Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

Multiple-object Tracking based on Monocular Camera and 3-D Lidar Fusion for Autonomous Vehicles

Hao Chen[1,2,3], Chunyue Xue[4], Shoubin Liu[3], Yuxiang Sun[5], Yongquan Chen[1,2*]

***Abstract*— This article describes a multi-object tracking method through sensor fusion with a monocular camera and a 3-D Lidar for autonomous vehicles. Specifically, several pairwise costs from information, such as locations, movements, and poses of 3-D cues, are designed for tracking. These costs can complement each other to reduce matching errors during the tracking process. Moreover, they are efficient to be on-line computed with embedded equipment. We feed the pairwise costs to the data-association framework, which is based on the Hungarian algorithm, and then do the back-end fusion for the tracking results. The experimental results on our autonomous sightseeing car demonstrate that our tracking method could achieve accurate and robust results in real-world traffic scenarios.**

I. Introduction

Nowadays, autonomous vehicles have attracted great interest from both the industry and academia. Multi-object tracking (MOT) is a critical capability for autonomous vehicles. By tracking multiple moving objects such as cars and people, we can predict the future movement of these objects, which is helpful for many autonomous vehicle applications, such as trajectory planning [1], autonomous exploration [2], localization [3] and mapping [4].

Recent years have witnessed great development of sensor fusion technologies [5]–[7]. Many MOT algorithms using different sensors have been proposed, some of which track objects only utilizing camera data. For example, [8] implements the 2-D multi-object tracking, while [9] makes use of the color and depth data from a stereo camera. Other algorithms implement the Lidar-based multi-object tracking such as [10], which only performs tracking using sparse 3-D point clouds. These algorithms mainly focus on the precision enhancing of the tracking results.

However, real-world environments are usually much more complicated. When only using Lidar, the sparse point clouds and limited features may increase the difficulty for detection and tracking. But with only cameras, the accuracy of location estimation in 3-D space may suffer from measurement errors. Moreover, vision-based tracking methods tend to suffer from tracking lost due to the limited field-of-view. For example, when the vehicle is driving at high speed or encounters a corner, the initially tracked objects are prone to exceed the effective range of the sensor, causing the loss of the object. In addition, when this object reappears within the field of view, it will be usually remarked with a new label, causing confusion for the tracking results.

Fig. 1: The overview of our autonomous sightseeing car. Our experiments are performed on the data collected using this autonomous vehicle.

In order to provide a solution to the above issue, we propose an on-line MOT method based on camera and 3-D Lidar sensor fusion. The large field of view of the 3-D Lidar could benefit the object tracking when they are lost from the camera. Specifically, we first use the data sequences from the monocular camera and the Lidar to separately detect and track objects based on the Lidar odometry. Then, we combine the results from two different sensors to obtain higher quality tracking results. The larger field of view of the Lidar complements the blind spots of the camera to keep the markers of tracked objects consistent. We evaluated our algorithm using our datasets collected in urban environments with an autonomous sightseeing car. The results showed that with the help of sensor fusion, our algorithm could keep continuous tracking of an object after its reappearance, and avoid the repeatedly marking issue to a certain degree. We summarize the contributions of this paper as follows:

- We propose a novel method leveraging the large field of view of 3-D Lidar to complement the narrow field of view of the camera for MOT.
- We develop a strategy based on multi-sensor fusion to alleviate the repeated marking issue when objects

[1]Shenzhen Institute of Artificial Intelligence and Robotics for Society.
[2]Institute of Robotics and Intelligent Manufacturing, The Chinese University of Hong Kong, Shenzhen, China.
[3]School of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen, China.
[4]School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, China.
[5]Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology.
*Corresponding author. email: yqchen@cuhk.edu.cn

978-1-7281-6321-5/19//$31.00 ©2019 IEEE

reappear in the camera field of view.

II. Related Work

A. Global Tracking

Global tracking [11]–[15] is one of popular methods to deal with tracking problems. They assume that all image frames can be utilized. At this level, the tracing problem can be modeled as a min-cost network flow model. One of the representative methods is the generalized minimum clique graphs. This method works with one object at a time while implicitly merging other objects. Another method is to construct small track segments (called tracklet) in the image stream, and then combine them into the whole track of the object globally.

B. Online Multi-object Tracking

Online tracker [8], [9], [16], [17] is another popular method of tracking. Online tracking only uses the information of the current frame and the past frame to track the object of the current frame, and the online tracker does not allow to change the previous tracking results. Usually, the online MOT is converted into a bipartite matching problem, which is solved by a Hungarian algorithm [18]. Both of them only use images as input to trackers, so they have congenital problems in some areas, such as detecting objects that repeatedly enter the field of view.

III. Problem Formulation

We adopt multi-sensor fusion online multi-object tracking paradigm where we assume that we are provided with a monocular video sequence of F_c frames $\{I_{f_c}\}$ for $f_c \in \{1 \dots F_c\}$. Then we have a set of object detection D_{f_c}, the object detection set consist of object detections $D^i_{f_c}$, where $i \in \{1 \dots N\}$ (N is the number of detections in frame f_c). Each $D^i_{f_c}$ is a parametrized as $D^i_{f_c} = (x^i_{f_c}, y^i_{f_c}, w^i_{f_c}, h^i_{f_c}, s^i_{f_c})$, where $(x^i_{f_c}, y^i_{f_c})$ corresponds to the top-left corner of the detection box in the image, $w^i_{f_c}$ is the bounding box width, $h^i_{f_c}$ is the bounding box height, and $s^i_{f_c}$ is the detectors confidence in the bounding box.

At the same time, we have a Lidar point clouds sequence of F_l frames $\{I_{f_l}\}$ for $f_l \in \{1 \dots F_l\}$. Then we have a set of point clouds object detection D_{f_l} , the object detection set consist of object detections $D^i_{f_l}$, where $i \in \{1 \dots N\}$ (N is the number of detections in frame f_l). Each $D^i_{f_l}$ is a parametrized as $D^i_{f_l} = (x^i_{f_l}, y^i_{f_l}, z^i_{f_l}, w^i_{f_l}, h^i_{f_l}, l^i_{f_l}, s^i_{f_l})$, where $(x^i_{f_l}, y^i_{f_l}, z^i_{f_l})$ corresponds to the center of the detection box in the image, $w^i_{f_l}$ is the bounding box width, $h^i_{f_l}$ is the bounding box height, $l^i_{f_l}$ is the bounding box length, and $s^i_{f_l}$ is the detectors confidence in the bounding box.

The multi-object tracking problem is to associate each bounding box to a object trajectory T_k such that the following constraints are met:

- All the object trajectories are made up of a series of bounding boxes that come from different frames.
- As long as the object is detected, the trajectory of the object will be tracked and generated.
- All spurious bounding box objects will not generate trajectories.

IV. Method

The core contribution of this paper is to utilize the wide viewing angle of 3-D Lidar to make up for the narrow viewing angle of the camera and reduce the repeated marks in the multi-object tracking process. We focus on urban automatic drive and show how to use 3-D Lidar and camera fusion to enhance tracking capability.

The costs in tracking algorithms include 2-D bounding box locations, appearance information (color histograms) of the image sequence, and 3-D bounding box locations, pose information of Lidar point clouds sequence. We think that using Lidar and camera fusion for tracking can effectively reduce repeated tagging without increasing computational overhead.

A. System Setup

Our work is mainly in the field of autonomous driving, where the video sequence is from a monocular camera, and the point clouds sequence is from a 3-D Lidar, which mounted on a car moving on the road plane, and the objects to be tracked also moving on the road. We use Lidar point clouds information combined with Lego-LOAM [19] to provide odometry information for frame-to-frame motion estimation. The estimating inter-frame motion by Lidar is more accurate than using visual information.

B. Monocular Camera Tracking

As for camera tracking, bounding box locations, appearance, shape, and pose are mainly used as the tracking cost in [20]. After acquiring pairwise costs, authors use a Hungarian architecture as a based tracker. Based on their work, we use more advanced object detectors to improve the object recognition rate, and use Lidar odometry as inter-frame estimation, which helps us further improve the tracking effect. Finally, we can obtain the trajectory tracking of multiple objects in the video sequence.

C. Lidar Tracking

1) 3D Cost: According to the above, we get the segmentation information of Lidar point clouds at each frame, assume we have detection $d^i_{f_l}$ in frame f_l, we wish to compute the pairwise affinity of $d^j_{f'_l}$ in frame f'_l. We use the Lidar odometry estimate the inter-frame motion. We can transport Lidar coordinates X_{f_l} in frame f_l to frame f'_l. The obtained Lidar coordinates $X_{f'_l}$ in in frame f'_l are projected down to obtain a 3-D search area in which potential matches for frame f'_l are expected to be found. There is 3D cost for two detections $d^i_{f_l}$ and $d^j_{f'_l}$ mathematical expression

$$C_{3D}(d^i_{f_l}, d^j_{f'_l}) = 1 - \frac{g(\xi, \varphi(X_{f_l{}^i}, s^i_{f_l})}{\varphi(X_{b^j_l}, s^j_{f_l})} \tag{1}$$

In fact, this cost measures a overlap of the 3D region of $d^j_{f'_l}$ in frame f'_l. $g(\xi, X)$ denotes a rigid-body motion $\xi \in se(3)$

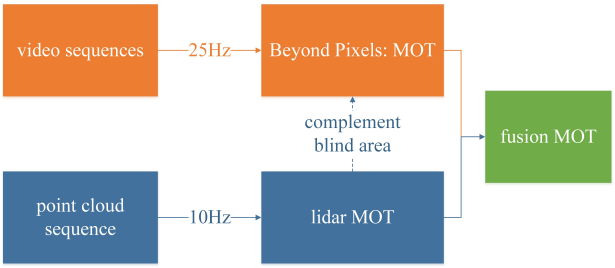

Fig. 2: The basic framework of the fusion algorithm.

applied to a 3-D point $X \in R^3$. $\xi(X, s)$ denotes the function that estimates the uncertainty of the 3D point S measured by Lidar and the detector confidence s.

This cost is only used to solve the possible position of the $d^j_{f'_l}$ in the frame f'_l, reducing the calculation requirement of matching between multiple objects.

2) Pose Cost: Pose is parametrized as an axis-angle vector ω. The pose information represents the orientation of the object. Generally, the rotation of the object between frames is not large, especially for vehicles. For detections $d^i_{f_l}$ and $d^j_{f'_l}$, the shape and pose cost is specified as

$$C_p(d^i_{f_l}, d^j_{f'_l}) = \eta_p \cdot \left\| \omega(d^i_{f_l}) - \omega(d^j_{f'_l}) \right\|^2_2 \tag{2}$$

Where η_p are normalization constants.

D. Tracking Fusion

We use the camera and Lidar to track the object, respectively, and then we combine the tracking results of the two to obtain the fusion track of multi-object. The basic framework of the algorithm is shown in Figure 2.

1) Lidar and Camera Fusion: The coordinate system of Lidar and camera can be unified by determining the external parameters (i.e., rotating translation matrix) of the two sensors through the joint calibration. Mathematically, the coordinates of an object in the camera coordinate system have the following transformation relationship with the Lidar coordinate system,

$$P_l = R \cdot P_c + T \tag{3}$$

P_c denotes the coordinates of an object in the camera coordinate system, and P_l denotes the coordinates of an object in the Lidar coordinate system. R denotes the rotating matrix, and T denotes the translation matrix of Lidar and camera. Through the above formula, obstacles in the coordinate system of two kinds of sensors can be united and marked uniformly.

2) Lidar Fills the Camera's Blind Spot: Since Lidar's range of perception is larger than that of the camera, the wide viewing angle of Lidar can be used to compensate for the camera's blind spot. We assume that in the process of multi-object tracking, some objects leave the camera's field of view, but are still in the range of Lidar perception. Then, Lidar can continuously track these objects, and even if they return to the camera's field of view, they will not be repeatedly marked.

3) Necessity of Camera Tracking: While using Lidar as an auxiliary mean, camera tracking is still the major and essential component of our tracking framework since the rich feature information provided by the camera can significantly enhance the object matching success rates between successive frames. Although Lidar can supplement the camera's field of view, the tracking interruptions may occur if using Lidar alone for long-duration object tracking due to the lack of matchable features.

V. Experimental Results and Discussions

We will implement the above algorithm on our own data set, using the basic Hungarian algorithm as the framework for multi-object tracking.

A. System Hardware

An overview of the system hardware is shown in figure 1. The experimental in this paper is validated using datasets gathered from Velodyne VLP-16 3-D Lidar and camera. The VLP-16 measurement range is up to 100m with an accuracy of $\pm$3cm. It has a vertical field of view (FOV) of $30°(\pm 15°)$ and a horizontal FOV of $360°$. The 16-channel sensor provides a vertical angular resolution of $2°$. The horizontal angular resolution varies from $0.1°$ to $0.4°$ based on the rotation rate. Throughout the paper, we choose a scan rate of 10Hz, which provides a horizontal angular resolution of $0.2°$.

B. System Overview

This paper is focused on the tracking-by-detection approach, so we need preprocessed video sequence and Lidar point clouds sequence with detection box. We use YOLO V3 [21] to detect the objects in each video frame, and use SqueezeSeg [22] to detect the objects in each Lidar frame. Both of them provide multiple detections per frame. At the same time, we apply a threshold to the object detection and prune the detection whose confidence is lower than the threshold. These detections will be input into our algorithm, then calculate the pairwise costs. In the end, they form the weight matrix, the bipartite matching algorithm that associates detections across two frames. We use Hungarian algorithm [23] to be the frame as bipartite matching.

C. Evaluation Metrics

To evaluate our algorithm, we used the conventional evaluation method in multi-object tracking. It mainly contains two indicators, Multi-Object Tracking Accuracy (MOTA) and Multi-Object Tracking Precision (MOTP). MOTA refers to the Multi-Object tracking accuracy,

$$MOTA = 1 - \frac{\sum_t (m_t + fp_t + mme_t)}{\sum_t g_t} \tag{4}$$

m_t means FN (False Negative), which refers to the positive sample predicted by the model to be negative, fp_t means TP (False Positive), which refers to the negative sample which predicted by the model to be positive, mme_t means IDS, which refers to the fragmentation for all frame objects. These three represent missing rates, misjudgments, and mismatches.

TABLE I: Comparison with other mainstream algorithms on tracking accuracy(MOTA) and precision(MOTP), mostly tracked(MT), ID switches(IDS), fragmentation(FRAG), etc.

	MOTA	MOTP	Recall	Precision	MT	IDS	FRAG
Deep Network Flow[1]	71.38	-	82.72	90.13	60.68	35	419
NOMT[5]	70.06	-	83.18	88.74	60.55	54	438
Ours	91.86	89.98	94.66	98.56	88.06	7	57

TABLE II: Comparison across various cues used for pairwise cost computation

	MOTA	MOTP	Recall	Precision	MT	IDS	FRAG
Camera	89.5	88.78	90.15	94.78	85.84	250	489
Camera + Lidar-3D	90.16	89.14	92.78	97.14	86.18	23	118
Camera + Lidar-3D + Lidar-pose	91.86	89.98	94.66	98.56	88.06	7	57

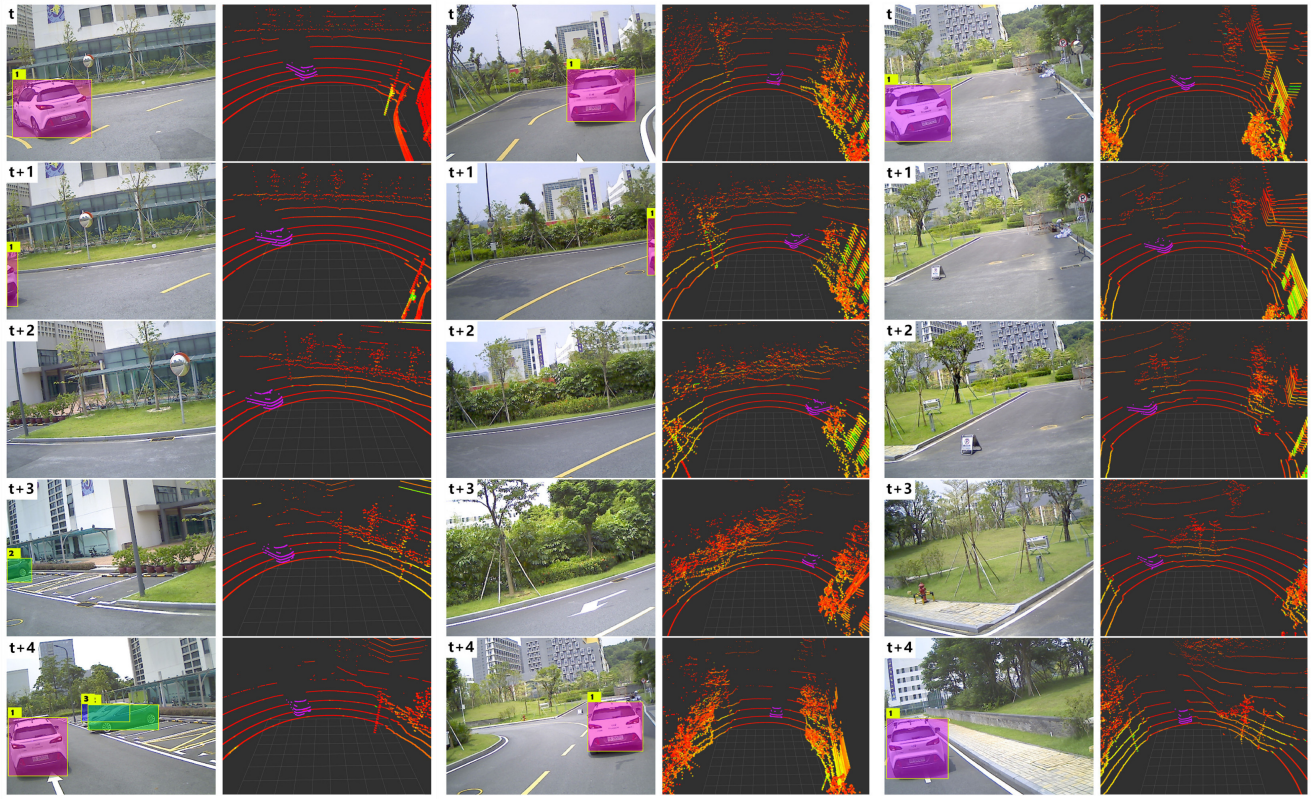

Fig. 3: Qualitative results in some challenging scenes. The left figures are the camera view, and the figures are best viewed in color. The right figures are the Lidar view.

g_t means T, which refers to the total number of true objects for all frames. MOTP refers to the positioning accuracy of the Multi-Object tracking.

$$MOTP = \frac{\sum_{i,t} d_t^i}{\sum_t c_t} \tag{5}$$

d_t^i represents the distance of the matching error of the object in frame t. c_t represents the number of matches between object and hypothesis in frame t.

D. Approaches Considered

We selected two popular multi-object tracking algorithms [24], [8] and compared them with our method. The first approach learns the pairwise costs of a network-based tracker in a neural network, and the second approach uses a complex set of artificial pairwise costs.

E. Experiment

In Table I, we compare the other two algorithms with our method, appropriately selected multiple cost weights, and achieved in terms of MOTA (91.86%) and MOTP (89.98%). At the same time, since we have introduced Lidar to supplement the blind area of the camera's field of view, the index of IDS and fragmentations has been significantly improved without introducing complex pairwise costs.

Then, we analyze the impact of each factor on the final

tracking effect, and the results are shown in Table II. This analysis shows that each element we choose has a significant impact on the overall performance. Compared with other multi-object tracking algorithms, we have added Lidar. The object can still be tracked by Lidar after it leaves the camera field of view. If the object enters the camera field of view again, it will not be repeatedly marked, thus decreases IDS and fragmentations.

Finally, we tested several challenging scenarios, as shown in figure 3. In this scenario, the object being tracked will leave the camera field of view for several frames, at which time the camera will lose the ability to track the object. Since we use Lidar, we can use Lidar to make a continuous track of the object, and after the object returns to the camera's view, it will not be repeatedly marked.

F. Results

This work is based on the camera and Lidar fusion tracking. The results show that this method can be used in challenging scenes. Table I analyzes the performance of our approach compared to traditional visual multi-object tracking, where our MOTA is over 90%. At the same time, IDS and fragmentations also greatly decrease without introducing complex artificial costs. The analysis in table II confirms the significant impact of the information provided by Lidar on improving tracking accuracy.

VI. Conclusion

The contribution of this paper is to add Lidar to the traditional multi-object tracking to enhance the tracking effect, not only taking advantage of the rich features of the camera,but also utilizing Lidar to make up the blind area of the camera and reduce the repeated marking of the same object. Although we only used the Hungarian algorithm as the algorithm framework, we still achieved a good tracking effect.

Acknowledgement

This paper is partially supported by Shenzhen Fundamental Research grant (JCYJ20180508162406177) and the National Natural Science Foundation of China (U1613216) from The Chinese University of Hong Kong, Shenzhen. This paper is also partially supported by funding from Shenzhen Institute of Artificial Intelligence and Robotics for Society.

References

[1] P. Cai, Y. Sun, Y. Chen, and M. Liu, "Vision-based trajectory planning via imitation learning for autonomous vehicles," in *2019 IEEE International Conference on Intelligent Transportation Systems*, 2019.

[2] H. Wang, Y. Sun, and M. Liu, "Self-supervised drivable area and road anomaly segmentation using rgb-d data for robotic wheelchairs," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4386–4393, 2019.

[3] Y. Sun, M. Liu, and M. Q.-H. Meng, "Improving rgb-d slam in dynamic environments: A motion removal approach," *Robotics and Autonomous Systems*, vol. 89, pp. 110–122, 2017.

[4] ——, "Motion removal for reliable rgb-d slam in dynamic environments," *Robotics and Autonomous Systems*, vol. 108, pp. 115–128, 2018.

[5] D. Xu, D. Anguelov, and A. Jain, "Pointfusion: Deep sensor fusion for 3d bounding box estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 244–253.

[6] A. Palffy, J. F. Kooij, and D. M. Gavrila, "Occlusion aware sensor fusion for early crossing pedestrian detection," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2019, pp. 1768–1774.

[7] Y. Sun, W. Zuo, and M. Liu, "Rtfnet: Rgb-thermal fusion network for semantic segmentation of urban scenes," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2576–2583, 2019.

[8] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3029–3037.

[9] A. Osep, W. Mehner, M. Mathias, and B. Leibe, "Combined image- and world-space tracking in traffic scenes," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 1988–1995.

[10] X. Weng and K. Kitani, "A baseline for 3d multi-object tracking," *CoRR*, vol. abs/1907.03961, 2019. arXiv: 1907.03961. [Online]. Available: http://arxiv.org/abs/1907.03961.

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

[12] A. Dehghan, S. Modiri Assari, and M. Shah, "Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4091–4099.

[13] L. Leal-Taixé, G. Pons-Moll, and B. Rosenhahn, "Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker," in *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, IEEE, 2011, pp. 120–127.

[14] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 9, pp. 1806–1819, 2011.

[15] V. Chari, S. Lacoste-Julien, I. Laptev, and J. Sivic, "On pairwise costs for network flow multi-object tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5537–5545.

[16] 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.

[17] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "Robust multiperson tracking from a mobile platform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, pp. 1831–1846, 2009.

[18] Y. Sun and M. Q.-H. Meng, "Multiple moving objects tracking for automated visual surveillance," in *2015 IEEE International Conference on Information and Automation*, IEEE, 2015, pp. 1617–1621.

[19] T. Shan and B. Englot, "Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 4758–4765.

[20] S. Sharma, J. A. Ansari, J. K. Murthy, and K. M. Krishna, "Beyond pixels: Leveraging geometry and shape cues for online multi-object tracking," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2018, pp. 3508–3515.

[21] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *ArXiv preprint arXiv:1804.02767*, 2018.

[22] B. Wu, A. Wan, X. Yue, and K. Keutzer, "Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2018, pp. 1887–1893.

[23] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[24] S. Schulter, P. Vernaza, W. Choi, and M. Chandraker, "Deep network flow for multi-object tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6951–6960.