Sensor systems

Robust Multiobject Tracking Using Mmwave Radar-Camera Sensor Fusion

Arindam Sengupta* , Lei Cheng, and Siyang Cao**

Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ 85721 USA
**Graduate Student Member, IEEE*
***Senior Member, IEEE*

Manuscript received 12 September 2022; accepted 3 October 2022. Date of publication 11 October 2022; date of current version 20 October 2022.

Abstract—With the recent hike in the autonomous and automotive industries, sensor-fusion-based perception has garnered significant attention for multiobject classification and tracking applications. Furthering our previous work on sensor-fusion-based multiobject classification, this letter presents a robust tracking framework using a high-level monocular-camera and millimeter wave radar sensor-fusion. The proposed method aims to improve the localization accuracy by leveraging the radar's depth and the camera's cross-range resolutions using decision-level sensor fusion and make the system robust by continuously tracking objects despite single sensor failures using a tri-Kalman filter setup. The camera's intrinsic calibration parameters and the height of the sensor placement are used to estimate a birds-eye view of the scene, which in turn aids in estimating 2-D position of the targets from the camera. The radar and camera measurements in a given frame is associated using the Hungarian algorithm. Finally, a tri-Kalman filter-based framework is used as the tracking approach. The proposed approach offers promising MOTA and MOTP metrics including significantly low missed detection rates that could aid large-scale and small-scale autonomous or robotics applications with safe perception.

Index Terms—Sensor systems, Sensor sapplications, Kalman filter, millimeter-wave (MmWave) radar, perception, sensor-fusion, tracking.

I. INTRODUCTION

Autonomous perception has been, in the recent past, predominantly driven using optical sensors on account of their high-resolution and advanced computer vision research. However, several testing mishaps due to the sensors' operational failures in poor illumination and occlusion [1], [2], has called for reevaluation of the over-emphasized reliance on just vision-based sensors. Millimeter-wave (mmWave) radars, also deployed on these vehicles, on the other hand, are operationally robust to scene lighting and weather conditions. However, they play the role of a secondary sensor, primarily due to its lower angular resolution compared to its optical counterparts. As complete replacement of optical solutions for perception is impractical, sensor-fusion schemes are being studied and developed to overcome the aforementioned limitations by making use of the complementary advantages offered by individual sensors.

Several radar-vision fusion-based tracking approaches have been proposed in the literature [3], [4], [5], [6], [7], [8]. However, these approaches have either attempted to improve localization accuracy, or attempted to improve the true detection rate and reduce false-alarm rates, i.e., false positives (FPs). Moreover, one of the key aspects of a safe autonomous perception to avoid catastrophic mishaps—missed detections or false negatives (FNs)—have not been explored or presented elaborately, except for [7] and [8]—where rather high FN rates of ≈55% and 16% have been reported. Furthermore, the fusion framework's adaptability to single-sensor failures has also not been explored adequately.

In our proposed approach, we not only aim to improve the localization accuracy when an object is detected by both the radar and camera, and also address the issue of missed detections due to sensor failures. While we can obtain the 2-D position of a target from the mmWave radar directly (following the signal processing chain), for the monocular camera image-stream, we used the detection bounding-box and inverse perspective mapping to estimate an object's 2-D position in space. Using the variance studies from our previous work [9] that demonstrated lower variance in down-range with radar and cross-range with camera, we used a high-level sensor fusion to obtain a better localization accuracy. A tri-Kalman Filter framework, one each for the individual sensors and one for the decision-level [10] fusion layer, was then used to continuously track objects and account for intermittent missed-detections from individual sensors. This layer also ensures that as long as one of the sensors detect an object, it can be tracked using the 2-D measurement from that specific sensor, making the proposed system robust to sensor failures.

This letter is organized as follows. Section II describes the proposed tracking framework, also summarized in Fig. 1. The experimental data collection, evaluation, and discussion of the tracking methodology is presented in Section III, and finally, Section IV concludes this letter.

II. TRACKING METHODOLOGY

In this letter, we map the object's position in 2-D ground plane in terms of the depth and lateral positions. In order to achieve robustness to single sensor failures, we need to ensure that we can obtain the depth and lateral positions of objects using individual sensors. While this is straightforward in the radar's case, estimating 2-D spatial position using monocular camera image require additional transformations. In our previous study [9], we had used a deep neural network to estimate the depth (Y) and cross-range (lateral) position (X) of a human using the bounding box coordinates from an uncalibrated monocular camera. However, it is possible to estimate a 2-D linear projection of the ground plane, also referred to as a "birds-eye view," using the sensor height and the estimated intrinsic properties [11] of a monocular camera with the aid of inverse perspective mapping [12], which we have used in

Corresponding author: Siyang Cao (e-mail: caos@arizona.edu).
(Arindam Sengupta and Lei Cheng are co-first authors).
Associate Editor: F. Costa.
Digital Object Identifier 10.1109/LSENS.2022.3213529

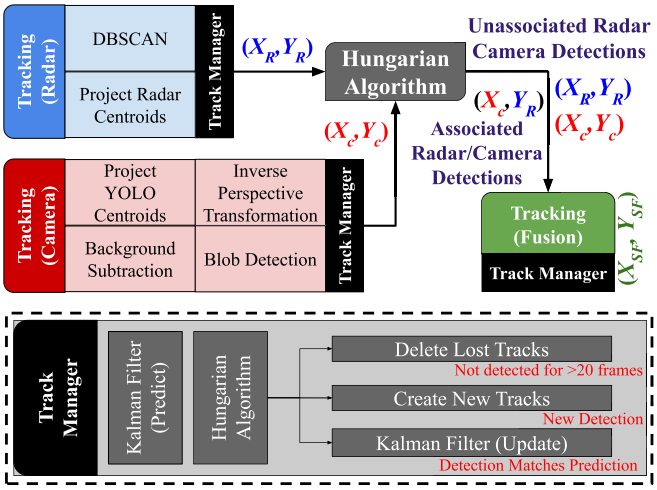

Fig. 1. Pictorial overview of the overall tracking framework.

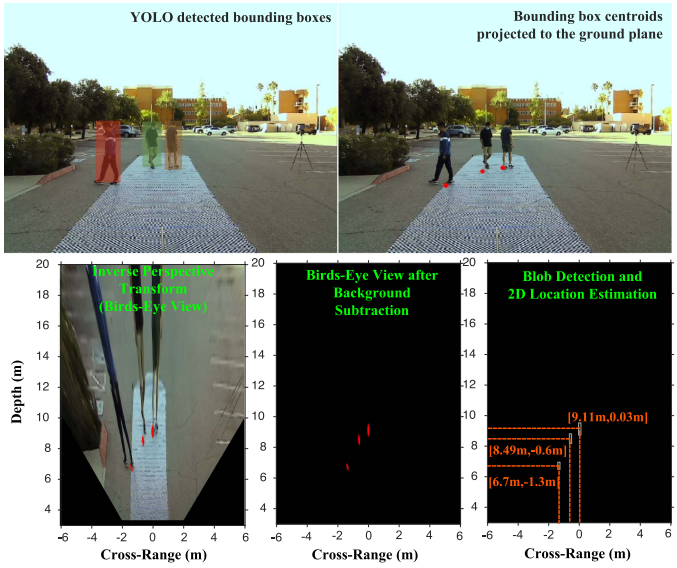

Fig. 2. Steps to obtain the 2-D spatial positions of objects from monocular camera image.

this study. The mmWave radar-camera cocalibration was carried out as described in [13].

To obtain the 2-D spatial position of the objects detected using the camera, 1) we first projected the bounding box centroid onto the ground plane using the lowest pixel level of the box; 2) transformed the projection into the birds-eye view; 3) and finally performed background subtraction and blob detection, as summarized in Fig. 2. To bypass the background estimation and subtraction stage, we used a blank image with the same dimensions as the camera image and then performed the projection and birds-eye view transformations, followed by blob detection. For the radar point clouds, we use density-based spatial clustering of applications with noise (DBSCAN) to cluster reflections pertaining to separate targets and use the cluster centroids as their 2-D positions.

The proposed sensor-fusion tracking system essentially consists of three trackers—one each for camera-based position (X_c, Y_c), radar-based position (X_R, Y_R), and a sensor-fused position (X_{SF}, Y_{SF}). In our previous study [9], we had shown that the radar offered a more stable

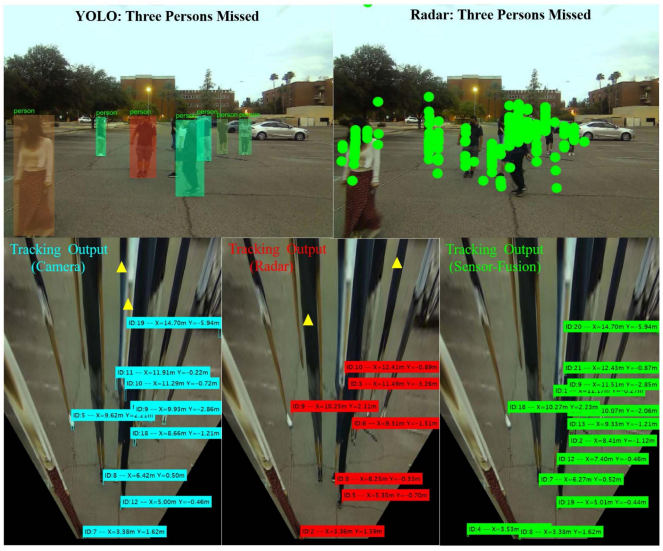

Fig. 3. Snapshot of the tracking process for a ten pedestrians scenario. The camera and radar sensors missed detecting five pedestrians (marked with triangles, with one being completely occluded and unmarked), due to occlusion. However, the proposed tri-Kalman filter aided sensor-fusion approach could successfully track all ten pedestrians.

and accurate depth estimation than cameras, while the lateral position from image data had a lower variance than the radar's measurement (on account of higher lateral resolution compared to radar). Therefore, a viable approach is to perform a high-level sensor fusion by using depth position from the radar ($Y_{SF} = Y_R$) and lateral position from the camera ($X_{SF} = X_C$) when an object is simultaneously detected/tracked by both sensors. Furthermore, we also assign $(X_{SF}, Y_{SF}) = (X_c, Y_c)$ or (X_R, Y_R), for unassociated camera or radar detections, respectively. This way, besides providing robustness to single sensor failures, as depicted in Fig. 3, the sensor fusion aims to lower the uncertainties in two-dimensions and in turn provide a higher tracking precision. The association is carried out by first computing an $N \times M$ cost-matrix, where N and M are the number of radar and camera detections respectively, with each element (n, m) in the matrix representing l^2-norm between the nth radar and mth camera detection. The Hungarian algorithm [14], [15] uses this cost-matrix to obtain an optimal assignment.

Each of the three trackers have a similar track management process. First, for newly detected objects in the scene, a new track is initialized with a unique track id. In the subsequent frames, a Kalman filter [16] first predicts the new position of the tracks based on past motion statistics and then uses Hungarian algorithm to match the predicted positions with the detections/measurements (using a formulated track-detection cost matrix) in the current frame. Note that while we have used a constant-velocity Kalman Filter model to prove our hypothesis, this study could also be extended to non-linear state estimation models such as extended/unscented Kalman filter that directly can take in $[Range, Az, El, V]$ from radar and $[u, v]$ from camera, while retaining the proposed framework. As the Kalman filter does not handle nonlinear state estimation, we have passed on position state vectors from the radar and camera in cartesian coordinate system.

There could be three outcomes in the Hungarian algorithm-based matching process: 1) matched detection-track pairs; 2) unmatched tracks; and 3) unmatched detections. Using the matched detection-track pairs, the Kalman filter then performs the update operation to generate the corrected positions of the tracks. For the unmatched tracks, there are no associated measurement or detection values; therefore,

there is no Kalman filter update stage in this case. Instead, the track is updated with the Kalman filter's predicted value to allow for continuous estimated tracking, while a running "invisibility" counter is incremented by a single frame. If a track is invisible for 20 consecutive frames, the track is deleted. Similar to the invisible counter, we have a track age (number of frames where a predict stage was carried out), and a visible count (frames where the object was detected by the sensor and both predict-update stages were carried out). A track is also deleted if it is reliability score ($RelScore$) (1) does not meet the user-defined threshold. In our study, we set this reliability threshold at 60%

$$RelScore = \frac{Visible\ Count}{Track\ Age} \times 100. \tag{1}$$

For the third scenario, i.e., unmatched detections, new track ids are assigned, and tracks are initialized. While all the aforementioned predict-update stages start for new tracks immediately from the next frames, they are not displayed on the monitor until they are classified as reliable tracks, i.e., at least present for five frames or more, as an additional stray-track filtering strategy.

III. DATA COLLECTION

To carry out an experimental validation of this study, we setup our experiment environment at a parking lot in a controlled environment. The controlled environment was pertinent in order to ensure that we have accurate ground truth information in terms of the number of subjects/objects in the scene and their trajectory. The TI 1843AWR mmWave radar-monocular camera system was mounted on a tripod and setup at a height of 1.64 m. The data collection was carried out for five different scenarios: 1) single pedestrian walking in a zigzag trajectory; 2) two pedestrians walking in diagonal trajectories crossing each other mid-path; 3) two pedestrians walking in crossing diagonal trajectories and one pedestrian walking in a straight line crossing the other two pedestrian trajectories; 4) one pedestrian and one vehicle; 5) 11 pedestrians walking in the horizontal, vertical, and diagonal directions, respectively. A checkerboard cloth was placed on the road and markers were placed on them to aid the volunteers follow the desired trajectories at controlled increments.

The data were collected on Nvidia Jetson Xavier GPU using custom robot operating system (ROS) packages for data acquisition from mmWave radar/camera, and intermediate processing, implemented in C, C++, and Python. The `usb_webcam` package reads the raw image from the camera and rectifies it using the intrinsic calibration matrix. The rectified image is then used by the `darknet_ros` package (YOLO) to identify objects in the scene and draw bounding boxes around it. On the other side, the mmWave radar scans the environment and returns processed radar reflections via the `mmwave_radar` package. The radar reflections, rectified image, and the bounding boxes are saved to an ROS bag file for offline processing on MATLAB. Each ROS message has a `timestamp` header which is used to associate data from both the sensors. The radar data are published point-by-point with a distinct `pointID` field in the data structure. The frame-wise radar point cloud is then generated by accumulating all the `pointIDs` starting from 0,1,2...N until the value resets to 0, signifying the start of a new frame.

A. Evaluating Tracking Performance

In order to evaluate the tracking performance of the proposed study, we use the standardized CLEAR Multiobject tracking (MOT) metrics [17], namely, $MOTA$ (Accuracy) and $MOTP$ (Precision). $MOTA$ is computed using the sum of the ratios of: 1) false negatives (FN_t) or missed objects in the scene; 2) false positives (FP_t) or incorrectly detected objects that are not present in the scene; 3) id switches ($IDSW_t$) or mismatched detection/track ids between objects in the scene, with respect to the ground truth (GT_t)—the total number of objects in the scene—in a given frame t, summed over all the frames, formally represented as

$$MOTA = 1 - \frac{\sum_t FN_t + FP_t + IDSW_t}{\sum_t GT_t} \tag{2}$$

where $\frac{\sum_t FN_t}{\sum_t GT_t}$, $\frac{\sum_t FP_t}{\sum_t GT_t}$, and $\frac{\sum_t IDSW_t}{\sum_t GT_t}$ are referred to as the FN rate (FNR), FP rate (FPR), and IDSW rate ($IDSWR$), respectively. $MOTP$ on the other hand measures the error of the trajectory mapping when an object is rightfully detected, and is represented as

$$MOTP = \frac{\sum_{i,t} d_t^i}{\sum_t GT_t} \tag{3}$$

where d_t^i is the distance between the tracked position and the ground truth position of the ith object in the tth frame. We evaluated the proposed tracking approach on the data from four scenarios, using the aforementioned CLEAR MOT metrics. Since the evaluation process is frame-by-frame and requires manual annotation of error metrics, we used 600 frames to evaluate each of the first four scenarios, and 255 frames for the 11 pedestrians scenario. The results are outlined in Table 1, expressed as FPR, FNR, and $IDSWR$, along with the $MOTA$ and $MOTP$ metrics.

B. Discussions and Challenges

The time cost per frame of the sensor fusion tracker is 113.00 ms and the frame rate is about 8 Hz. It can potentially achieve a practical $\geq$10 Hz frame rate with 1) faster C/C++ implementation, instead of Python, as in our case; and 2) higher processing power to enable faster YOLO throughput. The metrics in Table 1 clearly indicate that the sensor fusion tracker offers significantly lower false negatives (missed detections) than the individual sensors, and other approaches in the literature [7], [8], providing robust detection and tracking, while achieving SOTA comparable 26 cm MOTP. This is extremely critical in the drive towards safe and accident-free autonomous perception. While the radar tracker offered a marginally better $MOTP$, the FN metrics from the experiments suggest that they also missed substantial number of detections. This is due to 1) lower frame rate (2:1 ratio compared to camera) than the update rate and 2) MTI filtering out stationary targets. This can be alleviated by matching the camera-radar frame rate and disabling the MTI filter. However, the downside to this would include increased static clutter. In addition, the sensor-fusion tracker has a higher $IDSWR$ rate for object-dense scene. The primary reason is the radar's inability to distinguish and cluster objects that have similar spatial and motion statistics, which in turns results in correct detection-yet-incorrect assignment to track.

The higher number of FPs in the sensor-fusion tracker is due to the radar and camera detections from the same target being identified as unassociated or separate detections. One reason is the high volatility and discontinuity of radar points, which causes the centroid of the radar point cluster to often jump back and forth in front of and behind the target. Another critical factor is the bounding box from YOLO not being compact. These lead to the bounding box centroid being projected to a point much further away than the radar position measurements, as shown in Fig. 4. A combination of the compact bounding box challenge, coupled with missed radar detections resulted in a lower $MOTP$ from the sensor-fusion tracker than individual sensor trackers,

Table 1. CLEAR-MOT Metrics for the Five Tracking Scenarios

	Tracking (Camera Only)					Tracking (Radar Only)					Tracking (Sensor Fusion)				
	FNR	FPR	IDSWR	MOTA	MOTP	FNR	FPR	IDSWR	MOTA	MOTP	FNR	FPR	IDSWR	MOTA	MOTP
1	0.67%	5.00%	0.33%	94.00%	0.39m	53.00%	0.00%	0.00%	47.00%	0.17m	0.00%	11.83%	0.33%	87.83%	0.31m
2	26.83%	3.50%	0.25%	69.42%	0.52m	25.50%	0.00%	0.17%	74.33%	0.32m	1.67%	22.58%	0.33%	75.42%	0.27m
3	18.06%	10.22%	0.39%	71.33%	0.28m	33.61%	0.67%	0.22%	65.50%	0.19m	10.44%	12.89%	0.83%	75.83%	0.22m
4	5.48%	11.35%	0.20%	82.97%	0.27m	31.12%	2.54%	0.00%	66.34%	0.39m	2.35%	23.09%	0.20%	74.36%	0.32m
5	11.53%	14.90%	0.58%	72.99%	0.56m	27.13%	4.22%	1.17%	67.49%	0.24m	8.21%	16.23%	1.19%	74.37%	0.25m
Overall	MOTA			MOTP		MOTA			MOTP		MOTA			MOTP	
	74.26%			0.46m		66.49%			0.24m		75.83%			0.26m	

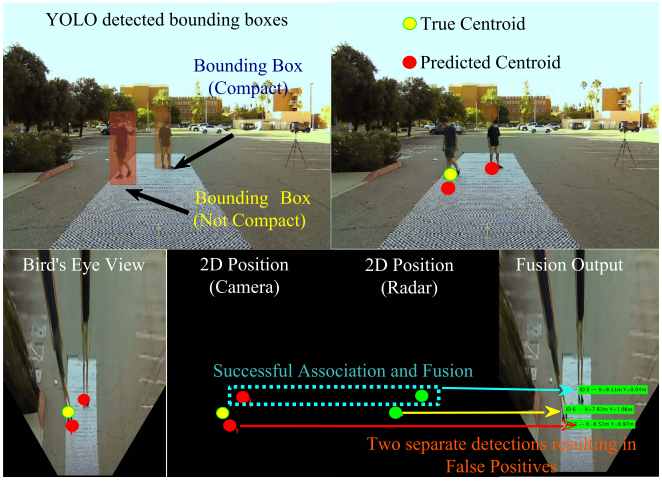

Fig. 4. Example frame depicting an FP detection in the sensor fusion tracker that arises due to noncompact bounding box being projected at a much further point, resulting in disparate detections.

contrary to our hypothesis. While YOLO has been used in this study with reliable results to provide proof-of-concept for our approaches, more sophisticated image processing/deep learning methods that can annotate compact bounding boxes will alleviate this challenge, which shall be explored in our future work.

IV. CONCLUSION

To summarize, we presented a tri-Kalman filter-based tracking approach using mmWave radar-camera sensor fusion to improve the localization accuracy by leveraging radar and camera's high-resolution components in measuring depth and cross-range, respectively, while being robust to single sensor failures and ensuring low FN rates, that can aid autonomous systems in collision avoidance and significantly reduce accidents due to missed detections. The proposed methodology is developed using first-principles and is extremely practical to deploy, thereby promoting rapid implementation for real-world development and testing. Future work on further improving the proposed framework's performance includes 1) exploring approaches to make the bounding boxes more compact and 2) studying non-Gaussian state-estimators such as extended Kalman filters and particle filters in the tracking framework.

ACKNOWLEDGMENT

This work was supported by the Sony Research Award Program. The authors would like to thank Shuting Hu and Qi Wen for their valuable help in experimental setup and data collection.

REFERENCES

[1] The New York Times, "Self-driving uber car kills pedestrian in Arizona, where robots roam," Mar. 2018. [Online]. Available: https://www.nytimes.com/2018/03/19/technology/uber-driverless-fatality.html

[2] The Tesla Team, "A tragic loss," Jun. 2016. [Online]. Available: https://www.tesla.com/blog/tragic-loss

[3] K.-E. Kim, C.-J. Lee, D.-S. Pae, and M.-T. Lim, "Sensor fusion for vehicle tracking with camera and radar sensor," in *Proc. 17th Int. Conf. Control, Autom. Syst.*, 2017, pp. 1075–1077.

[4] X. Wang, L. Xu, H. Sun, J. Xin, and N. Zheng, "On-road vehicle detection and tracking using MMW radar and monovision fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 7, pp. 2075–2084, Jul. 2016.

[5] S. Han, X. Wang, L. Xu, H. Sun, and N. Zheng, "Frontal object perception for intelligent vehicles based on radar and camera fusion," in *Proc. 35th Chin. Control Conf.*, 2016, pp. 4003–4008.

[6] R. Zhang and S. Cao, "Extending reliability of mmwave radar tracking and detection via fusion with camera," *IEEE Access*, vol. 7, pp. 137065–137079, 2019.

[7] F. J. Botha, C. E. van Daalen, and J. Treurnicht, "Data fusion of radar and stereo vision for detection and tracking of moving objects," in *Proc. Pattern Recognit. Assoc. South Afr. Robot. Mechatronics Int. Conf.*, 2016, pp. 1–7.

[8] F. A. Alencar, L. A. Rosero, C. Massera Filho, F. S. Osório, and D. F. Wolf, "Fast metric tracking by detection system: Radar blob and camera fusion," in *Proc. 12th Latin Amer. Robot. Symp. 3rd Braz. Symp. Robot.*, 2015, pp. 120–125.

[9] A. Sengupta, F. Jin, and S. Cao, "A DNN-LSTM based target tracking approach using mmWave radar and camera sensor fusion," in *Proc. IEEE Nat. Aerosp. Electron. Conf.*, pp. 688–693, 2019.

[10] M. Liggins, II, D. Hall, and J. Llinas, *Handbook of Multisensor Data Fusion: Theory and Practice*. Boca Raton, FL, USA: CRC, 2017.

[11] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[12] J. Jeong and A. Kim, "Adaptive inverse perspective mapping for lane map generation with slam," in *Proc. 13th Int. Conf. Ubiquitous Robots Ambient Intell.*, 2016, pp. 38–41.

[13] A. Sengupta, A. Yoshizawa, and S. Cao, "Automatic radar-camera dataset generation for sensor-fusion applications," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2875–2882, Apr. 2022.

[14] H. W. Kuhn, "The hungarian method for the assignment problem," *Nav. Res. Logistics Quart.*, vol. 2, no. 1/2, pp. 83–97, 1955.

[15] J. Munkres, "Algorithms for the assignment and transportation problems," *J. Soc. Ind. Appl. Math.*, vol. 5, no. 1, pp. 32–38, 1957.

[16] R. E. Kalman, "A new approach to linear filtering and prediction theory," *Trans. ASME, J. Basic Eng.*, vol. 83, pp. 95–108, 1961.

[17] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *EURASIP J. Image Video Process.*, vol. 2008, pp. 1–10, 2008.