Application of SORT on Multi-Object Tracking and Segmentation

Franz Koeferl * Johannes Link Bjoern Eskofier
Machine Learning and Data Analytics Lab
University of Erlangen-Nürnberg (FAU)

Abstract

Multiple object tracking and segmentation (MOTS) on monocular images using object detectors without any end-to-end learning of the tracking step has been a common problem historically. Including the posterior of the object detector into the tracking step proves to be difficult, because features from the detection step are reduced to only a segmentation mask, object probability, and class information. Based on this, solving tasks like combining of segmentation masks and the actual tracking step is still the main challenge. We adapt an existing simple online tracking method (SORT) based on bounding boxes. The tracking process predicts trajectory using a Kalman filter and matches tracks to detections using a simple IOU metric. The sMOTSA score on the test set of KITTI-MOTS are 64.1 (cars), 54.5 (pedestrian) and on the test set of MOTS20 is 56.8 (pedestrian).

1. Introduction

Simultaneous tracking and segmentation of objects has been gaining more and more attention in the last few years. Application areas like surveillance, competition monitoring, and autonomous driving benefit from online tracking approaches, as they require high responsiveness to changes in the environment. This increases the importance of high performing, generalizable, but also simple approaches for tracking. One important tracking method, published in the last years is SORT [3].

There are several key observations when applying SORT on tracking benchmarks: Trackings are mostly lost when objects are moving, with a high speed away or towards the camera or the camera itself moves with a small continuous speed towards objects. This can have various reasons: (1) SORT requires a small set of high confidence detections to function well. This means that essential detections could be lost, when pre-filtering is applied (2) The Kalman filter in SORT cannot predict the trajectory or the bounding boxes correctly. We explore both possibilities by applying various adaptations.

*Corresponding author: F. Koeferl (franz.koeferl@fau.de)

2. Related Work

SORT approached the problem of online tracking by a simple tracking method. High confidence predictions are matched to existing tracks using the Hungarian algorithm. Though the algorithm does not outperform the current state of the art, its simplicity is still an appealing alternative for tracking tasks.

DeepSORT [9] expanded SORT by adding the velocity information – similar to our approach – to tracking and used a Mahalanobis distance based matching, including appearance and Kalman filter state information. Though, now more complex, its approach is still simple enough for easy application.

Recent publications follow a more extensive approach by combining the detectors, segmentors, and the tracking to a single problem. Voigtlaender *et al*. [8] first formulated this by defining new evaluation metrics, new datasets, and a baseline method for future comparison.

MaskProp [2] solves this problem by adapting Mask R-CNN to video sequences. A mask propagation branch is introduced, which propagates tracking information from frame to frame. This approach proves to be robust to well-known artifacts in tracking, like motion blur and object occlusion.

Because the lack of segmentation labels is still a common issue in multi-object tracking and segmentation, Sun *et al*. [7] use a multi-instance semi-supervised learning approach for exploiting box-level annotation in tracking problems. They further use reinforcement learning to update templates used for matching.

3. Methods

Tracking, using existing detections with their respective segmentation masks is achieved by SORT. We apply various adaptations to increase the duration of tracking i.e. we

minimize the number of ID switches (IDS). This way, most other tracking scores should improve as well, because their computation is directly dependent on IDS [8].

On each frame SORT, first pre-filters the detections and then applies a matching step, which assigns each detection to an existing track. These matches are then filtered again to remove the wrong matches. The matched tracks' Kalman filter is updated, new tracks are added and old tracks are removed. In the following, we explain each step in more detail.

Pre-Filtering Each frame, the set of detections are reduced by removing all detections with a confidence score smaller than 0.5.

Matching These high confidence detections are then matched to existing tracks using the Hungarian algorithm [5]. The method requires a score between all detections and all tracks. SORT calculates this score by computing the IOU – Intersection-over-Union – metric over the bounding boxes of the detection and the bounding boxes of the tracks. The latter are predicted by the Kalman filter assigned to each track.

Filtering After matching the detections to the existing tracks, a filtering step is applied, which removes spurious and random matches. If the IOU of the match is below the filtering IOU threshold (0.3), the track and the detection are unmatched.

Kalman Each track has a Kalman filter assigned. If the track is matched, the filter is updated with the bounding box parameters of the detection.

Removing and Adding Tracks The remaining unmatched tracks and detections are updated as well. Each unmatched detection is added as a new track and initialized as a separate Kalman filter. Tracks are removed, if there has been no match since max_{age} frames.

Note: The output of SORT is independent of the actual tracks. Only, tracks which are tracked longer than min_{hits} frames are used in the output.

In the following, we briefly list our adaptations. In section 3.1, an alternative to pre-filtering by merging IOU and confidence scores in the matching and filtering step is explained. Section 3.2 describes a way to model the acceleration in the Kalman filter. Finally, we propose two simple heuristics for computing the merge confidence in section 3.3.

3.1. Removing pre-filtering

The pre-filtering step in SORT is potentially too restrictive, because the confidence is usually in regard to the class of the detected object, but not the probability of the existence of an object. This means, – even though the object is detected by the detector – SORT would ignore it for processing, because of its low confidence score c. Even worse: A high confidence score does not imply a higher probability, that the object is classified correctly [4], meaning the high confidence predictions still contain many wrong detections.

This could be mitigated, by removing the confidence based pre-filtering step completely. But this leads to many false-positive tracks (see section 4.4).

Though, the confidence is smaller, we suspect there still exists a correlation between the correctness of detections and the computed confidence. Based on this assumption, the matching and filtering score is updated to $\text{IOU}_{i,j} \cdot c_{i,j}$ for all detections i and tracks j.

3.2. Kalman Filter

The application of SORT resulted in an interesting problem. If an object is moving away or towards the camera, the bounding box prediction by the Kalman filter is too slow to adapt to the change of bounding box size. This means if a car is moving away from the camera the bounding box is shrinking faster, than the actual detection, which leads to a shrinking IOU score as well.

The Kalman filter is adapted to decrease this effect by taking the acceleration of the position and the bounding boxes into account. The acceleration should be able to model movement, in general, better. We update the Kalman equations of SORT accordingly.

3.3. Merge Confidence Heuristics

The merge confidence score for each detection is used to distinguish overlapping segmentation masks and tracks i.e. if two segmentation masks are overlapping, the segmentation with the higher score is retaining all its pixel for the final evaluation, whereas lower scores are only retaining pixels, not already used by the above masks for the evaluation. This way a one-to-one segmentation mapping between tracks and ground truth can be ensured.

As our baseline, we randomized the score values for each detection. We compared two heuristics: (1) Sorting the detections regarding their y-coordinate (height in the image) and assign their respective sorting position as their merge confidence score (2) Computing and assigning the height of the detections' bounding box as their merge confidence score.

The intuition behind (1) is, that most camera scenes contain a planar on which objects are placed, including the camera. In this situation, the closer the object is to the camera, the higher their respective y-coordinate should be. This means the y-coordinate can be used directly as a distinguishing feature. Heuristic (2) exploits the intuition, that far-away objects of similar size – this includes pedestrians and cars – appear smaller than closer objects. This means the height of the bounding box is a good indicator of the distance between the camera and object.

	KITTI MOTS		MOTSChallenge
	train	val	
# Sequences	12	9	4
# Frames	5,027	2,981	2,862
# Tracks Pedestrian	99	68	228
# Masks Pedestrian			
Total	8,073	3,347	26,894
Manually	1,312	647	3,930
# Tracks Car	431	151	-
# Masks Car			
Total	18,831	8,068	-
Manually	1,509	593	-

Table 1. Statistics of KITTI MOTS and MOTSChallenge datasets from Voigtlaender *et al*. [8]

4. Evaluation

The evaluation focuses on answering two questions: (1) How much does the merge confidence impact the final results? (2) Are our approaches competitive to the baseline?

To answer these questions, we first introduce the used metrics in section 4.2, further we explain the used datasets in section 4.3 and finally, show our results in section 4.4.

4.1. Setup and Training

As max_{age} and min_{hits} are critical hyperparameters, we applied a grid-search on both datasets simultaneously, averaged the sMOTSA score and used the best parameter combination for SORT without pre-filtering. For both datasets, parameters $min_{hits} = 8$ and $max_{age} = 7$ gave the best results. Note: We observed a significant change in performance, over the whole spectrum from 0 to 30 respectively. However, a small change of max_{age} or min_{hit} affected the results only slightly.

4.2. Metrics

The evaluation of tracking performance of multiple objects is – using only a single score – difficult. We utilize the metrics defined in [1, 8], where MOTS denotes 'multi-object tracking and segmentation' and MODS 'multi-object detection and segmentation':

- **sMOTSA**(↑): soft MOTS accuracy [8]
- **MOTSA**(↑): MOTS accuracy [8]
- **MOTSAL**(↑): MOTS accuracy with log. IDS [8]
- **MOTSP**(↑): MOTS precision [8]
- **MODSA**(↑): MODS accuracy [8]
- **MODSP**(↑): MODS precision [8]

	KITTI Car	KITTI Ped	MOTS Ped
With pre-filtering			
IDX	0.07	0.06	0.06
Height	0.02	0.11	0.04
Without pre-filtering			
IDX	2.48	4.94	2.36
Height	1.68	5.11	2.46

Table 2. Evaluation of Merge Confidence Computation. Shown are the differences of sMOTSA to the random baseline for computing the merge confidence using sorting (IDX) and height. The differences are averaged over all method and dataset configurations.

- **MT**(↑): number of mostly tracked trajectories i.e. at least 80% of the trajectory's life span has the same label
- **ML**(↓): number of mostly lost trajectories i.e. at most 20% of the trajectory's life span has the same label
- **IDS**(↓): number of times a tracking ID switches on a trajectory [1]

Metrics denoted with (↑) have better performance for higher values. If the metric is annotated with (↓), lower values mean better results.

4.3. Datasets

The datasets used in the evaluation are KITTI MOTS and MOTSChallenge from Voigtlaender *et al*. [8]. KITTI MOTS was captured in an autonomous driving setting i. e. camera on top of the car. The labels contain tracking and segmentation information for classes car and pedestrian. MOTSChallenge is based on video data in a pedestrian area and only contains the class pedestrian. The number of samples and the splits can be seen in Table 1.

The detections are provided by Voigtlaender *et al*. [8], which includes their respective segmentation masks. The detections were created using Mask R-CNN X152 of Detectron2 [10] and the segmentation masks refined using refinement net [6].

4.4. Results

The evaluation focuses on two aspects. The relative improvements by applying the merge confidence computations and the comparison of all adaptations.

The evaluation of our merge confidence values are visible in Table 2. We plotted only the sMOTSA score, however, the observations are similar for the remaining metrics as well. Both heuristics outperform the random computation of the merge-score, where the IDX approach is slightly better. It is also apparent, that the gain, compared to the approach with pre-filtering is much higher. This is probably due to the overall lower absolute scores, compared to the approaches with pre-filtering (see Table 3).

	sMOTSA	MOTSA	MOTSP	MOTSAL	MODSA	MODSP	MT	ML	IDS
With pre-filtering									
orig. SORT	**63.1**	**74.0**	86.3	**75.2**	**75.2**	86.5	**42.5**	**8.8**	319
Without pre-filtering									
orig. SORT	9.93	22.48	85.00	26.13	26.15	85.40	48.45	2.50	981
SORT	31.80	42.68	85.80	44.68	44.68	86.30	34.53	16.00	546
Kalman	33.85	44.65	85.90	46.50	46.53	86.40	35.33	17.33	505
IOUConf	56.58	66.18	**86.80**	67.33	67.35	**87.20**	28.50	21.90	**305**

Table 3. Evaluation of tracking performance. The labels are as follows: **orig. SORT** denotes SORT with original parametrization, **SORT** with our parametrization, **Kalman** is SORT with suggested Kalman filter adaptation, **IOUConf** is our computation method for matching. The upper table shows the results with pre-filtering and below without pre-filtering. We use the merge confidence computation IDX for all the results.

The performance for all adaptations are visible in Table 3. The conventional application of SORT performs best in regards to accuracy and recall. Disabling the pre-filtering step drops the sMOTSA score to 9.93, which matches our observations, that many wrong detections are used for tracking i.e. objects besides persons are tracked. Using this as a baseline, we improve the results by using the optimized hyperparameters by 21.87. The Kalman filter improves the accuracy further. The best method – without pre-filtering – is using the combination of IOU and confidence for matching. Although this approach does not reach the same accuracy score as the original application, it is slightly more precise. Intuitively, it is due to the reduced amount of ID switches.

Evaluation of the original SORT on the test sets achieved an sMOTSA score of 64.1 for cars and 54.5 for pedestrians on KITTI MOTS, and 54.3 on MOTSChallenge. The combined score of all datasets is 56.8.

5. Conclusion

We adapt the tracking approach SORT by postponing the pre-filtering step to a later stage. This simplification, however, is accompanied by worse tracking performance. Additionally, we evaluated simple heuristics for the computation of the merge confidence computation. The performance increases respectively by 0.02 and 2.48 on the sMOTSA score for class cars on the KITTI dataset. The performance increase is analogous to the other datasets and the other metrics.

6. Acknowledgment

Bjoern Eskofier gratefully acknowledges the support of the German Research Foundation (DFG) within the framework of the Heisenberg professorship programme (grant number ES 434/8-1).

References

[1] Keni Bernardin and Rainer Stiefelhagen. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. *EURASIP Journal on Image and Video Processing*, 2008(1):246309, 2008.

[2] Gedas Bertasius and Lorenzo Torresani. Classifying, segmenting, and tracking object instances in video with mask propagation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[3] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3464–3468, 2016.

[4] Heinrich Jiang, Been Kim, Melody Guan, and Maya Gupta. To trust or not to trust a classifier. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 5541–5552. Curran Associates, Inc., 2018.

[5] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[6] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for video object segmentation. In *Asian Conference on Computer Vision*, 2018.

[7] Mingjie Sun, Jimin Xiao, Eng Gee Lim, Bingfeng Zhang, and Yao Zhao. Fast template matching and update for video object tracking and segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[8] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[9] 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)*, pages 3645–3649, 2017.

[10] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.