FMCW Radar Point Cloud Multiperson Tracking Using a Kalman Filter–Based Approach

Bruno Santos, Arnaldo S. R. Oliveira, Nuno Borges Carvalho, Rui Fernandes, Andrea Cannizzaro, and Pedro Miguel Cruz

Abstract – In this article, we present a processing chain for multitarget tracking in an indoor scenario using a Texas Instruments IWR6843 radar development kit. The processing chain works on top of a radar point cloud and can detect and track people using a clustering algorithm and Kalman filtering. The detected and postprocessed data are then fused onto a live image acquisition from a surveillance camera as a complementary sensor.

1. Introduction

Radar is a widely used technology in military and civil applications, such as ballistic targeting, weather forecasting, and speed detectors [1]. With the growth of smart cities and industrial automation, radar is finding new applications as a sensor capable of providing reliable data without raising data protection concerns.

Emerging smart cities pushed surveillance cameras to forefront as one of the most viable sensors capable of detecting and tracking people and objects in real time by algorithms enabled through machine learning or artificial intelligence. Although the camera is widely used, it has some problems when visibility is low that can be caused by nonbeneficial atmospheric conditions. This is where radar enters as a complementary sensor to help the camera with point cloud data that are unaffected by those atmospheric conditions [2].

In this article, an application is presented that is capable of using radar data, detecting and tracking people within the radar field of view (FoV) and merging these data with real-time image streaming from a surveillance camera. The radar of choice was the Texas Instruments (TI) IWR6843ISK-ODS, which is a multiple-input multiple-output (MIMO), frequency-modulated continuous wave (FMCW) radar operating at the 60 GHz band and capable of providing real-time point cloud detections in three-dimensional (3D) Cartesian coordinates together with velocity information.

2. People Detection, Tracking, and Sensor Fusion

In this section, we describe a processing chain that can detect and track multiple persons in the FoV using point clouds and that is composed of four main steps: clustering, filtering, tracking, and sensor fusion (see Figure 1).

2.1 Clustering

The first stage is to find detections in the Cartesian data. To do so, clustering methods can identify a group of points in the spatial domain with an arbitrary shape, which is the case of data retrieved from the radar. The method chosen was density-based spatial clustering of applications with noise (DBSCAN) [3]. This method relies only on the minimum number of points (*min Pts*) and the maximum distance between them (*Eps*). The intuitive way for a human to visually detect a cluster is to find points that are closely spaced between them; this is the key idea behind DBSCAN. For each point detected in a given *Eps* radius, the neighborhood has to have a minimum number of points belonging to the spherical volume. After the clusters are identified, the centroid is calculated and used for tracking purposes.

2.2 Filtering

The clutter originating from multipath reflections creates clusters of points in different spatial coordinates, so a single person moving in a room can originate multiple clusters [4]. Typically, a multipath reflection means a longer distance traveled by the wave, and the power received by the antennas is inversely proportional to the traveled distance, meaning that the clutter will have a much smaller power. Because the multipath distance is usually longer than the direct distance, the average SNR of a cluster originated by a real target will be larger than the respective clutter SNR, so for a cluster to be considered a target, it will need to have an average SNR larger than a certain threshold that scales with distance to the radar.

Objects with an obstructed line of sight (no direct path) yet inside the radar's FoV may be detected if the reflections are so strong that the radar's processing chain does not consider it as noise. In those situations where the radar detects multipath reflections, the location of the detections is the direct path between the radar and the last bounce of the reflected wave. Such types of detections are rarely obtained, as they

Manuscript received 29 December 2021.

Bruno Santos is with the University of Aveiro, Campus Universitário de Santiago, 3810-193 Aveiro, Portugal; e-mail: b.santos@ua.pt.

Arnaldo S. R. Oliveira and Nuno Borges Carvalho are with the Instituto de Telecomunicações, University of Aveiro, Campus Universitário de Santiago, 3810-193 Aveiro, Portugal; e-mail: arnaldo.oliveira@ua.pt, nbcarvalho@ua.pt.

Rui Fernandes, Andrea Cannizzaro, and Pedro Miguel Cruz are with Bosch Security Systems S.A., Zona Industrial de Ovar, 3880-080 Ovar, Portugal; e-mail: ruifilipe.fernandes@pt.bosch.com, andrea.cannizzaro@pt.bosch.com, pedro.cruz4@pt.bosch.com.

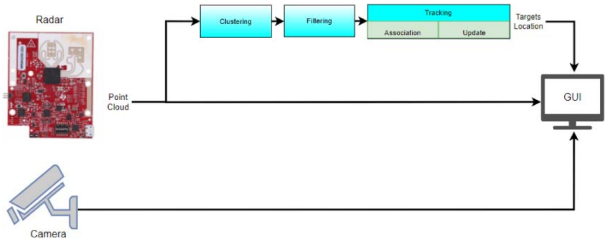


Figure 1. Setup installation and application flowchart.

difficultly trigger a cluster identification in the DBSCAN algorithm.

2.3 Tracking

The Kalman filter is a mathematical model that uses measurements over time that may contain noise and produce a prediction about the real values of the state vector, which contains information about the spatial position, velocity, and acceleration in a 3D Cartesian space. These predictions are based on previous measurements and assume a constant acceleration model [5]. The state transition matrix, F, is used to describe the dynamics of the system with t time between measurements. Since it is a constant acceleration model, the state change between each time step is described as follows:

$$\begin{aligned} X_k &= X_{k-1} + \dot{X}_{k-1}\Delta t + \ddot{X}_{k-1}\frac{\Delta t^2}{2} \\ \dot{X}_k &= \dot{X}_{k-1} + \ddot{X}_{k-1}\Delta t \\ \ddot{X}_k &= \ddot{X}_{k-1} \end{aligned} \tag{1}$$

where $X = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z}]^T$ represents the state vector that stores the measurement variables. The state transition matrix, F, is then defined as the following matrix (2) with the system behavior derived from the Newtonian equation $x = x_0 + v_0 t + \frac{1}{2}at^2$:

$$F = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 & 0.5T^2 & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 & 0 & 0.5T^2 & 0 \\ 0 & 0 & 1 & 0 & 0 & T & 0 & 0 & 0.5T^2 \\ 0 & 0 & 0 & 1 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

The state vector, X, at the kth measurement is

$$X_k = FX_{k-1} + w_k \tag{3}$$

$$Z_k = HX_{k-1} + e_k \tag{4}$$

where w_k is the process noise with covariance $\mathbf{Q}$, which represents the deviation between the actual state and the motion model. Z_k is the system measurement that can be extracted through the observation matrix, H_k, and e_k is the measurement noise.

2.3.1 *Predict Stage:* The first step is to predict an a priori state estimate, X_k, and covariance, P_k, as defined by

$$\begin{aligned} X_k &= FX_{k-1} + w_{k-1} \\ P_k &= F_k P_{k-1} F_k^T + Q_K \end{aligned} \tag{5}$$

2.3.2 *Update Stage:* Then the measurement residual, y_k, is calculated by using

$$y_k = z_k - HX_k \tag{6}$$

The residual covariance, S_k, is calculated through (7) and the respective Kalman filter gain, K_k, with (8), with R_k the measurement covariance matrix obtained from the measurement residual, y_k:

$$S_k = H_k P_k H_k^T + R_k \tag{7}$$

Then the Kalman gain matrix, K_k, is calculated as

$$K_k = P_k H_k^T S_k^{-1} \tag{8}$$

After the Kalman gain is calculated, the updated state and covariance estimation are reached through

$$\begin{aligned} X_k &= X_{k-1} + K_k y_k \\ P_k &= (I - K_k H_k) P_k \end{aligned} \tag{9}$$

The state transition, F_k, and observation matrix, H_k, are nonlinear differentiable equations. The extended Kalman filter approximates these functions into a linear approximation by taking the partial derivatives at point x_k [6]:

$$\begin{aligned} F_k &= \frac{\partial f}{\partial x}|_{X_k} \\ H_k &= \frac{\partial h}{\partial x}|_{X_k} \end{aligned} \tag{10}$$

The partial derivatives are computed through the Jacobian of the matrices, which means that the Jacobian is iteratively calculated for each time step.

As seen in Figure 2, the algorithm starts by predicting every existent track. The results of the predict stage are compared to the measured cluster centroid, and then the closest distance between the centroid and the prediction is the associated one. After the association stage, the measurement is used to calculate the update stage, and a new state vector for the track is obtained.

2.4 Sensor Fusion

The last part of the processing chain is to merge the 3D data onto a two-dimensional (2D) image provided by the surveillance camera. This is done through perspective projection, where a 3D point with Cartesian coordinates is projected onto a plane at a focal distance (f) [7]. This can be seen in Figure 3, and the

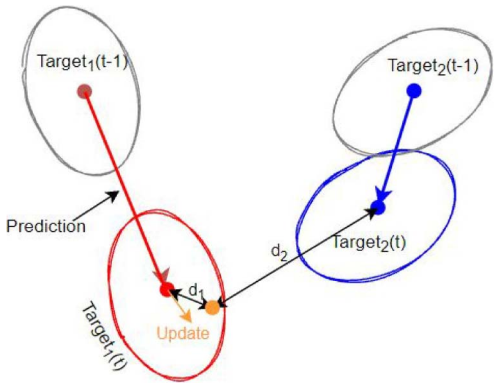


Figure 2. Tracking logic with the predict, associate, and update stages.

coordinates of the projection onto the plane can be calculated through triangle similarities:

$$\frac{x'}{f} = \frac{X}{Y} \implies x' = f\frac{X}{Y} \tag{11}$$

where x' is the coordinate of the projection onto the plane, X and Y are the 3D Cartesian coordinates of the target, and f is the focal distance. The coordinates of the z' can also be estimated the same way as the above:

$$\frac{z'}{f} = \frac{Z}{Y} \implies z' = f\frac{Z}{Y} \tag{12}$$

Then each projected point must be converted into a pixel location on an image.

3. Experimental Results

The configured parameters of the FMCW radar are presented in Table 1. The selected configuration was the common ground found to make possible the detection of multiple persons with large enough clusters so that the DBSCAN algorithm was able to separate the clusters into different targets and label them.

For tracking purpose, three tests were done:

1. A person walking in direction of the radar
2. A person doing an "8" walking shape
3. Two people walking in opposite directions

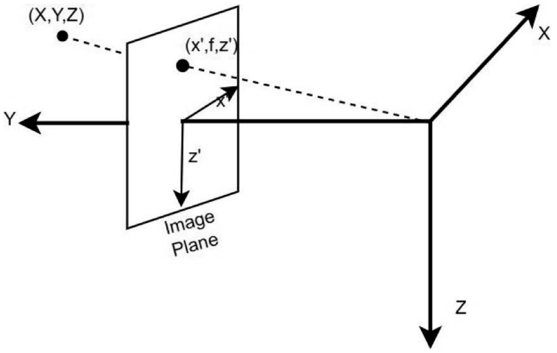


Figure 3. Perspective projection of a 3D point onto an image plane.

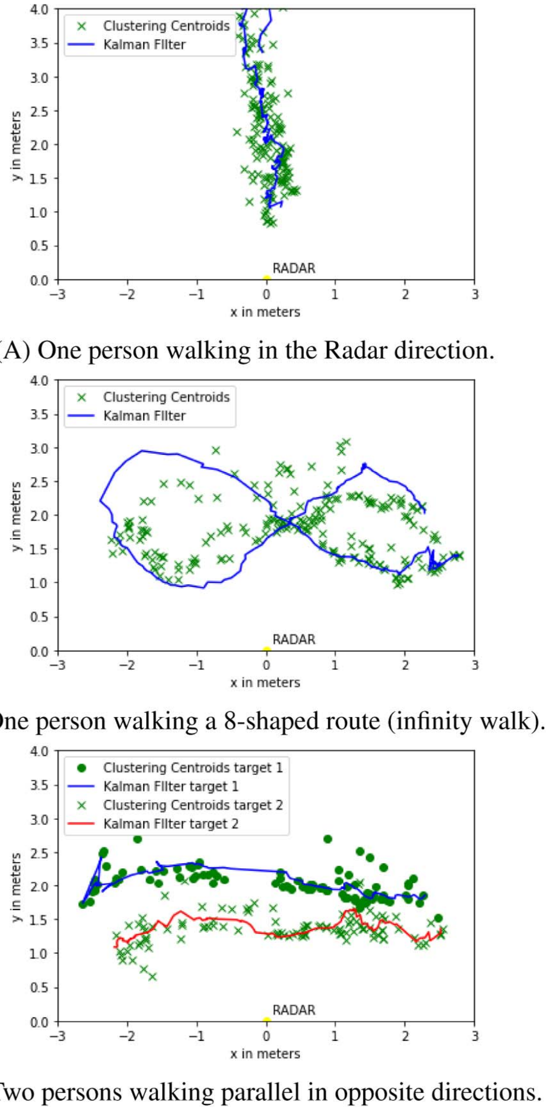
Table 1. Radar characteristics and parameters used for the experiments

Initial frequency	60.20	GHz
Ramp slope	74.95	MHz/µs
Bandwidth	2974	MHz
Range resolution	0.05	m
Maximum range	5.4	m
Velocity resolution	0.18	km/h
Maximum velocity	26.9	km/h
Sampling frequency	2.95	Msps

The results are shown in Figure 4.

The green points are the cluster centroids detected, and the blue line is the evolution of the Kalman state vector.

For the sensor fusion, the point clouds and the target location were projected onto an image, as seen in



(A) One person walking in the Radar direction.

(B) One person walking a 8-shaped route (infinity walk).

(C) Two persons walking parallel in opposite directions.

Figure 4. Tracking tests with real-time radar point cloud data and algorithm postprocessing.

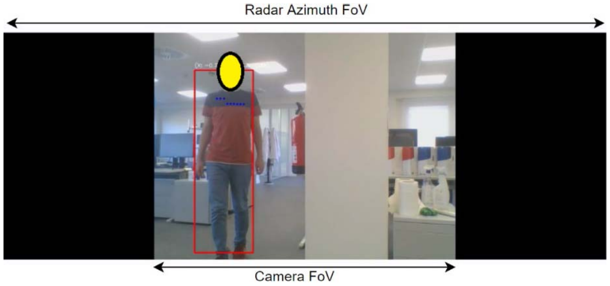


Figure 5. Sensor fusion of 3D data onto image with enhanced FoV.

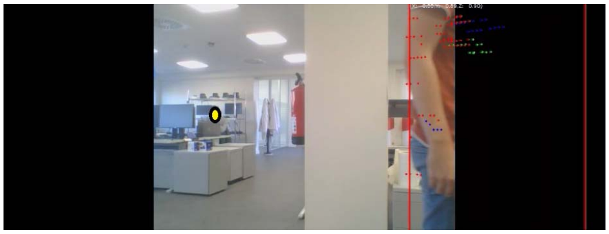


Figure 6. Enhancing the camera FoV with detection outside the camera FoV.

Figure 5. The red bounding box represents the target location, and the points are the point cloud projections.

Since the radar has a wider azimuth FoV, the black columns represent the camera blind side, which is enhanced by the radar data, as shown in Figure 6, being able to estimate a person outside the camera FoV.

4. Conclusion

A processing chain for multiperson detection and tracking using a MIMO FMCW radar was presented. This 3D point cloud information was also fused with a video camera for surveillance system improvement. The processing was evaluated in multiple scenarios where the tracking was able to follow the walking path and the data were projected onto a live image stream. The presented sensor fusion technique works in an asynchronous mode; that is, the radar tracking operates with a variable processing time between 10 and 20 frames per second, while the video stream runs at a fixed rate of 30 frames per second. This was implemented in a way to ensure that the video stream is not bottlenecked by the point cloud tracking process.

5. References

1. M. A. Richards, J. A. Scheer, W. A. Holm, "Introduction and Radar Overview" in Principles of Modern Radar, Vol. 1, SciTech Publishing, 2010, Chapter 1.
2. H. Zhou, P. Cao, and S. Chen, "A Novel Waveform Design for Multi-Target Detection in Automotive FMCW Radar," 2016 IEEE Radar Conference (RadarConf), May 2-6, 2016, Philadelphia, United States, pp. 1-5.
3. M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD'96), Portland, Oregon, August 1996, pp. 226-231.
4. M. A. Richards, Fundamentals of Radar Signal Processing, 2nd Edition, McGraw Hill Professional, 2014.
5. Texas Instruments Inc., 2017, "Tracking Radar Targets With Multiple Reflection Points", https://e2e.ti.com/cfs-file/__key/communityserver-discussions-components-files/1023/Tracking-radar-targets-with-multiple-reflection-points.pdf (Accessed 26 October 2021).
6. R. G. Brown and P. Y. C. Hwang, Introduction to Random Signals and Applied Kalman Filtering with Matlab Exercises, 4th Edition, Wiley, 2012.
7. "The Geometry of Perspective Projection," https://www.cse.unr.edu/~bebis/CS791E/Notes/PerspectiveProjection.pdf (Accessed 26 October 2021).