VisDrone-MOT2019: The Vision Meets Drone Multiple Object Tracking Challenge Results

Longyin Wen[1], Pengfei Zhu[2], Dawei Du[3], Xiao Bian[4], Haibin Ling[5], Qinghua Hu[2], Jiayu Zheng[2], Tao Peng[2], Xinyao Wang[1], Yue Zhang[1], Liefeng Bo[1], Hailin Shi[28], Rui Zhu[28], Ajit Jadhav[6], Bing Dong[7], Brejesh Lall[8], Chang Liu[9], Chunhui Zhang[10], Dong Wang[9], Feng Ni[7], Filiz Bunyak[11], Gaoang Wang[12], Guizhong Liu[13], Guna Seetharaman[27], Guorong Li[14], Håkan Ardö[15], Haotian Zhang[12], Hongyang Yu[16], Huchuan Lu[9], Jenq-Neng Hwang[12], Jiatong Mu[13], Jinrong Hu[17], Kannappan Palaniappan[11], Long Chen[17], Lu Ding[18], Martin Lauer[19], Mikael Nilsson[20], Noor M. Al-Shakarji[21,11], Prerana Mukherjee[6], Qingming Huang[14,16], Robert Laganière[22], Shuhao Chen[9], Siyang Pan[23], Vinay Kaushik[8], Wei Shi[24], Wei Tian[19], Weiqiang Li[13], Xin Chen[9], Xinyu Zhang[9], Yanting Zhang[23], Yanyun Zhao[23], Yong Wang[22], Yuduo Song[19], Yuehan Yao[7], Zhaotang Chen[17], Zhenyu Xu[7], Zhibin Xiao[25], Zhihang Tong[23], Zhipeng Luo[7], Zhuojin Sun[26]

[1]JD Digits, Mountain View, CA, USA.
[2]Tianjin University, Tianjin, China.
[3]University at Albany, SUNY, Albany, NY, USA.
[4]GE Global Research, Niskayuna, NY, USA.
[5]Stony Brook University, New York, NY, USA.
[6]Indian Institute of Information Technology, Sri City, India.
[7]DeepBlue Technology (Shanghai), Beijing, China.
[8]Indian Institute of Technology, Delhi, India.
[9]Dalian University of Technology, Dalian, China.
[10]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China.
[11]University of Missouri-Columbia, MO, USA.
[12]University of Washington, Seattle, WA, USA.
[13]Xi'an Jiaotong University, Xi'an, China.
[14]University of Chinese Academy of Sciences, Beijing, China.
[15]Axis Communications, Lund, Sweden.
[16]Harbin Institute of Technology, Harbin, China.
[17]Sun Yat-sen University, Guangzhou, China.
[18]Shanghai Jiao Tong University, Shanghai, China.
[19]Karlsruhe Institute of Technology, Karlsruhe, Germany.
[20]Centre for Mathematical Sciences, Lund University, Sweden.
[21]University of Technology, Baghdad, Iraq.
[22]University of Ottawa, Ottawa, Canada.
[23]Beijing University of Posts and Telecommunications, Beijing, China.
[24]INSKY Lab, Leotail Intelligent Tech, Shanghai, China.
[25]Tsinghua University, Beijing, China.
[26]YUNEEC Aviation Technology, Shanghai, China.
[27]U.S. Naval Research Laboratory, DC, USA.
[28]JD AI Research, Beijing, China.

Abstract

The Vision Meets Drone Multiple Object Tracking (MOT) Challenge 2019 is the second annual activity focusing on evaluating multi-object tracking algorithms on drones, held in conjunction with the 17*-th International Conference on Computer Vision (ICCV 2019). Results of* 12 *submitted MOT algorithms on the collected drone-based dataset are presented. Meanwhile, we also report the results of* 6 *state-of-the-art MOT algorithms, and provide a comprehensive analysis and discussion of the results. The results of all submissions are publicly available at the website:* `http://www.aiskyeye.com/`*. The challenge results show that MOT on drones is far from being solved. We believe the challenge can largely boost the research and development in MOT on drone platforms.*

1. Introduction

Multiple Object Tracking (MOT) aims to determine the identities and trajectories of multiple moving objects in a video, thus is a crucial step in video understanding. On the other hand, autonomous drone systems attract increasingly research in recent years because of its more flexibility than traditional fixed surveillance cameras.

Several previous benchmark datasets such as KITTI [20], MOTChallenge [28] and UA-DETRAC [53, 36, 35] are proposed for the MOT task. However, the challenges in those datasets are very different from that on drones for MOT algorithms, such as large viewpoint change and scales. Thus, these algorithms are not usually optimal for dealing with video sequences generated by drones. Some recent preliminary efforts [40, 45, 22, 16] 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 larger scale drone based benchmark [66] is proposed to further boost research on computer vision problems with drone platform.

As discussed in [53], the overall MOT system usually consists of object detection and multi-object tracking. It is more reasonable to evaluate complete MOT systems without common prior detection input. To this end, we organize a challenge workshop, “Vision Meets Drone Video Multiple Object Tracking” (VisDrone-MOT2019), in conjunction with the 17-th International Conference on Computer Vision (ICCV 2019) in Seoul, Korea. Different from VisDrone-VDT2018 [67] including MOT methods with common prior detection input, we invite researchers to submit the results of MOT systems on the benchmark dataset. The comparison of the submitted algorithms can be found on the challenge website: `www.aiskyeye.com/`.

2. Related Work

In this section, we review some recent multi-object tracking methods. Since similarity learning plays important role in the MOT task, we also review related person re-id methods, which calculates discriminative appearance features of objects for better tracking performance.

2.1. Multi-Object Tracking

The goal of the MOT task is to determine the target trajectories in sequences. Most of the previous methods are tracking-by-detection strategy based. In [55], a new data association method is developed based on hierarchical relation hypergraph, which formulates the MOT task as a dense neighborhoods searching problem on the dynamically constructed affinity graph. In [27], the Bilinear LSTM model is used to improve the learning of long-term appearance models of objects. Zhu *et al.* [65] embed single object tracking into data association methods to deal with noisy detections and frequent interactions between targets. Keuper *et al.* [26] develops a correlation co-clustering model for combining low-level grouping with high-level detection and tracking. In [52], both temporal and appearance information are combined in a unified framework. To exploit different degrees of dependencies among tracklets, Wen *et al.* [54] propose a new non-uniform hypergraph based MOT method. To minimize the number of switches, Maksai and Fua [37] propose an iterative scheme of building a rich training set to learn a scoring function that is an explicit proxy for the target tracking metric. Chu and Ling [12] develop an end-to-end network including feature extraction, affinity estimation and multi-dimensional assignment.

2.2. Person Re-identification

Person re-identification (ReID) aims to identify a person of interest at other time or place, which is widely applied in the MOT task. AlignedReID [61] extracts a global feature which is jointly learned with local features. Yang *et al.* [59] propose a weighted linear coding method to learn multi-level (*e.g.*, pixel-level, patch-level and image-level) descriptors from raw pixel data in an unsupervised manner. Sun *et al.* [50] learn discriminative features using a network named part-based convolutional baseline and a refined part pooling method. Si *et al.* [47] learn context-aware feature sequences and perform attentive sequence comparison simultaneously.

Instead of pairs of images, video-based ReID methods focus on pairs of video sequences. Gao and Nevatina [18] compare four different temporal modeling methods for video-based person reID, including temporal pooling, temporal attention, RNN and 3D convnets. Li *et al.* [29] propose a new spatiotemporal attention model that automatically discovers a diverse set of distinctive body parts. Recently, Chen *et al.* [10] aim to attend to the salient parts of persons in videos jointly in both spatial and temporal domains.

3. The VisDrone-MOT2019 Challenge

As discussed above, the VisDrone-MOT2019 Challenge focuses multi-object tracking without prior detection input. That is, participants are expected to submit multiple object tracking results based on their private detections. Besides, appearance or motion models from additional data are welcome.

3.1. The VisDrone-MOT2019 Dataset

The VisDrone-MOT2019 Dataset uses the same data as in the Visdrone-VDT2018 Challenge [67]. Specifically, it consists of 79 video clips with $33,366$ frames in total, which is divided into three 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). Since the dataset is extremely challenging, we focus on five selected object categories in this challenge, *i.e.*, *pedestrian*[1], *car*, *van*, *bus*, and *truck*. Some annotated example frames are shown in Figure 1.

Since we evaluate the peformance of the overall tracking system, we do not provide the common detection input for the tracker and encourage the participants to use their own detection methods. Similar to Task 4a in Visdrone-VDT2018 [67], we use the protocol of [41] 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 (*i.e.*, 0.25, 0.50, and 0.75). The MOT algorithm is ranked by averaging the mean average precision (mAP) per object class over different thresholds. Please refer to [41] for more details.

3.2. Submitted Trackers

There are in total 12 different multi-object tracking methods submitted to the VisDrone-MOT2019 Challenge. We summarize the submitted algorithms in Table 1, and present the descriptions of the algorithms in Appendix A. Given the Faster R-CNN [44] detection input, we also evaluate 6 baseline methods (*i.e.*, GOG [42], IHTLS [15], TBD [19], CMOT [5], H^{2}T [55], and CEM [39]) using the reasonable parameters. In addition, the MOT track winner of VisDrone-VDT2018 Challenge Ctrack [67] is compared in our experiment.

All the submitted MOT methods are tracking-by-detection based. Morover, recent state-of-the-art detectors are used to provide the detection input, such as Cascade R-CNN [8], CenterNet [64], R-FCN [13], FPN [31], RetinaNet [32] and Faster R-CNN [44]. To improve the data association accuray, the re-id strategy is used to generate discriminative feature between detections, including HMTT (A.4), IITD_DeepSort (A.5), SCTrack (A.7), T&D-OF (A.9), TNT_DRONE (A.10) and VCLDCN (A.12). To capture temporal coherency, single object trackers are combined into the MOT algorithm, including KCF (DBAI-Tracker (A.1)) and DaSiameseRPN (HMTT (A.4)). Another solution is exploit temporal features such as KLT (GGDTRACK (A.3)), optical flow (Flow-Tracker (A.2), T&D-OF (A.9)), motion patterns (TrackKITSY (A.11)) and LSTM (SGAN (A.8)). OS-MOT (A.6) is a non-deep learning based method including three main modules: feature extraction [14], data association [6], and model update.

4. Results and Analysis

The results of the submissions are presented in Table 2. DBAI-Tracker (A.1), TrackKITSY (A.11) and Flow-Tracker (A.2) achieve the top 3 AP score among all submissions, respectively. All of them are based on the detections from Cascade R-CNN [8]. To adapt to the VisDrone data with many small objects, they exploit not only robust appearance representation of the object, but also temporal coherency information by single object trackers or other low-level motion patterns.

Compared to the MOT-track winner of VisDrone-VDT2018 Challenge Ctrack [67], the top 6 submitted algorithms in this year achieve much higher accuracy. The baseline methods using the Faster R-CNN detections as input do not perform well. The best result is produced by CMOT with 14.22 AP score.

DBAI-Tracker (A.1) achieves top accuracy while maintaining good efficiency, *i.e.*, running $20 \sim 50$ fps with Tesla V100 GPU. In addition, GGDTRACK (A.3) achieves good performance while maintaining reasonable efficiency without GPU cards, *i.e.*, 25 fps.

4.1. Performance Analysis by Categories

We also report the accuracy of the trackers in different object categories, including AP$_{\text{car}}$, AP$_{\text{bus}}$, AP$_{\text{trk}}$, AP$_{\text{ped}}$ and AP$_{\text{van}}$. DBAI-Tracker (A.1) performs the best in all categories expect pedestrian. Moreover, it achieves much better AP score in categories with a small amount of training data, *e.g.*, bus and truck. We speculate that the improved Cascade R-CNN [8] are effective in such case. TrackKITSY (A.11) achieves the top AP$_{\text{ped}}$ score, demonstrating the effectiveness of the extracted motion patterns for tracking small objects. It also ranks the second place in the car, truck and van categories. Flow-Tracker (A.2) ranks the third place in the

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

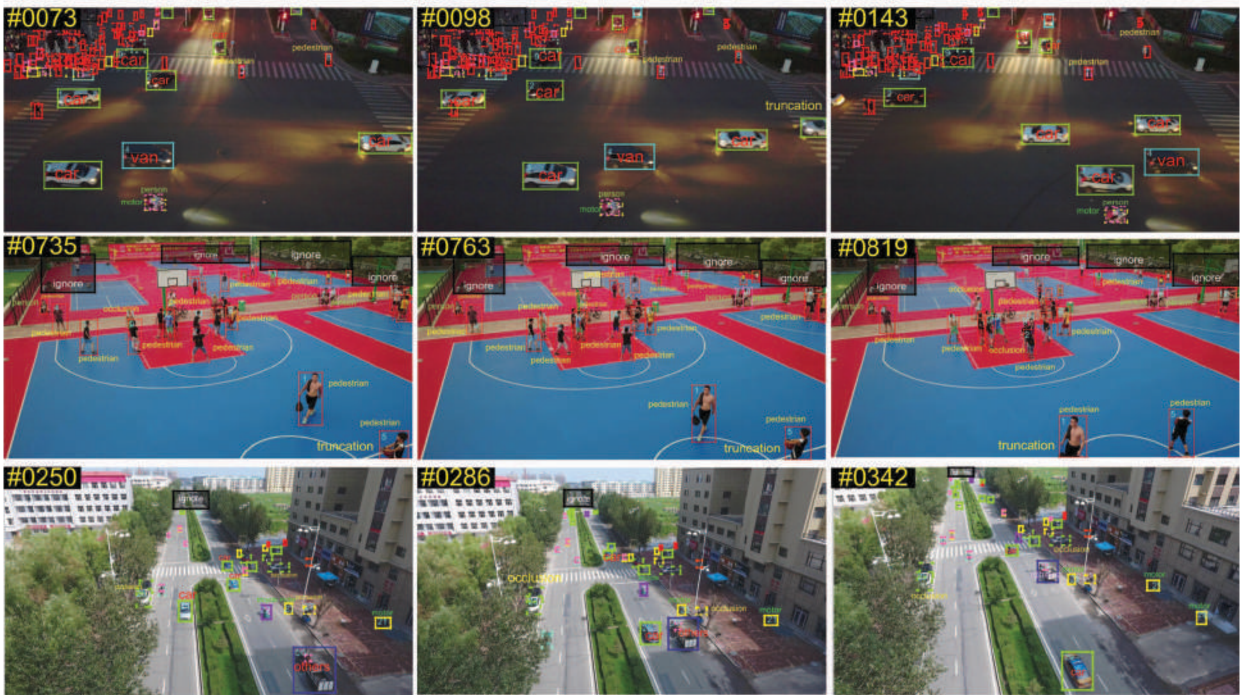

Figure 1. Some annotated example frames of MOT. The bounding boxes and the corresponding attributes of objects are shown for each sequence.

Table 1. The descriptions of the submitted MOT algorithms in the VisDrone-MOT2019 Challenge. GPUs and CPUs for training, implementation details (P for python and M for Matlab), framework, pre-trained datasets (A indicates Market1501 [62], C indicates COCO [33], M indicates MOT [38], O indicates OTB [58], U indicates CUHK [30], and × indicates that the methods do not use the pre-trained datasets) and the running speed (in FPS) are reported.

Method	GPU	CPU	Code	Framework	Pre-trained	Speed
DBAI-Tracker (A.1)	Tesla V100	Intel Xeon Platinum 8160	P	Cascade R-CNN [8]+GOG [42]	C	20 ∼ 50
Flow-Tracker (A.2)	GTX 1080Ti	Intel Xeon E5-1650v4@3.60GHz×12	P	Cascade R-CNN [8]+IoU Tracker [7]	C	5
GGDTRACK (A.3)	×	Intel Xeon E5-2650v3@2.30GHz(64GB)	P	Faster R-CNN [44]+DNF [46]	×	25
HMTT (A.4)	GTX TITAN X	Intel i7-4790K@4.00GHz	P	CenterNet [64]+IOU tracker [7]	C,O	0.4
IITD_DeepSort (A.5)	Tesla K80	Intel Xeon @1.70GHz×16	P	RetinaNet [32]+DeepSORT [57]	C	0.3
OS-MOT (A.6)	GTX980	Intel i7-6700K@4.00GHz×8(16GB)	M	auction assign [6]	×	5
SCTrack (A.7)	×	Intel i7-4720@2.60GHz	M	Faster R-CNN [44]+SCTrack [2, 1]	×	1.4
SGAN (A.8)	Titan X Pascal	Intel i7-6700@3.40GHz	P	Social-LSTM [3]	×	1.5
T&D-OF (A.9)	TITAN X MAXWELL	Intel i7-7700(48GB)	P	R-FCN [13]+MOTDT [11]	A,M,U	0.3
TNT_DRONE (A.10)	Quadro GV100/Titan Xp×2	Intel i7-7700K@4.20GHz	P,M	Faster R-CNN [43] +TrackletNet [52, 60]	M	3.2
TrackKITSY (A.11)	NVS5200M	Intel i7-6700@3.40GHz (16GB)	C++	Cascade R-CNN [8]+TrackCG [51]	×	10
VCLDAN (A.12)	GTX 1080Ti	Intel Xeon E5-2640@2.40GHz	P	DAN [49]	×	6.3

car, truck and van categories, which uses FlowNet [48] as a tracker to predict the locations of the unmatched tracks in several frames. Similarly, HMTT (A.4) ranks the second place in the bus and third place in pedestrian categories, which uses the state-of-the-art single object tracker DaSiameseRPN [68] to fill the gaps when matching IOU mechanism does not work.

4.2. Discussion

It is challenging to perform multi-object tracking on drones. The results of current submissions are far away from the requirements of practical applications. We can explore some effective techniques to follow:

- **Appearance representation.** According to the submitted MOT methods, the ReID models are useful in associating detections by exploiting discriminative features, *e.g.*, HMTT (A.4), IITD_DeepSort (A.5), SCTrack (A.7), T&D-OF (A.9), TNT_DRONE (A.10) and VCLDCN (A.12). The ReID models used in those algorithms are trained offline using external data such as Market1501 [62] and CUHK [30].

- **Motion representation.** Since the object motion pattern is complex within cameras on drones, it is important to construct robust motion model for object association, *e.g.*, KLT (GGDTRACK (A.3)), optical flow (Flow-Tracker (A.2), and LSTM (SGAN (A.8)).

Table 2. Multi-object tracking results on the VisDrone-MOT2019 `testing` set. $*$ indicates that the tracking algorithm is submitted by the VisDrone Team. The best three performers are highlighted by the red, green and blue fonts.

Method	AP	AP@0.25	AP@0.50	AP@0.75	AP_{car}	AP_{bus}	AP_{trk}	AP_{ped}	AP_{van}
DBAI-Tracker (A.1)	**43.94**	**57.32**	**45.18**	**29.32**	**55.13**	**44.97**	**42.73**	**31.01**	**45.85**
TrackKITSY (A.11)	**39.19**	**48.83**	**39.36**	**29.37**	**54.92**	**29.05**	**34.19**	**36.57**	**41.20**
Flow-Tracker (A.2)	**30.87**	**41.84**	**31.00**	**19.77**	**48.44**	26.19	**29.50**	18.65	**31.56**
HMTT (A.4)	28.67	39.05	27.88	19.08	44.35	**30.56**	18.75	**26.49**	23.19
TNT_DRONE (A.10)	27.32	35.09	26.92	19.94	38.06	22.65	33.79	12.62	29.46
GGDTRACK (A.3)	23.09	31.01	22.70	15.55	35.45	28.57	11.90	17.20	22.34
Ctrack† [67]	16.12	22.40	16.26	9.70	27.74	28.45	8.15	7.95	8.31
CMOT* [5]	14.22	22.11	14.58	5.98	27.72	17.95	7.79	9.95	7.71
IITD_DeepSort (A.5)	13.88	23.19	12.81	5.64	32.20	8.83	6.61	18.61	3.16
T&D-OF (A.9)	12.37	17.74	12.94	6.43	23.31	22.02	2.48	9.59	4.44
SCTrack (A.7)	10.09	14.95	9.41	5.92	18.98	17.86	4.86	5.20	3.58
VCLDAN (A.12)	7.50	10.75	7.41	4.33	21.63	0.00	4.92	10.94	0.00
GOG* [42]	6.16	11.03	5.30	2.14	17.05	1.80	5.67	3.70	2.55
TBD* [19]	5.92	10.77	5.00	1.99	12.75	6.55	5.90	2.62	1.79
CEM* [39]	5.70	9.22	4.89	2.99	6.51	10.58	8.33	0.70	2.38
H^{2}T* [55]	4.93	8.93	4.73	1.12	12.90	5.99	2.27	2.18	1.29
IHTLS* [15]	4.72	8.60	4.34	1.22	12.07	2.38	5.82	1.94	1.40
SGAN (A.8)	2.54	4.87	2.06	0.69	10.42	0.00	0.00	2.27	0.00
OS-MOT (A.6)	0.16	0.18	0.18	0.13	0.00	0.00	0.71	0.00	0.09

5. Conclusion

This paper concludes the VisDrone-MOT2019 Challenge, where 12 MOT algorithms are submitted. DBAI-Tracker (A.1), TrackKITSY (A.11) and Flow-Tracker (A.2) achieve the top three AP scores among all submissions, *i.e.*, 43.94, 39.19 and 30.87, respectively. Notably, they rely on state-of-the-art object detector, *i.e.*, Cascade R-CNN [8]. The VisDrone-MOT2019 Challenge was successfully held on October 27, 2019, which is a part of the "Vision Meets Drones: A Challenge" workshop in conjunction with the 17-th International Conference on Computer Vision (ICCV 2019). We hope this challenge can provide a unified platform for multiple object tracking evaluation on drones.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 61502332, 61876127 and 61732011, Natural Science Foundation of Tianjin Under Grants 17JCZDJC30800, Key Scientific and Technological Support Projects of Tianjin Key R&D Program 18YFZCGX00390 and 18YFZCGX00680 and JD Digits.

A. Submitted Trackers

In the appendix, we summarize 12 tracking methods submitted in the VisDrone-MOT2019 Challenge, which are ordered alphabetically.

A.1. DeepBlueAI-Tracker (DBAI-Tracker)

Zhipeng Luo, Yuehan Yao, Zhenyu Xu, Feng Ni and Bing Dong
{luozp, yaoyh, xuzy, nif, dongb}@deepblueai.com

DBAI-Tracker follows the pipeline of tracking by detection. Strong detection model is designed in Iou tracker [7]. GOG [42] and KCF [21] are also used. Our detection model is Cascade R-CNN [8] and IoU tracker [7]. We use FPN [31] based multi-scale feature maps to exploit robust representation of the object. Besides, GCNet [9] are used for better performance.

A.2. Multiple Object Tracking with Motion and Appearance Cues (Flow-Tracker)

Weiqiang Li, Jiatong Mu and Guizhong Liu
{lwq1230,m625329163}@stu.xjtu.edu.cn,
liugz@xjtu.edu.cn

Flow-Tracker is based on Cascade R-CNN [8] and IoU Tracker [7]. For detection, we use Cascade RCNN as base detector and the backbone is ResNet-101. In order to improve detection results, the deformable convolution was added in basic network. To supplement the training data, we use COCO train set to pretrain our detector and then fine-tune it on VisDrone2019-MOT train set. For tracking, we make some improvements on IoU Tracker. Our tracking framework can be divided into three parts. First, we use an optical flow network to predict the motion between two frames and predict the position of tracks on the current frame, which can solve the problem of camera motion. Then we compute IoU between the tracks and the detections. If it is higher than a thresh, we think they are matched. Second, we extract the appearance features of unmatched tracks and detections. Then we compute appearance distance and IoU distance between unmatched tracks and detections. If they meet the matching criteria at the same time, we think they are matched. Final, for those unmatched tracks, we use FlowNet [48] as a tracker to continue predicting their position for several frames. If they are matched successfully within these frames, we believe these tracks can continue; otherwise, we think these objects have disappeared. If the optical flow prediction is performed each frame, the tracking speed is 5 fps; and if the optical flow is predicted only when the camera is moving, the speed can reach 100 fps.

A.3. Costflow tracker Learning from Generalized Graph Differences (GGDTRACK)

Håkan Ardö and Mikael Nilsson
hakanad@axis.com, micken@maths.lth.se

The basic idea behind GGDTRACK [4] is to build a graph with object detections as vertices and use sparse optical flow feature point tracks, KLT-tracks[2], to connect these vertices with edges. Then a flow capacity of one is assigned to each edge and a network flow problem is solved. To allow objects to occlude each other, long range connections can be added to the graph. The problem is that during occlusion a lot of feature point tracks will jump from one object to the other, which means that the feature point tracks are not reliably in such situations. In order to address this issue, the common used linear motion model is utilized in this setup [34]. All feasible solutions to the network flow problem are embedded into a one dimensional feature space consisting of a score with the aim of making the score of the correct solution higher than all other solutions. Then a linear program is used during inference to efficiently search for the correct solution. We also introduce a data representation denoted generalized graph differences and show that it allows the training to be performed efficiently both in terms of training speed and data needs. The setup proposed is similar in sprit to recent works [17, 46]. However, they need to solve a linear program or a general convex problem respectively for each example during each step of the SGD-like optimisation, which is time consuming operations. Also, there is no need to approximate and reformulate the model as Schulter *et al.* [46] does. The small and efficient representation of generalized graph differences gives the potential for using larger graphs which is needed to fill in missing detections during, for example, occlusions by long range connections in the graphs. A key insight here is that lots of small generalized graph differences can be generated from a single annotated video sequence and be utilized as training data. This gives a good way to utilize the annotations as much as possible in order to avoid the need for extreme amounts of training data. We also show that by using average-pooling it is possible to use features for connecting detections that are derived from a varying number of feature point tracks of varying length.

A.4. A hierarchical multi-target tracker based on detection for drone vision (HMTT)

Siyang Pan, Zhihang Tong and Yanyun Zhao
{pansiyang, tongzh, zyy}@bupt.edu.cn

HMTT is based on CenterNet [64], IOU-tracker [7] and DaSiameseRPN [68]. During the stage of determining object position by adopting CenterNet, we first divide each classification into two categories, depending upon whether the object is photographed from right above. Then, we perform the detection results filtering by generating tracklets employing IOU-tracker with the aid of Hungarian algorithm. Using the bounding box results with tracklet-id abandoned, we restart the association stage. Differently from IOU-tracker, this time DaSiameseRPN and Kalman filtering are additionally employed to fill the gaps when matching with IOU does not work. Meanwhile, in case of camera's sudden move, SIFT points matching between consecutive frames estimates the affine transformation matrix, which assists bounding box association as well as single target tracking. Each trajectory's appearance feature gotten from OSNet [63] is used to measure its distance from other ones and we simply merge two trajectories if their distance is close enough.

A.5. Improved simple online and realtime tracking with a deep association metric (IITD_DeepSort)

Ajit Jadhav, Prerana Mukherjee, Vinay Kaushik and Brejesh Lall
{jadhavajit.j16,prerana.m}@iiits.in,
vinaykaushik15@gmail.com, brejesh@ee.iitd.ac.in

IITD_DeepSort is derived from DeepSORT [57]. The RetinaNet architecture [32] is used for object detection with modifications to the anchor parameters for improving small object detection as well as detection of objects with large variance in sizes. Increased range of scales helps in detection of objects across a wider variety in object sizes while incorporating finer scales improves the detection of small objects. Squeeze-and-Excitation(SE) [23] blocks are used to adaptively recalibrate channel-wise feature

[2] https://cecas.clemson.edu/~stb/klt/

responses by explicitly modelling interdependencies between channels. But instead of using the SE blocks in the ResNet50 architecture, we pass the features from the backbone feature layer to an SE block before feeding the features to the feature pyramid network. On the oother hand, a deep association metric is used along with the SORT algorithm [57] to improve the performance of SORT which helps to track objects through longer periods of occlusions, effectively reducing the number of identity switches. The network for deep assocaiation metric is trained using Deep Cosine Metric Learning for Person ReIdentification [56]. The object patches from the training set are resized to a size of 128×128 and are used as input for this network for training.

A.6. Auction algorithm for network flow problem (OS-MOT)

Yong Wang, Lu Ding, Robert Laganière, Zhuojin Sun, Chunhui Zhang and Wei Shi
ywang6@uottawa.ca, dinglu@sjtu.edu.cn,
laganier@eecs.uottawa.ca, harvards@gmail.com,
zhangchunhui@iie.ac.cn, weishi_insky@126.com

OS-MOT is composed of three main modules: feature extraction, data association, and model update. Specifically, targets are modeled by their visual appearance (via HOG feature) and their spatial location (via bounding boxes). The auction assign [6] algorithm is used for associating detections to targets. Finally, model updating is implemented.

A.7. Semantic Color Correlation Tracker (SCTrack)

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

SCTrack [2, 1] is a time-efficient detection-based multi-object tracking system. It employs a three-step cascaded data association scheme that combines a fast spatial distance only short-term data association, a robust tracklet linking step using discriminative object appearance models, and an explicit occlusion handling unit relying not only on tracked objects motion patterns but also on environmental constraints such as presence of potential occluders in the scene.

A.8. Long-Short Term Prediction for Tracking (SGAN)

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

SGAN uses the Social-LSTM [3] for long term prediction of the objects. At the same time, the appearance of the detections in adjacent frames are used for short term prediction of the objects. Then a GAN network use the two predictions generating the final position mask for the objects. The radius of the neighbourhood is 32 pixels. Correlation layer and 3 Convolutional layers are used in generating masks and 2 Convolutional layers are used for discriminating the ground-truth and generated mask.

A.9. Tracking by Detection with Optical Flow (T&D-OF)

Xinyu Zhang, Xin Chen, Shuhao Chen, Chang Liu, Dong Wang and Huchuan Lu
{chenxin3131,lcqctk0914,shuhaochn,
zhangxy71102}@mail.dlut.edu.cn,
{wdice,lhchuan}@dlut.edu.cn

T&D-OF is a modified version of MOTDT [11]. First, the R-FCN [13] based classifier is removed, and we add optical flow generated by FlowNetv2 [25] as additional cue for tracking. The ReID part of our model[3] is trained on MOT16 [38], Market1501 [62], CUHK01 and CUHK03 [30] datasets. We do not perform fine-tuning on the VisDrone data.

A.10. TrackletNet Tracker in Drone based scenes (TNT_DRONE)

Haotian Zhang, Yanting Zhang, Gaoang Wang, Tsung-wei Huang and Jenq-Neng Hwang
haotiz@uw.edu, zhangyt@bupt.edu.cn,
{gaoang,twhuang,hwang}@uw.edu

TNT_DRONE follows the “tracking by detection” scheme. The Faster R-CNN [43] is trained to detect the objects in the images. Given the detections in different frames, detection association is computed to generate tracklets for the Vertex Set V (denotes different tracklets). After that, each two tracklets are put into a novel TrackletNet [52, 60] to measure the connectivity, which formed the similarity on the Edge Set E. A graph model G can be derived from V and E. Finally, the tracklets with the same ID are grouped into one cluster using the graph partition approach [24].

A.11. Online multi-object tracking using joint domain information in traffic scenarios (TrackKITSY)

Wei Tian, Jinrong Hu, Yuduo Song, Zhaotang Chen, Long Chen and Martin Lauer

[3] https://github.com/longcw/MOTDT

{*wei.tian, martin.lauer*}*@kit.edu, utppm@student.kit.edu,*
hujr3@mail2.sysu.edu.cn, 761042366@qq.com,
chenl46@mail.sysu.edu.cn

TrackKITSY is based on the detections of the Cascade R-CNN [8]. Several modifications are applied to the original Cascade R-CNN to adapt to this dataset. First, to fit the big variance of bounding box aspect ratio, we add more anchors with different aspect ratios in the RPN. Second, photo metric distortion and random cropping are used as data augmentation in training. Third, lower IoU threshold is used in non-maximum-suppression (NMS) in the post-processing. The reason is that, according to our observation, the objects with valid annotation seldom overlap, while the overlapping objects are usually in the "ignored" region. Last, multi-scale training and testing are used to improve the precision. The tracking module is based on the work [51] with modifications adapted to the current dataset.

A.12. VCL's Deep Affinity Network (VCLDAN)

Zhibin Xiao
xzb18@mails.tsinghua.edu.cn

VCLDAN is based on the DAN tracker [49] and adds the score and category id information to the output. It can learn compact yet comprehensive features of pre-detected objects at several levels of abstraction, and perform exhaustive pairing permutations of those features in any two frames to infer object affinities. The open source implementation is available at `https://github.com/shijieS/SST.git`.

References

[1] N. M. Al-Shakarji, F. Bunyak, G. Seetharaman, and K. Palaniappan. Multi-object tracking cascade with multi-step data association and occlusion handling. In *AVSS*, pages 1–6, 2018.

[2] N. M. Al-Shakarji, G. Seetharaman, F. Bunyak, and K. Palaniappan. Robust multi-object tracking with semantic color correlation. In *AVSS*, pages 1–7, 2017.

[3] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, F. Li, and S. Savarese. Social LSTM: human trajectory prediction in crowded spaces. In *CVPR*, pages 961–971, 2016.

[4] H. Ardö and M. Nilsson. Multi target tracking by learning from generalized graph differences. *CoRR*, abs/1908.06646, 2019.

[5] S. H. Bae and K. Yoon. Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning. In *CVPR*, pages 1218–1225, 2014.

[6] D. P. Bertsekas. Auction algorithms for network flow problems: A tutorial introduction. *Comp. Opt. and Appl.*, 1(1):7–66, 1992.

[7] E. Bochinski, V. Eiselein, and T. Sikora. High-speed tracking-by-detection without using image information. In *AVSS*, pages 1–6, 2017.

[8] Z. Cai and N. Vasconcelos. Cascade R-CNN: delving into high quality object detection. In *CVPR*, pages 6154–6162, 2018.

[9] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. *CoRR*, abs/1904.11492, 2019.

[10] G. Chen, J. Lu, M. Yang, and J. Zhou. Spatial-temporal attention-aware learning for video-based person re-identification. *TIP*, 28(9):4192–4205, 2019.

[11] L. Chen, H. Ai, Z. Zhuang, and C. Shang. Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In *ICME*, pages 1–6, 2018.

[12] P. Chu and H. Ling. Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking. In *ICCV*, 2019.

[13] J. Dai, Y. Li, K. He, and J. Sun. R-FCN: object detection via region-based fully convolutional networks. In *NeuIPS*, pages 379–387, 2016.

[14] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.

[15] C. Dicle, O. I. Camps, and M. Sznaier. The way they move: Tracking multiple targets with similar appearance. In *ICCV*, pages 2304–2311, 2013.

[16] D. Du, Y. Qi, H. Yu, Y. Yang, K. Duan, G. Li, W. Zhang, Q. Huang, and Q. Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. In *ECCV*, 2018.

[17] D. Frossard and R. Urtasun. End-to-end learning of multi-sensor 3d tracking by detection. In *ICRA*, pages 635–642, 2018.

[18] J. Gao and R. Nevatia. Revisiting temporal modeling for video-based person reid. *CoRR*, abs/1805.02104, 2018.

[19] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun. 3d traffic scene understanding from movable platforms. *TPAMI*, 36(5):1012–1025, 2014.

[20] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *CVPR*, pages 3354–3361, 2012.

[21] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. High-speed tracking with kernelized correlation filters. *TPAMI*, 37(3):583–596, 2015.

[22] M. Hsieh, Y. Lin, and W. H. Hsu. Drone-based object counting by spatially regularized regional proposal network. In *ICCV*, 2017.

[23] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018.

[24] T.-W. Huang, J. Cai, H. Yang, H.-M. Hsu, and J.-N. Hwang. Multi-view vehicle re-identification using temporal attention model and metadata re-ranking. In *CVPRW*, 2019.

[25] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*, pages 1647–1655, 2017.

[26] M. Keuper, S. Tang, B. Andres, T. Brox, and B. Schiele. Motion segmentation & multiple object tracking by correlation co-clustering. *TPAMI*, 2018.

[27] C. Kim, F. Li, and J. M. Rehg. Multi-object tracking with neural gating using bilinear LSTM. In *ECCV*, pages 208–224, 2018.

[28] L. Leal-Taixé, A. Milan, I. D. Reid, S. Roth, and K. Schindler. Motchallenge 2015: Towards a benchmark for multi-target tracking. *CoRR*, abs/1504.01942, 2015.

[29] S. Li, S. Bak, P. Carr, and X. Wang. Diversity regularized spatiotemporal attention for video-based person re-identification. In *CVPR*, pages 369–378, 2018.

[30] W. Li, R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, pages 152–159, 2014.

[31] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 936–944, 2017.

[32] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *ICCV*, pages 2999–3007, 2017.

[33] T. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: common objects in context. In *ECCV*, pages 740–755, 2014.

[34] W. Luo, X. Zhao, and T. Kim. Multiple object tracking: A review. *CoRR*, abs/1409.7618, 2014.

[35] S. Lyu, M. Chang, D. Du, W. Li, Y. Wei, M. D. Coco, P. Carcagnì, and et al. UA-DETRAC 2018: Report of AVSS2018 & IWT4S challenge on advanced traffic monitoring. In *AVSS*, pages 1–6, 2018.

[36] S. L. S. Lyu, M. Chang, D. Du, L. Wen, H. Qi, Y. Li, Y. Wei, L. Ke, T. Hu, M. D. Coco, P. Carcagnì, and *et al.* UA-DETRAC 2017: Report of AVSS2017 & IWT4S challenge on advanced traffic monitoring. In *AVSS*, pages 1–7, 2017.

[37] A. Maksai and P. Fua. Eliminating exposure bias and metric mismatch in multiple object tracking. In *CVPR*, pages 4639–4648, 2019.

[38] A. Milan, L. Leal-Taixé, I. D. Reid, S. Roth, and K. Schindler. MOT16: A benchmark for multi-object tracking. *CoRR*, abs/1603.00831, 2016.

[39] A. Milan, S. Roth, and K. Schindler. Continuous energy minimization for multitarget tracking. *TPAMI*, 36(1):58–72, 2014.

[40] M. Mueller, N. Smith, and B. Ghanem. A benchmark and simulator for UAV tracking. In *ECCV*, pages 445–461, 2016.

[41] E. Park, W. Liu, O. Russakovsky, J. Deng, F.-F. Li, and A. Berg. Large Scale Visual Recognition Challenge 2017. http://image-net.org/challenges/LSVRC/2017.

[42] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In *CVPR*, pages 1201–1208, 2011.

[43] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NeurIPS*, pages 91–99, 2015.

[44] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *TPAMI*, 39(6):1137–1149, 2017.

[45] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *ECCV*, pages 549–565, 2016.

[46] S. Schulter, P. Vernaza, W. Choi, and M. Chandraker. Deep network flow for multi-object tracking. In *CVPR*, pages 2730–2739, 2017.

[47] J. Si, H. Zhang, C. Li, J. Kuen, X. Kong, A. C. Kot, and G. Wang. Dual attention matching network for context-aware feature sequence based person re-identification. In *CVPR*, pages 5363–5372, 2018.

[48] D. Sun, X. Yang, M. Liu, and J. Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *CVPR*, pages 8934–8943, 2018.

[49] S. Sun, N. Akhtar, H. Song, A. Mian, and M. Shah. Deep affinity network for multiple object tracking. *CoRR*, abs/1810.11780, 2018.

[50] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang. Beyond part models: Person retrieval with refined part pooling (and A strong convolutional baseline). In *ECCV*, pages 501–518, 2018.

[51] W. Tian, M. Lauer, and L. Chen. Online multi-object tracking using joint domain information in traffic scenarios. *TITS*, abs/1810.11780, 2019.

[52] G. Wang, Y. Wang, H. Zhang, R. Gu, and J. Hwang. Exploit the connectivity: Multi-object tracking with trackletnet. In *ACM MM*, 2019.

[53] L. Wen, D. Du, Z. Cai, Z. Lei, M. Chang, H. Qi, J. Lim, M. Yang, and S. Lyu. UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking. *CoRR*, abs/1511.04136, 2015.

[54] L. Wen, D. Du, S. Li, X. Bian, and S. Lyu. Learning non-uniform hypergraph for multi-object tracking. In *AAAI*, pages 8981–8988, 2019.

[55] L. Wen, W. Li, J. Yan, Z. Lei, D. Yi, and S. Z. Li. Multiple target tracking based on undirected hierarchical relation hypergraph. In *CVPR*, pages 1282–1289, 2014.

[56] N. Wojke and A. Bewley. Deep cosine metric learning for person re-identification. In *WACV*, pages 748–756, 2018.

[57] N. Wojke, A. Bewley, and D. Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*, pages 3645–3649, 2017.

[58] Y. Wu, J. Lim, and M. Yang. Object tracking benchmark. *TPAMI*, 37(9):1834–1848, 2015.

[59] Y. Yang, L. Wen, S. Lyu, and S. Z. Li. Unsupervised learning of multi-level descriptors for person re-identification. In *AAAI*, pages 4306–4312, 2017.

[60] H. Zhang, G. Wang, Z. Lei, and J. Hwang. Eye in the sky: Drone-based object tracking and 3d localization. In *ACM MM*, 2019.

[61] X. Zhang, H. Luo, X. Fan, W. Xiang, Y. Sun, Q. Xiao, W. Jiang, C. Zhang, and J. Sun. Alignedreid: Surpassing human-level performance in person re-identification. *CoRR*, abs/1711.08184, 2017.

[62] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *CVPR*, pages 1116–1124, 2015.

[63] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang. Omni-scale feature learning for person re-identification. *CoRR*, abs/1905.00953, 2019.

[64] X. Zhou, D. Wang, and P. Krähenbühl. Objects as points. *CoRR*, abs/1904.07850, 2019.

[65] J. Zhu, H. Yang, N. Liu, M. Kim, W. Zhang, and M. Yang. Online multi-object tracking with dual matching attention networks. In *ECCV*, pages 379–396, 2018.

[66] P. Zhu, L. Wen, X. Bian, H. Ling, and Q. Hu. Vision meets drones: A challenge. *CoRR*, abs/1804.07437, 2018.

[67] P. Zhu, L. Wen, D. Du, X. Bian, H. Ling, Q. Hu, and et al. Visdrone-vdt2018: The vision meets drone video detection and tracking challenge results. In *ECCVW*, pages 496–518, 2018.

[68] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu. Distractor-aware siamese networks for visual object tracking. In *ECCV*, pages 103–119, 2018.