Progress In Electromagnetics Research C, Vol. 155, 195–201, 2025

PIER C

(Received 11 February 2025, Accepted 29 April 2025, Scheduled 12 May 2025)

Online Targets Tracking and People Counting Using Multiple Distributed mmWave Radar Sensors

Uday K. Singh* and Rama R. Thipparaju

Department of ECE, SRM Institute of Science and Technology, Kattankulathur 603203, Tamilnadu, India

ABSTRACT: With the growing use of radar sensors, particularly in surveillance applications, there is an increasing need for real-time target tracking, especially in areas such as counting people. This paper offers a detailed description of the hardware setup, which is paired with a proposed fusion algorithm and the multiple-target tracking (MTT) algorithm for online target tracking. The fusion techniques introduced in this work combine data from spatially distributed Texas Instruments mmWave radar sensors by utilizing the likelihood of radar measurements. These sensors measure the positions of the reflectors, which are then visualized through the Robot Operating System (ROS). To support real-time target tracking and people counting, a connection is established between the ROS network and MATLAB. Finally, the measurements are processed in MATLAB using the proposed fusion technique alongside the existing MTT algorithm to generate accurate target tracks, which also enable people counting.

1. INTRODUCTION

Tracker plays a vital role in the overall radar signal processing chain, not only by tracking the movement of the maneuvering targets, but also by estimating their kinematic behavior [1–4]. Tracker plays an important role in autonomous vehicles [5], providing continuous monitoring of the environment surrounding it. To track multiple targets, the tracker employs data association techniques, which link measurements to tracks while avoiding false alarms and false measurements [6, 7]. Several data association methods, such as nearest neighbor (NN), joint probabilistic data association, and multiple hypothesis tracking, have been proposed in the literature [8–10]. However, most of the research on multi-target tracking (MTT) focuses on single-sensor systems. Single sensors are susceptible to missing target detections due to their limited field of view (FOV) and restricted radial range. To achieve broader surveillance coverage, multiple sensors can be employed. However, an efficient fusion technique is required to combine measurements from these sensors within the tracking framework. Existing Bayesian filtering algorithms such as Kalman filter, particle filter, and covariance intersection are significantly complex to be used in real time [6, 7]. In addition, in the event of data mismatch, these techniques are prone to diverge and yield unsatisfactory results. The other common approach to grouping measurements from distributed sensors and finding the single centroid is clustering algorithm. However, these algorithms are limited by the challenge of selecting appropriate parameter values, such as minimum distance. Without the right parameters, clustering-based fusion can lead to missed detections, especially in cases where sensors have limited FOV. Furthermore, to the best of the authors' knowledge, MTT techniques are currently mostly confined to offline analysis.

This paper addresses the limitations of existing methods by providing a detailed description of the hardware setup designed for the real-time tracking of maneuvering targets, specifically in indoor environments, using multiple distributed radar sensors. The setup was developed with the goal of performing fusion and MTT on-line, enabling real-time access to target tracks. As shown in Fig. 1, the system consists of several distributed mmWave IWR6843ISK radar sensors from Texas Instruments (TI) [11], synchronized with a precision of 1 millisecond [11, 12]. Robot Operating System (ROS) is utilized to acquire sensor measurements, visualize data, and support sensor synchronization and configuration. A connection is established between the ROS network and MATLAB to process sensor data in real time. Once this connection is configured, the measurements are processed for tracking using the proposed likelihood-based fusion and MTT algorithm from [13]. Unlike DBSCAN [1], the proposed fusion method incorporates measurement information based on their likelihoods and computes a weighted sum of the measurements for MTT tracking. The effectiveness of the proposed technique over DBSCAN-based fusion is demonstrated under sensors' limited FOV conditions, showing how the proposed fusion method successfully maintains the track, in contrast to DBSCAN-based fusion.

The key contributions of the presented work are as follows: A novel likelihood-based fusion technique is introduced to combine measurements from distributed radar sensors. The paper details the hardware setup developed to collect measurements from multiple radar sensors in real time. The proposed fusion technique is integrated with the MTT algorithm to generate target tracks and their counts in real time.

* Corresponding author: Uday Kumar Singh (udaykums@srmist.edu.in).

[1] It is important to note that all clustering algorithms share some limitation, and DBSCAN is used as a reference point for comparison with the other clustering methods.

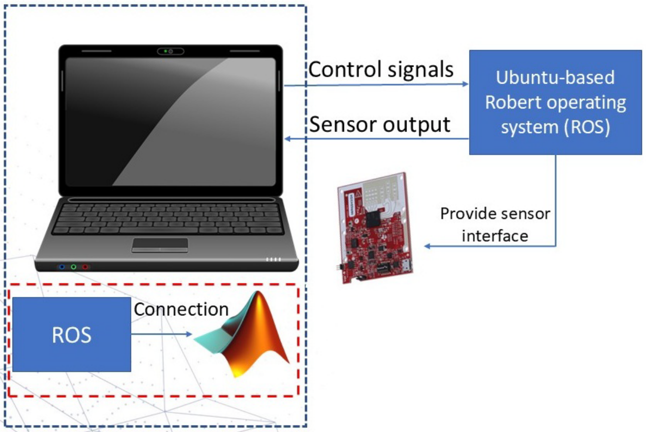

FIGURE 1. A block diagram depicting the process of acquiring measurements from TI's IWR6843ISK radar sensor and processing them in real-time to generate target counts and tracks.

The following subsections will provide an overview of the hardware setup, outline the proposed fusion technique, give a brief description of the MTT process, and present the result analysis based on real-time measurements.

Notations: Scalar variables (constants) are denoted by lower (upper) case letters. Vectors (matrices) are denoted by boldface lower (upper) case letters. The superscripts $(\cdot)^T$ denote the matrix transpose operation. $\mathbb{C}$ and $\mathbb{R}$ denote a set of complex and real numbers, respectively. $\mathbf{I}_n$ denotes the identity matrix of cardinality n.

2. DESCRIPTION OF HARDWARE SETUP

In this section, the hardware setup illustrated in Fig. 2 is discussed in detail. The three sensors: reference sensor, sensor 1, and sensor 2 are installed at different positions. For perfect alignment of the measurements obtained from the sensors, all sensors are raised at the same height. Sensor 1 and sensor 2 maintain a 90° angle in the clockwise and anticlockwise directions, respectively, w.r.t. the reference sensor z axis. Henceforth, after following the schematic shown in Fig. 3, the positions of sensor 1 and sensor 2 w.r.t. reference sensor are calculated manually. This strategic setting of the sensors results in

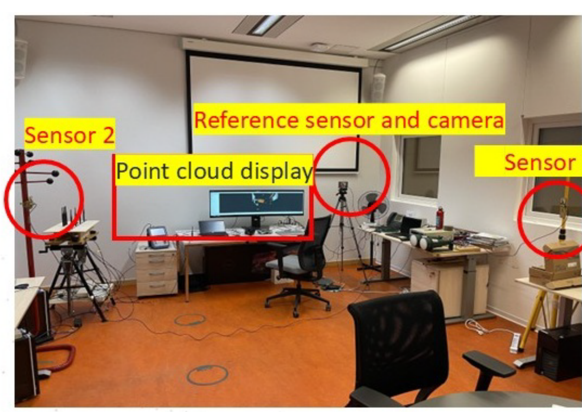

FIGURE 2. Setup to collect the measurements in real-time.

quantifying the rotation matrix, which in turn helps to transform the measurements to the reference sensor.

3. ONLINE MTT USING PROPOSED FUSION TECHNIQUE

This section details the proposed fusion methodology, in particular, and the MTT, in general. The steps involved in the implementation of the MTT algorithm are briefed in Fig. 4.

Let the number of targets located in the surveillance scene be denoted as Q. Also, let, at the kth coherent processing interval (CPI)/scan, the measurements reported by the sth sensor be given by $\mathbf{Z}_s \in \mathbb{R}^{M_s \times n_m} \forall s \in [s_0, \ldots, s_2]$, where s_0 corresponds to the reference sensor; s_1 and s_2 correspond to sensor 1 and sensor 2, respectively; M_s is the number of measurements received from the sth sensor; and n_m is the cardinality of each measurement. Referring to Fig. 3, it is explicit that geometrically the sensors are placed at distinct locations, and consequently, before further processing the measurements received from sensors s_1 and s_2 are needed to transform in the frame of reference of the reference sensor s_0. Having said that, the measurements received from the sensors s_1 and s_2 $(\mathbf{Z}_1, \mathbf{Z}_2)$ will be treated to be transformed into the s_0 frame of reference. As measurements are received in x, y, and z coordinates, $n_m = 3$, and it is also worth mentioning that M_s measurements from the sth sensor could consist of the detections from Q extended targets and clutter (spurious measurements due to reflections not belonging to the targets). The extended targets will result in multiple reflections; this happens because, for indoor scene applications, a finer range, velocity, and azimuth resolution are required. Consequently, the target spans multiple gates which leads to multiple reflections from the particular target. To address this effect, the DBSCAN algorithm is used next; the DBSCAN algorithm will group the measurements together and find the centroid for each group; and the resulting centroids will then be treated as the new point measurements good enough to be treated as the representative of the extended measurements. Referring to Fig. 4, the centroids of the clustered measurements are mentioned as $\mathbf{Z}_s^c \in \mathbb{R}^{M_s^c \times n_m} \forall s \in [s_0, \ldots, s_2]$, where it is trivial to mention that $M_s^c \ll M_s$ and each signal measurement out of M_s^c corresponds to a single reflector in the surveillance scene.

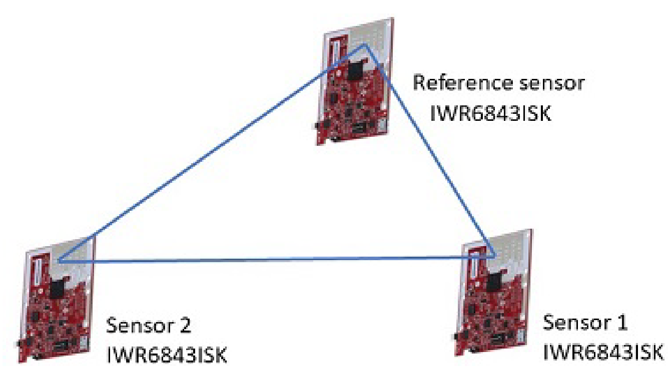

FIGURE 3. Geometric positions of the distributed IWR6843ISK sensors.

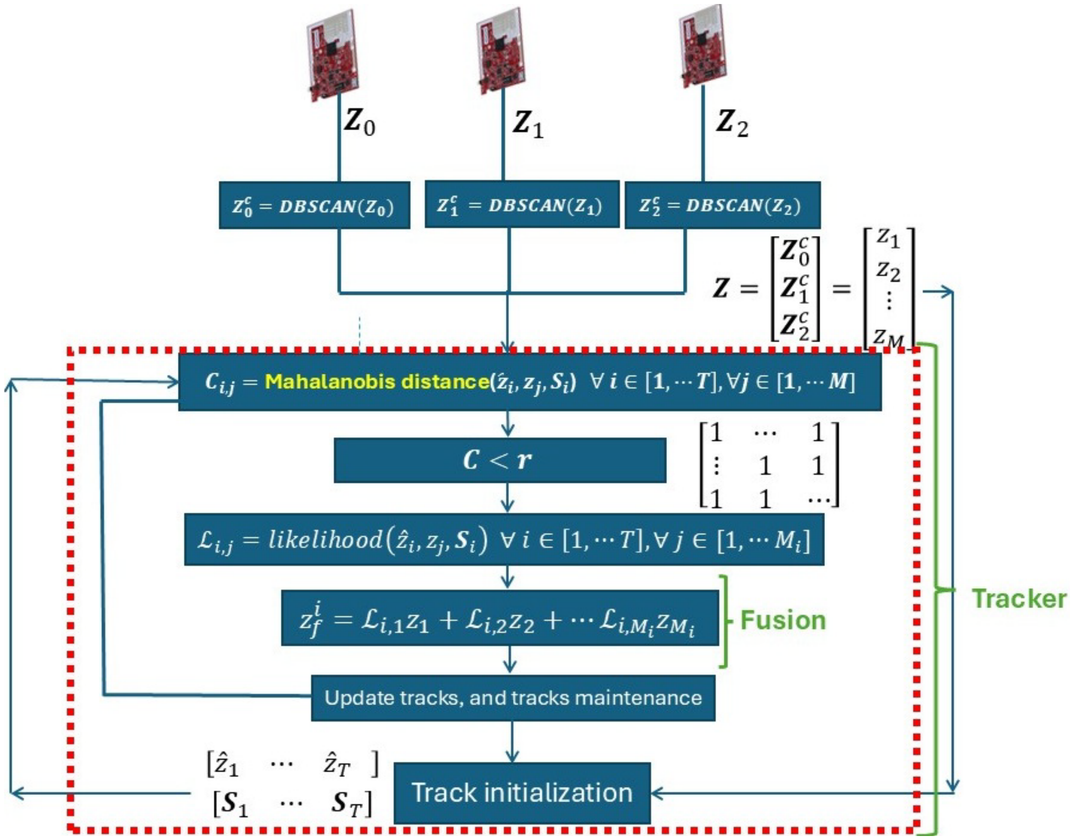

FIGURE 4. Block diagram depiction of the steps to process the measurements obtained by the radar sensor along with the role of proposed fusion technique in MTT.

Furthermore, as shown in Fig. 4, the point measurements $\mathbf{Z}_s^c \forall s$ are concatenated as $\mathbf{Z} \in \mathbb{R}^{(M_0^c+M_1^c+M_2^c)\times n_m} = [\mathbf{Z}_0^c, \mathbf{Z}_1^c, \mathbf{Z}_2^c]^T$, which is further expressed as $\mathbf{Z} = [\mathbf{Z}_0^c, \mathbf{Z}_1^c, \mathbf{Z}_2^c]^T = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_M]^T$, where $M = M_0^c + M_1^c + M_2^c$, and each $\mathbf{z}_j \in \mathbb{R}^{1\times n_m} \forall j \in [1, 2, \dots, M]$. Subsequently, as shown in Fig. 6, the $\mathbf{z}_j \forall j \in [1, 2, \dots, M]$ obtained by the procedure mentioned above will be further processed by the MTT algorithm in the following steps.

Track Initialization: In the beginning, since there are no existing tracks, the system processes $\mathbf{z}_j$ to determine whether there is a possibility to initialize a new track. The track initialization procedure shown in Fig. 4 takes some scans and follows the method described in [13]. If T tracks are successfully initialized, the measurements linked to these tracks are nominated as estimates $(\hat{\mathbf{z}}_1, \hat{\mathbf{z}}_2, \dots, \hat{\mathbf{z}}_T)$, and also each T track is initialized with the error covariance matrices $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_T$. Estimates of measurements with their associated error covariance matrix will be used in the subsequent data association step. After track initialization, $\mathbf{z}_j$ from subsequent scans will be processed further.

Data Association: In the MTT framework, the data association algorithm acts as a filter that filters out measurements that are associated with tracks. Essentially, this step decides which measurements of the M measurements are assigned to the existing tracks. For this purpose, the Mahalanobis distances between the measurements associated with the ith track and jth measurements $\forall\, i \in [1, 2, \dots, T]$ and $\forall j \in [1, 2, \dots, M]$, respectively will be calculated as

$$\mathbf{C}_{i,j} = (\hat{\mathbf{z}}_i - \mathbf{z}_j)\mathbf{S}_i(\hat{\mathbf{z}}_i - \mathbf{z}_j)^T. \tag{1}$$

In (1), if $\mathbf{S}_i = \mathbf{I}_{n_m}$, then $\mathbf{C}_{i,j}$ is reduced to the Euclidean distance. Hence, the Mahalanobis distance is the generalized form of a Euclidean distance for any $\mathbf{S}_i$ but identity matrix. Additionally, even if the measurements are identical in terms of Euclidean distance, they may differ substantially when being evaluated using the Mahalanobis distance. Next, a gate test will be applied to $\mathbf{C}_{i,j}$ to determine if it is smaller than the predefined gate radius r, i.e., $\mathbf{C}_{i,j} < r$. If this condition is met, the jth measurement is considered associated with the ith track and will be processed further. Otherwise, the jth measurement is discarded and considered to be the effect of clutter. This gating test will be performed for all values of i and j, and the results will be stored in the matrix $\mathbf{C}$ as a logical value of 1 if the test is passed and 0 if it is not as

$$\mathbf{C} \in [0, 1]^{T\times M} = \begin{bmatrix} 1 & 0 & \dots & 1 \\ 0 & 1 & \dots & 0 \\ 1 & 1 & \dots & 1 \end{bmatrix}. \tag{2}$$

From Equation (2), it is evident that for any given ith track (row of a matrix $\mathbf{C}$), the number of associated measurements (M_t) is marked with a 1, while the unassociated measurements, $M - M_t$, are represented as 0 in $\mathbf{C}$. The M_t measurements that pass the gating test for the ith track will be forwarded to the next stage for fusion at the proposed fusion center.

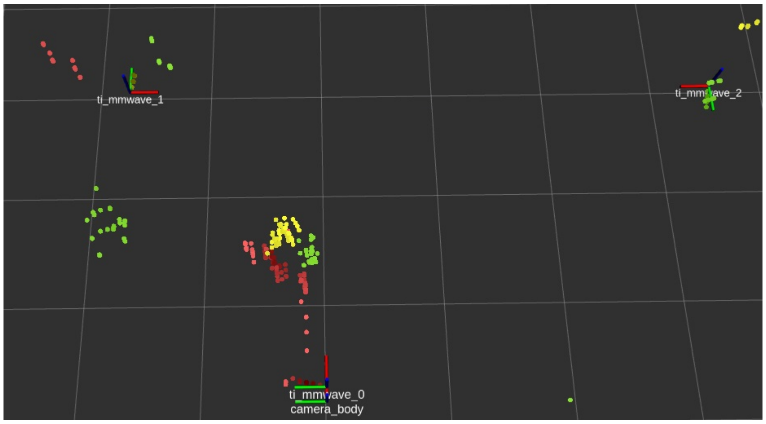

FIGURE 5. ROS display of measurements from reference sensor, sensor 1 and sensor 2, in red, yellow, and green, respectively.

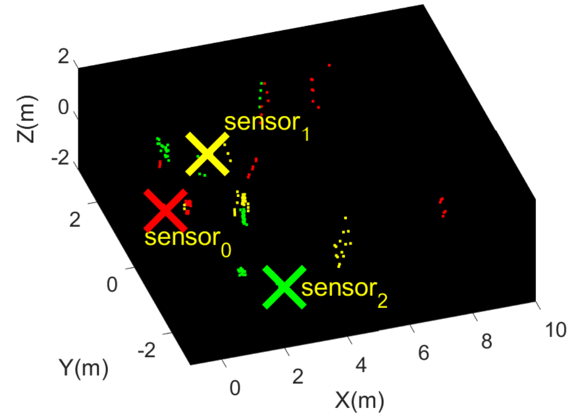

FIGURE 6. Measurements in MATLAB from reference sensor, sensor 1 and sensor 2, in red, yellow, and green, respectively. The depicted measurements corresponds to Fig. 5.

Proposed Fusion Technique: The M_t measurements $([\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{M_t}])$ assigned to the ith track can be processed individually to update the track information. However, processing each of these M_t measurements individually is computationally intensive. An alternative approach would be to treat the centroid (obtained by averaging the $([\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{M_t}])$) of the M_t measurements as a fusion of the measurements; however, this is a simplistic solution. Therefore, in the proposed fusion technique, the centroids of the measurements are obtained by fusing the measurements with their respective likelihoods, which are defined as $\forall i \in [1, 2, \ldots, T]$ and $\forall j \in [1, 2, \ldots, M_t]$ as follows:

$$\mathcal{L}_{i,j} = exp(-(\hat{\mathbf{z}}_i - \mathbf{z}_j)\mathbf{S}_i(\hat{\mathbf{z}}_i - \mathbf{z}_j)^T)\forall i, j \tag{3}$$

Subsequently, the M_t measurements associated with any ith track are fused as follows

$$\mathbf{z}_f^i = \mathcal{L}_{i,1}\mathbf{z}_1 + \mathcal{L}_{i,2}\mathbf{z}_2 + \ldots + \mathcal{L}_{i,M_t}\mathbf{z}_{M_t}, \tag{4}$$

where $\mathbf{z}_f^i$ represents the fused measurements from $([\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{M_t}])$, which will be used to update the ith track information. To obtain $\mathbf{z}_f^i$, all associated measurements are weighted by their corresponding likelihoods $[\mathcal{L}_{i,1}, \mathcal{L}_{i,2}, \ldots, \mathcal{L}_{i,M_t}]$. As a result, measurements closer to $\hat{\mathbf{z}}_\mathbf{i}$ will be given more weight in the fusion process than those farther away from $\hat{\mathbf{z}}_\mathbf{i}$. This approach ensures that the fused measurements, $\mathbf{z}_f^i$, appropriately account for the likelihoods by exploiting the contained information, in contrast to simpler methods.

Update tracks and tracks maintenance: Once $\mathbf{z}_f^i$ for all $i \in [1, 2, \ldots, T]$ are obtained, they are used to update the kinematic information of existing T tracks. This step also handles the deletion of tracks in cases where targets leave the scene or when there is a missed detection due to the sensor's limited FOV. For further details on track updates and deletions, the reader is encouraged to refer to [13].

The aforementioned steps will be repeated for all scans, resulting in continuous updates of the tracks.

4. RESULTS OF ONLINE ANALYSIS AND DISCUSSION

This section describes the method by which the hardware setup is used for the purpose of the online tracking. In addition, the obtained results are also discussed which are relevant to the performance of the proposed fusion technique in the MTT.

As shown in Fig. 5, ROS is used to access and display the measurements from the sensors. The measurements displayed in the ROS visualization tool are shown in Fig. 5. The TI IWR6843ISK radar sensors provide multidimensional measurements spanning x, y, and z in 3D coordinates, which are essentially the position of the reflectors relative to the individual sensor. The measurements are obtained after performing the basic signal processing steps that include dechirping, beam-forming, FFT (fast Fourier transform), and CFAR (constant false alarm rate) detections. As MTT and fusion algorithms are developed in MATLAB, the connection is built between the ROS network and MATLAB. The connection helps to access the measurements in real time from the sensors directly in MATLAB. The measurements in MATLAB are shown in Fig. 6, which is equivalent to the measurements shown in Fig. 5, in which the crosses represent the position of the sensors. After the connection between the ROS network and MATLAB is built, the MTT and fusion algorithms are applied to the received measurements.

For better clarity, Fig. 7 illustrates the measurements at different stages of the algorithm shown in Fig. 4. As shown, the measurements within the dashed red ellipse are the ones recorded by the reference sensor and correspond to the elements of the matrix $\mathbf{Z}_0$. The diamond shape then represents the centroid of the measurements enclosed within the solid red ellipse. Diamonds represent the centroid obtained by the conversion of extended measurements to point measurements, and they form the matrix $\mathbf{Z}_0^c$. Finally, the fusion of the centroids is enclosed within the solid blue ellipse and forms the vectors $\mathbf{z}_f^1$ and $\mathbf{z}_f^2$.

Referring back to Fig. 4, $\mathbf{z}_f^1$ and $\mathbf{z}_f^2$ are the fused measurements, which will be processed further to update the existing tracks.

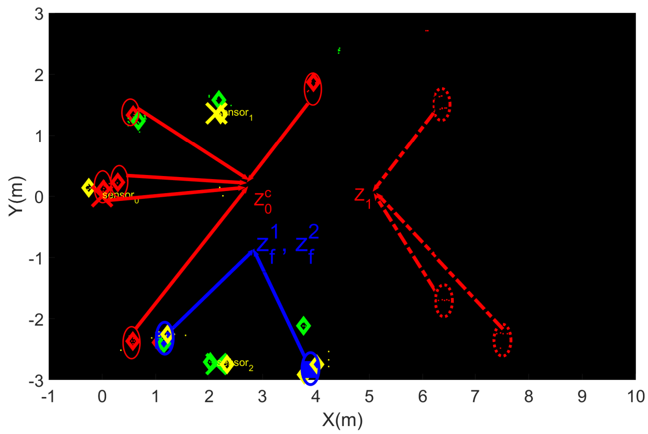

FIGURE 7. Measurements in MATLAB from reference sensor, sensor 1 and sensor 2, in red, yellow, and green, respectively. The figure also depicts the measurements at the various processing steps of the tracking algorithm depicted in Fig. 4.

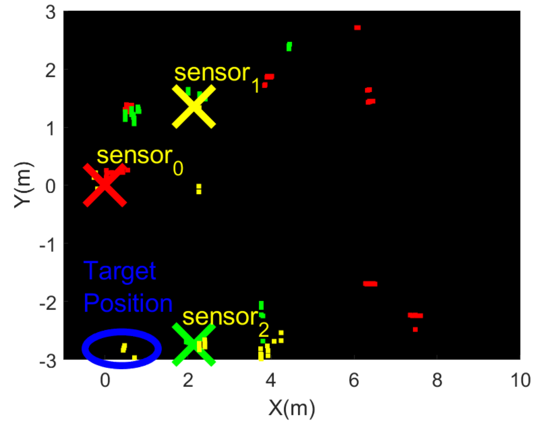

FIGURE 8. The figure depicting the position of the target out of reference sensor and sensor 2 FOV.

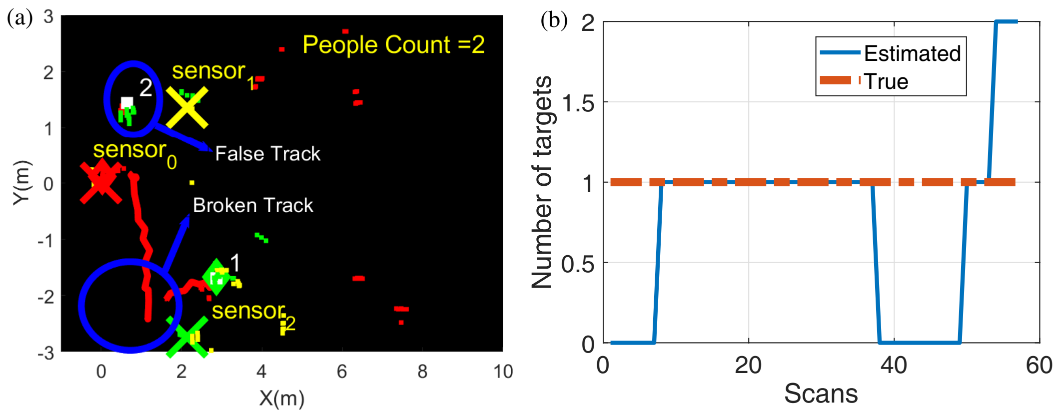

FIGURE 9. Tracking result for Scenario I using DBSCAN clustering for fusion. (a) Target track in red. (b) Number of estimated and true targets.

To access the effectiveness of the proposed likelihood-based fusion technique, two scenarios are considered. Scenario I: the target intentionally goes out of the sensor's FOV, and doing this, the Scenario I helps to validate the efficacy of the fusion techniques in the scenarios having a shortage of measurements. In Fig. 8, the measurements for this scenario are depicted, and the target position is encircled by a blue ellipse. It is evident from Fig. 8 that only a few measurements of sensor 1 (which are yellow in color) are available at this position, and the target is out of the FOV of sensor 0 and sensor 2. Scenario II: multiple targets are considered in the surveillance scene shown in Fig. 1, and this scenario validates the creditability of the proposed fusion technique in the scenario of multiple targets.

Scenario I: In this scenario, a target is present in the surveillance scene and gradually moves to a position where fewer measurements are reported; the corresponding tracking results are shown in Fig. 9, in which the MTT tracks the target's movement using the DBSCAN algorithm for the fusion of sensors' data. As shown in Fig. 9(a), when the target moves out of the FOV of sensor 0 and sensor 2, the lack of measurements causes the DBSCAN algorithm to fail to form a cluster. As a result, due to the absence of measurements, the tracker loses the developed track for a few scans which is re-established once the target returns to the FOV of all sensors. In addition to this, as depicted in Fig. 9(b) due to improper values of the DBSCAN parameters, a false track is also generated. The discontinuity of the track and the creation of the false track, stemming from the limitations of the DBSCAN algorithm, are further summarized/quantized in Fig. 9(b), in which between the 38th and 49th scans, the target is outside the FOV of sensors 0 and 2. During this period, the estimated number of target tracks drops to zero, only to resume as 1 when the target reenters the FOV of the sensors. However, after some scans, a false track is formed, incorrectly indicating the estimated number of targets as 2.

Next, the measurements corresponding to Scenario I are processed by the proposed likelihood-based fusion algorithm and MTT, and the obtained results are shown in Fig. 10. As shown in Fig. 10, contrary to DBSCAN, likelihood-based fusion does not suffer from the discontinuity of the track and the false track generation. This is because the likelihood-based framework leverages the average of all potential measurements weighted by their corresponding likelihoods.

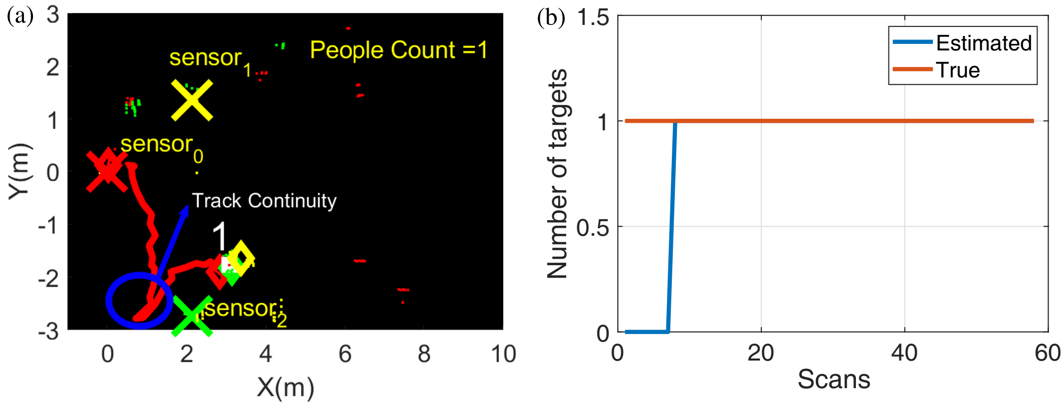

FIGURE 10. Tracking results for Scenario I using proposed likelihood-based fusion technique. (a) Target track in red. (b) Number of estimated and true targets.

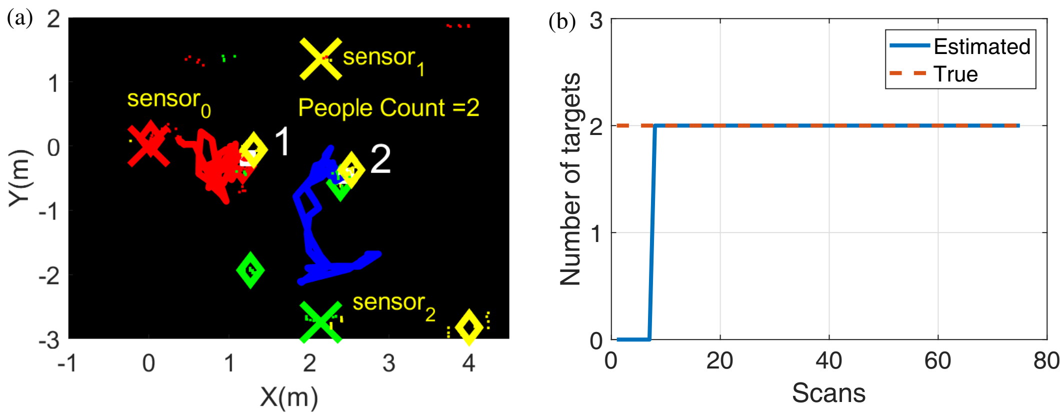

FIGURE 11. Tracking results for Scenario II using the proposed likelihood-based fusion technique. (a) Target tracks in red and blue for first target and second target, respectively. (b) Number of estimated and true targets.

As a result, if measurements from a single sensor are available, the proposed fusion technique will utilize them to keep the continuity of the track. Furthermore, in Fig. 10(b), the performance is summarized in terms of estimated tracks, which is 1 for all scans without any discontinuity and false track generation.

Scenario II: In this scenario, two targets are present in the surveillance scene, both maneuvering within the sensors' FOV throughout all the scans. The corresponding tracking results using the proposed technique are shown in Fig. 11. From Fig. 11, it is clear that the tracker generates tracks for the two targets, along with their respective counts. In addition, the tracker accurately estimates the number of people present in the surveillance scene. Furthermore, the view of the 2nd target from sensor 0 is occluded by the 1st target; however, the 2nd target remains within the FOV of sensor 1 and sensor 2, allowing them to continue to provide measurements that help update the track, as shown in Fig. 11(b). Additionally, in cases where the target is completely occluded from all sensors' fields of view, the MTT algorithm maintains the track by relying on previous measurements for a few scans. If no new measurements are received after several scans, the algorithm will eventually delete the corresponding track [13].

5. CONCLUSIONS

This paper has introduced a novel approach to tracking and counting people in real time using distributed radar sensors, supported by a fusion algorithm that takes advantage of the likelihood of measurements. The study demonstrated how the measurements from distributed radar sensors are collected and processed for the purpose of tracking and counting people in real time. ROS has played a role as an intermediate in collecting the measurements, and the tracking and fusion algorithm is developed in MATLAB. The presented results highlight the robustness of the proposed method, particularly in handling challenging scenarios such as target movement, leading to fewer measurements and sensor FOV limitations. Tracking performance analysis demonstrated that, while the DBSCAN-based fusion approach struggled with track discontinuities and false track generation, the likelihood-based fusion algorithm provided superior performance. It effectively maintained track continuity and avoided false track formation by utilizing all available sensor data, even under sparse conditions. The results confirm that the proposed method provides accurate and continuous tracking, offers significant improvements over traditional algorithms, and shows strong potential to be deployed for real-time surveillance and people-counting applications.

By default, the presented study assumes the universally accepted Gaussian distribution for the noise. This presents an opportunity to expand our work in the future, with the aim of enhancing the robustness of the proposed approach so that it becomes resilient to the effects of different noise distributions.

REFERENCES

[1] Richards, M. A., *Fundamentals of Radar Signal Processing*, McGraw-Hill New York, 2005.

[2] Tugac, S. and M. Efe, "Radar target detection using hidden Markov models," *Progress In Electromagnetics Research B*, Vol. 44, 241–259, 2012.

[3] Blackman, S. S., "Multiple-target tracking with radar applications," *Dedham*, 1986.

[4] Singh, U. K., M. Alaee-Kerahroodi, and M. R. B. Shankar, "RKHS based state estimator for radar sensor in indoor application," in *2022 IEEE Radar Conference (RadarConf22)*, 01–06, New York City, NY, USA, Mar. 2022.

[5] Jeong, S.-H., H.-Y. Yu, J.-E. Lee, J.-N. Oh, and K.-H. Lee, "A multi-beam and multi-range radar with FMCW and digital beam forming for automotive applications," *Progress In Electromagnetics Research*, Vol. 124, 285–299, 2012.

[6] You, H., J. Xiu, and X. Guan, *Radar Data Processing with Applications*, John Wiley & Sons, 2016.

[7] Bar-Shalom, Y., T. E. Fortmann, and P. G. Cable, "Tracking and data association," *The Journal of the Acoustical Society of America*, Vol. 87, No. 2, 918–919, 1990.

[8] Leung, H., Z. Hu, and M. Blanchette, "Evaluation of multiple radar target trackers in stressful environments," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 35, No. 2, 663–674, 1999.

[9] Livshitz, M., "Tracking radar targets with multiple reflection points," TI Internal Document, Texas Instruments, 2017.

[10] Huang, X., H. Cheena, A. Thomas, and J. K. P. Tsoi, "Indoor detection and tracking of people using mmWave sensor," *Journal of Sensors*, Vol. 2021, No. 1, 6657709, 2021.

[11] Texas Instruments, "IWR6843, IWR6443 single-chip 60 to 64 GHz mmWave sensor datasheet," 2022.

[12] Texas Instruments, "mmWave radar ROS node: Time sync of frames from multiple sensors," [Online]. Available: https://software-dl.ti.com/jacinto7/esd/robotics-sdk/08_06_00/docs/source/docs/multi_sensor_time_synchronization.html.

[13] Singh, U. K., M. Alaee-Kerahroodi, and M. R. B. Shankar, "Implementation and validation of multiple target tracking technique for indoor applications," in *2022 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 219–224, Gandhinagar, Gujarat, India, Dec. 2022.