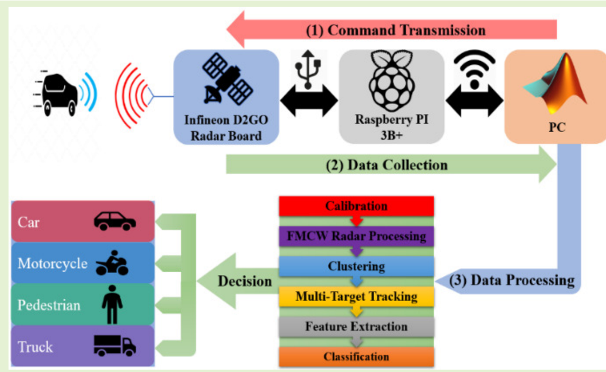
Cost-Efficient FMCW Radar for Multi-Target Classification in Security Gate Monitoring

Ali Rizik, *Student Member, IEEE*, Emanuele Tavanti, *Member, IEEE*, Hussien Chible, *Member, IEEE*, Daniele D. Caviglia, *Member, IEEE*, and Andrea Randazzo, *Senior Member, IEEE*

***Abstract*—A prototype of a radar system for classification of multiple targets passing through a road gate is presented in this paper. It allows to identify different types of targets, i.e., pedestrians, motorcycles, cars, and trucks. The developed system is based on a low-cost 24 GHz off-the-shelf FMCW radar, combined with an embedded Raspberry Pi PC for data acquisition and transmission to a remote processing PC, which take care of detection and classification. The processing chain relies upon a tracking algorithm to follow the targets during traversal, combined with a classification scheme based on support vector machines. The approach has been validated with experimental data acquired in different scenarios, showing good identification capabilities.**



***Index Terms*— FMCW radar, electromagnetic scattering, range-Doppler processing, edge computing, target classification.**

I. Introduction

The emergence of the autonomous driving cars along with the ongoing advancements in the autonomous vehicles technology, drove the research for developing a pedestrian recognition system to ensure pedestrians safety [1]–[4]. In addition, the continuously increasing threat of national terrorism and criminal acts is now considered the main driving force for researchers to develop more reliable surveillance systems to be installed in urban environments [5]. Optical and radar sensors are nowadays considered the most promising technologies to be used in such systems [6]. In fact, unlike vision sensors, radar apparatuses are more robust in bad lighting and weather conditions, which can severely affect the performance of optical techniques [7], [8]. Different types of radars were used in target detection and classification applications. In particular, the Frequency Modulated Continuous Wave radars (FMCW) are the mostly adopted ones in different applicative field, and in particular in automotive applications [9]–[14], since they are able to simultaneously provide both range and velocity. Such an information can be obtained through different processing scheme, e.g., by using a 2D FFT technique [15]. Moreover, unlike pulse and Ultra-Wide Band (UWB) radars, FMCW systems require lower sampling rates and lower peak-to-average power ratio to detect distance and speed of multiple moving targets [16], [17]. Many different approaches for FMCW radar target classification have been presented in the literature. Some of them adopt classical machine learning techniques with features extracted from the FFT spectrums or the Range-Doppler maps. For example, authors in [18] and [19] presented a human-vehicle classification system based on a 24 GHz FMCW radars sensor, and a support vector machine classifier (SVM) [20]. The system showed a good classification performance using features extracted from the range and velocity profiles of the targets. Another human-vehicle classification system based on SVM and a 77 GHz FMCW radar sensor was introduced in [21], where features based on the Radar Cross Section (RCS) were used. In addition, a vehicle classification system based on radar measurements was presented in [22], where height and length profile-based features were used.

Deep learning classification techniques applied on radar measurements for radar target classification has also been adopted in the literature [23]–[25]. A moving target classification system based on automotive radar and deep neural networks was presented in [26]. Another radar target classification system based on FMCW radar and recurrent neural networks (RNN) were presented in [27] and [28]. However, unlike classical machine learning techniques, for optimal performance, deep learning networks usually require a

Manuscript received April 1, 2021; revised June 1, 2021; accepted June 25, 2021. This work was partially supported by Regione Liguria through the POR-FESR project "GenovaSicura" under grant no. G33D0180000D90007.

D. D. Caviglia, A. Randazzo, A. Rizik, and E. Tavanti are with the Department of Electrical, Electronic, Telecommunications Engineering and Naval Architecture (DITEN), University of Genoa, 16145 Genoa, Italy (e-mail: daniele.caviglia@unige.it, andrea.randazzo@unige.it, ali.rizik@edu.unige.it, emanuele.tavanti@edu.unige.it).

H. Chible is with the MECRL – EDST Lebanese University (e-mail: hchible@ul.edu.lb).

huge amount of data to perform the training step. Indeed, this implies the need of a big memory capacity, especially with the largest networks. This puts some limits on the required hardware specifications to run deep learning algorithms with big training data, especially at the GPU level. Also, a bigger training set means that much more training time is required.

The extraction of micro-Doppler features was another method used by researchers in the literature for radar target classification [29]. A Human-Robot classification system based on 25 GHz FMCW radar using micro-Doppler features was introduced in [30]. Another target classification system based on micro-Doppler signatures was introduced in [31]. Although micro-Doppler features proved to be good in radar target classification, they are difficult to extract. Indeed, extraction of micro-Doppler features requires having long observation periods along with a large number of consequent observations to be processed. This fact makes impossible to extract such features with low-cost radar devices.

In this paper, a multi-target classification system is proposed based on a short-range low-cost 24 GHz FMCW radar sensor. It is worth noting that most of the systems mentioned before mainly deals with single target situation. Based on the authors' knowledge, FMCW radar systems that deals with multi-target classification problems has been rarely reported in the literature. The proposed system consists of a Distance2Go radar board developed by Infineon technologies [32], a Raspberry Pi 3B+ mini-PC, and a portable PC running MATLAB software. The radar board continuously transmits the radar signal and receives echo signals reflected by the targets. The Raspberry Pi collects the targets echo signals and transmits them to the remote PC for processing. The proposed system is equipped with a novel processing chain that allows the detection of multiple moving targets in cluttered environment. The detected targets are tracked and finally classified among one of four classes, namely trucks, cars, pedestrians, and motorcycles. The proposed system was tested in two different cluttered environments. The cost of the measurement setup presented in this paper is around just 300 € (radar board, ~200 €; Raspberry PI 3B+, ~50 €, optional camera, ~10 €; power bank along with the boxing and the other expenses, ~50 €), which is one of the key features of the system.

The structure of the paper is organized as follows. The architecture of the proposed system along with a brief illustration of the FMCW processing is presented in Section II. A detailed explanation of the proposed algorithm processing chain (calibration, clustering, tracking, feature extraction, and classification) is provided in Section III. Section IV shows the system performance in multi-target classification. Finally, conclusions are drawn in Section V.

II. SYSTEM ARCHITECTURE

A. System Architecture

A block diagram of the developed system is shown in Fig. 1. It consists of three main parts:

- A radar board
- A Raspberry PI 3B+ embedded PC [30]
- A control PC

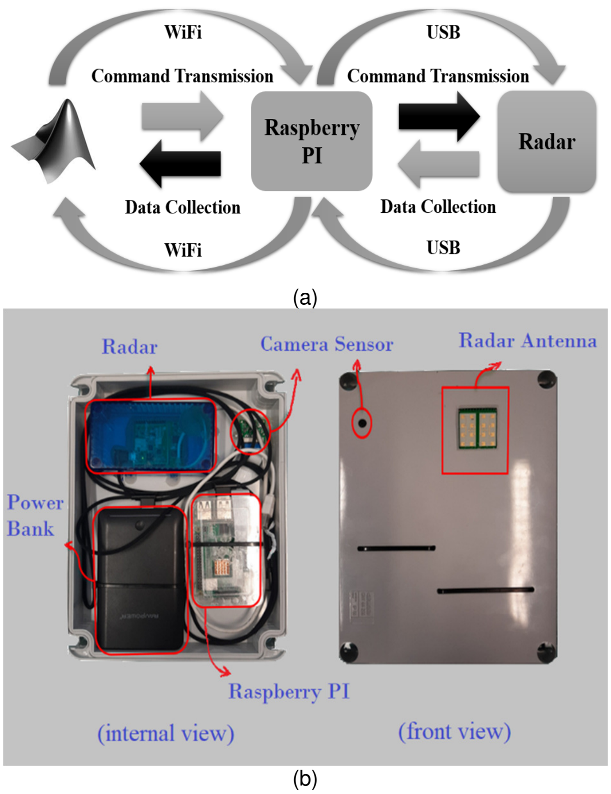


Fig. 1. (a) Block diagram and (b) picture of the developed radar system.

The radar device is based on the Distance2Go demo board developed by Infineon [32], which is composed of two parts: the radar main board and the debugger board. The radar main board contains a BGT24MTR11 radar transceiver, equipped with a pair of arrays of microstrip patch antennas (a transmitting and a receiving one) characterized by a 12 dBi gain and $20° \times 40°$ beamwidths (defining in this way the Field of View, FOV). The board also includes a frequency control part for managing the timing and the number of transmitted chirps. In the receiving stage there is also an analogue amplification part dedicated to signal filtering and amplification, as well as a quadrature down-conversion mixer. Finally, it contains an XMC4200 micro-controller that controls the whole operation of the adopted radar board. In particular, it manages the data communication between board and PC and can be used for performing simple online signal processing tasks (provided that a proper firmware is adopted). In the developed system, the default firmware that enables the board to only capture the radar raw data without any other processing capabilities has been used. All the processing chain, including the FMCW radar processing, have been implemented on the control PC in a MATLAB environment.

The communication between the PC and the radar board is performed via a serial communication through the USB port. The debugger board can be used for programming the XMC micro-controller. Table I reports the parameters adopted in the radar board. The range (ΔR) and velocity (ΔV) resolutions indicate the ability of the radar to distinguish between two closely separated targets in the range and velocity domains, respectively.

The second block of the system is a Raspberry PI 3B+ embedded PC, which was used as a link between the PC and the radar board. The Raspberry PI collects the data from the radar board and transmits it to the main PC via a Wi-Fi network. The communication between the radar and the PI was done through the USB ports of the PI.

The final part of the prototype consists of a PC running MATLAB, which controls the radar board via the Raspberry PI by means of the Raspberry PI toolbox. This toolbox helps creating a virtual port for accessing the USB ports of the Raspberry PI. The PC collects the measured radar signals and performs the data processing and classification.

B. FMCW Radar Module

In the developed system, the board operates as a FMCW radar with a saw-tooth like chirp signal and a linear frequency modulation. Specifically, a multi-chirp sequence FMCW approach is adopted, in which a frame composed by $N_c = 21$ subsequent chirps is transmitted and processed to extract the range and Doppler information of the targets.

As it is well known, the received signal is a copy of the transmitted one delayed by a time $\tau = 2R/c$, where c is the speed of light and R is the target range (a single localized target is assumed here for sake of simplicity). In the FMCW radar receiver module, the transmitted and received signal passes through an I/Q demodulator at the output of the receiver. The signal provided by the I/Q mixer (after a low-pass filter that removes the high frequency components) is called the Intermediate Frequency (IF) signal (or beat signal) and, neglecting higher order terms and Range-Doppler coupling, can be expressed as [33]:

$$IF(t_c) = A_b e^{j2\pi\left[\frac{B}{T_c}\tau t_c - \frac{2v_r}{\lambda_0} n_c T_{crp}\right]} \quad (1)$$

where $t_c \in [0, T_c]$ is the relative time inside a single up chirp period, n_c is the chirp index inside the sequence of N_c transmitted chirps, A_b is a complex coefficient, B is the sweep bandwidth, T_c is the duration of the up-chirp, T_{crp} is the chirp repetition period , and v_r is the radial velocity of the target. In particular, T_{crp} is the sum of the chirp time T_c, the ramp down chirp time T_{rd}, and the PLL steady state time T_{pll_s}, where $T_{rd} = 200\ \mu s$, and $T_{pll\_s} = 400\ \mu s$. The chirp time T_c, the chirp slope a, and the sampling frequency f_s are given in Table I.

The IF signal is sampled by considering, for each chirp, $N_s = 64$ samples. Consequently, a data matrix of dimensions $N_c \times N_s$ is formed from the output of the I/Q receiver. The board provides such a data matrix as a data frame, which can be processed and analyzed to obtain the target's information. In particular, a 2D FFT processing [15] is applied on the collected data frames to simultaneously obtain the range and speed of the target. Such an operation is performed on the baseband signals available at the output of the I/Q receiver. The first FFT is applied along each chirp to provide the range bins and is called range-FFT. Specifically, the range information can be obtained searching for the peaks corresponding to the frequency f_b of the beat signal. Indeed, when using a small chirp time T_c, the Doppler component is usually very small and is negligible with respect to the size of the range bins. Consequently, the beat frequency (i.e., the derivative of the phase of the signal IF) can be approximated as $f_b = B\tau/T_c$ and therefore the target range can be deduced with the following formula:

$$R = \frac{cT_c f_b}{2B} \quad (2)$$

TABLE I
RADAR SENSOR PARAMETERS

Symbol	Description	Value
B	Sweep Bandwidth	200 MHz
f_0	Center Frequency	24.025 GHz
f_s	Sampling Frequency	42667 Hz
R_{max}	Maximum range	25 m
V_{max}	Maximum Velocity	5.4 Km/hr
ΔR	Range resolution	0.75 m
ΔV	Velocity resolution	0.4 Km/hr
N_s	Number of samples/chirp	64
N_c	Number of chirps/frame	21
T_c	Up-chirp time	1.5 ms
a	Chirp Slope	133 MHz/s
T_{crp}	Chirp repetition time	2100 µs

The second FFT is applied along the chirps for a single range bin to provide the Doppler information, and it is called Doppler-FFT. The matrix obtained at the end of the 2D FFT is called Range-Doppler map and holds the signature derived from the echo signal of each target. The approach for the 2D FFT processing is illustrated in Fig. 2.

It is worth noting that the 2D FFT processing is a well-known technique in the FMCW community and consequently several commercial radars provide an integrated support to the 2D FFT processing, which may be transparent to the user. However, in order to have a full control of the processing chain, in the present implementation of the system the whole processing chain is fully performed on the control PC starting from the radar raw ADC data frame provided by the radar board.

The proposed setup permits the use of a low-cost radar

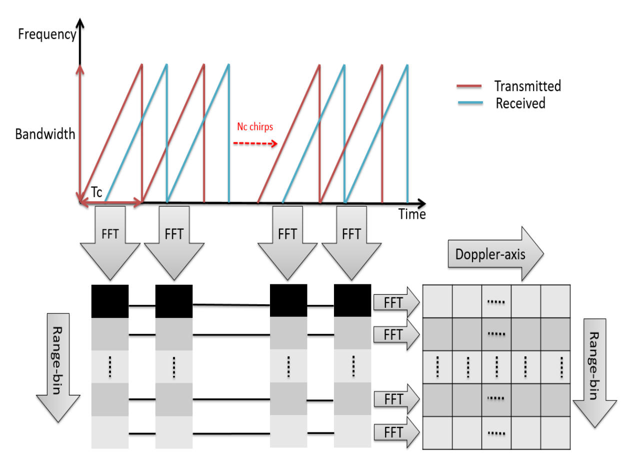


Fig. 2. Basic concept of the 2D FFT processing in multi-chirp sequence FMCW.

hardware in a quite complex and advanced radar application, i.e., the detection and classification of multiple targets passing through a gate. It also offers the option to perform all the processing scheme in real-time, either on the Raspberry Pi embedded PC or using a portable PC. The possibility of performing some simple online processing tasks, like FMCW processing, directly on the radar board adds additional flexibility to the system. This all offered in a cost-efficient, portable, and power-friendly hardware.

III. Data Processing Algorithms

Fig. 3 shows a flowchart describing the main blocks of the developed data processing algorithm. The developed approach has been specifically designed for extracting, from the measurements provided by a low-cost radar, a set of multi-target features to be used to classify different objects passing through the monitored gate. To this end, after acquisition of the raw data, a background subtraction algorithm is used to calibrate the measurements. Then, the FMCW processing based on the 2D FFT analysis [15] starts whenever a signal with an amplitude higher than a proper range detection threshold is detected. The Range-Doppler matrix obtained from the 2D FFT Processing then undergoes clustering with the DBSCAN algorithm. After that, an ad-hoc designed peak to cluster assignment step is used to confirm the detections coming from real targets in the range spectrum, and later on to separate the contributions of each target in the spectrum in order to facilitate the subsequent extraction of the range-spectrum-based features. This step also decreases the false alarm rate as other detections coming from noise or clutters are discarded. The indices of the confirmed detections are then fed to a specifically-designed multi-target tracking algorithm, based on the α-β filter, that assigns a unique target-ID to each target. The tracking filter provides the feature extraction block with the target IDs and indices, whereby each of the features is assigned to its respective target. The measured features are then fed to a multi-class SVM for classification. At the end, the detected target will be classified as either a truck, motorcycle, car, or a pedestrian.

Details on each processing block are described below.

A. Background Removal with Frame Subtraction

Background Subtraction (BS) technique is a popular image processing scheme used for moving target recognition. In the developed system, each radar data frame is represented as a $N_s \times N_c$ data matrix and can be then considered as a digital image. BS considers the foreground detection as the difference between a newly recorded image and an old image used as a reference (often called the “Background model” or “reference model”) [34]. In particular, standard methods, which are often also provided by commercial radars, rely upon the use of a fixed reference background. For the application at hand, however, this results in a not optimal removal of the contribution of the background, leading to the appearance of clutters whenever a moving target appear in the FOV of the radar. Consequently, a modified BS algorithm has been developed for calibration and moving object detection.

The main difference with respect to standard BS is that the reference frame is not fixed. Indeed, since the electromagnetic scattering response of the scenario may vary during time (e.g.,

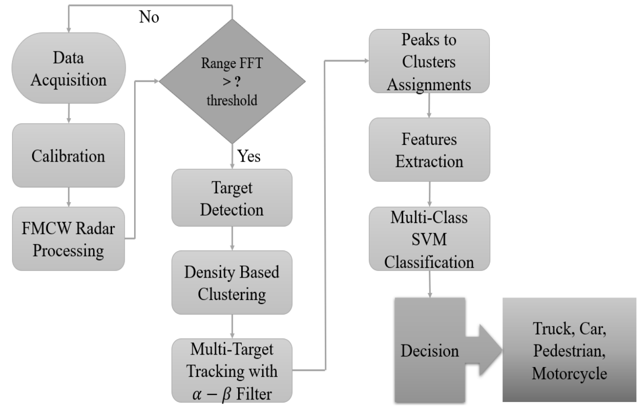


Fig. 3. Flowchart of the developed data processing algorithm.

due to changes in the static objects present in the scene), a varying and adaptive background reference model is used. In particular, the previous frame is used as reference model, i.e., the calibrated frame at time step k is given by:

$$Y(k) = F(k) - F(k-1) \tag{3}$$

where, $F(k)$ and $F(k-1)$ are the frames recorded at time steps k and $k-1$. The data frame considered here is the raw data matrix obtained before the 2D FFT processing, which is directly provided by the radar board and represent the input data to the processing chain. However, it is worth noting that BS could also be performed after the FFT computation, leading to comparable results. Besides, it is also worth highlighting that the subtraction is carried out on the complex-valued data provided by the I/Q receiver. In this way, the target response, which changes from a frame to another, can sum constructively or destructively, thus the resulting peak does not necessarily have a lower amplitude. However, the static response usually has a similar response, thus allowing the removal of the corresponding clutters. Consequently, the adopted frame subtraction technique allows to significantly decrease stationary clutters in the radar image, while keeping the scattering response of moving targets. Indeed, although the target amplitude may be different, the overall information needed for the subsequent classification is retained. Nevertheless, if a target stops when passing through the monitored area, its response would be canceled by BS. However, the tracking algorithm adopted in the processing chain still maintain a lock on the target, allowing to recover its response when it restarts moving and to perform the classification. It is also worth noting that, with respect to other approaches relying on the removal of the zero-Doppler areas in the range-Doppler map, which are often used in radar applications, BS has the advantage of removing additional noise (e.g., due to random changes in the scenario or to other object with non-strictly zero Doppler response that may be present).

Fig. 4 exemplifies the application of the proposed clutter removal technique in the case of a single target located at 11.7 m from the radar and departing with a speed of 2 Km/h. In particular, Fig. 4(a) and Fig. 4(c) show the magnitude of the range FFT (i.e., obtained through a Fourier transform of the chirp with the maximum FFT amplitude of the measured frame) before and after BS. It can be clearly seen that before applying

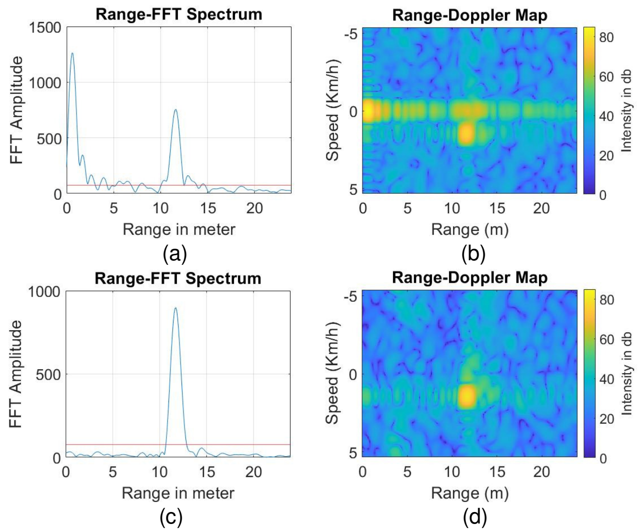


Fig. 4. Example of results of the range FFT and Range-Doppler map (a)-(b) before and (c)-(d) after calibration with the frame subtraction algorithm.

the frame subtraction technique, the obtained range-FFT spectrum has, in addition to the main target peak at 11.7 m, several small other peaks related to clutters and one big peak close to zero caused by the leakage between the transmitting and receiving antennas. In the corresponding Range-Doppler maps in Fig. 4(b) and Fig. 4(d) several artifacts are present, and a very high peak clutter in the zero-speed axis is also present. The target is represented by a weaker peak at 11.7 m and 2 Km/h. On the other hand, the range-FFT spectrum, after frame subtraction was applied, shows only one strong peak related to the target at 11.7 m. Moreover, a single sharp peak related only to the moving target, with no contribution from the clutters appearing on the zero-speed axis, is present in the Range-Doppler map.

B. Clustering with DBSCAN Algorithm

Considering the resolution of the adopted radar board along range and Doppler axes (given in Table I), as well as the dimensions of the targets and the dynamics of their movements, the targets will appear as extended clouds of points on the Range-Doppler map. In addition to that, targets with extended range profiles like cars will produce multiple peaks at different indices in the range-FFT spectrum.

Consequently, an appropriate clustering algorithm is needed, in order to assign the indices of the points on the cluster maps to their respective peaks, which are related to each target on the spectrum. To this end, the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [35] is adopted. It is worth emphasizing that only the points at which the intensity of the map is greater than or equal to a proper threshold value th_{DBSCAN} selected by the user are considered in the DBSCAN method to remove spurious components in the Range-Doppler map that may be not fully removed by the clutter and leakage removal method. Moreover, the DBSCAN algorithm requires the definition of two more parameters: the minimum number M of neighbor required to identify a core point and the neighborhood search radius ε.

There are two particular characteristics that made DBSCAN the best choice for the envisioned application: First, there is no need for any prior knowledge about the number of clusters as DBSCAN can automatically determine the number of obtained clusters. Second, DBSCAN clustering has the potential to recognize any spurious aggregation of points that are isolated and consequently should not be included in the clusters; they are assigned a class value of -1.

DBSCAN provides the true range points values along with their indices for each target, which is very helpful especially in the multi-target situation. These values are very crucial for the next steps related to multi-tracking and features extraction.

C. Multi-Target Tracking with α-β Filter

In order to perform multi-target classification, it is necessary to define a proper set of features for each target at every time frame. In particular, such features are extracted from the range and Doppler spectra and are related to the range and Doppler profiles, range and Doppler spreads, and the reflectivity of the targets, as detailed in Section III.E. Then, the features of each target across several frames are used to form the final feature vector that is used for classification. Consequently, there is the need of performing a multi-target tracking to correctly follow the targets throughout the various frames.

An ad-hoc tracking algorithm is proposed in this paper, which is based on the solution of a one-dimensional tracking problem for each different target present in the FOV of the radar, using only the radar range measurements as inputs. At first, it creates separate track files for each of the radar detections, which are initialized with the first radar range measurements and zero speed. Subsequently, detection and prediction are performed, followed by an association phase, in order to use the radar measurements to predict the position of the targets on the next frame. Such predictions are compared with the new radar measurements to determine the track files the new measurements should be assigned to. After that, it uses a credit system to determine and control the lifetime of the created track files of each target. It also provides an estimate of the target's speed, which is not directly possible with the radar hardware since its ability to unambiguously detect speed is limited to 5.4 Km/h.

In particular, the α-β filter [36] is used to predict the range and speed of each target starting from the measured range provided by the radar and from the values estimated at the previous frame, assuming a Newton's motion model with constant velocity. Fig. 5 illustrates the considered α-β filter structure. In the figure, $y(k)$ is the current range measurement, while α and β are filter parameters, and T_f is the frame sampling time. Specifically, the filter is composed by two stages, namely update and prediction, that are described by the following equations:

1. Update step:

$$R(k|k) = R(k|k-1) + \alpha[y(k) - R(k|k-1)] \tag{4}$$

$$v(k|k) = v(k|k-1) + \frac{\beta}{T_f}[y(k) - R(k|k-1)] \tag{5}$$

where $R(k|k)$ and $v(k|k)$ are the corrections, based on the current distance measurement, of the estimated distance and speed, respectively.

2. Prediction step:

$$R(k+1|k) = R(k|k) + T_f v(k+1|k) \quad (6)$$

$$v(k+1|k) = v(k|k) \quad (7)$$

where $R(k+1|k)$ and $v(k+1|k)$ are the predictions for the next frame, based on the current state of the filter $R(k|k)$, and $v(k|k)$, respectively.

The overall algorithm for multi-target tracking is summarized as follows:

1. Initialization: in the first frame, a track is initialized for each target identified by DBSCAN, whose initial state is given by the distance measured by the radar and zero speed. Furthermore, for each of these sets a common credit value of value 6 is assigned (illustrated below).
2. Detection: when a new frame is acquired, it is analyzed to detect the targets that are present in the scene.
3. Association: the new detections are associated with existing tracks or are used to initialize new tracks. In particular, the association between an existing track and a new survey takes place by solving an assignment problem using the Hungarian algorithm [37], where the cost function is given by the difference between the distance corresponding to the survey itself and the distance predicted by the track in step 4. The detections not associated with any existing tracks are used to initialize new tracks, whose initial status is given by the measured distance and zero speed. A common credit value is also set for the new tracks. The cost is given as following:

$$Cost_k^i = R_{k-1}^i - y_k^i \quad (8)$$

where, R_{k-1}^i is the i^{th} range prediction of the i^{th} target at the previous time frame, and y_k^i is the i^{th} range detection measured of the i^{th} target on the current time frame. The cost is calculated for each new track, and tracks associations are based on lowering the value of the corresponding cost matrix. In other words, each new detection is associated with the track where the value of the cost in minimal.
4. Correction: the traces associated with detections are updated through the correction operation indicated in (4)-(5).
5. Credit system: The credit system is used to check the activity of a certain target track file. The active tracks get new detections at every new frame, and so they get new predictions and associations. This permits the targets to have a lifetime measure based on their history. Every new target starts with an initial given credit that is constant for all targets. Whenever a target track misses an association with the new detections it loses a credit. On the other hand, whenever a target track gets an association, it gains a credit. When the target track runs out of credits it is then considered a dead and inactive track and so it is deleted.
6. Prediction: the operations in (6)-(7) are applied to predict the status of all the active tracks (associated or not) in the next frame. When no detection for the target track at the current time frame is recorded, only a prediction is performed using (6) and (7). In that case, the predicted velocity stays constant, and the predicted range is based on that predicted velocity. If no detections are associated with a target track for some number of frames (e.g., equals to the initially assigned credit value), it results in a linear prediction or 'coasting' target. Coasting is allowed for a short period of time after which the target track is deleted.
7. If the collection of frames is finished, the algorithm proceeds to the next processing block, otherwise start again from step 2 with $k \rightarrow k+1$.

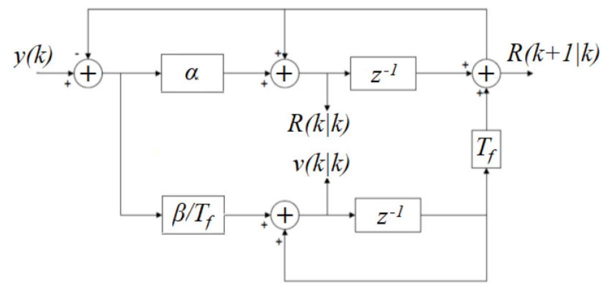


Fig. 5. α-β tracking filter implementation.

The tracking information provided by the algorithm are then used for extracting the features for each target alone, which are subsequently used for the classification of the different targets, as discussed in the following sections.

It is worth remarking that the parameters α and β are constant values selected by the user. Such parameters, whose values must reside in the range (0,1) for stability reasons, significantly affect the dynamic characteristics of the filter. Some methods for defining their optimal values are available for specific problems, e.g., [36]. However, it is also possible to heuristically find suitable values by performing some empirical analysis of their meaning. Indeed, the choice of α and β should be based on the accuracy of the measurements returned by the radar module of the distance and velocity of the targets occupying the illuminated scene: if this accuracy is high, then high values can be adopted for the two parameters, while otherwise low values allow to obtain a greater smoothing effect on the state estimate. Choosing high values for α means that more priority is put on range measurements rather than previous range predictions, whereas high values for β result in higher priority on the velocity measurements other than the previously predicted velocity.

D. Feature Extraction

To achieve a real time classification system, it is needed to look for very simple features that can be extracted in an easy fashion without any complex computation that can take time and resources. In that regard, the feature vector is built using information that can be directly extracted from the range and

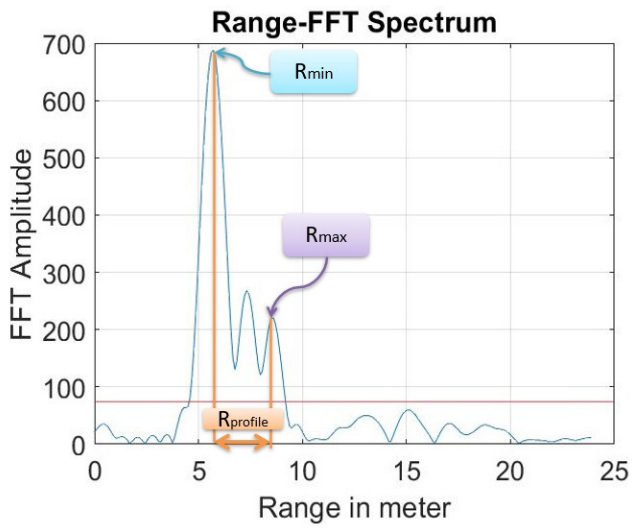


Fig. 6. Example of computation of the range spread in the presence of a car. The horizontal red line indicates the threshold beyond which the peaks of the spectrum are considered.

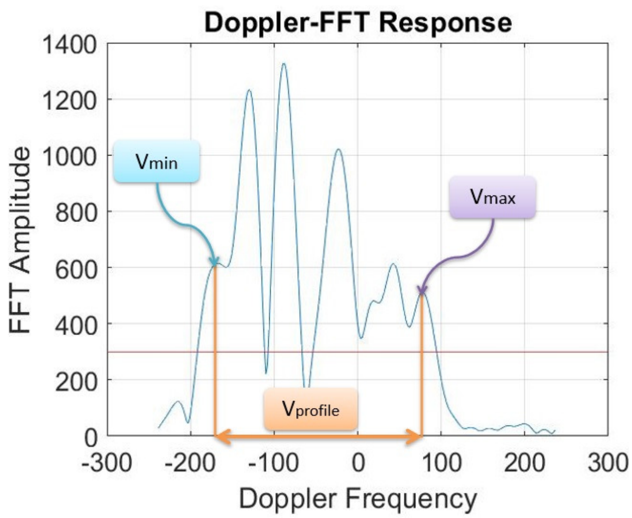


Fig. 7. Example of computation of the velocity spread in the presence of a pedestrian. The red line indicates the threshold beyond which the peaks of the spectrum are considered.

Doppler profiles of the target. In particular, the range profile is the measure of the spread caused by the target in the range spectrum. This spread mainly occurs due to a change in radial velocity, or whenever the length of the target along the range axis of the radar is higher than the radar range resolution. Similarly, the velocity profile is the extension that the target presents along the Doppler spectrum. This extension is caused by the radial velocities of the target's reflection points.

In the developed system, the method presented in [18] is adopted, which allows a very simple and direct approach to calculate the range and Doppler profiles of the targets. Specifically, the range profile is computed as:

$$R_{profile} = R_{max} - R_{min} \tag{9}$$

where R_{max} and R_{min} are respectively the maximum and minimum distances of the peaks (corresponding to reflection points of the target) in the spectrum of the range-FFT. It is worth remarking that only the peaks that are over a predefined threshold th_{range} are used in the computation of $R_{profile}$. Fig. 6 exemplifies this calculation when a single car is present. Note that $R_{profile}$ is 0 in the presence of a spectrum containing a single peak, as for example usually happens for pedestrians.

Since the maximum velocity measured by the radar is limited to $V_{max} = 5.4$ Km/h (as indicated in Table I), an aliasing phenomenon is present in the Doppler spectrum, especially in case of vehicles; thus, the calculation of the velocity spread is not carried out directly on the output of the Doppler-FFT. First, it is verified in which Doppler half-axis (positive or negative) the greatest spectral content is present, after which only the 3/4 of spectrum that contains the selected half-axis are considered. This process can be formulated as follows. Consider S_D the original Doppler spectrum obtained after the 2D FFT analysis, N_{pos} and N_{neg} are the numbers of positive and negative Doppler frequency samples contained in S_D respectively, and N_{DFFT} is the number of Doppler FFT points. The new Doppler spectrum S'_D after the deletion is:

$$S'_D = \begin{cases} S_D U(n + \frac{3 \times N_{DFFT}}{4}), & N_{pos} > N_{neg} \\ S_D U(\frac{3 \times N_{DFFT}}{4} - n), & N_{pos} < N_{neg} \end{cases} \tag{10}$$

where n is the discrete index on the Doppler frequency axis and $U(n)$ is the unit step function.

Finally, the Doppler profile is computed by applying the following formula:

$$V_{profile} = V_{max} - V_{min} \tag{11}$$

where V_{max} and V_{min} are respectively the maximum and minimum radial velocities of the peaks, corresponding to the reflection points of the target, identified in the chosen spectrum portion. Fig. 7 exemplifies this calculation when a single pedestrian is present in the scenario.

Using (9) and (11) to measure the range and Doppler profiles works fine for a single target situation. However, applying (9) directly in a multi-target scenario leads to wrong range profile estimations. This is because in the multi-target case it is always expected to have multiple peaks in the range spectrum that are coming from different targets. Hence, we should determine first the indices of the peaks corresponding to each target in the spectrum before applying (9).

In the developed procedure, this is achieved by using the results of the DBSCAN clustering algorithm [35]. However, before the calculation of the range profile it is necessary to perform a "peaks to clusters assignments". This step assigns the reflections of the targets on the range spectrum to their respective clusters on the Range-Doppler map.

This step can be formulated as follows. Let C_i be the i^{th} cluster obtained by DBSCAN. A target that presents multiple reflection points generates multiple peaks in the spectrum located at different ranges along the range axis. All the peaks p_j^i with ranges R_j^i belonging to the i^{th} cluster C_i are associated to a set $P_i = \{p_1^i, p_2^i, \dots, p_j^i\}$. The peaks that are not associated with any cluster are neglected, which highly decreases the probability of false alarm. The tracking algorithm benefits hugely from the “peaks to clusters assignments” step by introducing only one tracking file for targets having multiple peaks. The tracking of such targets is based on tracking the range of the maximum peak in the spectrum.

Fig. 8 exemplifies the results of using DBSCAN to cluster targets (pedestrian and car) on the Range-Doppler map, and it also shows the scattering from each target on the range spectrum. Comparing the images, it can be seen that the green cluster can be associated to the three peaks located in the same positions of its points, while the red cluster can be associated with the first peak.

Unlike the range profile, the calculation of the velocity profile in the multi-target case follows the same procedure as in the single target case. This is because the Doppler spectrum is obtained at each range bin. In other words, each target will generate its own Doppler spectrum which is usually distinct from the others. In fact, it may occur in some cases that many targets move in the same range bin. However, this will not be problematic as it is not the common case, and it doesn’t usually last for more than one frame.

In order to create a more robust feature vector, the previous quantities, which are just related to the extent of the target response in the range and Doppler spectra, have been integrated with other global information. In particular, the mean and variance of the parts of the spectrum relevant to each tracked target have been included in the feature vector. Indeed, both range and Doppler spectra exhibit some random variations, due to target movements, residual clutter components that survived

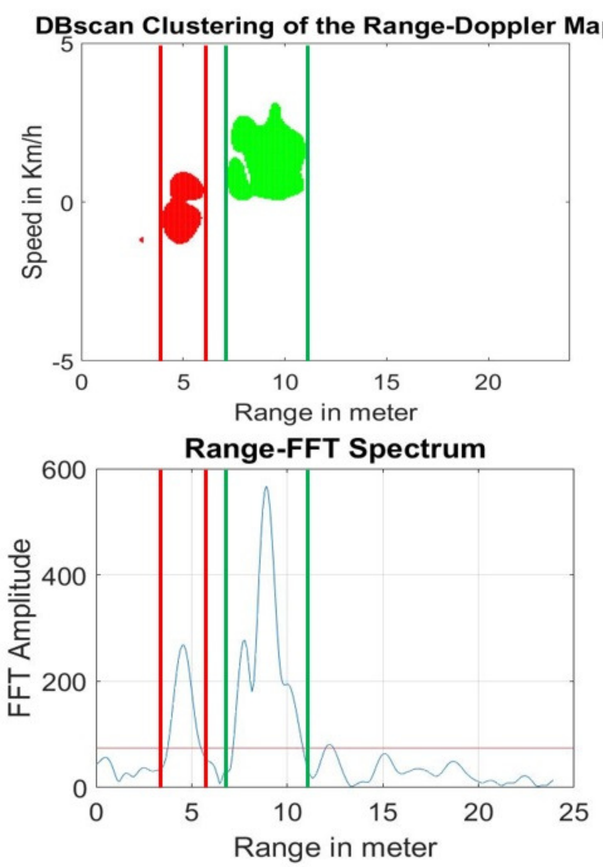


Fig. 8. Car (green) and pedestrian (red) clustering with DBSCAN along with their respective contributions in the range-FFT spectrum.

the removal method, and various sources of noise. For this reason, such contributions can be considered as random variables. Based on this consideration, the following preparatory quantities for the definition of the feature vector are also introduced (only the spectral components of amplitude greater than a threshold th_{range} are considered) [18], [38]:

- R_1, variance of the target spectrum contained in the range-FFT;
- R_2, standard deviation of the target spectrum contained in the range-FFT;
- R_3, average of the target spectrum contained in the range-FFT;
- V_1, variance of the target spectrum contained in the Doppler-FFT;
- V_2, standard deviation of the target spectrum contained in the Doppler-FFT;
- V_3, average of the target spectrum contained in the Doppler-FFT.

It is worth remarking that the standard deviations and variances used in the feature vector are closely related quantities. However, as often done in machine learning classification [18], [19], a feature vector containing redundant information has been adopted, since it may strengthen the classification.

The measurements of V_1, V_2, and V_3 are simple, since at each bin, a single target is present with a single Doppler spectrum, thus targets at different range bins will have their own separate Doppler spectrums. However, computing R_1, R_2, and R_3 in the multi-target case is more complicated because of the mixed contributions of multiple targets in the range spectrum. Consequently, to calculate the values of R_1, R_2, and R_3, an approximated range spectrum of each target is obtained depending on the range bins indices obtained from the

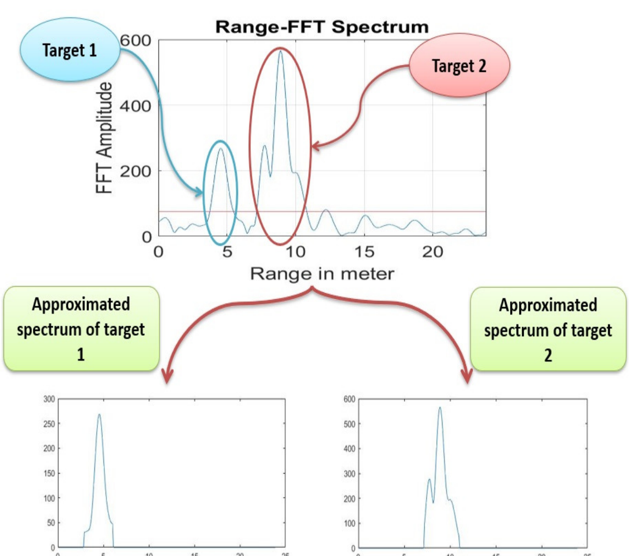


Fig. 9. Approximation of the range-FFT spectrum of each target separately from the contribution of the other targets in the range spectrum.

DBSCAN clustering. Only the part that is above the range threshold was approximated for each target while the other parts are set to zero. This process is illustrated in Fig. 9.

To clarify things up, the process can be formulated as follows: Let j be the index of the points on the range axis, i be the cluster index, and S_{RFFT} is the range-FFT spectrum. Now consider $S_R^{(i)}$ as the i^{th} approximated range spectrum of the i^{th} detection based on the i^{th} cluster C_i, i.e.,

$$S_R^{(i)} = \begin{cases} S_{RFFT}, & \forall R_j \in C_i \\ 0, & otherwise \end{cases} \tag{12}$$

Finally, two more features are added to the previous ones, namely the measure of reflectivity and the average target radial speed estimated by the tracking algorithm. The reflectivity measure is proportional to the Radar Cross Section (RCS), which is a measure of the ability of the target to reflect the radar signal in the direction of the radar receiver. The reflectivity γ was measured following the procedure presented in [39]. The radial speed v_{avg} of a target is estimated by averaging the set of radial velocities in the target track file obtained using (5).

The components of the feature vector for classification are given by the means of the quantities listed above. which are calculated on the set of available frames.

The tracking algorithm provides the proposed feature extraction block with the correct indices of the targets along with their target IDs across the different frames. Based on this information, the extracted features at each new time frame are assigned to their true respective target. When a target track file is deleted by the tracking algorithm, the information about that target is no more provided to the feature extraction block, and so no more features are assigned to that target ID. However, the collected feature set of that target is saved to be averaged and classified later on, after which it is also deleted.

E. Classification with OVA SVM:

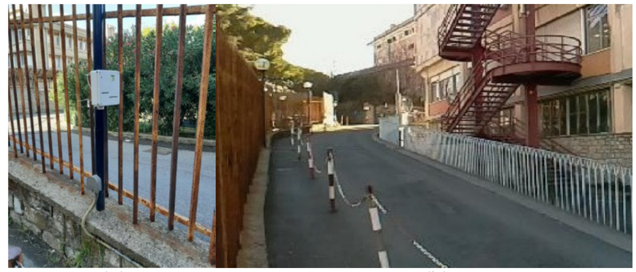
Support Vector Machine (SVM) [20] is a well-known algorithm that behaves well in many classification problems. On this basis, for our problem of the classification of objects passing through the monitored gate, we chose to adopt the One-vs-All Multiclass SVM [40]. This is a generalization of binary SVM capable of handling the multi-class scenario. Binary SVM is based on the construction of a hyperplane within the data space that maximizes the distance from the points of both classes that are closest to the plane itself (maximizing the margin).

As detailed in the previous Section, the feature vector $\mathbf{x}_j$ at the j^{th} time frame upon which the classification was carried on is defined as

$$\mathbf{x}_j = \left[R_{profile}, R_1, R_2, R_3, V_{profile}, V_1, V_2, V_3, \gamma, v_{av}\right]^t \tag{13}$$

The feature vector was normalized using a mean normalization scheme. In particular, the l-th component of normalized feature vector $\mathbf{z}_j$ is obtained as follows:

$$z_j^l = \frac{x_j^l - \bar{x}_j^l}{x_{j,max}^l - x_{j,min}^l} \tag{14}$$



(a) (b)

Fig. 10. (a) Positioning of the external unit in the first test scenario. (b) Picture of the environment of the first test scenario.

where $\bar{x}_j{}^l$, $x_{j,max}^l$, and $x_{j,min}^l$ are the mean, maximum, and minimum value of the l-th component of $\mathbf{x}_j$ in the training set.

As for the OVA SVM method, the Gaussian kernel has been adopted, which is defined as follows:

$$K(\mathbf{x}_a, \mathbf{x}_b) = e^{-\rho\|\mathbf{x}_a - \mathbf{x}_b\|_2^2} \tag{15}$$

where ρ is a parameter to be tuned for the problem at hand.

IV. Experimental Results

A. Training of the System

In order to collect an adequate amount of data for training and testing the system, its external unit was mounted on a pole at a height of 1.5 m, close to an internal road, as shown in Fig. 10. A total of 120 different cases of data were collected for 4 classes of targets: trucks, cars, motorcycles, and pedestrians. Each target has 30 samples that were fed to the processing chain and then to the classifier.

The available dataset is split into two parts. 100 samples are used for the training set (25 observations for each class) while the remaining 20 make up the test set (used after the optimization of the classifier to assess the results). The feature vector was calculated for each of such samples. The classifier was trained considering only the data of the single target scenario. The complete set of the parameters used during the training phase is presented in Table II.

Table II
Set of Parameters Adopted in the Processing Chain

Symbol	Quantity	Value
Th_R	Range detection threshold	75 mV
Th_D	Doppler detection threshold	300 mV
r_{min}	Minimum detection range	1 m
r_{max}	Maximum detection range	25 m
K_R	Shape factor for the Kaiser window on the range dimension	3.2
K	Shape factor for the Kaiser window on the Doppler dimension	4
Th_{DBSCAN}	Threshold for DBSCAN clustering	505 mV
M	Minimum number of neighbor points for DBSCAN clustering	115
ε	Neighborhood search radius for DBSCAN clustering	1.2
C_{NA}	Cost of non-assignment for the Hungarian algorithm	2.8
α	Alpha tracking parameter	0.9
β	Beta tracking parameter	0.3
T_f	Frame sampling time	200 ms

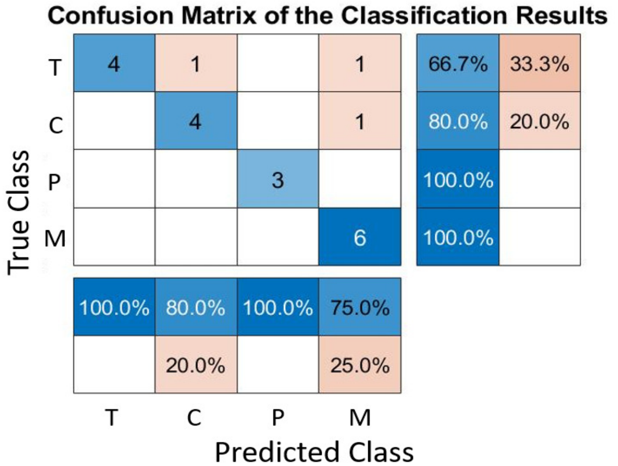


Fig. 11. Confusion matrix obtained by applying the trained OVA-SVM to the single-target test data.

The OVA-SVM has been trained by using the Bayesian global optimization algorithm [41] available in the MATLAB "Statistics and Machine Learning Toolbox". Moreover, a K-fold cross-validation scheme, with $K = 10$, has been adopted to find suitable values of the hyper parameters C and ρ, where C is the SVM regularization parameter [42], [43]. Fig. 11 shows the confusion matrix obtained by applying the SVM obtained after this optimization on the test set, where class T stands for trucks, C stands for cars, P stands for pedestrians, and M for motorcycles. These must give the method good generalization skills, i.e. that it may be able to obtain good classification performances even on elements that are not part of the training set.

The obtained classifier is then used for the multi-target classification, as detailed in the following Section.

B. Multi-Target Validation in the Testing Scenario

After training the OVA-SVM, the performance of the system in classifying multiple targets have been evaluated by considering the same scenario. The experimental validation was performed on 93 multi-target samples, which belong to a different dataset from the one used for training. Fig. 12 shows the classification performance of the proposed method on the whole dataset. The perfect classification score indicates that the proposed method was able to classify all the targets correctly, while the imperfect classification indicates that not all of the targets were classified correctly. The bad classification score indicates that none of the targets were correctly classified.

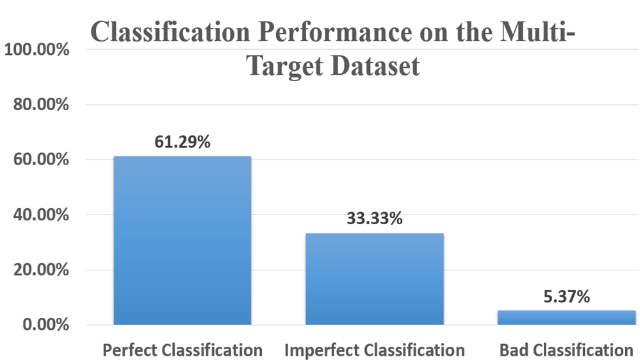


Fig. 12. Performance of the proposed method in classifying targets from the multi-target dataset.

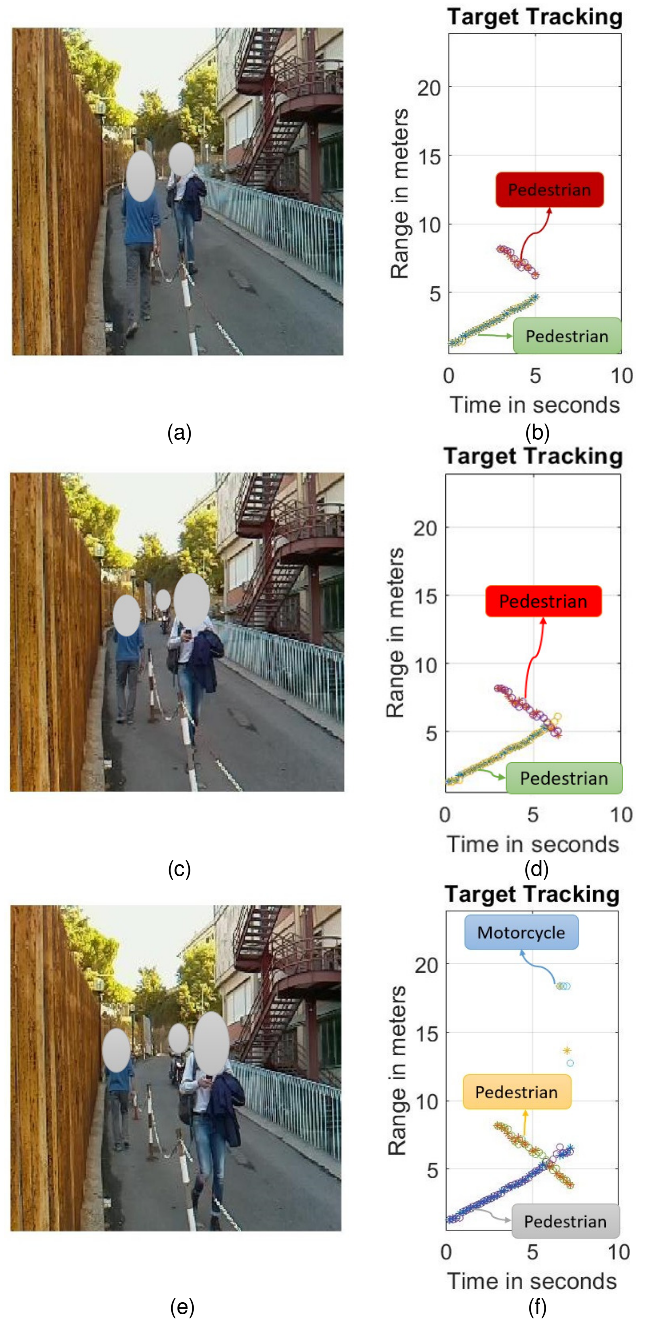


Fig. 13. Camera images and tracking of test case 1. The circle symbol indicates the predicted path by the alpha-beta filter, while the star symbol indicates the measured path by the radar.

Some specific examples are discussed in detail in the following to better analyze the behavior of the whole processing chain. The first test case consists of two pedestrians, one of whom is approaching and one moving away, and subsequently an approaching motorcycle. Fig. 13 illustrates some moments of the recorded event and the related tracking results (the different colors are used to denote different classes of targets in the image).

In Fig. 13(a) and Fig. 13(c) there are the two pedestrians before and after the instant in which they found themselves at the same distance from the radar. From Fig. 13(b) and Fig. 13(d) it can be seen how the method is able to follow the trend at constant

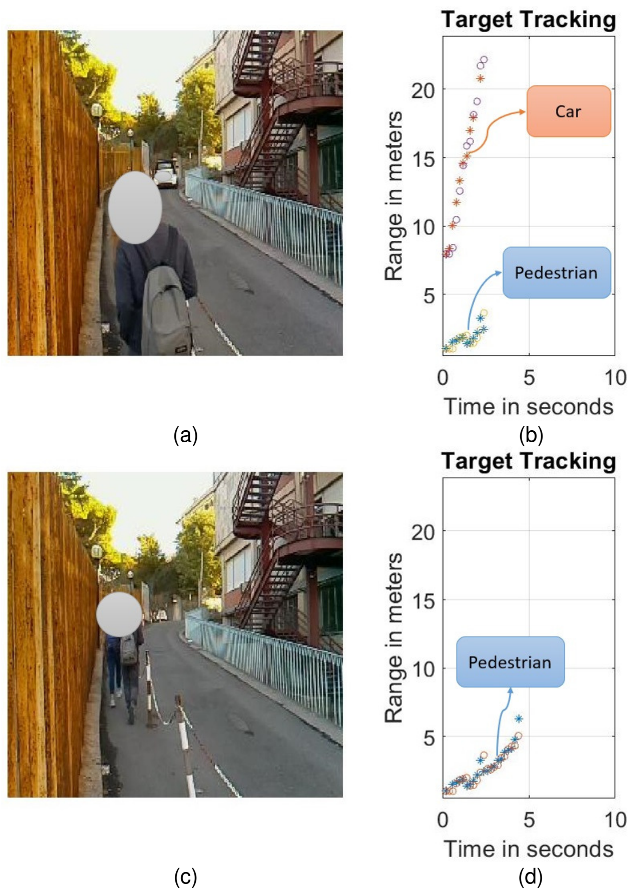


Fig. 14. Camera images and tracking of test case 2. The circle symbol indicates the predicted path by the alpha-beta filter, while the star symbol indicates the measured path by the radar.

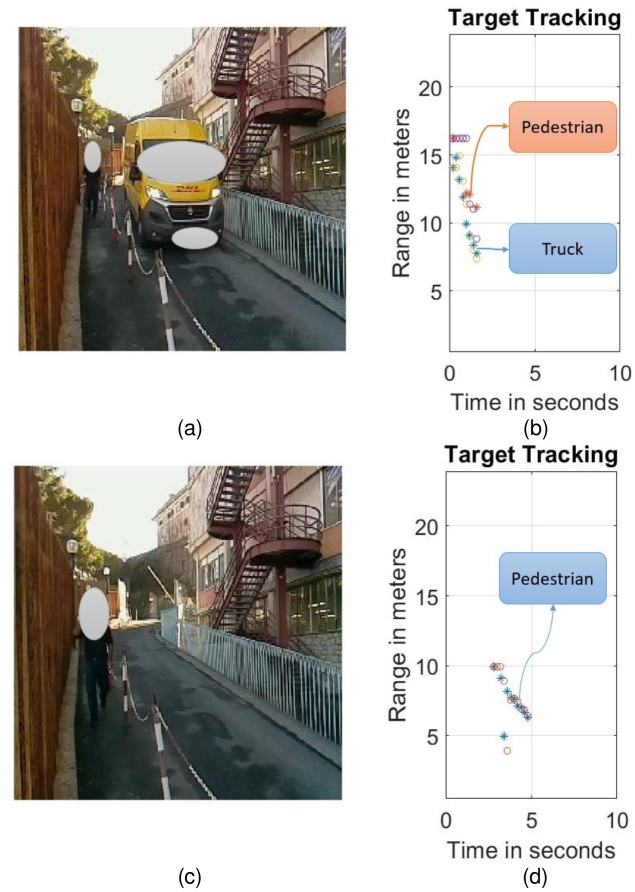


Fig. 15. Camera images and tracking of test case 3. The circle symbol indicates the predicted path by the alpha-beta filter, while the star symbol indicates the measured path by the radar.

speed of their trajectories. In Fig. 13(e) the motorcycle appears. However, due to its high speed, it is present in recording for few frames, as it can be seen from the scarcity of points linked to it in Fig. 13(f). Despite this, all three targets are correctly classified as shown in Table III, which also indicates the aggregation losses calculated for each class and the estimated average radial speeds. Note that recognition occurs with wide margins compared to competing classes. The estimated average radial speeds are compatible with the motion of the targets in the scene.

In the second considered test case, there are a pair of pedestrians moving away from the radar and a car also moving away (Fig. 14(a)). Following the disappearance of the car from the scene, the recording ends with the presence of pedestrians only (Fig. 14(c)). Since the two pedestrians are very close to each other (the separation distance is lower than the range resolution) and move with almost coincident trajectories, they appear on the Range-Doppler map as a single cluster and therefore the tracking is not able to distinguish the two underlying real targets, as we can see both in Fig. 14(b) and in Fig. 14(d). However, the method can follow their overall motion at constant speed. In Fig. 14(b) it is also clear the presence of the car, whose track is characterized by a greater slope than pedestrians because of its higher speed. We can also note in Fig. 14(d) how the method correctly deactivates the track relating to the car when it leaves the radar range. The result of the classification, the aggregation losses, and the estimated average radial speeds are shown in Table III.

TABLE III
RECOGNITION OF THE TYPE OF TARGET MADE ON THE STORED TRACKS, AGGREGATION LOSS AND ESTIMATED AVERAGE RADIAL SPEEDS. P = PEDESTRIAN, M = MOTORCYCLE, T = TRUCK, C = CAR.

Real class	*Predicted class*	*Aggregation loss*				*Average radial velocity* [km/h]
		C	*P*	*T*	*M*	
TEST 1						
P	*P*	*0.55*	*0.06*	*0.68*	*0.44*	*2.5*
P	*P*	*0.57*	*0*	*0.76*	*0.65*	*-3.2*
M	*M*	*0.67*	*0.50*	*0.71*	*0.03*	*-33.7*
TEST 2						
P	*P*	*0.66*	*0*	*0.62*	*0.82*	*2.3*
C	*C*	*0.01*	*0.79*	*0.49*	*0.84*	*26.5*
TEST 3						
P	*P*	*0.63*	*0*	*0.7*	*0.56*	*-5.8*
T	*T*	*0.46*	*1.06*	*0.04*	*0.94*	*-15.3*

In the third test case, a pedestrian and a truck are present. In Fig. 15(a)-(b), the tracking method has been able to follow both targets throughout their stay in the illuminated scene. In addition, the track related to the truck is correctly deactivated when it leaves the radar range and only the pedestrian is tracked until the end of the event recording, as can be seen in Fig. 15(c)-(d). The aggregation losses and the estimated average radial speeds reported in Table III highlight the robustness of the classification and the goodness of the tracking.

(a) (b)

Fig. 16. (a) Positioning of the external unit in the second test scenario. (b) Picture of the environment of the second test scenario.

C. System Validation in a Different Scenario

To confirm the robustness of the proposed method, another experiment was done in a totally different environment and under different conditions. The radar was mounted on a pole near a side public road at 3 m height and at about 5 m distance from the street, tilted with a 45-degree angle toward the street (Fig. 16). A total of 37 samples were collected for this test. The training of the OVA-SVM was not performed, since the old training parameters found for the first scenario were used. However, the different environmental conditions required to retune some of the parameters of the algorithms used in the developed method. In particular, the higher distance of the target from the radar compared to the first scenario required to decrease the range and DBSCAN detection thresholds for obtaining an improved detection of the target contribution in the range spectrum and the resulting DBSCAN clusters. On the other hand, the higher lateral movement of the targets with respect to the radar compared to the longitudinal one in the first scenario imposed higher variations in the Doppler spectrum. This is mainly due to the greater variation in the radial velocity of the targets, and thus a higher Doppler threshold should be adopted to avoid detections caused by noise. Such considerations, together with preliminary empirical tests on some targets, have been used to retune the aforementioned parameters. The values of the modified parameters are shown in Table IV. As in the previous test, Fig. 17 shows the performance of the proposed method on the whole dataset. Perfect, imperfect, and bad classification have the same meaning as presented before.

Also in this second scenario, the behavior of the method has been analyzed in detail for some significant cases. The first test

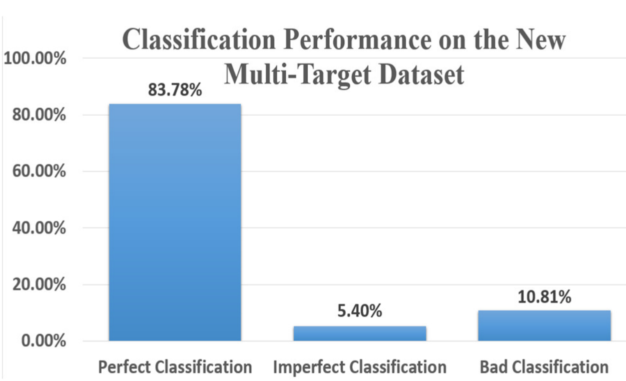


Fig. 17. Performance of the proposed method in classifying targets from the second multi-target dataset.

TABLE IV
MODIFIED PARAMETERS ADOPTED IN THE PROCESSING CHAIN (EXPERIMENT 2)

Symbol	Quantity	Value
Th_R	Range detection threshold	50 mV
Th_D	Doppler detection threshold	500 mV
Th_{DBSCAN}	Threshold for DBSCAN clustering	260 mV

case (Test 1) concerns the presence of two targets, of which one is a pedestrian departing and the other is an approaching car (Fig. 18 (a)-(c)). From Fig. 16 it can be clearly seen the narrow angle of the radar in this new setup. However, the presence of the targets can be still clearly seen. The method was able to track the two targets perfectly and follow the trend even after the occlusion of the two targets (Fig. 18(b)-(d)). The aggregation losses and the classification results are shown in Table V. The reason behind the lower difference margin of the aggregation loss compared with the previous tests is because the model used here is the previously trained model, built on a different scenario. However, the method still shows sufficient robustness to perform the classification.

Finally, the last test case (Test 2) shown in Fig. 19 concerns a motorcycle and a car approaching the radar (Fig. 19(a)). The two targets are successfully detected and tracked, as it can be seen from Fig. 19(b). The motorcycle track was deleted after it

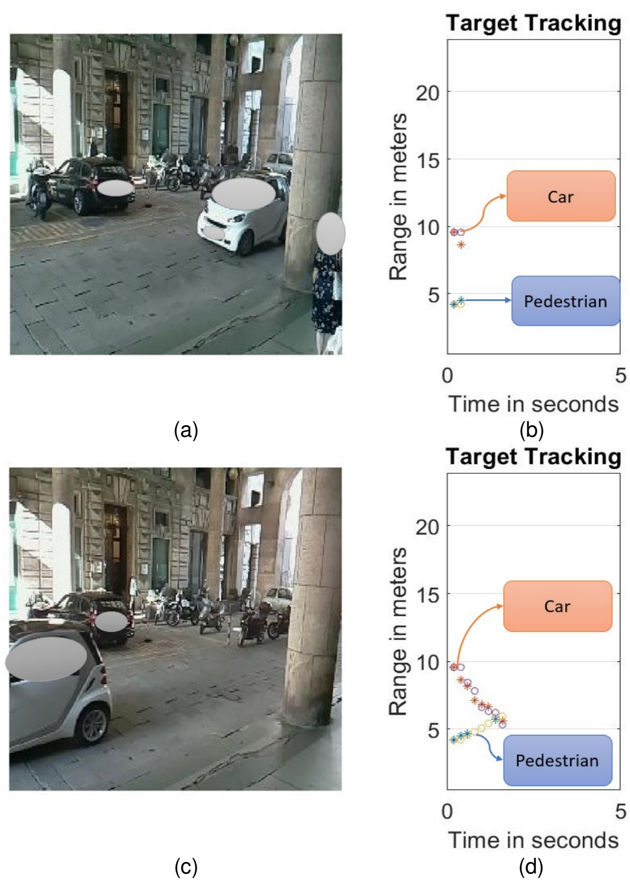


Fig. 18. Camera images and tracking obtained from the second scenario (first test case). The circle symbol indicates the predicted path by the alpha-beta filter, while the star symbol indicates the measured path by the radar.

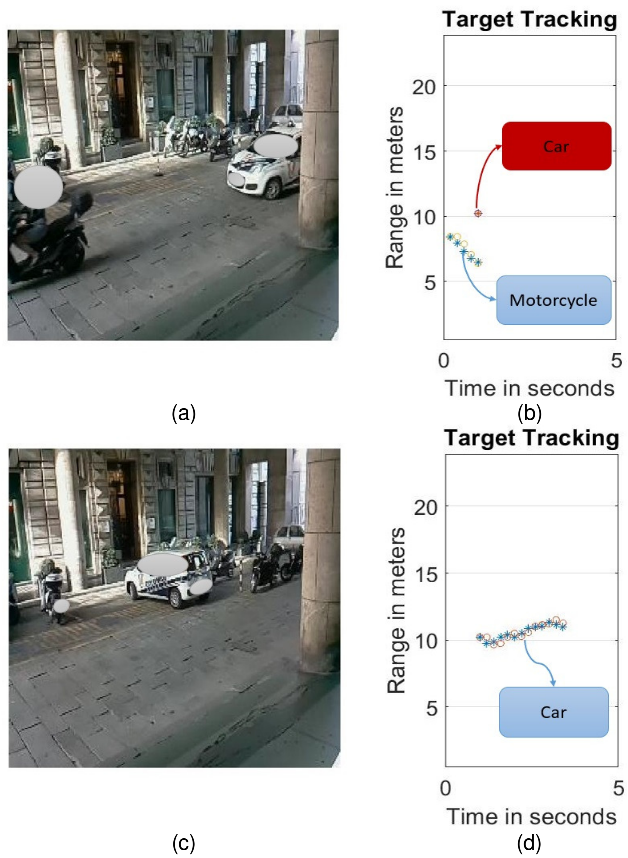


Fig. 19. Camera images and tracking obtained from the second scenario (third test case). The circle symbol indicates the predicted path by the alpha-beta filter, while the star symbol indicates the measured path by the radar.

left the scene for several frames, while the car was still tracked until it parked on the side of the road (Fig. 19(c)-(d)). The two targets were correctly classified by the system and the robustness of the classification is shown in Table V. It is however worth noting that also in this second case the classifier margins are lower than in the first scenario. In particular, the motorcycle is quite close to be classified as pedestrian. This behavior can be related to diversity of the scenario used to train the classifier. Indeed, since the positioning of the radar has been changed, the range and Doppler features may be different. Neverthless, the approach seems to be quite robust with respect to such changes.

TABLE V
RECOGNITION OF THE TYPE OF TARGET MADE ON THE STORED TRACKS, AGGREGATION LOSS AND ESTIMATED AVERAGE RADIAL SPEEDS. P = PEDESTRIAN, M = MOTORCYCLE, T = TRUCK, C = CAR.

Real class	*Predicted class*	*Aggregation loss*				*Average radial velocity [km/h]*
		C	*P*	*T*	*M*	
TEST 1						
P	*P*	*0.7*	*0.14*	*0.52*	*0.36*	*1.94*
C	*C*	*0.18*	*0.37*	*0.34*	*0.4*	*-6.4*
TEST 2						
M	*M*	*0.45*	*0.255*	*0.63*	*0.25*	*-5.16*
C	*C*	*0.1*	*0.71*	*0.4*	*0.82*	*2.1*

V. CONCLUSION

In this paper, a cost-efficient multi-target classification system for smart gate monitoring was proposed. The system consists of a low-cost FMCW radar and Raspberry Pi 3B+. The targets data were collected by the system and processed using a custom MATLAB software. In this study, four classes of targets were considered (trucks, cars, pedestrians, and motorcycles), and the classification is performed using features extracted from the range and Doppler spectra. The system was trained only on the single target data set, then the performance of the system was tested in two different scenarios with different conditions. In the first scenario, the system was tested on a multi-target dataset that was collected in the same environmental conditions as of the training data and the system showed a good accuracy in classifying the multiple targets. In the second scenario, no new training has been done and the already trained system was tested on a new multi-target dataset that was collected by the system in different environmental conditions. Also in this case, the system showed a good performance in classifying the different passing targets.

ACKNOWLEDGMENT

The authors would like to thank Alessandro Delucchi and Roberto Vio (Darts Engineering Srl, Genoa, Italy) for their kind support during the development phase and the experimental activities presented in this article.

REFERENCES

[1] B. Pottier, L. Rasolofondraibe, and S. Kerroumi, “Pedestrian detection strategy in urban area: capacitance probes and pedestrians’ signature,” *IEEE Sens. J.*, vol. 17, no. 17, pp. 5663–5668, Sep. 2017, doi: 10.1109/JSEN.2017.2718734.

[2] X. Zhao, P. Sun, Z. Xu, H. Min, and H. Yu, “Fusion of 3D LIDAR and camera data for object detection in autonomous vehicle applications,” *IEEE Sens. J.*, vol. 20, no. 9, pp. 4901–4913, May 2020, doi: 10.1109/JSEN.2020.2966034.

[3] T. Gandhi and M. M. Trivedi, “Pedestrian protection systems: Issues, survey, and challenges,” *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 413–430, Sep. 2007, doi: 10.1109/TITS.2007.903444.

[4] A. Rasouli and J. K. Tsotsos, “Autonomous vehicles that interact with pedestrians: A survey of theory and practice,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 900–918, Mar. 2020, doi: 10.1109/TITS.2019.2901817.

[5] P. Withington, H. Fluhler, and S. Nag, “Enhancing homeland security with advanced UWB sensors,” *IEEE Microw. Mag.*, vol. 4, no. 3, pp. 51–58, Sep. 2003, doi: 10.1109/MMW.2003.1237477.

[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, doi: 10.1109/ACCESS.2019.2942382.

[7] E. Hyun, Y. S. Jin, and J. H. Lee, “Design and development of automotive blind spot detection radar system based on ROI pre-processing scheme,” *Int. J. Automot. Technol.*, vol. 18, no. 1, pp. 165–177, Feb. 2017, doi: 10.1007/s12239-017-0017-5.

[8] A. Mukhtar, L. Xia, and T. B. Tang, “Vehicle detection techniques for collision avoidance systems: A review,” *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2318–2338, Oct. 2015, doi: 10.1109/TITS.2015.2409109.

[9] T. Lee, S. Jeon, J. Han, V. Skvortsov, K. Nikitin, and M. Ka, “A simplified technique for distance and velocity measurements of multiple moving objects using a linear frequency modulated signal,” *IEEE Sens. J.*, vol. 16, no. 15, pp. 5912–5920, Aug. 2016, doi: 10.1109/JSEN.2016.2563458.

[10] T. Lee, V. Skvortsov, M. Kim, S. Han, and M. Ka, “Application of W-band FMCW radar for road curvature estimation in poor visibility conditions,” *IEEE Sens. J.*, vol. 18, no. 13, pp. 5300–5312, Jul. 2018, doi: 10.1109/JSEN.2018.2837875.

[11] M. Rameez, M. Dahl, and M. I. Pettersson, "Autoregressive model-based signal reconstruction for automotive radar interference mitigation," *IEEE Sens. J.*, vol. 21, no. 5, pp. 6575–6586, Mar. 2021, doi: 10.1109/JSEN.2020.3042061.

[12] T. Yang, H. Shi, J. Guo, and M. Liu, "FMCW ISAR autofocus imaging algorithm for high-speed maneuvering targets based on image contrast-based autofocus and phase retrieval," *IEEE Sens. J.*, vol. 20, no. 3, pp. 1259–1267, Feb. 2020, doi: 10.1109/JSEN.2019.2947559.

[13] M. Ash, M. Ritchie, and K. Chetty, "On the application of digital moving target indication techniques to short-range FMCW radar data," *IEEE Sens. J.*, vol. 18, no. 10, pp. 4167–4175, May 2018, doi: 10.1109/JSEN.2018.2823588.

[14] F. Alimenti *et al.*, "Noncontact measurement of river surface velocity and discharge estimation with a low-cost Doppler radar sensor," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 5195–5207, Jul. 2020, doi: 10.1109/TGRS.2020.2974185.

[15] M. Kronauge and H. Rohling, "New chirp sequence radar waveform," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 4, pp. 2870–2877, Oct. 2014, doi: 10.1109/TAES.2014.120813.

[16] J.-J. Lin, Y.-P. Li, W.-C. Hsu, and T.-S. Lee, "Design of an FMCW radar baseband signal processing system for automotive application," *SpringerPlus*, vol. 5, no. 1, p. 42, Dec. 2016, doi: 10.1186/s40064-015-1583-5.

[17] T. Mitomo, N. Ono, H. Hoshino, Y. Yoshihara, O. Watanabe, and I. Seto, "A 77 GHz 90 nm CMOS transceiver for FMCW radar applications," *IEEE J. Solid-State Circuits*, vol. 45, no. 4, pp. 928–937, Apr. 2010, doi: 10.1109/JSSC.2010.2040234.

[18] S. Heuel and H. Rohling, "Pedestrian classification in automotive radar systems," in *Proc. 13th Int. Radar Symp.*, Warsaw, Poland, May 2012, pp. 39–44, doi: 10.1109/IRS.2012.6233285.

[19] S. Heuel and H. Rohling, "Pedestrian recognition based on 24 GHz radar sensors," in *Ultra-Wideband Radio Technologies for Communications, Localization and Sensor Applications*, R. Thom, Ed. InTech, 2013.

[20] M. Awad and R. Khanna, "Support Vector Machines for Classification," in *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*, M. Awad and R. Khanna, Eds. Berkeley, CA: Apress, 2015, pp. 39–66.

[21] S. Lee, Y.-J. Yoon, J.-E. Lee, and S.-C. Kim, "Human–vehicle classification using feature-based SVM in 77-GHz automotive FMCW radar," *IET Radar Sonar Navig.*, vol. 11, no. 10, pp. 1589–1596, Oct. 2017, doi: 10.1049/iet-rsn.2017.0126.

[22] I. Urazghildiiev *et al.*, "Vehicle classification based on the radar measurement of height profiles," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 245–253, Jun. 2007, doi: 10.1109/TITS.2006.890071.

[23] R. Khanna, D. Oh, and Y. Kim, "Through-wall remote human voice recognition using doppler radar with transfer learning," *IEEE Sens. J.*, vol. 19, no. 12, pp. 4571–4576, Jun. 2019, doi: 10.1109/JSEN.2019.2901271.

[24] A. Bhattacharya and R. Vaughan, "Deep learning radar design for breathing and fall detection," *IEEE Sens. J.*, vol. 20, no. 9, pp. 5072–5085, May 2020, doi: 10.1109/JSEN.2020.2967100.

[25] X. Huang, J. Ding, D. Liang, and L. Wen, "Multi-person recognition using separated micro-Doppler signatures," *IEEE Sens. J.*, vol. 20, no. 12, pp. 6605–6611, Jun. 2020, doi: 10.1109/JSEN.2020.2977170.

[26] A. Angelov, A. Robertson, R. Murray-Smith, and F. Fioranelli, "Practical classification of different moving targets using automotive radar and deep neural networks," *IET Radar Sonar Navig.*, vol. 12, no. 10, pp. 1082–1089, Oct. 2018, doi: 10.1049/iet-rsn.2018.0103.

[27] S. Kim, K. Lee, S. Doo, and B. Shim, "Automotive radar signal classification using bypass recurrent convolutional networks," in *Proc. IEEECIC Int. Conf. Commun. China*, Changchun, China, Aug. 2019, pp. 798–803, doi: 10.1109/ICCChina.2019.8855808.

[28] Y. Kim, I. Alnujaim, S. You, and B. J. Jeong, "Human Detection Based on Time-Varying Signature on Range-Doppler Diagram Using Deep Neural Networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 3, pp. 426–430, Mar. 2021, doi: 10.1109/LGRS.2020.2980320.

[29] V. C. Chen, Fayin Li, Shen-Shyang Ho, and H. Wechsler, "Micro-doppler effect in radar: phenomenon, model, and simulation study," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 1, pp. 2–21, Jan. 2006, doi: 10.1109/TAES.2006.1603402.

[30] S. Abdulatif, Q. Wei, F. Aziz, B. Kleiner, and U. Schneider, "Micro-Doppler based human-robot classification using ensemble and deep learning approaches," in *Proc. 2018 IEEE Radar Conf.*, Oklahoma City, OK, Apr. 2018, pp. 1043–1048, doi: 10.1109/RADAR.2018.8378705.

[31] Jiajin Lei and Chao Lu, "Target classification based on micro-doppler signatures," in *Proc. 2005 IEEE Int. Radar Conf.*, Arlington, VA, USA, 2005, pp. 179–183, doi: 10.1109/RADAR.2005.1435815.

[32] Infineon Technologies, "Distance2Go – XENSIV™ 24 GHz radar demo kit with BGT24MTR11 and XMC4200 32-bit ARM® Cortex™-M4 MCU for ranging, movement and presence detection," Application Note AN543, 2020. [Online]. Available: https://www.infineon.com/cms/en/product/evaluation-boards/demo-distance2go/.

[33] M. A. Richards, *Fundamentals of radar signal processing*, Second edition. New York, NY: McGraw-Hill, 2014.

[34] J. Wang, Y. Lu, L. Gu, C. Zhou, and X. Chai, "Moving object recognition under simulated prosthetic vision using background-subtraction-based image processing strategies," *Inf. Sci.*, vol. 277, pp. 512–524, Sep. 2014, doi: 10.1016/j.ins.2014.02.136.

[35] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. Second Int. Conf. Knowl. Discov. Data Min.*, Portland, Oregon, Aug. 1996, pp. 226–231.

[36] P. R. Kalata and K. M. Murphy, "α-β target tracking with track rate variations," in *Proc. Twenty-Ninth Southeast. Symp. Syst. Theory*, Cookeville, TN, USA, 1997, pp. 70–74, doi: 10.1109/SSST.1997.581581.

[37] J. Munkres, "Algorithms for the assignment and transportation problems," *J. Soc. Ind. Appl. Math.*, vol. 5, no. 1, pp. 32–38, Mar. 1957, doi: 10.1137/0105003.

[38] S. Liaqat, S. A. Khan, M. B. Ihsan, S. Z. Asghar, A. Ejaz, and A. I. Bhatti, "Automatic recognition of ground radar targets based on target RCS and short time spectrum variance," in *Proc. 2011 Int. Symp. Innov. Intell. Syst. Appl.*, Istanbul, Turkey, Jun. 2011, pp. 164–167, doi: 10.1109/INISTA.2011.5946127.

[39] S. Lee, S. Kang, S.-C. Kim, and J.-E. Lee, "Radar cross section measurement with 77 GHz automotive FMCW radar," in *Proc. 27th IEEE Annu. Int. Symp. Pers. Indoor Mob. Radio Commun.*, Valencia, Spain, Sep. 2016, pp. 1–6, doi: 10.1109/PIMRC.2016.7794738.

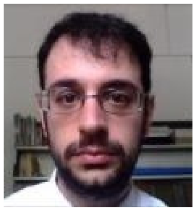
[40] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, "An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes," *Pattern Recognit.*, vol. 44, no. 8, pp. 1761–1776, Aug. 2011, doi: 10.1016/j.patcog.2011.01.017.

[41] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, Dec. 2012, pp. 2951–2959.

[42] A. Rojas-Dominguez, L. C. Padierna, J. M. Carpio Valadez, H. J. Puga-Soberanes, and H. J. Fraire, "Optimal Hyper-Parameter Tuning of SVM Classifiers With Application to Medical Diagnosis," *IEEE Access*, vol. 6, pp. 7164–7176, 2018, doi: 10.1109/ACCESS.2017.2779794.

[43] E. Tuba, M. Tuba, and D. Simian, "Adjusted bat algorithm for tuning of support vector machine parameters," in *Proc. 2016 IEEE Congr. Evol. Comput.*, Vancouver, Canada, Jul. 2016, pp. 2225–2232, doi: 10.1109/CEC.2016.7744063.

Ali Rizik (Student Member, IEEE) received the B.S. degree in electronic and the M.Sc. degree in signal, telecom, image, and speech processing, from the Faculty of Science, Lebanese University, in 2014 and 2016, respectively. He is currently pursuing the Ph.D. degree in science and technology for electronic and telecommunication engineering with the Department of Naval, Electrical, Electronic, and Telecommunications Engineering (DITEN), University of Genoa. From April to August 2016, he was an Internship with ULCO University, France. His main research interest includes machine learning applications, radar signal processing, and radar target classification for security applications.



Emanuele Tavanti (Member, IEEE) received the Laurea degree in electronic engineering and the Ph.D. degree in science and technology for electronic and telecommunications engineering, from the University of Genoa, Genoa, Italy, in 2015 and 2019, respectively. He is currently a Research Fellow with the Department of Electrical, Electronic, Telecommunications Engineering, and Naval Architecture of the University of Genoa. His research activity is focused on computational methods for direct and inverse electromagnetic scattering, biomedical applications of bioimpedance measurement technologies, and security applications of radar technologies.

Hussien Chible (Member, IEEE) received the degree in electrical engineering, section Communication and Electronics Engineering, from the Beirut Arab University, Lebanon, in 1992, and obtained the BAU University Award. He obtained the Ph.D. degree from the University of Genoa, Italy, in 1997 in Electronic and Computer Engineering. He then joined Lebanese University, Beirut, Lebanon, where he is currently a full Professor and He is now the head of the research Centre at the faculty of tourism, also he is the founder of MECRL TEAM at the PhD school of science and technology. He obtained the full professor title at Lebanese University in 2009. He has many publications in microelectronics, and tourism. His scientific interests focus on microelectronics analog CMOS VLSI circuits, designing full custom IC, Neural Network, E-skin. Also, his interests are in tourism field.

Daniele D. Caviglia (Member, IEEE) graduated in electronic engineering and specialized in computer engineering from the University of Genoa, Genoa, Italy, in 1980 and 1982, respectively. In 1983, he joined the Institute of Electrical Engineering, University of Genoa, as an Assistant Professor in electronics. Since 1984, he has been with the Department of Biophysical and Electronic Engineering (DIBE), as Associate Professor (1992) and Full Professor (2000). He was Director of DIBE from 2002 to 2008. He is currently with the Department of Electrical, Electronics and Telecommunications Engineering and Naval Architecture (DITEN), University of Genoa, teaching courses in electronic systems for telecommunications, electronic devices and circuits, and microelectronics. His research interests include the design of electronic circuits and systems for telecommunications, electronic equipment for health and safety, and energy harvesting techniques for Internet of Things (IoT) applications. He is also active in the development of innovative solutions for environmental monitoring, for civil protection and security applications.

Andrea Randazzo (Senior Member, IEEE) received the laurea degree (cum laude) in telecommunication engineering and the Ph.D. degree in information and communication technologies from the University of Genoa, Genoa, Italy, in 2001 and 2006, respectively. He is currently a Full Professor of electromagnetic fields at the Department of Electrical, Electronic, Telecommunication Engineering, and Naval Architecture, University of Genoa. He has coauthored the book Microwave Imaging Methods and Applications (Artech House, 2018) and more than 270 articles published in journals and conference proceedings. His primary research interests are in the field of microwave imaging, inverse-scattering techniques, radar processing, numerical methods for electromagnetic scattering and propagation, electrical tomography, and smart antennas.