Multi-Object Tracking in Drone Videos

Daniel Stadler

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
daniel.stadler@kit.edu

Abstract

In this report, three popular methods for multi-pedestrian tracking are extended to a multi-category setting and tested on a large drone-based dataset. A thorough comparison of the algorithms is presented and a common shortcoming is identified. Building on this, a new tracking-by-detection based approach is developed that outperforms the other methods by a large margin. In addition, a state-of-the-art object detection model is adapted for the drone imagery, since no public detections are available for the dataset.

1 Introduction

Multi-object tracking (MOT) in drone videos has several applications ranging from sports analysis to traffic surveillance. Challenges arise not only from complicated scenes with occlusions or fast moving objects but also from different camera altitudes and angles resulting in a large variance of object size and appearance. To solve these problems, most MOT approaches follow the tracking-by-detection paradigm, where the tracking of objects is divided into two subtasks – detection and association. This procedure has the advantage that the vast improvements of deep learning based object detectors from the last years can be directly applied. After detecting objects in each frame of a video independently, the goal of the subsequent association step is to link detections of the same object to tracks with a unique ID. This is the part, where the main differences of existing MOT frameworks lie. While some simple methods only use the bounding box information of the detections [2, 3, 4], other more sophisticated approaches use separate networks adopted from person re-identification to extract appearance features of the underlying objects [1, 14]. Further ideas that emerged from the analysis of persons are to use human pose information [7] or the interaction of people [10] to assist the association of detections to tracks. Apart from that, the movement of objects often is considered with a motion model (MM) following a simple constant velocity assumption (CVA) between two consecutive frames when the sampling rate is high or by applying a Kalman filter, for example.

In this report, three approaches originally designed for multi-person tracking are extended in order to treat multiple object categories. After a qualitative comparison, a new tracking method is developed upon the identification of a common shortcoming of the existing approaches. Furthermore, a state-of-the-art object detector is trained to boost the performance of the applied tracking-by-detection based methods and to allow a fair comparison. The superior performance of the proposed tracker is shown through experiments on a large drone-based video dataset.

2 State-of-the-Art MOT

In this chapter, three popular MOT frameworks originally developed for person tracking are described and extended to the multi-category context. Whereas the first two follow the predominant tracking-by-detection paradigm, the latter one proposes a new concept to perform the association step implicitly. After a qualitative comparison, a new approach is developed that builds upon the weaknesses of the existing approaches.

2.1 Tracking-by-Detection Approaches

One of the simplest trackers is the IOU tracker [3]. It calculates the Intersection over Union (IoU) of all possible assignments and follows a greedy matching strategy that starts to link the detection and track with the highest IoU. Whereas in this tracker, no visual information is used, the advanced V-IOU tracker [4] leverages a KCF [8] as single object tracker to bypass missing detections. Since the KCF works class-agnostic, the V-IOU tracker can be directly applied in a multi-category setting, linking only detections from the same class together.

As a second tracking-by-detection based method, Deep SORT [14] is chosen, since it is one of the most popular MOT frameworks. As a further development of the SORT algorithm [2], it models the movement of objects with a Kalman filter and uses the motion information by calculating the Mahalanobis distance between predicted Kalman states and new detections. However, this motion metric is not used directly in the association step but only to restrict the area of possible matches for a track, i.e., as a gating function. For associating detections to tracks within their gating area determined by the motion metric, appearance features are extracted using a CNN adopted from person re-identification. The visual features of a track and a detection are compared and if their distance is below a threshold they can be linked. Instead of a greedy matching strategy, the Hungarian algorithm [9] is used for an optimal assignment. Although the separate model for generating appearance features was only trained on person data, it is found that the extracted features are also suitable for comparing objects of other categories like cars or buses. Therefore, Deep SORT also can simply be extended to track multiple categories.

2.2 Tracktor

In contrast to the aforementioned trackers, Tracktor [1] goes beyond the tracking-by-detection procedure. Detections of the previous frame are used as additional region proposals in the second stage of a Faster R-CNN [12] detector and regressed to the new positions in the current frame. Hence, no association step is needed and the tracking is done implicitly. Tracks are stopped if the score of the classification branch falls below a threshold and the detector runs in parallel to start additional tracks when new objects appear in the scene. Since the regression does not work when large object displacements are present, the method is extended with a CVA as MM and a camera motion compensation (CMC) model that applies the Enhanced Coefficient Maximization technique from [6]. Additionally, a re-identification model checks for new detections whether they belong to earlier interrupted tracks, so that occlusions can be bypassed, leading to the improved version Tracktor++. Although the method is developed for tracking persons, it can be extended to multi-category tracking. For this, the score vector of the classification branch is taken when regressing a track and if its maximum does not correspond to the same category as in the previous frame the track is stopped.

2.3 Own Approach: PAS Tracker

The goal is to build a tracker that uses as much information as possible in the association step and combines the different cues of objects like **p**osition, **a**ppearance and **s**ize in a sophisticated way in order to use all of the information at the same time. In contrast, the previously described algorithms do not use all available information and apply one cue after another: The V-IOU tracker considers appearance information only in the post processing basing the association solely on IoU, whereas Deep SORT takes position information only for gating and relies mainly on appearance features for linking detections. Tracktor++ takes appearance of objects only into account to retrieve lost tracks, but does not use this source of information in the association step.

To overcome the aforementioned limitations, the similarity measure between the position of a detection and a track should fulfill the following three requirements. First, the center of objects shall be directly compared instead of using the IoU, which is not accurate enough for densely packed small objects as often present in the drone context. Second, a similar gating mechanism to inhibit impossible matches as in Deep SORT is desirable. Therefore, the position similarity has to be zero for too large displacements. Third, in order to enable a straightforward combination with other similarity measures, the metric should be normalized between zero and one. Given the position of a detection $\mathbf{p}_D$ and the position of a track $\mathbf{p}_T$ (after MM and CMC) with center coordinates $\mathbf{p} = (x, y)$, the position similarity s_p is calculated as follows:

$$s_\mathrm{p} = \max(1 - \lambda_\mathrm{p} ||(\mathbf{p}_T - \mathbf{p}_D) \oslash \mathbf{z}_D||, 0) \tag{2.1}$$

$\oslash$ denotes the element-wise division, $||$ the Euclidean norm and $\mathbf{z}_D = (w, h)$ the size of the detection, i.e. the width and height of the bounding box. The normalization w.r.t. the object size accounts for varying camera altitudes that lead to differently large displacements in the image. The hyperparameter λ_p tunes the size of the gating area, where the position similarity is not zero. A good choice of this value is related to the displacements of objects between frames, thus to the velocity of the moving objects and the camera frame rate ($24fps$ in the VisDrone MOT dataset [15]); λ_p is empirically set to 0.3.

As a second similarity measure, the size of objects is compared. Similar to the position information, the IoU also reflects size similarity but is not very accurate, since it does not measure position and size similarity independently. To get a maximum similarity score of one, the following formula to calculate the size similarity s_z is used:

$$s_\mathrm{z} = 1 - ||(\mathbf{z}_T - \mathbf{z}_D) \oslash (\mathbf{z}_T + \mathbf{z}_D)|| \tag{2.2}$$

For a visual comparison of detections and tracks, the improvements from the person re-identification community are leveraged using a state-of-the-art model from [11]. With this model, 2048-dimensional feature vectors are extracted for a detection $\boldsymbol{\theta}_D$ or a track $\boldsymbol{\theta}_T$. Then, the appearance similarity s_a is calculated as cosine similarity like in the Deep SORT framework:

$$s_\mathrm{a} = \frac{\boldsymbol{\theta}_T \cdot \boldsymbol{\theta}_D^\mathrm{T}}{||\boldsymbol{\theta}_T|| \cdot ||\boldsymbol{\theta}_D||} \tag{2.3}$$

Since s_a also gets one for maximum similarity, the three aforementioned metrics can be easily combined to a joint similarity measure s in order to use all the information at the same time in the association step:

$$s = s_\mathrm{p} \cdot s_\mathrm{a} \cdot s_\mathrm{z} \tag{2.4}$$

This similarity is calculated for each track-detection pair and an optimal assignment is achieved with the Hungarian algorithm. A CVA is taken as MM, since

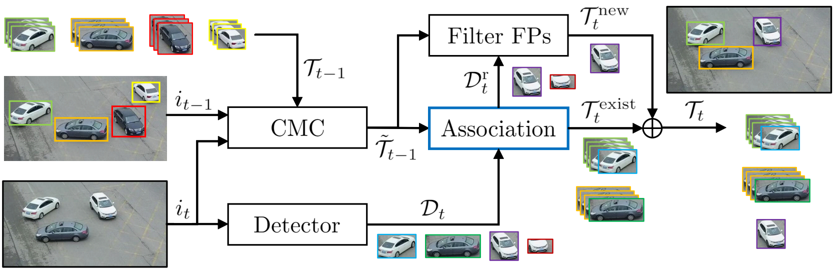
$\mathcal{T}_{t-1}$
i_{t-1}
CMC
i_t
Detector
$\tilde{\mathcal{T}}_{t-1}$
Filter FPs
$\mathcal{D}_t^{\mathrm{r}}$
Association
$\mathcal{D}_t$
$\mathcal{T}_t^{\mathrm{new}}$
$\mathcal{T}_t^{\mathrm{exist}}$
$\mathcal{T}_t$

Figure 2.1: Workflow of the proposed PAS tracker [13]. A detector takes as input the current frame i_t and generates a set of detections $\mathcal{D}_t$. Then, a CMC model calculates a transformation using the current frame i_t and the previous frame i_{t-1}. This transformation is applied on the previously found tracks $\mathcal{T}_{t-1}$ yielding the tracks $\tilde{\mathcal{T}}_{t-1}$ that are aligned with the current frame. For each track-detection pair, a similarity measure is calculated and the detections with a high similarity are assigned to the existing tracks $\mathcal{T}_t^{\mathrm{exist}}$. Before starting new tracks $\mathcal{T}_t^{\mathrm{new}}$, the remaining un-assigned detections $\mathcal{D}_t^r$ go into a module that filters false positive detections.

in drone-based imagery usually a large frame rate is available and a Kalman filter yielded no better results in the experiments. To compensate for fast camera movements, the same CMC model as in Tracktor++ is adopted. As a further component, a simple yet effective module to filter false positive detections in crowded scenarios is introduced, since these cause many ID switches. For each new detection that has not been assigned in the association step, its overlap with existing tracks is computed and the detection is removed if the overlap is too high (>0.8) arguing that it is unlikely for objects to appear at positions where already other tracks are present. The complete workflow of the PAS tracker is visualized in Figure 2.1.

3 Experiments

At first, an overview of the dataset on which the presented tracking methods have been evaluated is given. Next, the applied object detector and the adaptations made to cope with the drone-based imagery are described. Finally, the results of the conducted experiments are presented.

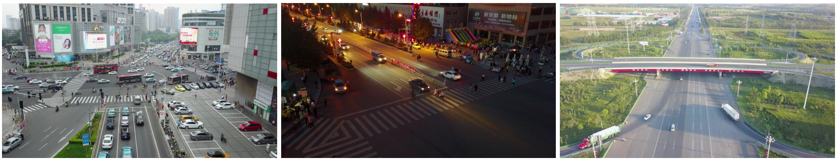

Figure 3.1: Example images of the VisDrone MOT dataset [15].

3.1 Dataset

To analyze the performance of state-of-the-art MOT methods in the context of drone imagery, a suitable dataset is needed. For this purpose, the VisDrone MOT dataset [15] is chosen, since it is the largest drone-based dataset for MOT that is publicly available. The dataset consists of 96 videos comprising about 40,000 frames with resolutions up to 3840 × 2160 pixels and is divided into 4 splits – *train* (56), *val* (7), *test-dev* (17) and *test-challenge* (16). Note that the annotations of the test-challenge split are hidden and used for a yearly challenge hosted by the VisDrone team. Therefore, the test-dev split is used for evaluation. The five categories *pedestrian*, *car*, *van*, *truck* and *bus* are evaluated as it is done in the challenge. Figure 3.1 shows some example images of VisDrone MOT. The dataset is very challenging due to a high variance in camera altitude and viewing angle leading to diverse object appearances and sparse object distributions. Furthermore, both day and night scenes exist.

3.2 Object Detector

Since no public detections are provided with the dataset, an own detector is trained on the train split of VisDrone MOT. A Cascade R-CNN [5] is used, as the drone images comprise a lot of small objects where this network has its strengths performing the bounding box regression several times with increasing accuracy requirements during training. To adapt the Cascade R-CNN to the dataset, the training is performed on patches of 600 × 600 pixels and the default anchor sizes are halved to account for the small object sizes. Similarly, to consider the larger number of objects in one image, the number of proposals is doubled. To further

Table 3.1: Results of the Cascade R-CNN detector with different test-strategies.

Cascade R-CNN	AP	$AP_{0.5}$	$AP_{0.75}$	AP_s	AP_m	AP_l
Baseline	35.4	62.5	35.2	12.3	38.9	54.7
+ more proposals	35.8	63.6	35.4	12.5	39.2	54.8
+ multi-scale testing	38.4	67.7	37.6	16.7	42.4	55.7
+ horizontal flipping	**39.2**	**68.5**	**38.4**	**17.8**	**43.2**	**56.6**

improve the detection performance, multi-scale testing and horizontal flipping are used. The influence of these strategies is evaluated on the test-dev split of VisDrone MOT and the resulting average precisions (APs) are summarized in Table 3.1.

3.3 Tracking Results

For a fair comparison, all evaluated tracking methods use the same set of detections generated by the Cascade R-CNN detector with all test-time improvements (see Table 3.1). For Tracktor++, the default Faster R-CNN is exchanged with the trained Cascade R-CNN to use the superior detector also for the proposal regression that implicitly performs the association. Similarly, for all methods, the same re-identification model from [11] and CMC model from [6] are taken, if applicable. The tracking results on the test-dev split are shown in Table 3.2. Note that the AP for MOT differs from the AP for object detection.

The PAS tracker outperforms the other methods by a large margin for both short- ($AP_{0.25}$), middle- ($AP_{0.5}$) and long-term tracking ($AP_{0.75}$) as well as for all object categories. The V-IOU tracker performs the worst, since it only uses IoU for the association. The IoU is not as accurate as the position and the size similarity of the PAS tracker, especially in crowded scenes with small object sizes. This is reflected in the very low AP_{ped} value for pedestrian tracking, as in the VisDrone MOT dataset pedestrians often appear in groups. The Deep SORT algorithm does not rely on IoU but bases its association solely on appearance similarity. However, the extraction of appearance features is harmed by nearby

Table 3.2: Comparison of the PAS tracker with other tracking approaches from the literature. Note that all methods use the same object detector.

Tracker	AP	$\text{AP}_{0.25}$	$\text{AP}_{0.5}$	$\text{AP}_{0.75}$	AP_{car}	AP_{bus}	AP_{trk}	AP_{ped}	AP_{van}
V-IOU	26.4	34.5	29.2	15.6	40.9	36.4	22.7	7.8	24.4
Deep SORT	33.2	51.0	35.0	13.6	31.9	58.3	30.0	21.3	24.5
Tracktor++	34.3	48.6	35.5	18.8	50.6	40.0	32.8	20.2	27.8
PAS	**50.8**	**66.1**	**52.5**	**33.8**	**62.7**	**81.2**	**43.9**	**30.3**	**35.9**

overlapping objects under occlusion and no precise position information is available in the association. In an ablative experiment, it is found that the precise position similarity has the most impact on the tracking performance in the PAS tracker. Whereas Tracktor++ achieves state-of-the-art results on other tracking benchmarks, it struggles in the VisDrone MOT dataset, mainly at small objects and in crowded scenes, since the bounding box regression is sensitive to jumping onto nearby objects. Using position, appearance and size information at the same time, the PAS tracker achieves state-of-the art performance on the VisDrone MOT dataset.

4 Conclusion

In this report, three popular MOT methods originally developed for person tracking are extended to multi-category trackers and tested on a dataset of drone-based imagery. For this, a Cascade R-CNN detector is adapted to the drone images to improve detection performance. After a detailed comparison of the existing trackers, it is found that none of them takes full advantage of object cues and a new tracker that uses position, appearance and size information at the same time in the association step is designed. The proposed PAS tracker outperforms the other approaches by a large margin. In future works, other combination possibilities of the available object information should be investigated.

References

[1] P. Bergmann, T. Meinhardt, and L. Leal-Taixé. “Tracking Without Bells and Whistles”. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019.

[2] A. Bewley et al. “Simple Online and Realtime Tracking”. In: *2016 IEEE International Conference on Image Processing (ICIP)*. 2016.

[3] E. Bochinski, V. Eiselein, and T. Sikora. “High-Speed Tracking-by-Detection Without Using Image Information”. In: *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. 2017.

[4] E. Bochinski, T. Senst, and T. Sikora. “Extending IOU Based Multi-Object Tracking by Visual Information”. In: *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. 2018.

[5] Z. Cai and N. Vasconcelos. “Cascade R-CNN: Delving Into High Quality Object Detection”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018.

[6] G. D. Evangelidis and E. Z. Psarakis. “Parametric Image Alignment Using Enhanced Correlation Coefficient Maximization”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30.10 (2008).

[7] R. Girdhar et al. “Detect-and-Track: Efficient Pose Estimation in Videos”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[8] J. F. Henriques et al. “High-Speed Tracking with Kernelized Correlation Filters”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37.3 (2015).

[9] H. W. Kuhn and B. Yaw. “The Hungarian Method for the Assignment Problem”. In: *Naval Research Logistics Quarterly* (1955).

[10] L. Lan et al. “Interacting Tracklets for Multi-Object Tracking”. In: *IEEE Transactions on Image Processing* 27.9 (2018).

[11] H. Luo et al. “Bag of Tricks and a Strong Baseline for Deep Person Re-Identification”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2019.

[12] S. Ren et al. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (2017).

[13] D. Stadler, L. W. Sommer, and J. Beyerer. “PAS Tracker: Position-, Appearance- and Size-Aware Multi-object Tracking in Drone Videos”. In: *Computer Vision - ECCV 2020 Workshops*. 2020.

[14] N. Wojke, A. Bewley, and D. Paulus. “Simple Online and Realtime Tracking with a Deep Association Metric”. In: *2017 IEEE International Conference on Image Processing (ICIP)*. 2017.

[15] P. Zhu et al. “Vision Meets Drones: Past, Present and Future”. In: *arXiv preprint arXiv:2001.06303* (2020).