EagerMOT: Real-time 3D Multi-Object Tracking and Segmentation via Sensor Fusion

Aleksandr Kim[*] Aljoša Ošep Laura Leal-Taixé
Technical University of Munich

Abstract

*Multi-object tracking (MOT) enables mobile robots to perform well-informed motion planning and navigation by localizing surrounding objects and predicting their future motion. Existing methods detect and track targets in 3D space using a depth sensor (*e.g. *LiDAR) but only up to a limited range due to the sparsity of the signal. On the other hand, RGB cameras provide a dense and rich visual signal to localize even distant objects in the image domain but at the loss of 3D localization capability. In this paper, we propose a tracking formulation that eagerly integrates all available object observations from both sensor modalities in order to obtain a well-informed interpretation of the scene dynamics. Using images, we can identify distant incoming objects, while depth estimates allow for precise trajectory localization as soon as objects are within the range. Our method is general enough to obtain state-of-the-art results for several tasks related to multi-object tracking and segmentation (MOTS) while running in real-time at 90 FPS on a commodity CPU – a fraction of the cost of competing tracking methods.*

1. Introduction

For safe robot navigation and motion planning, mobile agents need to be aware of surrounding objects and be able to foresee their future states. To this end, they need to to detect, segment and – especially critical in close proximity – precisely localize objects in 3D space across time. Furthermore, to ensure safety of all traffic participants, such methods should be efficient, reliable, and explainable. As shown by Weng and Kitani [13], even a simple method with linear motion models and 3D overlap-driven two-frame data association can achieve competitive tracking performance when using a strong 3D object detector [10]. However, compared to image-based, methods that rely on depth sensors (*e.g.* LiDAR) are more sensitive to false positives originating from partial occlusions and can operate only within a limited distance due to signal sparsity. Image-based MOT and MOTS methods, on the other hand, leverage a rich visual signal to minimize false positives, gain robustness to partial occlusions, and localize objects with pixel-precision, even when they are too far to correctly estimate their depth [12, 9]. However, these methods track objects only in the image domain and reliable 3D localization remains a challenge.

In this paper, we present EagerMOT, a tracking framework that fuses all available object observations from 3D and 2D object detectors to obtain the most complete understanding of the scene. Using cameras, our method identifies and keeps track of all targets in the image domain, while 3D detections allow for precise trajectory and motion estimation as soon as objects can be reliably localized in 3D space. This is achieved by associating object detections, originating from different sensor modalities, and a tracking formulation that allows to update track states even when only partial (*i.e.*, only image-based or LiDAR-based) object evidence is available.

When reporting only objects, reliably localized in 3D space, our method establishes a new state-of-the-art on the official KITTI tracking benchmark [3] while being significantly faster than competing methods (≈90 FPS). When evaluated for the MOTS task, our EagerMOT yields performance, on-par with MOTSFusion [6] while being ≈20 times faster.

2. Related work

2D-based MOT. The majority of the existing vision-based tracking methods rely on recent advances in the field of object detection [8, 4] to detect and track object in the image domain. TrackR-CNN [12] extends Mask R-CNN [4] with 3D convolutional networks to improve temporal consistency of the detector and an object re-identification head to use as the data association cue. Tracktor [1] repurposes the regression head of Faster R-CNN [8] to follow targets by predicting their bounding boxes in future frames.

3D-based MOT. The recent AB3DMOT paper [13] proposed a simple, yet well-performing 3D MOT method; however, due to its strong reliance on 3D-based detections,

[*] Correspondence to: aleksandr.kim@tum.de

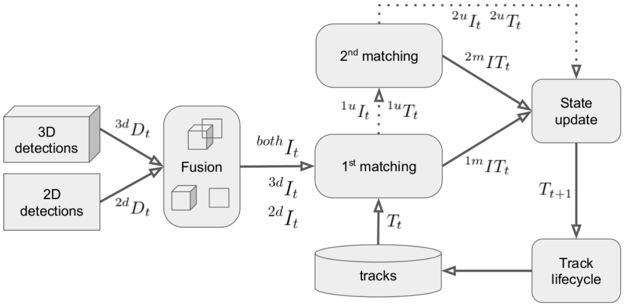

Figure 1. Overview of our tracking framework.

it is susceptible to false positives and struggles with bridging longer occlusion gaps. Our methods presents a framework that combines the localization accuracy of 3D-based detectors with the precision of 2D object detectors.

Fusion-based Methods. Fusing object evidence from 2D and 3D during tracking is an under-explored area. Ošep *et al*. [7] propose a stereo vision-based approach. At its core is a tracking state filter that maintains track position jointly, in 3D and image domain, and can update them using only partial object evidence. In contrast, our method treats different sensor modalities independently. We track targets in bsoth domains simultaneously, but we do not explicitly couple their 2D-3D states.

MOTSFusion [6] fuses optical flow, scene flow, stereo-depth, and 2D object detections to track objects in 3D space. Different to that, our method relies only on object detections obtained from two complementary sensor modalities. As as a result, we (i) achieve comparable 2D MOTS results and (ii) are able to track objects in 3D space with significantly higher accuracy while (iii) being much faster.

3. Approach

Our EagerMOT framework combines complimentary 2D and 3D (*e.g*. LiDAR) object evidence while remaining real-time and suitable for deployment on resource-constrained systems. As input at each frame, our method only takes a 3D point cloud, a set of 3D object detections $^{3d}D_t$, and a camera image with 2D detections $^{2d}D_t$.

A general overview of the pipeline is illustrated in Fig. 2 and shows its main components: (i) fusion of 3D and 2D evidence that merges detections belonging to the same object, (ii) two-stage matching that links detections across time to build tracks, (iii) state update that enables motion forecasting, and (iv) a track lifecycle module that deletes obsolete tracks and reports information about confirmed ones.

In this paper, we show the merit of our method by combining LiDAR-based object detectors, image-based detectors, and instance segmentation models. However, our approach is not limited to aforementioned sensor modalities.

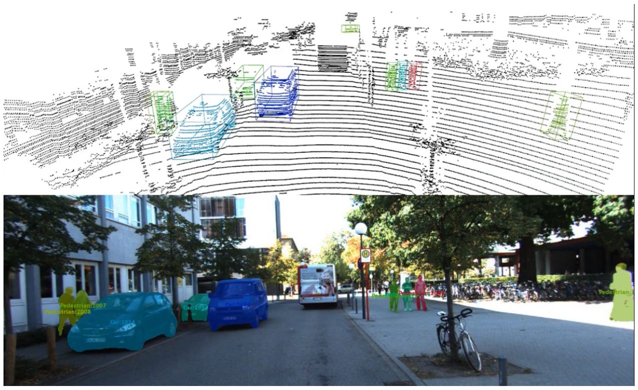

Figure 2. Qualitative MOTS results on the KITTI test (Seq. 18).

3.1. Fusion

At input, we obtain two sets of object evidence. Both sets provide object localization and semantic information. LiDAR-based object detections $^{3d}D_t$ are parametrized as 3D object-oriented bounding boxes, while image-based object detections $^{2d}D_t$ are defined by a 2D bounding box and an optional pixel-precise segmentation mask (for MOTS) in the image domain. First, we aim to establish a bi-partite matching between the two sets.

The fusion module performs this task by associating detections in $^{3d}D_t$ to detections in $^{2d}D_t$ via the Hungarian algorithm [5] and producing a set of fused object instances I_t=$\{I_t^0, ..., I_t^i\}$. The cost matrix for the algorithm is computed based on the overlap between each pair of $^{3d}D_t^i$ and $^{2d}D_t^i$. Matched detections with overlap above a threshold θ_{fusion} form fused instances $^{both}I_t \subseteq I_t$ containing both properties: a precise 3D location of the object and its 2D information *e.g*. bounding box and segmentation mask. Fusion criterion depends on the sources of information at hand (see below). Remaining detections form instances $^{3d}I_t^i \subseteq I_t$ and $^{2d}I_t^i \subseteq I_t$, containing only one of the properties. We refer to them as *partial observations*, or *partial object evidence*. Note that, $^{both}I_t \subseteq {}^{3d}I_t$ and $^{both}I_t \subseteq {}^{2d}I_t$.

Fusion criteria. For box-level MOT, the overlap between each pair of detections is defined in 2D space as the intersection over union (IoU) between a 2D bounding box $^{2d}D_t^i$ and a 2D box projection of a 3D bounding box $^{3d}D_t^i$.

For MOTS, 2D detections $^{2d}D_t$ are localized with pixel-level precision. To make use of this additional information, the overlap between detections is defined over the points in the input point cloud. For each bounding box $^{3d}D_t^i$, we determine the set of points it encloses. For each segmentation mask $^{2d}D_t^i$, we find the set of points that are covered by the mask when they are projected onto image space. Then, the overlap between each $^{3d}D_t^i$ and $^{2d}D_t^i$ is computed as the IoU between their point sets, which yields more accurate matching.

3.2. Matching

During each frame t, fused instances I_t enter a two-stage matching process to update existing tracks T_t with new 3D and/or 2D information.

Track parameterization. Following [7], tracks T_t are parameterized simultaneously, but independently, in 3D and 2D space. A track's 3D state is represented by a 3D object-oriented bounding box and a 3D velocity vector, while its 2D state is represented by a 2D bounding box and an optional 2D segmentation mask. Note than these states do not have to be fully observed, *i.e.*, tracks might have only 3D information ${}^{3d}T_t \subseteq T_t$, only 2D information ${}^{2d}T_t \subseteq T_t$, or both ${}^{both}T_t \subseteq T_t$.

Two-stage data association. In the first matching stage, instances ${}^{3d}I_t$ and tracks ${}^{3d}T_t$ are associated using the Hungarian algorithm. Each track's motion model predicts its current 3D bounding box position (using a Kalman filter), and the cost matrix for this association is computed using 3D box IoU between the instances' 3D bounding boxes and predicted locations of existing tracks (*c.f.* [13]).

Matched instance-track pairs ${}^{1m}IT_t = \{(I_t^i, T_t^j), ...\}$ are confirmed if their IoU is above a certain threshold θ_{3d}. Remaining matches are discarded and those instances and tracks are also considered unmatched: ${}^{1u}I_t$, ${}^{1u}T_t$ with ${}^{1m}IT_t \cup {}^{1u}I_t \cup {}^{1u}T_t = {}^{3d}I_i \cup {}^{3d}T_i$.

In the second stage, all so far unmatched instances and tracks $(I_i \cup T_i) \setminus {}^{1m}IT_t$ are again considered for association. This matching stage is identical to the first one but uses 2D box IoU as its criterion. Given this formulation, all types of instances and tracks have the potential to find their pairings - either by using their 2D detection directly or by projecting their 3D state to image domain. As before, associations with IoU below a minimum matching threshold θ_{2d} are considered invalid and are treated as unmatched instances/tracks.

This second matching stage addresses a few common scenarios that are ignored by methods that use only one set of detections: (i) tracks can recover from a partial occlusion when a 3D detector fails but a 2D detector still detects the object; (ii) tracks representing objects that exit LiDAR sensing area continue to be tracked and are updated by the 2D object evidence; (iii) when distant but tracked object enters LiDAR sensing range, its track can smoothly initialize a motion model and start modelling object state in 3D space in addition to the image domain.

State update. Matched pairs and unmatched instances and tracks are treated the same way regardless of how many association stages they participated in:

(i) All tracks erase their 2D information;

(ii) Matched instances update their states: new measurements for 3D Kalman filters and new 2D states;

(iii) Unmatched instances start new tracks.

In our current implementation, we do not modify or predict motion of 2D bounding boxes assuming a high frame rate to guarantee sufficient overlap between static boxes observed in consecutive frames. Adding a prediction model for the 2D state may further improve the results and remains our future work.

3.3. Track lifecycle

Following AB3DMOT, we employ a simple set of rules to manage object trajectories and their lifecycle. Tracks that have not been updated with any type of instance for a certain number of frames Age_{max} are discarded. Tracks that have had a certain number of matches F_{min} are considered confirmed and report their latest position estimates. To further make use of 2D detections, if a track's latest matched instance has a score higher than a certain threshold δ_{early}, its required F_{min} threshold is lowered.

For MOTS, overlapping masks are sorted based on their distance to the ego vehicle. Closer masks take priority over more distant masks. If a 3D location is not available for a particular track, its mask's distance is assumed to be infinite. If multiple masks with infinite distance are present, they are sorted by their detection score.

4. Experiments

4.1. Settings

Datasets. We evaluate our performance on three tasks (i) KITTI MOTS [12], (ii) KITTI 2D MOT [3], and (iii) KITTI 3D MOT. For MOTS, we follow the evaluation protocol proposed by [12]. We evaluate 2D MOT on the official KITTI MOT benchmark using CLEAR-MOT evaluation measures [2]. For 3D MOT, we follow the definition of the validation split and evaluation measures from [13]. All tasks are evaluated for the *car* and *pedestrian* classes.

3D detections. For our final model we use a pretrained state-of-the-art PointGNN [11] 3D object detector. Additionally, we use detections provided by [13] to ensure a fair comparison and to measure the impact of detection quality on tracking performance.

2D detections and segmentation masks. We evaluate our method using (i) pre-computed object segmentations provided by [6] and (ii) those provided by the TrackRCNN [12] network for a fair comparison of our 3D-based tracking system to their appearance-based model.

4.2. Results

MOT performance. In Table 1, we compare our 3D MOT performance to [13]. *"Ours (base)"* denotes the variant where the object detections provided by the baseline are

Method	sAMOTA	MOTA	MOTP	Rec	Prec	IDs
car Ours	93.43	**96.23**	**80.41**	**96.55**	**100**	0
car Ours (base)	**94.93**	93.46	78.39	94.68	**100**	0
car AB3DMOT	91.78	83.35	78.43	92.17	93.86	0
ped Ours	**94.33**	**93.61**	**73.23**	**93.83**	**99.97**	6
ped Ours (base)	83.70	83.06	66.17	83.45	**100**	**0**
ped AB3DMOT	73.18	66.98	67.77	72.82	93.28	1

Table 1. 3D MOT evaluation on the KITTI val set.

Method	MOTA	MOTP	Rec	Prec	IDs	FPS
Ours	**87.17**	85.17	**91.19**	96.45	31	90
TuSimple	86.62	83.97	90.50	97.99	293	2
MOTSFusion	84.83	85.21	88.76	**98.02**	275	2
AB3DMOT	83.84	**85.24**	88.32	96.98	**9**	**213**

Table 2. 2D MOT evaluation on the KITTI test set (car).

also used in our framework. With the first association stage being identical to the one of [13], this experiment highlights the significant contribution of the second-stage matching to final model performance. *"Ours"* denotes our best performing variant that uses detections from PointGNN and MOTSFusion as input. As expected, we observe that better 3D detections from PointGNN further improve tracking results, most notably for the *pedestrian* class.

In Table 2, we compare our 2D MOT performance on the KITTI test set to two (peer-reviewed) state-of-the-art methods. Our method achieves better results at a much higher speed.

MOTS performance. In Table 3, we show our MOTS performance on the test set and compare to the only two published methods. Using the same masks as MOTSFusion, we obtain comparable results at a much higher frame rate. *"Ours (TrackRCNN)"* denotes the variant where 2D detections and masks from [12] are used. Since the final variant (*"Ours"*) uses exactly the same detections but improved masks, the difference in their results shows how much impact mask quality has on final MOTS performance. Greyed out methods are other participants in the challenge and may use stronger detections.

Ablation experiments. In Table 4, we show a few variations of our framework. We include different input detection combinations to illustrate the framework's flexibility. Moreover, the variant *"GNN+TRCNN Reid"* shows framework's performance when appearance vectors from TrackRCNN are used in the second stage matching instead of simple box IoU, effectively demonstrating tracking using both location- and appearance-based association models.

For our standard variant *"GNN+MF Ours"*, ≈14% of all car matches were found during the second stage, ≈23% for pedestrian matches. The variant *"GNN+MF 1stg"* shows framework performance without it. These ablation results show how different components predictably influence the framework's overall performance and demonstrate its generalization to multiple input sources and association mechanisms.

Method	sMOTSA	MOTSA	MOTSP	IDs
car MOTSFusion	**75.00**	**84.10**	89.30	**201**
car Ours	74.50	83.50	**89.60**	457
car TrackRCNN	67.00	79.60	85.10	692
car PointTrack++	82.80	92.60	89.70	270
car LIFTS	79.60	89.60	89.00	114
ped MOTSFusion	**58.70**	**72.90**	**81.50**	279
ped Ours	58.10	72.00	**81.50**	**270**
ped TrackRCNN	47.30	66.10	74.60	481
ped PointTrack++	68.10	83.60	82.20	250
ped LIFTS	64.90	80.90	81.00	206

Table 3. MOTS evaluation on the KITTI MOTS test set.

		car			ped	
Configuration	sMOTSA	MOTSA	MOTSP	sMOTSA	MOTSA	MOTSP
GNN+MF Ours	85.6	94.4	90.8	59.2	72.7	82.6
GNN+MF 1 stg	74.2	81.6	91.0	58.1	70.8	82.8
GNN+TRCNN	77.2	89.0	87.1	49.1	67.4	75.6
GNN+TRCNN Reid	77.1	88.9	87.1	49.6	68.0	75.6
base+MF	85.4	94.2	90.8	58.4	71.4	**82.9**
base+TRCNN	76.9	88.7	87.1	49.1	66.9	75.9

Table 4. Ablation study on KITTI MOTS val set.

5. Conclusion

We presented a real-time tracking framework powered by fused frame-level detections and a simple two-stage association module capable of achieving state-of-the-art results on a commodity CPU. Through experiments we show our method's generalization to multiple tracking tasks, different sets of 3D and 2D detections and even different association cues. We hope that our framework will serve as a baseline for future research into accurate and efficient tracking.

Acknowledgements. This project was funded by the Humboldt Foundation through the Sofja Kovalevskaja Award.

References

[1] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixé. Tracking without bells and whistles. In *ICCV*, 2019.

[2] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: The clear mot metrics. 2008:1:1–1:10, 2008.

[3] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *CVPR*, 2012.

[4] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017.

[5] Harold W Kuhn. The hungarian method for the assignment problem. *Naval Res. Logist. Quart*, pages 83–97, 1955.

[6] Jonathon Luiten, Tobias Fischer, and Bastian Leibe. Track to reconstruct and reconstruct to track. *arXiv:1910.00130*, 2019.

[7] Aljoša Ošep, Wolfgang Mehner, Markus Mathias, and Bastian Leibe. Combined image- and world-space tracking in traffic scenes. In *ICRA*, 2017.

[8] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[9] Sarthak Sharma, Junaid Ahmed Ansari, J. Krishna Murthy, and K. Madhava Krishna. Beyond pixels: Leveraging geometry and shape cues for online multi-object tracking. In *ICRA*, 2018.

[10] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *CVPR*, 2019.

[11] Weijing Shi and Ragunathan (Raj) Rajkumar. Point-gnn: Graph neural network for 3d object detection in a point cloud. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[12] Paul Voigtlaender, Michael Krause, Aljoša Ošep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. MOTS: Multi-object tracking and segmentation. In *CVPR*, 2019.

[13] Xinshuo Weng and Kris Kitani. A baseline for 3d multi-object tracking. *arXiv arXiv:1907.03961.*