2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)
Windsor Oceanico Hotel, Rio de Janeiro, Brazil, November 1-4, 2016

3D Object Tracking in Driving Environment: a short review and a benchmark dataset

Pedro Girão, Alireza Asvadi, Paulo Peixoto, and Urbano Nunes

Abstract— **Research in autonomous driving has matured significantly with major automotive companies making important efforts to have their autonomous or highly automated cars available in the near future. As driverless cars move from laboratories to complex real-world environments, the perception capabilities of these systems to acquire, model and interpret the 3D spatial information must become more robust. Object tracking is one of the challenging problems of autonomous driving in 3D dynamic environments. Although different approaches are proposed for object tracking with demonstrated success in driving environments, it is very difficult to evaluate and compare them because they are defined with various constraints and boundary conditions. The appearance modeling for object tracking in the driving environments, using a multimodal perception system of autonomous cars and advanced driver assistance systems (ADASs), and the evaluation of such object trackers are the research focus of this paper. A benchmark dataset, called 3D Object Tracking in Driving Environment (3D-OTD), is also proposed to facilitate the evaluation of object appearance modeling in object tracking methods.**

I. Introduction

Object tracking is an essential component in the perception pipeline of autonomous cars and ADASs. Using tracking, an ego-vehicle can make a prediction about its surrounding objects locations and behaviors, and based on that make proper decisions and plan next actions. There is an extensive research literature on object tracking in image sequences [1]–[6]. These surveys are mainly focused on 2D object tracking algorithms, which work with images from monocular cameras. Generally, autonomous vehicles [7]–[9] are equipped with a number of on-board sensors *e.g.*, mono and stereo cameras, thermal, night vision, LIDAR, Radar, Inertial Navigation System (INS), Global Positioning System (GPS), and Inertial Measurement Unit (IMU), to have a multimodal robust observation of the scene. Such a sensor setup makes autonomous cars more capable, with a robust sensory perception in comparison with a single monocular observation.

This paper reviews different object tracking techniques and approaches that have been developed for autonomous driving with a focus on using stereo vision and 3D LIDAR, which are the first options to acquire 3D spatial information in the IV/ITS context [10], including when they are fused with other sensors. In this work, a benchmark dataset is constructed out of the '*KITTI Object Tracking Evaluation*' [9], where several sequences are defined and labeled according to their attributes and challenging factors. Two baseline object trackers were also implemented and evaluated. Experiments with various evaluation criteria were performed for the performance analysis. The evaluation scripts, source codes for the baseline object trackers and the ground-truth data, corresponding to this work, are available online[1].

Institute of Systems and Robotics (ISR), Department of Electrical and Computer Engineering (DEEC), University of Coimbra, Portugal pmo.girao@gmail.com; {asvadi, peixoto, urbano}@isr.uc.pt

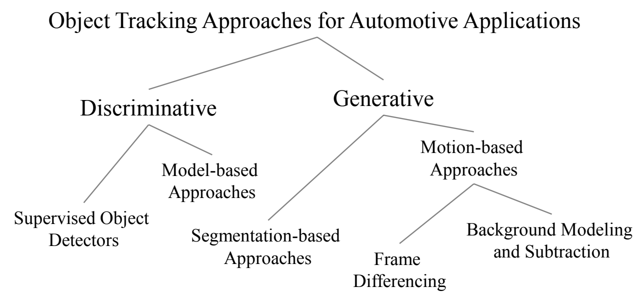

Fig. 1: Taxonomy of object tracking using 3D sensors.

The remainder of this paper is organized as follows. In the next section, a brief overview of object tracking algorithms for autonomous driving, using 3D sensors is presented. Section III is devoted to describing the benchmark dataset and its characteristics. Section IV presents the baseline 3D object trackers, while Section V focuses on the selected metrics and evaluation methodology. Evaluation results are detailed in Section VI, and finally, Section VII brings some concluding remarks.

II. Overview of 3D Object Tracking Methods for Autonomous Driving

Fig. 1 presents a taxonomy of object tracking methods in driving environments. Generally, object tracking algorithms can be divided into two categories based on representation scheme for the object [2]:

1) Tracking-by-detection (or Discriminative) Approaches: Discriminative object trackers localize the object using a pre-trained (supervised) detector (*e.g.*, DPM [22]) that learns a decision boundary between the appearance of the target object and other objects and obstacles, and next link-up the detected positions over time. Many approaches [23]–[28] have been proposed for discriminative object tracking based on monocular cameras, with most of them focused on the data association problem. An overview of such approaches is given in the '*KITTI Object Tracking Evaluation Benchmark*' [9], MOT15 [29] and MOT16 [30]. In a supervised manner

[1] https://sites.google.com/site/amshmi12/downloads

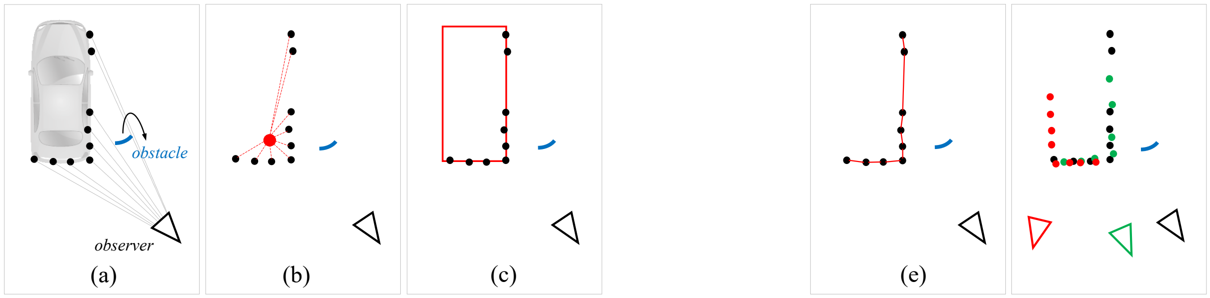

Fig. 2: Some approaches for the appearance modeling of a target object. (a) represents a scan of a vehicle which is split up by an occlusion from top view [11] , (b) the centroid (point model) representation, (c) 2D rectangular [12] or 2.5D box [13] shape based representations, (d) 2.5D grid [14], 3D voxel grid [15], [16] or octree data structure-based representation [17], [18], (e) object delimiter-based representation [19], and (f) 3D reconstruction of the shape of the target object [20], [21].

and using 3D-LIDAR, Azim and Aycard [17] proposed a method for detection and tracking moving objects. Petrovskaya and Thrun [12] proposed vehicle tracking by fitting a 2D rectangular object model to object proposals, where the object proposal is performed by moving evidence detection. However, the requirement of having all object categories being previously known and trained limits the application of discriminative approaches.

2) Model-free (or Generative) Approaches: In order to have a reliable perception system for autonomous cars in real-world driving scenarios a generic object tracker [14], [19], [31]–[33] is also required. A generic tracker should be able to track all kinds of objects, even if their existence is not previously predicted or trained. Generative methods build a model to describe the appearance of an object and then look for its next occurrence by searching for the region most similar to the model. To handle the appearance variations of the target object, the object model is often updated online. The simplest representation of a target object considers the centroid of object points, so called the '*point model*'. The point model is feasible even with a few number of object points, however, a richer appearance modeling can be exploited to capture objects physical properties (Fig. 2).

Some approaches [12], [14]–[16] detect and track generic objects based on their motion. This group of methods is the most widely used and are closely related to the Detection and Tracking of Moving Objects (DATMO) [35] approaches. These methods are unable to detect stationary objects which potentially can move. Moving object detection can be achieved by detecting changes that occur between two or three consecutive observations (which can be interpreted as '*frame differencing*') [12]. Moving object detection can also be achieved by building a consistent static model of the scene, called the background model, and then finding deviations from the model in each incoming frame [14], [17]. This process can be referenced as '*background modeling and subtraction*'. The background model is usually a short-term map of the surrounding environment of the ego-vehicle. Generally, the static background model is built by combining the ego-vehicle localization data and a representation of 3D sensor inputs such as: PCD [12], 2.5D elevation grid [14], [19], 3D voxel grid [15], [16] or octree data structure-based representation [17], [18]. Ego-motion estimation usually is achieved using Visual Odometry [16], INS (GPS/IMU) [14], [19], variants of Iterative Closest Point (ICP) scan matching algorithm [15] or a combination of them [17]. Moosmann and Stiller [21] used a local convexity based segmentation method for object hypotheses detection. A combination of KF and ICP is used for tracking moving objects and a classification method for managing tracks. Their method includes the 3D reconstruction of the shape of moving objects. Hosseinyalamdary et al. [11] used prior Geospatial Information System (GIS) map to reject outliers. They tracked moving objects in a scene using KF, with Constant Velocity process model (CV-KF) and used ICP for pose estimation. Dewan et al. [34] detect motions between consecutive scans using RANSAC and use a Bayesian approach to segment and track multiple objects in 3D-LIDAR data.

The majority of these approaches have only been developed for detection and tracking of moving objects. However, in real-world applications, static objects should also be taken into account. Segmentation-based approaches are proposed to partition the PCD into perceptually meaningful regions that can be used for object detection. Ošep et al. [31] used the PCD generated from a disparity map (obtained from a stereo camera pair) to find and track generic objects. They suggested a two-stage segmentation approach for multi-scale object proposal generation, followed by Multi Hypothesis Tracking (MHT) at the level of the object proposals. Vatavu et al. [19] built a Digital Elevation Map (DEM) from PCD obtained from a stereo vision system. They segmented on-ground obstacles by extracting free-form object delimiters. The object delimiters are represented by their positions and geometries, and then tracked using particle filters. KFs are used for adapting object delimiter models.

In another approach, Held et al. [20] combined a PCD with a 2D camera image to construct an up-sampled colored PCD. They used a color-augmented search algorithm to align the colored PCDs from successive time frames. Assuming a known initial position of the object, they utilized 3D shape, color data and motion cues in a probabilistic framework to perform joint 3D reconstruction and tracking. They showed

TABLE I: Some of the recent 3D object tracking methods for autonomous driving applications.

Ref.	3D Perception Sensor	Ego-motion Estimation	Tracking Approach	Object Representation	Object Search Mechanism	Object Model update
[19]	Stereo Vision	GNSS, INS	Generative	Object Delimiters	Particle Filter	KF
[15]	Multi-Layer LIDAR	ICP	Generative	Voxel	EKF	No Update
[11]	3D-LIDAR, GIS Map	INS	Generative	PCD	CV-KF	No Update
[31]	Stereo Vision	V-Odometry	Generative	Voxel	KF and MHT	Weighted ICP
[21]	3D-LIDAR	-	Generative	PCD	CV-KF	ICP, Accumulation
[20]	3D LIDAR, Camera	INS	Generative	Colored PCD	CV-KF	ICP, Accumulation
[34]	3D-LIDAR	DGPS/IMU	Generative	PCD	Bayesian approach	No Update
[16]	Stereo Vision	V-Odometry	Generative	Voxel	KF	No Update
[17]	3D-LIDAR	INS, ICP	Discriminative	Octree	KF and GNN	No Update
[12]	3D-LIDAR	INS	Discriminative	2D Rectangle	Particle Filter	CV-KF
[14]	3D LIDAR	INS	Generative	Elevation Grid	CV-KF and Gating	No Update

that the accumulated dense model of the object leads to a better object velocity estimate.

A summary of the most representative tracking approaches is provided in Table I.

III. Dataset and Experimental Setup

Previous attempts to propose object tracking benchmarks for automotive applications were mostly based on monocular cameras [29], [30], or were just focused on the data association problem [23]. In this paper a new framework to evaluate the performance of the appearance modeling of a target object using 3D sensors, based on the '*KITTI Object Tracking Evaluation*' dataset is proposed.

In the original KITTI dataset [9], objects are annotated with their tracklets, and generally, the dataset is more focused on the evaluation of the data association problem in discriminative approaches. Our goal is to provide a tool for the assessment of object appearance modeling in both the discriminative and generative methods. Therefore, instead of tracklets, the full track of each object is extracted. A benchmark dataset with 50 annotated sequences is constructed out of the '*KITTI Object Tracking Evaluation*' to facilitate the performance evaluation. In the constructed benchmark dataset, each sequence denotes a trajectory of only one target object (*i.e.*, if one scenario includes two target objects, it is considered as two sequences). The general specifications of each sequence and the most challenging factors are extracted and reported in Table II. The table contains the description of the scene, sequence, and objects including the *number of frames* for each sequence, *object type*: car '*C*', pedestrian '*P*' and cyclist '*Y*', *object and Ego-vehicle situations*: moving '*M*' or stationary '*S*', *scene condition*: roads in urban environment '*U*' or alleys and avenues in downtown '*D*'. The object width (Im-W) and height (Im-H) in the first frame (in pixels), and width (PCD-W), height (PCD-H), and length (PCD-L) in the first PCD (in meters) of each sequence are also reported. Each of these sequences is categorized according to the following challenges: *occlusion (OCC)*, *object pose (POS)* and *distance (DIS)* variations to the ego-vehicle, and changes in the *relative velocity (RVL)* of the object to the ego-vehicle.

IV. Baseline 3D Object Tracking Algorithms

As a starting point for the benchmark, two generative 3D-LIDAR-based methods were implemented as baselines for evaluation purposes. The baseline methods take LIDAR PCDs as the input (after a ground removal process). The initial position of the Object's 3D Bounding Box (3*D-OBB*) is known, the size of the 3*D-BB* is assumed fixed during the tracking, and the '*point model*' is used for the object representation.

A. Baseline KF 3D Object Tracker (3D-KF)

A 3D Constant Acceleration (CA) KF with a Gating Data Association (DA) is used for the robust tracking of the object centroid in the consecutive PCDs. The state of the filter is $x = \left[x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z}\right]^T$, where $\dot{x}, \dot{y}, \dot{z}$ and $\ddot{x}, \ddot{y}, \ddot{z}$ are velocity and acceleration in x, y, z location, respectively. To eliminate outliers and increase the robustness of the process, the search area is limited to a gate in the vicinity of the predicted KF location from the previous step. If no measurement is available inside the gate area, the predicted KF value is used. Experiments with different gate sizes (1×3*D-OBB*, 1.5×3*D-OBB* and 2×3*D-OBB*) were performed to conclude that the gate size of 1×3*D-OBB* provides a better result.

B. Baseline MS 3D Object Tracker (3D-MS)

In the 3D-MS approach, the Mean Shift (MS) iterative procedure is used to locate the object, as follows:

1) The shift-vector between the center of 3*D-BB* and centroid of point set P inside the 3*D-BB* is computed.
2) The 3*D-BB* is translated using the shift-vector.
3) Iterate steps 1 and 2 until convergence: The MS iteratively shifts the 3*D-BB* until the object is placed entirely within the 3*D-BB*. MS is considered converged when the centroid movement $|m_k| < 0.5$m or the maximum number of iterations is met.

We conducted an experiment with different maximum number of iterations (3, 5 and 10) and observed that a maximum of 3 iterations provides a better result. The object orientation is achieved by subtracting the current estimated location and the previous location of the object.

TABLE II: Detailed information and challenging factors for each sequence.

ID	No. Frames	Obj.	Obj. Status	Ego. Status	Scene Cond.	Im-W	Im-H	PCD-H	PCD-W	PCD-L	OCC	POS	DIS	RVL
1	154	C	M	M	U	178	208	1.73	0.82	1.78	*	*	*	
2	154	Y	M	M	U	154	127	2.00	1.82	4.43		*	*	
3	101	C	S	M	U	93	42	2.19	1.89	5.53	*		*	
4	18	C	S	M	U	77	52	1.52	1.55	3.57			*	
5	58	C	S	M	U	19	17	1.54	1.66	4.14			*	
6	144	P	S	M	U	16	42	1.72	0.73	0.55			*	
7	78	C	M	M	U	54	21	1.48	1.59	3.46	*		*	*
8	78	C	M	M	U	193	77	3.01	2.59	11.84		*		*
9	122	C	M	M	U	100	302	1.59	1.65	3.55				*
10	314	C	M	M	U	152	87	1.64	1.67	3.63		*	*	*
11	297	C	M	M	U	36	36	1.62	1.62	4.50			*	
12	101	Y	M	M	U	8	26	1.64	0.33	1.57			*	
13	42	Y	M	M	U	15	36	1.64	0.33	1.57	*		*	
14	136	C	M	M	U	95	34	1.47	1.35	3.51	*	*	*	*
15	38	C	S	M	U	98	34	1.45	1.63	4.20	*			
16	51	C	M	M	U	52	34	1.57	1.65	4.10		*	*	
17	42	C	M	M	U	22	19	1.85	1.67	4.09			*	
18	31	C	S	M	U	52	34	1.45	1.60	4.22			*	
19	24	C	M	M	U	30	32	3.43	2.81	7.02			*	
20	390	C	M	M	U	18	13	1.25	1.59	3.55			*	
21	36	C	S	M	U	28	32	2.71	1.89	5.77			*	
22	65	C	S	M	U	76	28	1.72	1.73	4.71			*	
23	56	C	M	M	U	152	57	3.52	2.89	10.81	*	*	*	
24	474	C	M	M	U	274	97	3.52	2.89	10.81	*	*	*	*
25	63	P	M	M	U	16	30	1.63	0.40	0.83			*	
26	99	Y	M	M	D	39	39	1.81	0.59	1.89		*	*	*
27	41	P	M	M	D	25	42	1.53	0.61	0.73			*	
28	323	Y	S	M	U	25	38	1.72	0.78	1.70			*	
29	188	C	M	M	U	30	21	1.44	1.74	4.23	*	*	*	
30	51	C	M	M	U	126	37	1.50	1.54	4.09	*	*	*	*
31	41	P	M	S	D	70	105	1.63	0.66	0.89	*			
32	131	P	M	S	D	46	65	1.76	0.90	1.11	*	*		*
33	132	P	M	S	D	43	72	1.89	0.84	1.05	*	*		
34	140	P	M	S	D	33	63	1.83	0.73	1.16	*	*		
35	141	P	M	S	D	27	58	1.70	0.65	1.10	*	*		
36	112	P	M	S	D	33	66	1.84	0.78	1.03	*	*		
37	31	Y	M	S	D	35	47	1.84	0.50	1.60	*		*	
38	112	P	M	S	D	20	54	1.67	0.44	0.75	*		*	
39	145	P	M	S	D	19	53	1.95	0.62	0.74	*		*	
40	54	P	M	S	D	101	160	1.71	0.48	0.93	*			
41	45	P	M	S	D	196	224	1.64	0.55	0.94	*			
42	264	C	M	M	U	28	24	1.40	1.54	3.36	*		*	
43	71	P	M	M	D	89	124	1.61	0.91	0.91	*			
44	125	P	M	M	D	36	62	1.64	0.88	0.49	*		*	
45	146	V	S	M	D	45	56	2.56	2.05	5.86			*	
46	156	P	M	M	D	25	48	1.88	0.95	0.94			*	
47	45	P	M	M	D	29	58	1.67	0.70	0.94	*		*	
48	188	P	M	M	D	31	67	1.76	0.76	1.01		*	*	
49	359	P	M	M	D	28	51	1.80	0.90	0.94		*	*	
50	360	P	M	M	D	26	49	1.72	0.84	0.85		*	*	

V. Quantitative Evaluation Methodology

Different metrics have been proposed for the evaluation of object tracking methods [5], [6], [36]. For the quantitative evaluation, two assessment criteria are used: 1)- The precision plot of overlap success, and 2)- The precision plot of orientation success. The overlap rate (the intersection-over-union metric) in 3D is given by,

$$O_{3D} = \frac{volume(3D\text{-}BB \cap 3D\text{-}BB^{g})}{volume(3D\text{-}BB \cup 3D\text{-}BB^{g})} \qquad (1)$$

where $3D\text{-}BB^{g}$ is the ground-truth 3D bounding boxes available in the KITTI dataset. The overlap rate ranges from 0 to 1. To be correct (to be considered a success), the overlap ratio O_{3D} must exceed 0.25, which is a standard threshold. The percentage of frames with a successful occurrence is used as a metric to measure tracking performance.

The ground-truth for the orientation of the object in the KITTI dataset is given by the Yaw angle (Yaw angle describes the heading of the object, and corresponds to the rotation around Z-axis) The orientation error can be computed by,

$$E_{\theta} = |\overrightarrow{\theta} - \overrightarrow{\theta}^{g}| \qquad (2)$$

where $\overrightarrow{\theta}^{g}$ is the ground-truth orientation of the object. The precision plot of orientation is given by the percentage of frames with E_{θ} less than a certain threshold (this value is empirically set as 10 degrees).

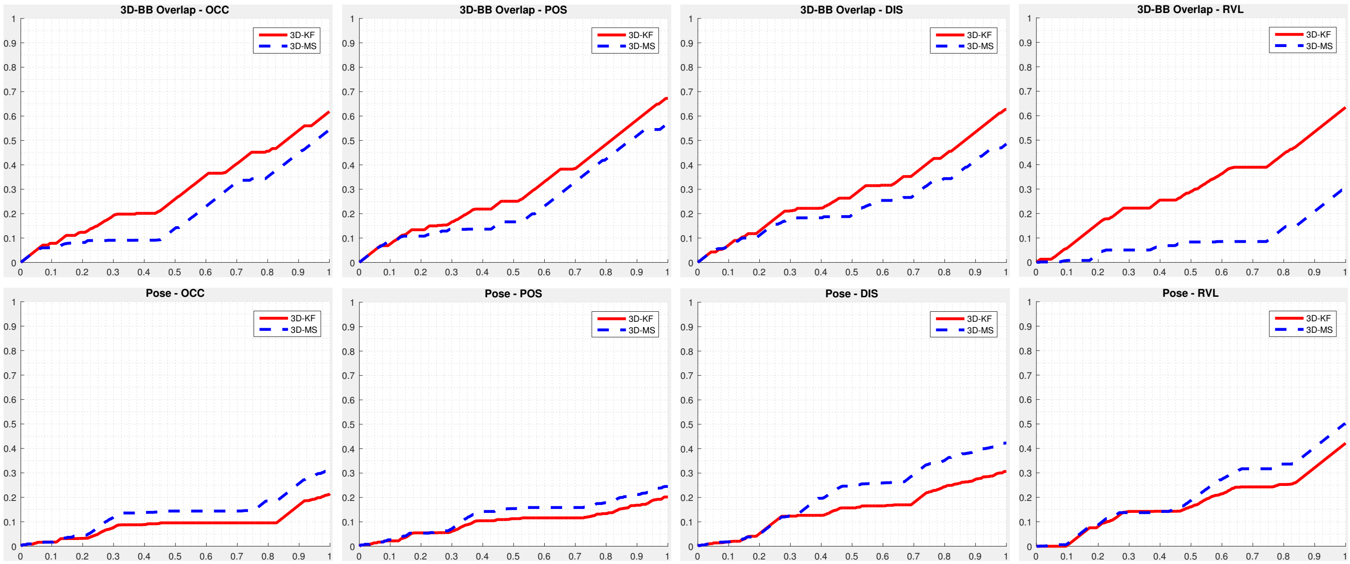

Fig. 3: The precision plot of 3D overlap rate and orientation error based on *OCC, POS, DIS and RVL* challenges.

VI. Evaluation Results and Analysis of Metric Behaviors

The metrics for the two baseline trackers (3D-MS and 3D-KF) are computed based on *OCC, POS, DIS and RVL* challenges and plotted in Fig. 3, where X-axis denotes normalized number of the frames in all the sequences and Y-axis shows normalized cumulative sum of the successful cases (i.e. each frame in which the 3D BB overlap or orientation error condition is met gets added to the cumulative sum). The 3D-KF achieves higher success rate because the 3D-MS tracker may diverge to a denser nearby object (a local minima) instead of tracking the target object. Interestingly, 3D-KF performs much better in the *RVL* challenge because of a more accurate estimation of the object dynamics. However, the 3D-MS tracker has a higher precision in the orientation estimation. The average computation time of baseline trackers is about 15 fps. The experiment was carried out using a quad core 3.4 GHz processor with 8 GB RAM under MATLAB R2015a.

A. A Comparison of Base-line Trackers with the State-of-the-art Computer Vision based Object Trackers

3D-LIDAR sensors are opening their way for high-level perception tasks in computer vision, like object tracking, object recognition, and scene understanding. We found it interesting to compare our baseline trackers (3D-MS and 3D-KF) with two high-ranking state-of-the-art computer vision based object trackers (SCM [37], and ASLA [38]) in the Object Tracking Benchmark [6]. SCM and ASLA run at about 1 fps and 6.5 fps, respectively. The precision plot is given by the percentage of successful occurrences (localization error less than 20 pixels [6]), and is presented in Fig. 4.

We found that our baseline trackers, benefiting from highly reliable 3D-LIDAR data, have superior performance over the state-of-the-art approaches in Computer Vision field. This is

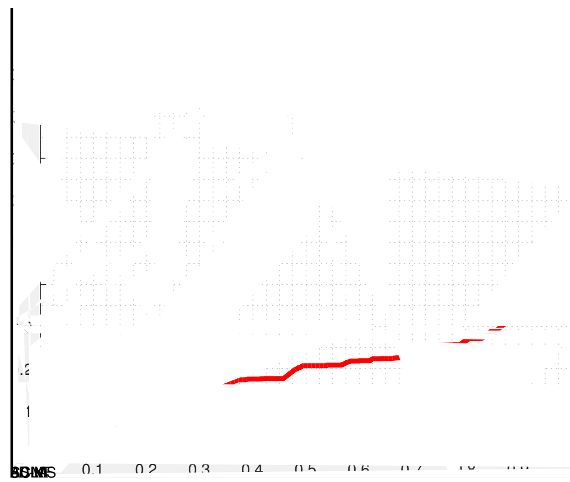

Fig. 4: The precision plot of location error.

because, in autonomous driving scenarios, ego-vehicle and objects are often moving. Therefore, object size and pose undergo severe changes (in the RGB image), which can easily mislead visual object trackers.

VII. Conclusion and Future Directions

In this work, we presented a brief survey for 3D object tracking in driving environments. A benchmark dataset based on the '*KITTI Object Tracking Evaluation*', a quantitative evaluation methodology, and two baseline trackers are provided for performance assessment.

We encourage other authors to evaluate their 3D object tracking methods using the proposed evaluation benchmark, and make their results available in order to facilitate the quantitative comparison of future approaches. An extension of the dataset and codes to include more sequences and trackers remains an area for future work. Fusion of reliable

3D-LIDAR data with mature visual object tracker in the computer vision field could also be a promising direction for future work.

ACKNOWLEDGMENTS

This work has been supported by the FCT project "AMS-HMI2012 - RECI/EEIAUT/0181/2012" and project "ProjB-Diagnosis and Assisted Mobility - Centro-07-ST24-FEDER-002028" with FEDER funding, programs QREN and COMPETE.

REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *Acm computing surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.

[2] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: A review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, 2011.

[3] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, "A survey of appearance models in visual object tracking," *ACM transactions on Intelligent Systems and Technology (TIST)*, vol. 4, no. 4, p. 58, 2013.

[4] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1442–1468, 2014.

[5] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernandez, T. Vojir, G. Hager, G. Nebehay, and R. Pflugfelder, "The visual object tracking vot2015 challenge results," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 1–23.

[6] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.

[7] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[8] M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. Hoffmann, B. Huhnke *et al.*, "Junior: The stanford entry in the urban challenge," *Journal of field Robotics*, vol. 25, no. 9, pp. 569–597, 2008.

[9] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[10] A. Broggi, P. Grisleri, and P. Zani, "Sensors technologies for intelligent vehicles perception systems: A comparison between vision and 3d-lidar," in *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*. IEEE, 2013, pp. 887–892.

[11] S. Hosseinyalamdary, Y. Balazadegan, and C. Toth, "Tracking 3d moving objects based on gps/imu navigation solution, laser scanner point cloud and gis data," *ISPRS International Journal of Geo-Information*, vol. 4, no. 3, pp. 1301–1316, 2015.

[12] A. Petrovskaya and S. Thrun, "Model based vehicle detection and tracking for autonomous urban driving," *Autonomous Robots*, vol. 26, no. 2-3, pp. 123–139, 2009.

[13] J. Choi, S. Ulbrich, B. Lichte, and M. Maurer, "Multi-target tracking using a 3d-lidar sensor for autonomous vehicles," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 881–886.

[14] A. Asvadi, P. Peixoto, and U. Nunes, "Detection and tracking of moving objects using 2.5 d motion grids," in *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*. IEEE, 2015, pp. 788–793.

[15] T. Miyasaka, Y. Ohama, and Y. Ninomiya, "Ego-motion estimation and moving object tracking using multi-layer lidar," in *Intelligent Vehicles Symposium, 2009 IEEE*. IEEE, 2009, pp. 151–156.

[16] A. Broggi, S. Cattani, M. Patander, M. Sabbatelli, and P. Zani, "A full-3d voxel-based dynamic obstacle detection for urban scenario using stereo vision," in *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, Oct 2013, pp. 71–76.

[17] A. Azim and O. Aycard, "Layer-based supervised classification of moving objects in outdoor dynamic environment using 3d laser scanner," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, June 2014, pp. 1408–1414.

[18] J. Ćesić, I. Marković, S. Jurić-Kavelj, and I. Petrović, "Short-term map based detection and tracking of moving objects with 3d laser on a vehicle," in *Informatics in Control, Automation and Robotics*. Springer, 2016, pp. 205–222.

[19] A. Vatavu, R. Danescu, and S. Nedevschi, "Stereovision-based multiple object tracking in traffic scenarios using free-form obstacle delimiters and particle filters," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 16, no. 1, pp. 498–511, 2015.

[20] D. Held, J. Levinson, and S. Thrun, "Precision tracking with sparse 3d and dense color 2d data," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1138–1145.

[21] F. Moosmann and C. Stiller, "Joint self-localization and tracking of generic objects in 3d range data," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1146–1152.

[22] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 9, pp. 1627–1645, 2010.

[23] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun, "3d traffic scene understanding from movable platforms," *Pattern Analysis and Machine Intelligence (PAMI)*, 2014.

[24] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[25] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1201–1208.

[26] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 1, pp. 58–72, 2014.

[27] P. Lenz, A. Geiger, and R. Urtasun, "Followme: Efficient online min-cost flow tracking with bounded memory and computation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4364–4372.

[28] J. Hong Yoon, C.-R. Lee, M.-H. Yang, and K.-J. Yoon, "Online multi-object tracking via structural constraint event aggregation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1392–1400.

[29] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "Motchallenge 2015: Towards a benchmark for multi-target tracking," *arXiv preprint arXiv:1504.01942*, 2015.

[30] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "Mot16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*, 2016.

[31] A. Ošep, A. Hermans, F. Engelmann, D. Klostermann, , M. Mathias, and B. Leibe, "Multi-scale object candidates for generic object tracking in street scenes," in *ICRA*, 2016.

[32] A. Teichman and S. Thrun, "Tracking-based semi-supervised learning," *The International Journal of Robotics Research*, vol. 31, no. 7, pp. 804–818, 2012.

[33] R. Kaestner, J. Maye, Y. Pilat, and R. Siegwart, "Generative object detection and tracking in 3d range data," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 3075–3081.

[34] A. Dewan, T. Caselitz, G. D. Tipaldi, and W. Burgard, "Motion-based detection and tracking in 3d lidar scans," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, Stockholm, Sweden, 2016.

[35] A. Petrovskaya, M. Perrollaz, L. Oliveira, L. Spinello, R. Triebel, A. Makris, J.-D. Yoder, C. Laugier, U. Nunes, and P. Bessiere, "Awareness of road scene participants for autonomous driving," in *Handbook of Intelligent Vehicles*. Springer, 2012, pp. 1383–1432.

[36] L. Čehovin, A. Leonardis, and M. Kristan, "Visual object tracking performance measures revisited," *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1261–1274, 2016.

[37] W. Zhong, H. Lu, and M.-H. Yang, "Robust object tracking via sparse collaborative appearance model," *IEEE Transactions on Image Processing*, vol. 23, no. 5, pp. 2356–2368, 2014.

[38] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 1822–1829.