.1), CFE-SSDv2 (A.2), RetinaNet_s (A.3) and RD (A.4). The EODST (A.5) detector is a combination of the image object detector and visual tracker, and the FGFA+ (A.6) detector is an end-to-end learning framework for video object detection. We summarize the submitted algorithms in Table 2, and present a brief description of the submitted algorithms in Appendix A.

Results and Analysis. The results of the submitted algorithms are presented in Table 3. CFE-SSDv2 (A.2) achieves the best performance of all submissions, which design a comprehensive feature enhancement module to enhance the features for

Table 2. The descriptions of the submitted video object detection algorithms in the VisDrone-VDT2018 challenge. The running speed (in FPS), GPUs for training, implementation details, training datasets and the references on the video object detection task are reported.

Method	Speed	GPU	Code	Datasets	Reference
CERTH-ODV (A.1)	1	GTX1070	Python	MS-COCO VisDrone-VDT	FRCNN [39]
CFE-SSDv2 (A.2)	1	TitanXP×4	Python	VisDrone-VDT MS-COCO	SSD [29]
RetinaNet_s (A.3)	25	GTX1080Ti	Pytorch	VisDrone-VDT	RetinaNet [27]
RD (A.4)	1.5	TitanXP×3	Caffe	VisDrone-VDT	RefineDet [57]
EODST (A.5)	1	Titan	Caffe	VisDrone-VDT	SSD [29]
FGFA+ (A.6)		GTX1080	Python Matlab	VisDrone-VDT	FGFA [59]

small object detection. In addition, the multi-scale inference strategy is used to further improve the performance. The EODST (A.5) detector produces the second best results, closely followed by FGFA+ (A.6). EODST (A.5) considers the concurrence of objects, and FGFA+ (A.6) employs the temporal context to improve the detection accuracy. RD (A.4) performs slightly better than FGFA+ (A.6) in AP_{50}, but produces worse results on other metrics. CERTH-ODV (A.1) performs on par with RetinaNet_s (A.3) with the AP score less than 10%.

Table 3. Video object detection results on the VisDrone-VDT2018 `testing` set. The submitted algorithms are ranked based on the AP score.

Method	AP [%]	AP_{50} [%]	AP_{75} [%]	AR_1 [%]	AR_{10} [%]	AR_{100} [%]	AR_{500} [%]
CFE-SSDv2	**21.57**	**44.75**	**17.95**	**11.85**	**30.46**	**41.89**	**44.82**
EODST	16.54	38.06	12.03	10.37	22.02	25.52	25.53
FGFA+	16.00	34.82	12.65	9.63	19.54	22.37	22.37
RD	14.95	35.25	10.11	9.67	24.60	29.72	29.91
CERTH-ODV	9.10	20.35	7.12	7.02	13.51	14.36	14.36
RetinaNet_s	8.63	21.83	4.98	5.80	12.91	15.15	15.15

3.3 Multi-object Tracking

Given an input video sequence, multi-object tracking aims to recover the trajectories of objects. Depending on the availability of prior object detection results in each video frame, we divide the multi-object tracking task into two sub-tasks, denoted by MOT-a (without prior detection) and MOT-b (with prior detection). Specifically, for the MOT-b task, we provide the object detection results of the Faster R-CNN algorithm [40] trained on the VisDrone-VDT2018 dataset in the

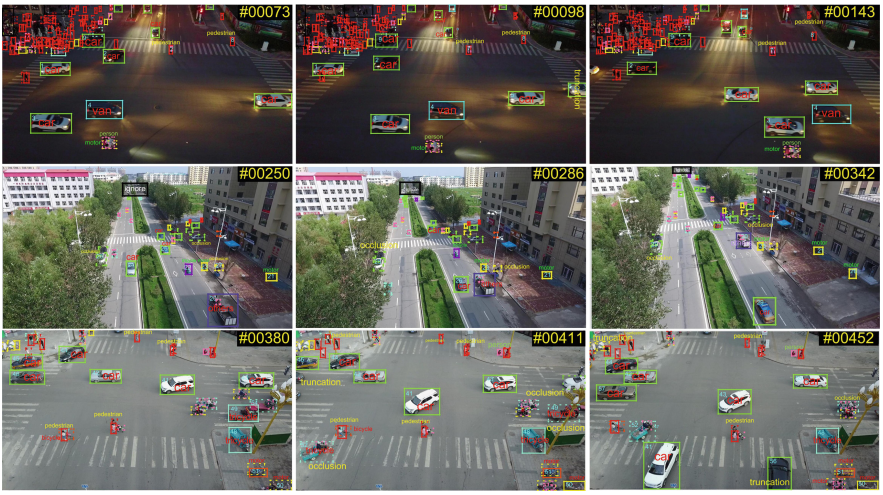

Fig. 3. Some annotated example video frames of multiple object tracking. The bounding boxes and the corresponding attributes of objects are shown for each sequence.

VisDrone2018 challenge, and require the participants to submit the tracking results for evaluation. Some annotated video frames of the multi-object tracking task are shown in Fig. 3.

Evaluation Protocol. For the MOT-a task, we use the tracking evaluation protocol of [35] to evaluate the performance of the submitted algorithms. Each algorithm is required to produce a list of bounding boxes with confidence scores and the corresponding identities. We sort the tracklets (formed by the bounding box detections with the same identity) according to the average confidence over the bounding box detections. A tracklet is considered correct if the intersection over union (IoU) overlap with ground truth tracklet is larger than a threshold. Similar to [35], we use three thresholds of evaluation, *i.e.*, 0.25, 0.50, and 0.75. The performance of an algorithm is evaluated by averaging the mean average precision (mAP) per object class over different thresholds. Please refer to [35] for more details.

For the MOT-b task, we follow the evaluation protocol of [31] to evaluate the performance of the submitted algorithms. Specifically, the average rank of 10 metrics (*i.e.*, MOTA, MOTP, IDF1, FAF, MT, ML, FP, FN, IDS, and FM) is used to rank the algorithms. The MOTA metric combines three error sources, *i.e.*, FP, FN and IDS. The MOTP metric is the average dissimilarity between all true positives and the corresponding ground truth targets. The IDF1 metric indicates the ratio of correctly identified detections over the average number of ground truths and the predicted detections. The FAF metric indicates the average number of false alarms per frame. The FP metric describes the total number of tracker outputs which are the false alarms, and FN is the total number

of targets missed by any of the tracked trajectories in each frame. The IDS metric describes the total number of times that the matched identity of a tracked trajectory changes, while FM is the times that the trajectories are disconnected. Both the IDS and FM metrics describe the accuracy of the tracked trajectories. The ML and MT metrics measure the percentage of tracked trajectories less than 20% and more than 80% of the time span based on the ground truth respectively.

Table 4. Multi-object tracking results **without prior object detection in each video frame** on the VisDrone-VDT2018 `testing` set. The submitted algorithms are ranked based on the AP metric.

Method	AP	AP@0.25	AP@0.50	AP@0.75	AP_{car}	AP_{bus}	$AP_{tr.k}$	AP_{ped}	AP_{van}
Ctrack	**16.12**	**22.40**	**16.26**	**9.70**	27.74	**28.45**	**8.15**	7.95	8.31
deep-sort_d2	10.47	17.26	9.40	4.75	**29.14**	2.38	3.46	7.12	**10.25**
MAD	7.27	12.72	7.03	2.07	16.23	1.65	2.85	**14.16**	1.46

Table 5. Multi-object tracking results **with prior object detection in each frame** on the VisDrone-VDT2018 `testing` set. The submitted algorithms are ranked based on the average rank of the ten metrics. $*$ indicates that the tracking algorithm is submitted by the committee.

Method	Rank	MOTA	MOTP	IDF1	FAF	MT	ML	FP	FN	IDS	FM
V-IOU	2.7	40.2	74.9	56.1	0.76	297	514	11838	74027	**265**	**1380**
TrackCG	2.9	**42.6**	74.1	**58.0**	0.86	323	395	14722	68060	779	3717
GOG_EOC	3.2	36.9	**75.8**	46.5	**0.29**	205	589	**5445**	86399	354	1090
SCTrack	3.8	35.8	75.6	45.1	0.39	211	550	7298	85623	798	2042
Ctrack	3.9	30.8	73.5	51.9	1.95	**369**	**375**	36930	**62819**	1376	2190
FRMOT	4.0	33.1	73.0	50.8	1.15	254	463	21736	74953	1043	2534
GOG* [36]	-	38.4	75.1	45.1	0.54	244	496	10179	78724	1114	2012
IHTLS* [11]	-	36.5	74.8	43.0	0.94	245	446	14564	75361	1435	2662
TBD* [15]	-	35.6	74.1	45.9	1.17	302	419	22086	70083	1834	2307
H^2T* [53]	-	32.2	73.3	44.4	0.95	214	494	17889	79801	1269	2035
CMOT* [3]	-	31.5	73.3	51.3	1.42	282	435	26851	72382	789	2257
CEM* [33]	-	5.1	72.3	19.2	1.12	105	752	21180	116363	1002	1858

Trackers Submitted. There are in total 8 different multi-object tracking methods submitted to the VisDrone-VDT2018 challenge. The VisDrone committee also reports 6 baseline methods (*i.e.*, GOG (B.9) [36], IHTLS (B.13) [11], TBD (B.10) [15], H^2T (B.14) [53], CMOT (B.12) [3], and CEM (B.11) [33]) using the default parameters. If the default parameters are not available, we select the reasonable values for evaluation. The Ctrack (B.7), TrackCG (B.5) and V-IOU (B.6) trackers aim to exploit the motion information to improve tracking performance. GOG_EOC (B.2), SCTrack (B.3) and FRMOT (B.4) are designed to learn discriminative appearance features of objects to help tracking. Another two trackers MAD (B.1) and deep-sort_v2 (B.8) combines the detectors

(*e.g.*, RetinaNet [27] and YOLOv3 [38]) and tracking algorithms (*e.g.*, DeepSORT [54] and CFNet [50]) to complete the tracking task. We summarize the submitted algorithms in Table 6, and present the descriptions of the algorithms in Appendix B.

Table 6. The descriptions of the submitted algorithms in the multi-object tracking task in the VisDrone-VDT2018 challenge. The running speed (in FPS), CPU and GPU platforms information for training and testing, implementation details (*i.e.*, P indicates Python, M indicates Matlab, and C indicates C/C++), training datasets, and the references on the multi-object tracking task are reported. The $*$ mark is used to indicate the methods are submitted by the VisDrone committee.

Method	Task	Speed	CPU	GPU	Code	Datasets	Reference
MAD (B.1)	a	1.35	E5-2620	TitanXP	P	VisDrone-VDT	CFNet [50]
GOG_EOC (B.2)	b	1	i7-6700	TitanXP	P,M	UAVDT [12]	GOG [36]
SCTrack (B.3)	b	2.90	i7-4720	-	M	-	SCTrack [2]
FRMOT (B.4)	b	5	-	TitanXP	P	VOC 2007	FRCNN [39]
TrackCG (B.5)	b	10	i7-6700	-	C	-	TrackCG [47]
V-IOU (B.6)	b	20 – 200	i7-6700	-	P	-	IOU [6]
Ctrack (B.7)	a/b	15	i7-6700HQ	-	M	-	Ctrack [48]
deep-sort_v2 (B.8)	a	25	-	GTX1080Ti	P	MS-COCO VisDrone-VDT	DSORT [54]
GOG* (B.9)	b	564.80	i7-3520M	-	M	-	GOG [36]
IHTLS* (B.13)	b	16.30	i7-3520M	-	M	-	IHTLS [11]
TBD* (B.10)	b	0.70	i7-3520M	-	M	-	TBD [15]
CMOT* (B.12)	b	1.39	i7-3520M	-	M	-	CMOT [3]
CEM* (B.11)	b	7.74	i7-3520M	-	M,C	-	CEM [33]
H^2T^* (B.14)	b	1.56	i7-3520M	-	C	-	H2T [53]

Results and Analysis. The results of the submissions of the MOT-a and MOT-b tasks are presented in Tables 4 and 5, respectively.

As shown in Table 4, Ctrack (B.7) achieves the top AP score among all submissions in the MOT-a task. In terms of different object categories, it performs the best in the bus and truck categories. We suspect that the complex motion models used in Ctrack (B.7) are effective in tracking large size objects. Deepsort_d2 (B.8) produces the best results for cars and vans. Since these two categories of objects usually move smoothly, the IOU similarity and deep appearance features are effective to extract the discriminative motion and appearance features of these objects. MAD (B.1) produces the top AP_{ped} score, which demonstrates the effectiveness of the model ensemble strategy.

As shown in Table 5, we find that V-IOU (B.6) produces the top average rank of 2.7 over the 10 metrics. The TrackCG method (B.5) achieves the best MOTA and IDF1 scores among all submissions. GOG_EOC (B.2) considers the exchanging context of objects to improve the performance, which performs much better than the original GOG method (B.9) in terms of the MOTP, IDF1, FAF, ML, FP, IDS and FM metrics, and ranks at the third place. Ctrack (B.7) performs on par with SCTrack (B.3), but produces better MT, ML and FN scores.

Ctrack (B.7) uses the aggregation of prediction events in grouped targets and the stitching procedure by temporal constraints to help tracking, which is able to recover the target objects with long-time disappearance in the crowded scenes.

To further analyze the performance of the submissions thoroughly in different object categories, we present the MOTA and IDF1 scores of 5 evaluated object categories (*i.e.*, *car*, *bus*, *truck*, *pedestrian*, and *van*) in Figs. 4 and 5. The top

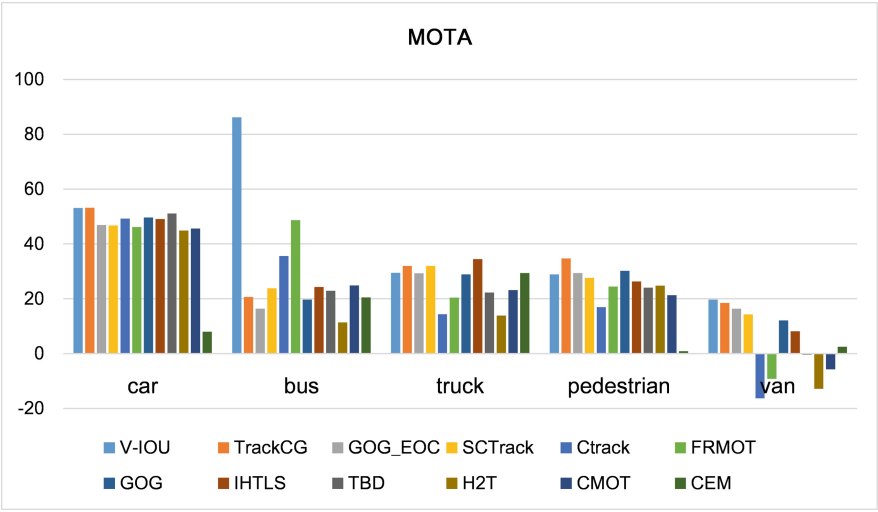

Fig. 4. Comparisons of all the submissions based on the MOTA metric for each object category.

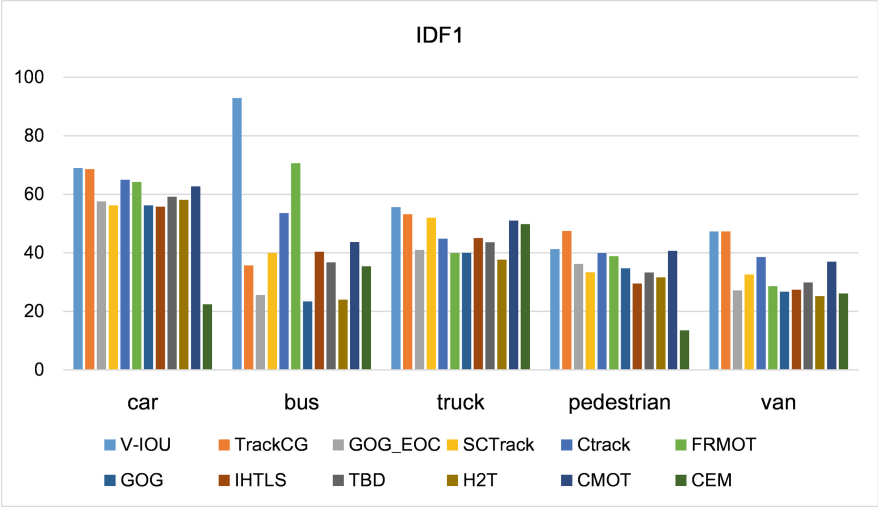

Fig. 5. Comparisons of all the submissions based on the IDF1 metric for each object category.

two best trackers V-IOU (B.6) and TrackCG (B.5) produce the best results in all categories of objects. We also observe that V-IOU (B.6) and FRMOT (B.4) produce the best results in the bus category, which may be attributed to the effectiveness of the IOU and deep feature based similarities in tracking the large size objects.

4 Conclusions

This paper concludes the VisDrone-VDT2018 challenge, which focuses on two tasks, *i.e.*, (1) video object detection, and (2) multi-object tracking. A large-scale video object detection and tracking dataset is released, which consists of 79 challenging sequences with 33,366 frames in total. We provide fully annotations of the dataset with annotated bounding boxes and the corresponding attributes such as object categories, occlusion status and truncation ratios. 6 algorithms are submitted to the video object detection task and 14 algorithms are submitted to the multiple object tracking (*i.e.*, 3 methods do not use the prior object detection in video frames and 12 methods use the prior object detection in video frames). The CFE-SSDv2 (A.2) method achieves the best results in the video object detection task, Ctrack (B.7) achieves the best results in the MOT-a task, and V-IOU (B.6) and TrackCG (B.5) perform better than other submitted methods in the MOT-b task. The **VisDrone-VDT2018** challenge was successfully held on September 8, 2018, which is a part of the VisDrone2018 challenge workshop. We hope this challenge is able to provide a unified platform for video object detection and tracking evaluations on drones. Our future work will focus on revising the dataset and evaluation kit based on the feedbacks from the community.

Acknowledgements. This work was supported in part by the National Natural Science Foundation of China under Grant 61502332 and Grant 61732011, in part by Natural Science Foundation of Tianjin under Grant 17JCZDJC30800, in part by US National Science Foundation under Grant IIS-1407156 and Grant IIS-1350521, and in part by Beijing Seetatech Technology Co., Ltd and GE Global Research.

A Submissions in the Video Object Detection Task

A.1 CERTH's Object Detector in Videos (CERTH-ODV)

Emmanouil Michail, Konstantinos Avgerinakis, Panagiotis Giannakeris, Stefanos Vrochidis, Ioannis Kompatsiaris
{*michem,koafgeri,giannakeris,stefanos,ikom*}*@iti.gr*

CERTH-ODV is based on the Inception ResNet v2 Faster R-CNN [40] method pretrained on the MSCOCO dataset. The model is fine-tuned on the training set of the Visdrone-VID2018 dataset. Training images are selected every 5 frames to avoid overfitting. Since pedestrian and cars are dominant compared to other classes, to balance the number of the object classes, we remove several thousand car and pedestrian ground-truths. For training, we used the Inception ResNet v2 Faster R-CNN model pretrained on the MSCOCO dataset.

A.2 SSD with Comprehensive Feature Enhancement (CFE-SSDv2)

Qijie Zhao, Feng Ni, Yongtao Wang
{*zhaoqijie,nifeng,wyt*}*@pku.edu.cn*

CFE-SSDv2 is an end-to-end one-stage object detector with specially designed novel module, namely the Comprehensive Feature Enhancement (CFE) module. We first improve the original SSD model [29] by enhancing the weak features for detecting small objects. Our CFE-SSDv2[4] model is designed to enhance detection ability for small objects. In addition, we apply multi-scale inference strategy. Although training on the input images with the size of 800×800, we expand the input images to the size of 2200×2200 in testing, leading to further improvements in detection accuracy, especially for small objects.

A.3 Some Improvement on RetinaNet (RetinaNet_s)

Jianfei Zhao, Yanyun Zhao
{*zjfei,zyy*}*@bupt.edu.cn*

RetinaNet_s is based on the RetinaNet50 model [27]. We change the anchor size to detect more small objects. For the same reason, we add a conv layer in FPN's P3 and P4 where the higher feature add to the lower feature. We also use the multi-scale training and multi-scale testing techniques, and the Soft-NMS [8] algorithm in post processing.

A.4 RefineDet with SEResNeXt-50 Base Network (RD)

Oliver Acatay, Lars Sommer, Arne Schumann
{*oliver.acatay,lars.sommer,arne.schumann*}*@iosb.fraunhofer.de*

RD is a variant of the RefineDet detector [27], and uses the novel Squeeze-and-Excitation Network (SENet) [21] as the base network. Specifically, we train the detector with SEResNeXt-50 as the base network and adapt the anchor sizes and training parameters to the dataset.

A.5 Efficient Object Detector with the Support of Spatial and Temporal Information (EODST)

Zhaoliang Pi, Yinan Wu, Mengkun Liu
{*zhaoliangpi_xdu,18710706807,18700919098*}*@163.com*

EODST is based on the SSD detector [29] and ECO tracker [10]. Our method consists of three main components: (1) still-image object detection, (2) visual tracking, (3) false positive analysis. Specifically, our still-image object detectors adopt the SSD framework. To deal with the imbalance problem of classes, we crop the objects from the training data to generate more training samples and

[4] https://github.com/qijiezhao/CFENet.

balance the samples among each class as possible. Then we test the images (with contrast, clarity or brightness enhanced) in multi-scales and merge the detection result of cropped images using NMS technique. Afterwards, we use the tracking algorithms from ECO and associate still-image object detection. For each object class in a video clip, we track high-confidence detection objects bidirectionally over the temporal dimension. Additionally, we consider the relationship of contextual regions, *i.e.*, features of different contextual regions validate each other (like bicycle and people, motor and people). We conduct box refinement and false positive suppression by inference according to temporal and contextual information of videos.

A.6 Modified Flow-Guided Feature Aggregation for Video Object Detection Based on Image Segmentation (FGFA+)

Jie Gao, Yidong Bai, Gege Zhang, Dan Wang, Qinghua Ma
ggzhang_1@stu.xidian.edu.cn, baiyidong@sina.cn

FGFA+ is the improved variant of an efficient method for frames detection [59]. However, the emerging problems can be listed as follows: (1) nearly all images from training set are taken under the sunset, while many images from testing set are in the night time. (2) According the fact that quite a large regions are ignored so that the objects in them are not necessary to be detected accurately. In order to solve these problems, the contributions are listed as follows:

(1) The frames are enhanced in contrast and brightness before they are used for training in FGFA. (2) The ignored regions are set to black so that FGFA can extract obvious features for training process. (3) For such object with both high evolutions in two classes, it may be correctly classified using NMS. (4) NMS is necessary when we merge the whole images for submission to restore the cutting images.

B Submissions in the Multi-object Tracking Task

B.1 Multi-object Tracking Algorithm Assisted by Detection (MAD)

Wei Song, Yuxuan Li, Zhaoliang Pi, Wenhua Zhang
522545707@qq.com

MAD is mainly based on YOLOv3 [38] and CFNet [50]. To determine the initial tracking position of objects, we adopt the detection strategy combining YOLOv3 and RetinaNet. YOLO has a good detection effect for usual objects but is not ideal for smaller and denser objects, yet the advantages of RetinaNet are detecting dense small objects well. To deal with small objects, we first expand the three object categories (*i.e.*, *van*, *truck*, *bus*), including rotation, deformation, and brightness adjustment. Second, we train a model separately for them. Therefore, we train three models: (1) YOLO for *pedestrian* and *car*, (2) YOLO for *van*, *truck* and *bus*, (3) RetinaNet. After the inference is completed, repeating objects are removed by NMS.

B.2 Globally-Optimal Greedy Algorithms with the Harmony Model Exchanging Object Context (GOG_EOC)

Hongyang Yu, Guorong Li, Qingming Huang
hyang.yu@hit.edu.cn, {liguorong,qmhuang}@ucas.ac.cn

Our method is based on the Globally-Optimal Greedy algorithms (GOG) [36]. For the graph built in the GOG tracker, we change the cost of connecting the detections between two frames. Specifically, the cost consists of the object detection overlap and the context harmony degree. The proposed context harmony degree measures the detections harmony with Exchanging Object Context (EOC) patches via the Siamese network.

B.3 Semantic Color Tracker (SCTrack)

Noor M. Al-Shakarji, Filiz Bunyak, Guna Seetharaman, Kannappan Palaniappan
{nmahyd,bunyak}@mail.missouri.edu, gunasekaran.seetharaman@rl.af.mil, palaniappank@missouri.edu

SCTrack is a detection-based multi-object tracking system [2] that uses a multi-step data association approach to ensure time-efficient processing while preserving tracking accuracy. The system relies on a robust but discriminative object appearance model combined with a novel color correlation cost matrix to maintain object identities in time.

B.4 Faster-RCNN Features for Multiple Object Tracking (FRMOT)

Elena Luna, Diego Ortego, Juan C. San Miguel, José M. Martínez
{elena.luna,diego.ortego,juancarlos.sanmiguel,josem.martinez}@uam.es

FRMOT is composed of five main modules: feature extraction, data association, track management, model update and spatial prediction. In this framework, targets are modeled by their visual appearance (via deep features) and their spatial location (via bounding boxes). Firstly, we describe the appearance of each bounding box by using off-the-shelf features from the pre-trained deep neural network Faster R-CNN. Secondly, we use Kalman filtering for predicting the spatial position of targets with constant velocity motion and linear observation model. Thirdly, we use the Hungarian algorithm for associating detections to targets. Notably, each match between targets and detections is employed to determine the tracks (*i.e.*, sequential information of targets over time). We employ two counters for each target for handling initialization and suppression of trackers. One counter focuses on the number of consecutive frames where the target matches any detection. Another counter focuses on the number of consecutive frames where the target is unmatched. To update the target model, we perform two strategies. The spatial target model is updated via the update step of each corresponding Kalman filter. The appearance model update consists in maintaining a buffer/gallery of the last n samples previously associated appearance descriptors of the target.

B.5 Multi-object Tracking with Combined Constraints and Geometry Verification (TrackCG)

Wei Tian, Zhiming Ma, Martin Lauer
wei.tian@kit.edu, zhiming0405@sjtu.edu.cn, martin.lauer@kit.edu

TrackCG is based on the work [48] with modifications adapted to the current dataset. This algorithm is separated into two stages. In the first stage, it mainly estimates the state of a target based on the motion pattern of grouped objects and builds short tracklets from individual detections. In the second stage, it deploys graph models for long range association, which means associating tracklets to construct tracks. Additionally, according to [47], we deploy a regression method to coarsely estimate the ground plane to filter out outliers. In our experiment, this filtering procedure is also combined with criteria like track length, average object size and score, ratio of consecutive frames in the track, etc.

B.6 Visual Intersection-Over-Union Tracker (V-IOU)

Erik Bochinski, Tobias Senst, Thomas Sikora
{bochinski,senst,sikora}@nue.tu-berlin.de

V-IOU is based on the IOU tracker [6] which associates detections to tracks solely by their spatial overlap (Intersection-over-Union) in consecutive frames. The method is further improved by visual tracking to continue a track if no detection is available. If a valid detection can be associated again, visual tracking is stopped and the tracker returns to the original IOU tracker functionality. Otherwise, the visual tracking is aborted after *ttl* frames. For each new track, visual tracking is performed backwards for a maximum of *ttl* previous frames or until the track can be merged with a finished one if the IOU criteria of [6] is satisfied. This extension is made to efficiently reduce the high amount of fragmentation of the tracks produced by the original IOU tracker. V-IOU can be used in association with a wide range of visual single-object trackers. In our evaluation, we consider Medianflow [22] and KCF [19] achieving state-of-the-art performance at processing speeds of 20 and 209 fps respectively. Please refer to [7] for further details.

B.7 Constrained Track (Ctrack)

Wei Tian, Zhiming Ma, Martin Lauer
wei.tian@kit.edu, zhiming0405@sjtu.edu.cn, martin.lauer@kit.edu

Ctrack is based on two ideas to deal with multiple object tracking, including the aggregation of prediction events in grouped targets and the stitching procedure by temporal constraints. Thanks to these strategies, we are able to track objects in crowded scenes and recover the targets with long time disappearance. Specifically, we analyze the motion patterns within grouped targets in the light of aggregated prediction events. Additionally, we use a stitching procedure based on graph modeling to link separated tracks of the same target. Please refer to [48] for more details.

B.8 More Improvements in Detector and Deep-sort for Drones (deep-sort_d2)

Jianfei Zhao, Yanyun Zhao
{zjfei,zyy}@bupt.edu.cn

Deep-sort_d2 is based on RetinaNet50 [27] and Deep-SORT [54]. For detection, we use a RetinaNet50 [27], and we change the anchor size to detect more small objects. For the same reason, we add a conv layer in fpn's p3 and p4 where the higher feature add to the lower feature. We also use multi-scale training and multi-scale testing, meanwhile we use the Soft-NMS [8]. For tracking, we make some improvement on the deep sort algorithm [54]. The algorithm can be divided into four steps. First, we compute iou distance between the tracks which appear on the last frame and the detections, if the distance is lower than a strict thresh, we think they are matched. And if the unmatched detections are more than the matched detections, we think the camera moved suddenly or rotated, then we will change the parameters and strategies in the other steps. Second, we get the detections appearance features from an AlignedReID net [58], and we use a cascade strategy to matching the unmatched tracks and unmatched detections from last step. Then we compute the IOU distance again between the unmatched tracks and unmatched detections with a higher thresh than the first step. Final, if the camera does not move, for every two matches, which matched track appeared in last three frames, we would switch their detections' positions if their relative angle were changed. For every tracks, we use the Gaussian Process Regressor to process the continuous part. Besides, we compute the average position to fill the fragmentations.

B.9 Globally-Optimal Greedy Algorithms for Tracking a Variable Number of Objects (GOG)

Submitted by the VisDrone Committee

GOG formulates the multi-object tracking problem as the integer linear program (ILP). Specifically, the model is based on the min-cost flow network, which is efficient in the greedy manner. It allows us to handle long sequences with large number of objects, even in complex scenarios with long-term occlusion of objects. Please refer to [36] for further details.

B.10 3D Traffic Scene Understanding From Movable Platforms (TBD)

Submitted by the VisDrone Committee

TBD is a probabilistic generative model for multi-object traffic scene understanding from movable platforms. The model extracts a diverse set of visual cues in the form of vehicle tracklets, including vanishing points, semantic scene labels, scene flow, and occupancy grids. For each of these cues, the likelihood functions

are proposed, which are integrated into a probabilistic generative model and are learnt from the training data using contrastive divergence. Please refer to [15] for further details.

B.11 Continuous Energy Minimization for Multitarget Tracking (CEM)

Submitted by the VisDrone Committee

CEM is an offline multi-object tracking algorithm as minimization of a continuous energy over all target locations and all frames of a time window. Thus the existence, motion and interaction of all objects of interest in the scenes are represented by a suitable energy function. To solve the non-convex energy minimization problem, we introduce a number of jump moves which change the dimension of the current state, thereby jumping to a different region of the search space, while still decreasing the energy. Please refer to [33] for further details.

B.12 Robust Online Multi-object Tracking Based on Tracklet Confidence and Online Discriminative Appearance Learning (CMOT)

Submitted by the VisDrone Committee

CMOT is an online multi-object tracking method based on the tracklet confidence using the detectability and continuity of the tracklet. According to the confidence values of tracklets, reliable tracklets with high confidence are locally associated with online-provided detections, while fragmented tracklets with low confidence are globally associated with other tracklets and detections. The proposed online discriminative appearance learning can handle similar appearances of different objects in tracklet association. Please refer to [33] for further details.

B.13 The Way They Move: Tracking Multiple Targets with Similar Appearance (IHTLS)

Submitted by the VisDrone Committee

IHTLS is a tracking by detection multi-object tracking method, which uses motion dynamics as a cue to distinguish targets with similar appearance. Specifically, it formulates the problem as a generalized linear assignment (GLA). Then, the efficient IHTILS algorithm is employed to estimate these similarity measures. Please refer to [11] for further details.

B.14 Multiple Target Tracking Based on Undirected Hierarchical Relation Hypergraph (H^2T)

Submitted by the VisDrone Committee

H^2T formulates the multiple object tracking as a data association problem. Specifically, hierarchical dense neighbourhoods searching is performed on the dynamically constructed undirected affinity hypergraph. The nodes denote the tracklets of objects and the hyperedges describe the appearance and motion relationships among different tracklets across the temporal domain, which makes the tracker robust to the spatially close targets with similar appearance. Please refer to [53] for further details.

References

1. Mot17 challenge. https://motchallenge.net/
2. Al-Shakarji, N.M., Seetharaman, G., Bunyak, F., Palaniappan, K.: Robust multi-object tracking with semantic color correlation. In: IEEE International Conference on Advanced Video and Signal-Based Surveillance, pp. 1–7 (2017)
3. Bae, S.H., Yoon, K.: Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning. In: Proceedings of IEEE Conference on Computer Vision and Pattern (2014)
4. Barekatain, M., et al.: Okutama-action: an aerial view video dataset for concurrent human action detection. In: Workshops in Conjunction with the IEEE Conference on Computer Vision and Pattern (2017)
5. Bertasius, G., Torresani, L., Shi, J.: Object detection in video with spatiotemporal sampling networks. CoRR abs/1803.05549 (2018). http://arxiv.org/abs/1803.05549
6. Bochinski, E., Eiselein, V., Sikora, T.: High-speed tracking-by-detection without using image information. In: IEEE International Conference on Advanced Video and Signal-Based Surveillance, pp. 1–6 (2017)
7. Bochinski, E., Senst, T., Sikora, T.: Extending IOU based multi-object tracking by visual information. In: AVSS. IEEE (2018)
8. Bodla, N., Singh, B., Chellappa, R., Davis, L.S.: Soft-NMS - improving object detection with one line of code. In: Proceedings of the IEEE International Conference
9. Choi, W.: Near-online multi-target tracking with aggregated local flow descriptor. In: Proceedings of the IEEE International Conference Computer Vision, pp. 3029–3037 (2015)
10. Danelljan, M., Bhat, G., Khan, F.S., Felsberg, M.: ECO: efficient convolution operators for tracking. In: Proceedings of IEEE Conference on Computer Vision and Pattern
11. Dicle, C., Camps, O.I., Sznaier, M.: The way they move: tracking multiple targets with similar appearance. In: Proceedings of the IEEE International Conference
12. Du, D., et al.: The unmanned aerial vehicle benchmark: object detection and tracking. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11214, pp. 375–391. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01249-6_23
13. Feichtenhofer, C., Pinz, A., Zisserman, A.: Detect to track and track to detect. In: Proceedings of the IEEE International Conference Vision, pp. 3057–3065 (2017)

14. Galteri, L., Seidenari, L., Bertini, M., Bimbo, A.D.: Spatio-temporal closed-loop object detection. IEEE Trans. Image Process. **26**(3), 1253–1263 (2017)
15. Geiger, A., Lauer, M., Wojek, C., Stiller, C., Urtasun, R.: 3D traffic scene understanding from movable platforms. IEEE Trans. Pattern Anal. Mach. Intell. **36**(5), 1012–1025 (2014)
16. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: Proceedings of IEEE Conference on Computer Vision and Pattern
17. Girshick, R.B., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern (2014)
18. Han, W., et al.: Seq-NMS for video object detection. CoRR abs/1602.08465 (2016)
19. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 583–596 (2015)
20. Hsieh, M., Lin, Y., Hsu, W.H.: Drone-based object counting by spatially regularized regional proposal network. In: ICCV (2017)
21. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. CoRR abs/1709.01507 (2017). http://arxiv.org/abs/1709.01507
22. Kalal, Z., Mikolajczyk, K., Matas, J.: Forward-backward error: automatic detection of tracking failures. In: ICPR, pp. 2756–2759 (2010)
23. Kalogeiton, V., Ferrari, V., Schmid, C.: Analysing domain shift factors between videos and images for object detection. TPAMI **38**(11), 2327–2334 (2016)
24. Kang, K., et al.: Object detection in videos with tubelet proposal networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern
25. Leal-Taixé, L., Milan, A., Reid, I.D., Roth, S., Schindler, K.: MOTChallenge 2015: towards a benchmark for multi-target tracking. CoRR abs/1504.01942 (2015)
26. Li, B., Wu, T., Zhu, S.-C.: Integrating context and occlusion for car detection by hierarchical And-Or model. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 652–667. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10599-4_42
27. Lin, T., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference
28. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
29. Liu, W., et al.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
30. Lyu, S.L.S., et al.: UA-DETRAC 2017: report of AVSS2017 & IWT4S challenge on advanced traffic monitoring. In: AVSS, pp. 1–7 (2017)
31. Milan, A., Leal-Taixé, L., Reid, I.D., Roth, S., Schindler, K.: MOT16: a benchmark for multi-object tracking. CoRR abs/1603.00831 (2016) arXiv preprint arXiv:1603.00831
32. Milan, A., Rezatofighi, S.H., Dick, A.R., Reid, I.D., Schindler, K.: Online multi-target tracking using recurrent neural networks. In: Association for the Advancement of Artificial Intelligence, pp. 4225–4232 (2017)
33. Milan, A., Roth, S., Schindler, K.: Continuous energy minimization for multitarget tracking. IEEE Trans. Pattern Anal. Mach. Intell. **36**(1), 58–72 (2014)

34. Mueller, M., Smith, N., Ghanem, B.: A benchmark and simulator for UAV tracking. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 445–461. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_27
35. Park, E., Liu, W., Russakovsky, O., Deng, J., Li, F.F., Berg, A.: Large Scale Visual Recognition Challenge 2017. http://image-net.org/challenges/LSVRC/2017
36. Pirsiavash, H., Ramanan, D., Fowlkes, C.C.: Globally-optimal greedy algorithms for tracking a variable number of objects. In: Proceedings of IEEE Conference on Computer Vision and Pattern
37. Prest, A., Leistner, C., Civera, J., Schmid, C., Ferrari, V.: Learning object class detectors from weakly annotated video. In: CVPR, pp. 3282–3289 (2012)
38. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. CoRR abs/1804.02767 (2018). http://arxiv.org/abs/1804.02767
39. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: NIPS, pp. 91–99 (2015)
40. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2017)
41. Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9914, pp. 17–35. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48881-3_2
42. Robicquet, A., Sadeghian, A., Alahi, A., Savarese, S.: Learning social etiquette: human trajectory understanding in crowded scenes. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 549–565. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_33
43. Russakovsky, O., et al.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2015)
44. Sadeghian, A., Alahi, A., Savarese, S.: Tracking the untrackable: learning to track multiple cues with long-term dependencies. In: Proceedings of the IEEE International Conference
45. Son, J., Baek, M., Cho, M., Han, B.: Multi-object tracking with quadruplet convolutional neural networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern
46. Tang, S., Andriluka, M., Andres, B., Schiele, B.: Multiple people tracking by lifted multicut and person re-identification. In: Proceedings of IEEE Conference on Computer Vision and Pattern (2017)
47. Tian, W., Lauer, M.: Fast cyclist detection by cascaded detector and geometric constraint. In: IEEE International Conference on Intelligent Transportation Systems, pp. 1286–1291 (2015)
48. Tian, W., Lauer, M.: Joint tracking with event grouping and temporal constraints. In: IEEE International Conference on Advanced Video and Signal-Based Surveillance, pp. 1–5 (2017)
49. Tian, Y., Luo, P., Wang, X., Tang, X.: Pedestrian detection aided by deep learning semantic tasks. In: CVPR, pp. 5079–5087 (2015)
50. Valmadre, J., Bertinetto, L., Henriques, J.F., Vedaldi, A., Torr, P.H.S.: End-to-end representation learning for correlation filter based tracking. In: Proceedings of IEEE Conference on Computer Vision and Pattern, pp. 5000–5008 (2017)
51. Wen, L., et al.: UA-DETRAC: a new benchmark and protocol for multi-object detection and tracking. CoRR abs/1511.04136 (2015)

52. Wen, L., Lei, Z., Lyu, S., Li, S.Z., Yang, M.: Exploiting hierarchical dense structures on hypergraphs for multi-object tracking. TPAMI **38**(10), 1983–1996 (2016)
53. Wen, L., Li, W., Yan, J., Lei, Z., Yi, D., Li, S.Z.: Multiple target tracking based on undirected hierarchical relation hypergraph. In: CVPR, pp. 1282–1289 (2014)
54. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: Proceedings of IEEE International Conference on Image
55. Xiang, Y., Alahi, A., Savarese, S.: Learning to track: Online multi-object tracking by decision making. In: Proceedings of the IEEE International Conference
56. Yoon, J.H., Lee, C., Yang, M., Yoon, K.: Online multi-object tracking via structural constraint event aggregation. In: Proceedings of IEEE Conference on Computer Vision and Pattern
57. Zhang, S., Wen, L., Bian, X., Lei, Z., Li, S.Z.: Single-shot refinement neural network for object detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern
58. Zhang, X., et al.: AlignedReID: surpassing human-level performance in person re-identification. CoRR abs/1711.08184 (2017). http://arxiv.org/abs/1711.08184
59. Zhu, X., Wang, Y., Dai, J., Yuan, L., Wei, Y.: Flow-guided feature aggregation for video object detection. In: Proceedings of the IEEE International Conference