Delft University of Technology

Distributed radar fusion and recurrent networks for classification of continuous human activities

Guendel, Ronny; Fioranelli, Francesco; Yarovoy, Alexander

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH

Distributed radar fusion and recurrent networks for classification of continuous human activities

Ronny G. Guendel ⬤ | Francesco Fioranelli ⬤ | Alexander Yarovoy

Microwave Sensing, Signals and Systems (MS3) Group, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands

Correspondence

Ronny G. Guendel, Microwave Sensing, Signals and Systems (MS3) Group, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands.
Email: rguendel@tudelft.nl

Abstract

Continuous Human Activity Recognition (HAR) in arbitrary directions is investigated in this paper using a network of five spatially distributed pulsed Ultra-Wideband radars. While activities performed continuously and in unconstrained trajectories provide a more realistic and natural scenario for HAR, the network of radar sensors is proposed to address the issue of unfavourable or occluded perspectives when using only a single sensor. Different techniques to combine the relevant information from the multiple radars in the network are investigated, focussing on signal level fusion directly applied on Range-Time maps, and the selection of radar nodes based on location and velocity of the target derived from multilateration processing and tracking. Recurrent Neural Networks with and without bidirectionality are used to classify the activities based on the micro-Doppler (μD) spectrograms obtained for sensor fusion techniques. To assess classification performances, novel evaluation metrics accounting for the continuous nature of the sequence of activities and inherent imbalances in the dataset are proposed and compared with existing metrics. It is shown that the conventional accuracy metric may not capture all the important aspects for continuous HAR, and the proposed metrics can be considered for a more comprehensive evaluation.

1 | INTRODUCTION

Monitoring Activities of Daily Living (ADL) by radar has gained attention for safe and independent ageing-in-place of older and vulnerable subjects. This includes recordings of critical events such as falls, monitoring abnormalities in movements and activities, and in general, providing an appraisal of wellbeing in terms of cognitive and physical state [1–3].

As Human Activity Recognition (HAR) by radar is typically exploiting micro-Doppler (μD) signatures of human movements, distributed networks with multiple cooperating radars have attracted significant interest to address the issue of reduced μD signatures recorded at unfavourable aspect angles [4–9]. However, the focus has been often on the classifier's architectures, that is, neural networks [3], including those in our

work [10]. To the best of our knowledge, there are only limited experimental studies on the most suitable number and topology of the different radar sensors for HAR, and on the most effective fusion techniques to combine their data [6, 11]. Hence, such questions remain widely open in HAR.

Griffiths et al. [12, 13] investigated drone payload classification using 3 multistatic radar nodes and achieved superior results by voting-based decision fusion among independent classifiers (named as 'binary voting' and 'threshold voting'). The same authors [14, 15] also investigated the usage of handcrafted features of μD spectrograms (e.g. their centroid and bandwidth) for other classification tasks based on the same multistatic radar network of 3 nodes. These tasks included gait analysis of individuals alone and in pairs to identify whether they were armed or unarmed in outdoor surveillance scenarios. Even in this case, decision fusion appeared to provide the best

classification results, but little investigation was devoted to lower level signal fusion approaches.

Unlike the aforementioned studies, this paper investigates novel lower-level fusion approaches applied to a network of five spatially distributed monostatic radars simultaneously observing a surveillance area. The proposed fusion schemes aim to combine data from a selection of different radar nodes prior to the generation of the μD spectrogram used for classification. Specifically a fusion method based on *incoherent fusion* of the Range-Time (RT) domain data from each radar node is investigated. While rather simple in terms of computation, this approach proves to be the most effective in terms of overall classification results, with the following potential advantages given:

- Minimising the number of classifiers to be defined and trained in the pipeline to one.
- Avoiding complex methods to combine the partial decisions from separate classifiers.
- Using one single μD spectrogram representation for the whole network, containing information from all nodes with computational simplicity.

Furthermore, two approaches are evaluated to select a subset of the available radar nodes for the subsequent fusion process. The first approach is the *orthogonal radar selection*, whereby two radar nodes with orthogonal lines of sight are selected in order to capture μD signatures along both the radial and tangential directions. While this selection is static and done once and for all, with the second approach, a dynamic selection of the radar nodes is performed. In this *weighted radar selection*, the location, velocity, and heading of the target are first estimated by multilateration processing [16] combined with a simple alpha-beta-(gamma) $-$ α, β, (γ) tracking filter. Then, a weighting function is implemented to select at any given time the most suitable radar for classification, that is, the closest radar to the target (thus assuming the highest SNR), or the one radar with the most favourable aspect angle to the trajectory (thus assuming the richest micro-Doppler information).

The aforementioned different radar data fusion methods are explored in this paper in the context of classifying continuous sequences of human activities, as opposed to the more conventional classification of artificially separated activities [6, 17, 18]. These continuous activities represent more realistic and natural scenarios to evaluate radar-based HAR algorithms, where the transitions between different activities can happen at any time and undefined instants. If performed along unconstrained trajectories and directions, the classification of such activities can clearly benefit from the multiperspective views of distributed radar networks.

However, as discussed in our preliminary results in [19], HAR on continuous sequences benefit from additional, alternative performance evaluation metrics beyond simple accuracy or quantities directly extracted from confusion matrices, regardless of the nature of the radar used for recording, that is, monostatic or distributed/multistatic. Specifically, four aspects of continuous HAR data need to be considered.

First, the *Continuity*. The activities are performed in a natural way—a continuous sequence, where transitions between them are not only happening at arbitrary times, but also extended in time. So, it is difficult to pinpoint precisely the time instant where one activity ends and the following activity starts, even in the ground truth. Then the presence of *Misalignments*. As a consequence of the difficulty to estimate precisely the time instant of activity transitions, misalignments between ground truth and predictions label can happen, that is, time offsets between ground truth and predictions. Depending on the overall goal of the HAR system, one needs to establish how important such misalignments are in terms of the performance evaluation of classification algorithms. Furthermore, the possible *Interruptions*. As an activity occupies an extended number of time bins, an ideal prediction would have the corresponding correct label for all of them. However, there may be cases where the classifier returns temporary short fluctuations (i.e. outliers) in the predicted label for one or a limited number of time bins. This fluctuation of the predicted label is generally overlooked when classifying human activities as artificially separated, 'snapshot' images, and not captured when using conventional evaluation metrics for continuous-time sequences. Finally, the *Imbalance* in the dataset. When evaluating realistic sequences of activities, imbalances in the dataset will naturally appear due to different occurrences and time spans of activities during an observation period. As in this paper, a typical example can be the prevalence of the 'walking' class while participants move about in the room to perform single instances of other in-place activities. Furthermore, another typical situation is the small amount of available samples for critical activities, such as 'falling'.

To account for the continuous nature of human activities and provide a more insightful performance analysis based on the aforementioned aspects, this paper discusses a collection of 10 possible evaluation metrics with their advantages and disadvantages. These metrics are used to evaluate the different radar data fusion methods on an experimental dataset with 5 radar nodes, 15 participants, and 9 labelled activity classes. The presented methods are validated for single human target cases. Although more complex scenarios exist, such as multiple individuals in the field of view or individuals with domestic animals, these are left for future research work.

To summarise, the main contribution of this paper are:

- Novel fusion methods to combine data from radars in a spatially distributed network are investigated, focussing on signal level fusion and selection of a subset of nodes to improve HAR performances.
- Original classification performance metrics are shown to account for the continuous nature of human activities. These metrics are used to evaluate an experimental dataset containing data from 5 radar nodes, with 15 participants performing unconstrained and continuous sequences of 9 activities.
- Four Recurrent Neural Networks (RNNs) are used as classifiers for HAR, namely Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM), including their

bidirectional implementation. These architectures are considered very suitable for HAR based on continuous sequences, as they can take directly as input a µD sequence.

The rest of the paper is organised as follows. Section 2 describes the experimental setup, the collected dataset, and the class set distribution plus their separability, followed by Section 3 showing the multilateration tracking approach. Section 4 presents the proposed fusion schemes for distributed radar sensors. The evaluation metrics are introduced in Section 5 and afterwards the hyperparameter tuning for the used RNN in Section 6. The experimental results are presented in Section 7, and final remarks are given in Section 8.

2 | EXPERIMENTAL SETUP AND COLLECTED DATASET

2.1 | Experimental setup

Five Ultra-Wide Band (UWB) radar nodes by Humatics P410 (former PulsON) are simultaneously employed with coded waveform capabilities minimising interference between nodes. The in-phase back-scattered signal is recovered by filter banks, with the quadrature component obtained by Hilbert transform. The Pulse Repetition Frequency (PRF) f_{PRF} is equal to 122 Hz (corresponding to a Pulse Repetition Interval [PRI] of 8.2 ms). The unambiguous Doppler frequency results in ± 61 Hz (± 2.2 m/s), and the radar filterbanks have a time-of-flight sampling rate of $\tau = 61$ ps. The range resolution (r_{res}) by using a bandwidth of $B = 2.2$ GHz is 68 mm according to $r_{res} = \frac{c}{2 \cdot B}$.

The radar nodes are deployed over a circular baseline, with a spacing of approximately 45° between them, covering a surveillance area of about 4.39 m as shown in Figure 1.

2.2 | Dataset and class distribution

Continuous sequences of activities are recorded with 15 participants available in a public dataset[1] [20]. Test and training data procedure is performed by excluding one participant from the training data for testing. The procedure is well known as *leave one person out* (L1PO). For all participants, each of the collected recording has a total duration of 2 min, with all activities performed at predefined locations (sequence type A), and at freely chosen locations within the surveillance area (sequence type B). It should be noted that the participants were free to move in unconstrained directions between performing each activity, and to face random directions in terms of aspect angles to the five radars.

While nine activities were recorded, these were grouped into 5 classes, namely: (I) 'translation' activities (i.e., essentially walking); (II) 'stationary' activity (i.e., essentially the position of
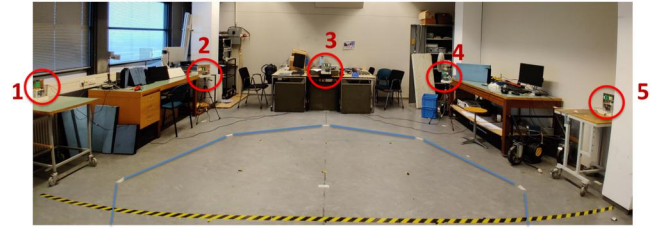
FIGURE 1 Distributed radar network with five radars observing a surveillance area of about 4.39 m in diameter at the MS3 laboratory, TU Delft

standing between two other activities without performing any specific movement); (III) 'in-place' activities (namely, sitting down, standing up from sitting, bending while sitting and standing); (IV) 'falling' (including both falling from standing or walking); and (V) 'standing up from falling'.

This decision in terms of grouping activities together into 'macro-classes' came from limited practical benefits gained by too fine-grained separation of very similar in-place classes such as 'bending from walking' and 'bending from standing', or by distinguishing 'falling from walking' and 'falling from standing'. This also reduces in part the imbalance or skewness of the dataset by grouping together minority classes of activities for which few samples were available, especially the 'in-place' activities. The t-distributed Stochastic Neighbor Embedding (tSNE) representation for the features extracted from the µD domain of the original 9 classes and for the grouped 5 'macro-classes' are presented in Figure 2 after feature scaling performed.

Furthermore, the pie diagram in Figure 3 illustrates the sample distribution for the collected dataset after the class grouping. As mentioned in the introduction, continuous activities recorded in semi-natural conditions may lead to imbalanced datasets where walking is predominant compared to in-place activities or critical activities like fall events. This is visualised with the 'walking' class being about 50% of the dataset. Without resorting to generation of synthetic data to correct for the imbalance [21], its effect in performance assessment is addressed by the parameter optimisation process shown in Section 6, where the dispersion using the standard deviation of the F_1 score across classes has been included in the metrics evaluation process by minimising the F_1 score fluctuation as shown in Equation (31).

3 | MULTILATERATION AND TRACKING APPROACH

This section introduces multilateration based localisation to determine the target's position and is followed by an alpha-beta-(gamma) $- \alpha, \beta, (\gamma)$ filter that is applied to estimate the smoothed target's location, velocity, and acceleration. These are then used in the selection of the radar nodes to be involved in the data fusion process prior to classification. The multilateration processing and the tracking approach with the $\alpha, \beta, (\gamma)$ filter can be considered as subsequent processing 'blocks'
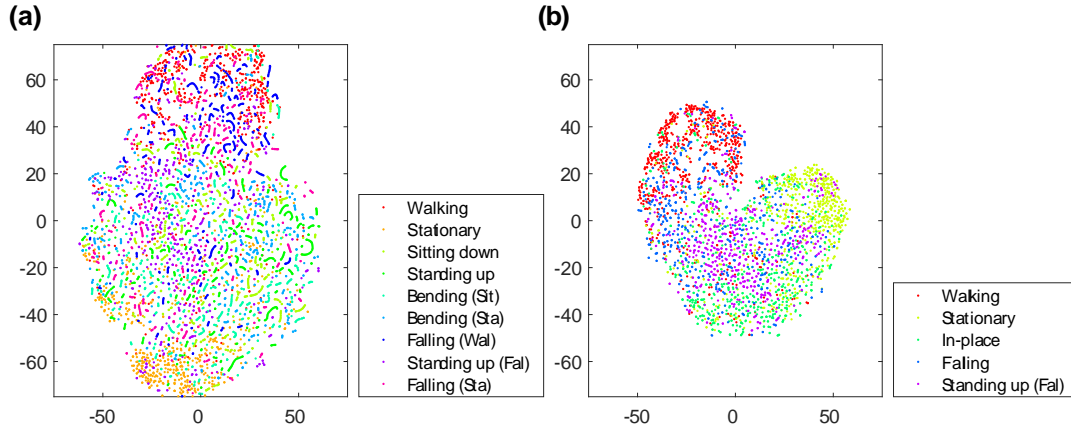
FIGURE 2 The t-SNE distribution shown for the original 9 classes and their combinations into 5 'macro-classes' to be classified. (a) before class combination; (b) after combining classes to 'macro-classes', such as, in-place ← [sitting down, standing up, and bending] falling ← [falling from walking, and falling from standing]
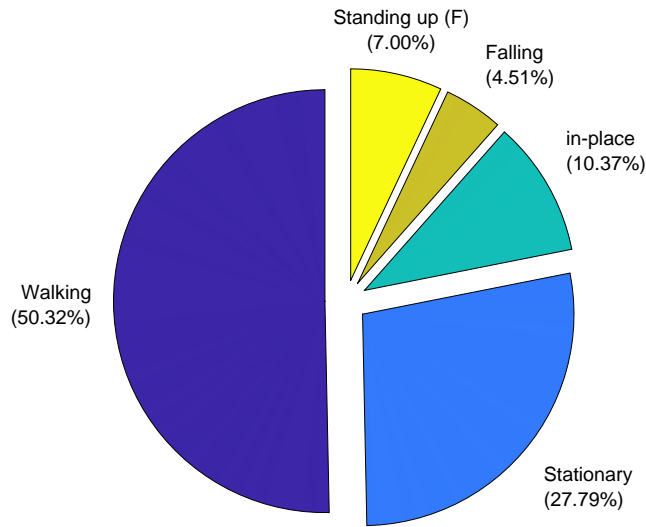
FIGURE 3 The pie diagram shows the imbalance of classes with the minority and majority classes of 'falling' and 'walking' below 5% and above 50%, respectively

FIGURE 4 The distributed layout for the radar network uses five nodes. The target's location is estimated by the intersections of the determined ranges of each node using multilateration processing

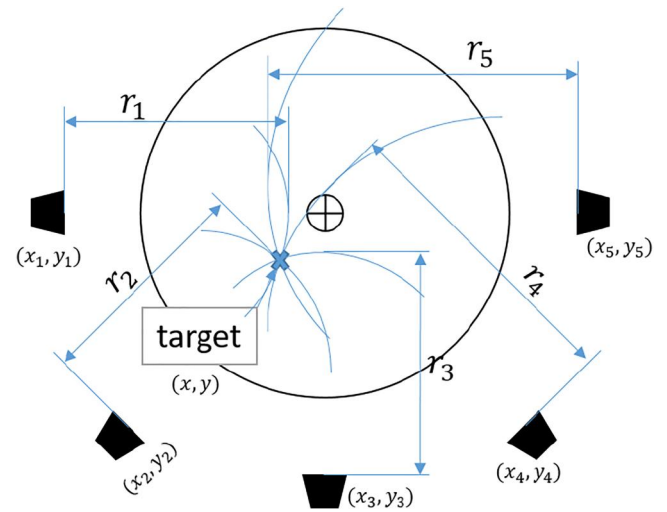one after the other. Specifically, while the target's location can be estimated directly from the multilateration processing, the tracking filter is used for smoothing the location and additionally extracting the single target's velocity and acceleration. Although more advanced methods such as various implementations of the Kalman filter may be applied, in this paper the simpler α, β, (γ) filter provides satisfactory results for the next stage of selection of radar nodes based on the target's state, which is described in Section 4.3.

3.1 | Multilateration positioning

Multilateration processing is applied to estimate the target's location in a multi-sensor system. Hence, the system is an over-

determined equation system, as it can easily be seen in Figure 4. The peak power of the range pulse as in [22] provides the radial range, r_{nx} of the nth radar with known radar node location (x_{nx}, y_{nx}). The target's estimated position $(\tilde{x}, \tilde{y})$ can be expressed as follows:

$$\begin{cases} (\tilde{x} - x_1)^2 + (\tilde{y} - y_1)^2 & = r_1^2 \\ (\tilde{x} - x_2)^2 + (\tilde{y} - y_2)^2 & = r_2^2 \\ \vdots \quad \vdots \quad \vdots & \vdots \\ (\tilde{x} - x_{Nx})^2 + (\tilde{y} - y_{Nx})^2 & = r_{Nx}^2 \end{cases} \quad (1)$$

Equation (1) is linearised for five radars such as:

$$\begin{cases} (\tilde{x} - x_1)^2 + (\tilde{y} - y_1)^2 - (\tilde{x} - x_5)^2 - (\tilde{y} - y_5)^2 & = r_1^2 - r_5^2 \\ (\tilde{x} - x_2)^2 + (\tilde{y} - y_2)^2 - (\tilde{x} - x_5)^2 - (\tilde{y} - y_5)^2 & = r_2^2 - r_5^2 \\ \quad\vdots \qquad\qquad \vdots \qquad\qquad\qquad \vdots \qquad\qquad \vdots \\ (\tilde{x} - x_4)^2 + (\tilde{y} - y_4)^2 - (\tilde{x} - x_5)^2 - (\tilde{y} - y_5)^2 & = r_4^2 - r_5^2 \end{cases} \quad (2)$$

Then, Equation (2) can then be rewritten in matrix notation as follows:

$$\mathbf{A} = \begin{bmatrix} 2(x_1 - x_5) & 2(y_1 - y_5) \\ \vdots & \vdots \\ 2(x_4 - x_5) & 2(y_4 - y_5) \end{bmatrix} \quad (3)$$

$$\mathbf{b} = \begin{bmatrix} x_1^2 - x_5^2 + y_1^2 - y_5^2 + r_5^2 - r_1^2 \\ \vdots & \vdots & \vdots \\ x_4^2 - x_5^2 + y_4^2 - y_5^2 + r_5^2 - r_4^2 \end{bmatrix} \quad (4)$$

The over-determined equation system is solved by an ordinary least squares (OLS) estimation that minimises the error. Thus, the target coordinates can be calculated as follows:

$$\tilde{\mathbf{x}}_n = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = \left(\mathbf{A}^T \mathbf{A} \right)^{-1} \mathbf{A}^T \mathbf{b} \quad (5)$$

where $\left(\mathbf{A}^T \mathbf{A} \right)^{-1} \mathbf{A}^T$ is the Moore–Penrose inverse and $(\cdot)_n$ is the discrete-time instant index [16, 23, 24].

3.2 | Alpha beta (gamma) – α, β, (γ) filter

The output of α, β, (γ) filter is used to estimate the location, velocity and acceleration of the single target in the scene, respectively. The tracking filter is jointly used for track smoothing, essential for slow performing movements as those in HAR.

The estimated and unfiltered location is indicated as $\tilde{\mathbf{x}}_n = [\tilde{x}, \tilde{y}]^T$ attained from Equation (5) in Section 3.1. The filter model is given by:

$$\mathbf{X}_{n+1,n} = \mathbf{F} \mathbf{X}_{n,n} \quad (6)$$

with the prediction state $\mathbf{X}$, while $\mathbf{F}$ defines the target kinematic. The kinematic dependencies are expressed such as:

$$\hat{\mathbf{x}}_{n+1,n} = \hat{\mathbf{x}}_{n,n} + \hat{\dot{\mathbf{x}}}_{n,n} \Delta t + \hat{\ddot{\mathbf{x}}}_{n,n} \frac{\Delta t^2}{2} \quad (7a)$$

$$\hat{\dot{\mathbf{x}}}_{n+1,n} = \hat{\dot{\mathbf{x}}}_{n,n} + \hat{\ddot{\mathbf{x}}}_{n,n} \Delta t \quad (7b)$$

$$\hat{\ddot{\mathbf{x}}}_{n+1,n} = \hat{\ddot{\mathbf{x}}}_{n,n} \quad (7c)$$

with $\hat{\dot{\mathbf{x}}}$ and $\hat{\ddot{\mathbf{x}}}$ being the velocity and acceleration, respectively. The target states in Equation (7) can be formulated in matrix notation such as:

$$\mathbf{X}_{n+1,n} = \begin{bmatrix} \hat{\mathbf{x}}_{n+1,n} \\ \hat{\dot{\mathbf{x}}}_{n+1,n} \\ \hat{\ddot{\mathbf{x}}}_{n+1,n} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_{n,n} \\ \hat{\dot{\mathbf{x}}}_{n,n} \\ \hat{\ddot{\mathbf{x}}}_{n,n} \end{bmatrix} \quad (8)$$

$$= \mathbf{F} \mathbf{X}_{n,n}$$

$$\mathbf{X}_{n+1,n} = \begin{bmatrix} \hat{x}_{n+1,n} \\ \hat{y}_{n+1,n} \\ \hat{\dot{x}}_{n+1,n} \hat{\dot{y}}_{n+1,n} \\ \hat{\dot{y}}_{n+1,n} \\ \hat{\ddot{x}}_{n+1,n} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta t & 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & 1 & 0 & \Delta t & 0 & \frac{\Delta t^2}{2} \\ 0 & 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\times \begin{bmatrix} \hat{x}_{n,n} \\ \hat{y}_{n,n} \\ \hat{\dot{x}}_{n,n} \hat{\dot{y}}_{n,n} \\ \hat{\dot{y}}_{n,n} \\ \hat{\ddot{x}}_{n,n} \end{bmatrix} \quad (9)$$

The track update equations enclosing the present measurement, $\tilde{\mathbf{x}}_n$, are given by:

$$\hat{\mathbf{x}}_{n,n} = \hat{\mathbf{x}}_{n,n-1} + \alpha \left(\tilde{\mathbf{x}}_n - \hat{\mathbf{x}}_{n,n-1} \right) \quad (10a)$$

$$\hat{\dot{\mathbf{x}}}_{n,n} = \hat{\dot{\mathbf{x}}}_{n,n-1} + \beta \left(\frac{\tilde{\mathbf{x}}_n - \hat{\mathbf{x}}_{n,n-1}}{\Delta t} \right) \quad (10b)$$

$$\hat{\ddot{\mathbf{x}}}_{n,n} = \hat{\ddot{\mathbf{x}}}_{n,n-1} + \gamma \left(\frac{\tilde{\mathbf{x}}_n - \hat{\mathbf{x}}_{n,n-1}}{\frac{1}{2} \Delta t^2} \right) \quad (10c)$$

with the residual error included as $\tilde{\mathbf{x}}_n - \hat{\mathbf{x}}_{n,n-1}$. Δt is the time between instantaneous measurements and refers to the PRI. The location-, velocity-, and acceleration parameters of the

tracking filter, α, β, and γ, respectively, are found empirically [25]. The prediction states from the tracking filter are used for the selection of radar nodes and subsequent data fusion, as described in Section 4.3.

4 | RADAR DATA FUSION APPROACHES

This section presents the proposed signal and feature fusion approaches for the network of five UWB radars for HAR. The notation in this section is defined as in Table 1, and the feature vector length obtained from one µD spectrogram is defined by k with feature scaling performed. Furthermore, Figure 5 shows schematically the proposed fusion approaches presented in this work and compared to the case of a single radar sensor being used.

4.1 | Signal level fusion

The received radar echoes in fast time provide the target's radial range. The main lobe is typically associated with the target's position and the sidelobes defining the noise floor, assuming sufficient SNR conditions. The simple summation of the complex Range-Time (RT) matrices from all radar nodes implements incoherent signal-level fusion as:

$$X(m, t) = \frac{1}{Nx} \sum_{nx=\#1}^{Nx} \chi(m, t)^{(nx)} \quad (11)$$

The obtained RT matrix, $X(m, t)$, contains information from all radar nodes, as shown in the pipeline in Figure 6. This resulting matrix is further used to calculate a µD spectrogram to be fed as input to the classifier of choice after feature scaling.

The Short-time Fourier transform (STFT) is applied on the RT, $X(m, t)$, as in [26]. The resulting µD spectrogram, $\Psi(m', t')$, contains the Doppler/velocity information of the target from all nodes, where m' refers to the µD spectrogram frequency bins and t' indicates the slow-time bins, respectively. A variety of STFT window sizes and overlap values were tested for the best performance between clutter suppression and clarity of limb motions. Clutter cancelation is performed by subtracting the average Doppler frequencies from the µD spectrogram, with satisfying classification achieved with the STFT overlap of 10 samples (82 ms → t'), and a Hanning window size of 150 samples (1.23 s) [22, 27].

Afterwards, the proposed method uses directly the slow-time bins of the resulting µD spectrogram as feature vectors for classification based on Recurrent Neural Networks (RNN) [28], rather than using sliding windows or other techniques to segment the flow of continuous activities.

4.2 | Feature level fusion

Feature level fusion is applied on the individual µD spectrograms, indicated as $\psi_{nx}(m', t')$. Thus, the STFT is applied on

TABLE 1 Notation for radar data fusion approaches in Section 4

k/K	Individual/total feature vector length
nx/Nx	Individual/total radar nodes with {#1, …, #5}
χ_{nx}/X	Individual/combined range-time (RT) map
ψ_{nx}/Ψ	Individual/combined µD spectrogram feature map
m/t	RT map fast time –, and the slow-time index
m'/t'	µD spectrogram frequency bin –, and the slow-time index

the individual RT plot from each radar node, $\chi_{nx}(m, t)$, as computed in [26], where $(\cdot)_{nx}$ indicates the radar. Feature fusion concatenates the individual µD spectrograms from each radar node as:

$$\Psi(m', t') = \left[\psi_1(m', t')^T, \psi_{nx}(m', t')^T, \ldots, \psi_{Nx}(m', t')^T \right]^T \quad (12)$$

with, t', the slow time bins of the µD spectrogram. It should be noted that the slow-time sampling t of the RT map, $\chi_{nx}(m, t)$, depends on the PRF of the radars, whereas the slow-time sample t' of the µD spectrogram, $\psi_{nx}(m', t')$, depends on the STFT window and overlap parameters.

Both the proposed feature fusion and the signal level fusion presented in Section 4.1 use information from all the radar nodes in the network. However, with feature fusion, the length of the resulting feature vector is $Nx \times k$, thus Nx times long compared to signal level fusion (where in our case Nx is equal to five, the total number of radar nodes in the network). Hence, an advantage of signal level fusion may be the dimensionality reduction of the resulting feature vector.

4.3 | Weighted radar selection over time

As discussed in the previous section, feature fusion increases the dimensionality of the feature vector by the number of radar nodes used. Hence, there could be an advantage in reducing the number of radars used in the fusion process by only keeping those providing relevant information for classification. In the proposed weighted radar selection, this process is not done once and for all, but dynamically adjusted based on the target behaviour. The position and movement of the target are used to select the most suitable radar with respect to (w.r.t.) the proposed weight function in Equations (13)–(15).

First, the prediction states of the alpha beta (gamma) – α, β, (γ) filter from Section 3.2 provides the location, velocity, and acceleration of the target $[\hat{\mathbf{x}}_{n+1,n}, \dot{\hat{\mathbf{x}}}_{n+1,n}, \ddot{\hat{\mathbf{x}}}_{n+1,n}]$. As the position of the radar nodes $\mathbf{x}_{nx} = [x_{nx}, y_{nx}]^T$ is assumed known, the Euclidean distance computes the length of the vector differences and defines the target's distance to each node such as $\|\vec{\eta}_{nx}\|_{l_2} = \|\mathbf{x}_{nx} - \hat{\mathbf{x}}_{n+1,n}\|_{l_2}$. It should be noted that the discrete time index, $(\cdot)_{n+1,n}$, will be neglected for convenience to improve the readability.
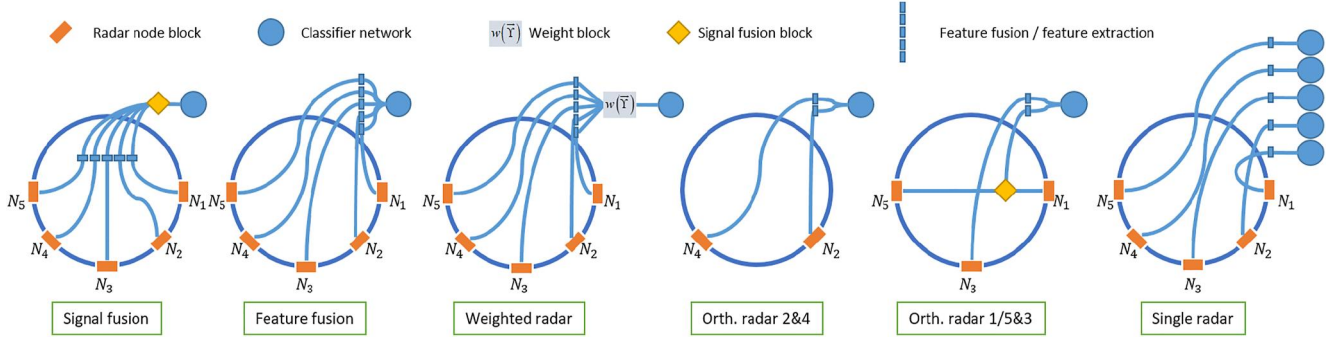
FIGURE 5 Sketches of the proposed fusion approaches of 'signal fusion', 'feature fusion', 'weighted radar selection', and 'orthogonal radar selection' (in two forms) compared with 'single radar' classification applied for HAR in a distributed radar network
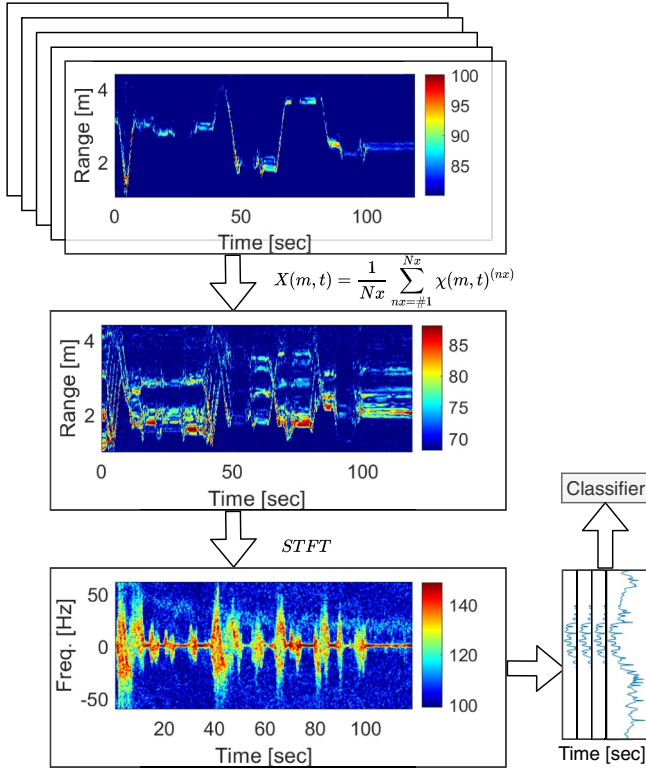
FIGURE 6 Pipeline of incoherent signal fusion, from individual Range-Time (RT) maps to one µD spectrogram used for the Recurrent Neural Network (RNN) classifier

Then, a weighting function can be computed as:

$$\mathbf{w}(\eta) = \frac{\frac{1}{\|\vec{\eta}_{nx}\|_{l_2}^4}}{\sum_{nx=\#1}^{Nx} \frac{1}{\|\vec{\eta}_{nx}\|_{l_2}^4}} = \frac{1}{\sum_{nx=\#1}^{Nx} \|\vec{\eta}_{nx}\|_{l_2}^{-4}} \begin{bmatrix} \frac{1}{\|\vec{\eta}_1\|_{l_2}^4} \\ \frac{1}{\|\vec{\eta}_2\|_{l_2}^4} \\ \vdots \\ \frac{1}{\|\vec{\eta}_{Nx}\|_{l_2}^4} \end{bmatrix} \quad (13)$$

where the term $1/(\cdot)^4$ is related to the SNR distance-power relationship inspired by the radar equation [29]. The denominator, $\sum_{nx=\#1}^{Nx} \|\vec{\eta}_{nx}\|_{l_2}^{-4}$, is a normalisation term such that $\sum_{nx=\#1}^{Nx} w_{nx}(\eta) = 1$, for $\mathbf{w}(\eta) = [w_1(\eta), w_{nx}(\eta), \ldots, w_{Nx}(\eta)]^T$. The radar associated with the maximum value of the weighting function at a given time is the one selected for the subsequent classification process, typically the one physically closest to the target and typically providing the highest SNR. Only its feature vector is forwarded to the RNN classifier as sketched in Figure 5.

Two further weight functions are formulated accounting for the target's aspect angle, $\cos(\phi_{nx})$, and radial velocity, Υ_{nx}, respectively. As previously shown, the first partial derivative of the predicted state, $\hat{\mathbf{x}}_{n+1,n}$ is the velocity. The projection of the velocity onto the line-of-sight vector, $\vec{\eta}_{nx}$, is the radial velocity to each radar node such as, $\Upsilon_{nx} = \frac{\hat{x} \cdot \vec{\eta}_{nx}}{\|\vec{\eta}_{nx}\|_{l_2}^2} \vec{\eta}_{nx}$. The weight function of the person's velocity to the radar nodes is computed similar to Equation (13) such as:

$$\mathbf{w}(\Upsilon) = \frac{\left\|\vec{\Upsilon}_{nx}\right\|_{l_2}}{\sum_{nx=\#1}^{Nx} \left\|\vec{\Upsilon}_{nx}\right\|_{l_2}} \quad (14)$$

with $\|\cdot\|_{l_2}$ (the l2-norm) as the absolute velocity w.r.t. the radar nodes.

The target's aspect angle to the radar nodes is determined by the angle between the velocity vector, $\hat{\mathbf{x}}_{n+1,n}$, and the line-of-sight vector, $\vec{\eta}_{nx}$ such as, $\cos(\phi_{nx}) = \frac{\hat{x} \cdot \vec{\eta}_{nx}}{\|\hat{x}_{nx}\|_{l_2} \cdot \|\vec{\eta}_{nx}\|_{l_2}}$. The weight function of the target's aspect angle to the radar nodes is then computed as:

$$\mathbf{w}(\cos(\phi_{nx})) = \frac{|\cos(\phi_{nx})|}{\sum_{nx=\#1}^{Nx} |\cos(\phi_{nx})|} \quad (15)$$

with the angle term $\cos(\phi_{nx})$ ranging between [-1,1] and 0 indicating a tangential movement to the radar node ($|\cdot|$ denotes the absolute value). It should be noted that the

aspect angle of facing the radar, $|\cos(0)| = 1$, as well as facing away, $|\cos(2\pi)| = 1$, are expected to provide the highest Doppler response and radar cross-section for HAR (i.e., directly facing back or torso of the person). The radar node associated with the maximum value of the weighting function $\mathbf{w}(\Upsilon)$ or $\mathbf{w}(\cos(\phi_{nx}))$ at a given time is the one selected for the subsequent classification process, essentially the one perceiving the highest radial velocity or located at the most superior aspect angle, respectively.

It should be noted that during this research test which combines the aforementioned weight functions as $\mathbf{w} = \mathbf{w}(\eta) \cdot \mathbf{w}(\Upsilon) \cdot \mathbf{w}(\cos(\phi_{nx}))$ were performed, but led to limited performance improvement, hence they are not reported in this paper.

4.4 | Orthogonal radar (static) data fusion

It has been found that classification of unconstrained HAR using a single radar suffers from the lack of recognition capabilities of orthogonal movements, that is, tangential to the line of sight. Hence, with orthogonal radar fusion a pair of 2 nodes are deliberately chosen in the network with lines of sight along radial and tangential directions. As shown in Figure 5, this orthogonal pair of radars is separated by a quarter circle (90°). The choice of this pair of radars is static, that is, not changing during the measurement. An example of a resulting 2D-μD spectrogram is shown in Figure 7, with velocity (Doppler) components in principle covering all standard movement directions needed for unconstrained HAR.

Two possible selections are considered in this paper:

- *Orthogonal radars 2&4.* The chosen subset of radar nodes $nx = \{\#2, \#4\}$ with the RT maps $\chi_{\#2}(m, t)$, and $\chi_{\#4}(m, t)$ are used to extract the μD spectrograms by applying the STFT. The resulting μD spectrograms $\psi_{\#2}(\text{m',t'})$, and $\psi_{\#4}(\text{m',t'})$ are combined via feature fusion as $\Psi(\text{m', t'}) = \left[\psi_{\#2}(\text{m', t'})^T, \psi_{\#4}(\text{m', t'})^T\right]^T$ and then used for classification. The feature vector domain doubled in size compared to using a single radar or the signal level fusion in Section 4.1.
- *Orthogonal radars one-fifth&3.* Here the orthogonal radar nodes #1 and #5 are first incoherently fused at signal level as in Section 4.1, as: $X_{\#1,\#5}(m,t) \leftarrow 1/N \sum_{nx=\#1}^{\#5} \chi_{nx}(m,t)$. Then the STFT applied on the concatenated RT domain $\left(X_{\#1,\#5}(m,t)\right)$ obtains the spectrogram as: $\psi_{\#1,\#5}(\text{m', t'}) \leftarrow \text{STFT}\left(X_{\#1,\#5}(m,t)\right)$. Finally, feature fusion is performed together with node #3 needed for generating the 2D-μD spectrogram of the target of interest such as $\Psi_{\#1,\#5,\#3}(2\text{m', t'}) = \left[\psi_{\#1,\#5}(\text{m', t'})^T, \psi_{\#3}(\text{m', t'})^T\right]^T$. Thus, the concatenated features are used for subsequent classification with a feature vector length of $2 \times \text{k} := 2 \times \text{len(m')}$.
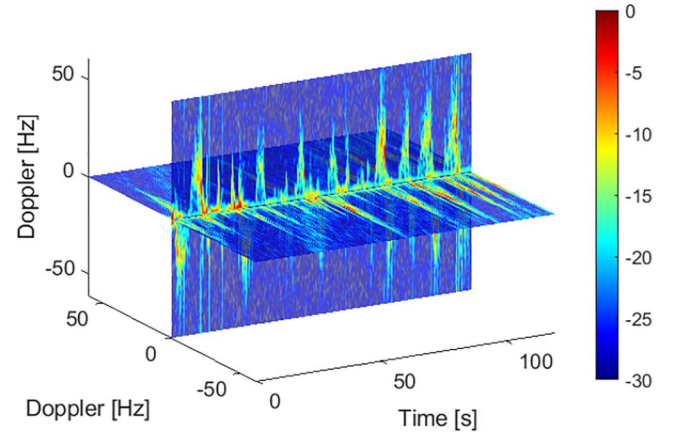

FIGURE 7 The 2D orthogonal spectrogram spanning an x-and y-plane over time domain generating the 2D-μD spectrogram

4.5 | Summary of radar data fusion approaches

This section introduced different fusion methods for combining data collected by distributed radar nodes. Low level *signal fusion* combines directly the RT data of different radars and generates one single μD spectrogram for classification, hence a feature vector with size equivalent to just using only one radar. *Feature fusion* calculates and concatenates the spectrograms from different nodes, hence enlarging the size of the feature vector. Rather than selecting all nodes all the time for feature fusion, only pairs of *orthogonal radars* have been considered. Furthermore, a weighted radar selection over time has been formulated, whereby the most suitable radar node is selected at any given time based on a criterion mapped to a weighting function (i.e. the closest radar to the target, the one with the most favourable aspect angle, or velocity presented). The proposed methods are summarised in Table 2, and their performances will be compared to just using one single radar for classification, as if there was no distributed radar network.

5 | EVALUATION METRICS

In this section, evaluation metrics for continuous and unconstrained HAR introduced in our earlier work [19] are described in detail using the notation in Table 3. These metrics will be used in the following sections for performance assessment.

5.1 | Accuracy

To compare predictions and ground truth labels the identity function for each class c $I^{(c)}\left(\hat{y}_p, y_p\right)$ is introduced to measure incorrect predictions:

Fusion method	Feature size	Notes
Signal fusion	K	Best case reference
Feature fusion	5×k	Longest feature vector
Weighted radar fusion	K	Tracker needed
Orth. fusion {#2,#4}	2×k	Not all nodes included
Orth. fusion {#1,#5,#3}	2×k	Not all nodes included
Single radar	K	Not suitable for unconstrained HAR

TABLE 2 Overview of the fusion methods for distributed radar and the relative feature vector length together with the author's objective notes

TABLE 3 Notation for metric definitions in Section 5

$y, A/\hat{y}, \hat{A}$	Ground truth/predicted label/area
$\bar{y}/\bar{\hat{y}}$	Mean ground truth/mean predicted samples
$s/\hat{s}$	Ground truth/predicted block
$(\cdot)_p, P$	Sample, set of samples
$(\cdot)^{(c)}$	Class index (later neglected for readability)
$tp; tn; fp; fn$	True/false positive/negative rate

$$I^{(c)}(\hat{y}_p, y_p) = \begin{cases} 0 \leftarrow \hat{y}_p = y_p \\ 1 \leftarrow \hat{y}_p \neq y_p \end{cases} \quad (16)$$

The number of misclassifications is provided by:

$$M^{(c)} = \frac{1}{P} \sum_{p=1}^{P} I^{(c)}(\hat{y}_p, y_p) \quad (17)$$

with the resulting accuracy being equal to:

$$A^{(c)} = 1 - M^{(c)} \quad (18)$$

When evaluating classification performances, the accuracy metric does not capture inequalities of false negative (fn) and false positive (fp), and does not account for imbalanced datasets. This may lead to overlook drops in performance [30].

5.2 | F_β score (F_1) with precision, recall and specificity

The F_β score (F_1) provides a more concise metric accounting for fn and fp imbalances, and consists of a combination of precision and recall. Together with precision, recall, and F_β score (F_1), the specificity can be also computed as:

$$\text{precision} = \frac{tp}{tp + fp} \quad (19a)$$

$$TPR = \text{sensitivity} = \text{recall} = \frac{tp}{tp + fn} \quad (19b)$$

$$TNR = \text{specificity} = \frac{tn}{tn + fp} \quad (19c)$$

$$F_\beta^{(c)} = (1 + \beta^2) \times \frac{precision \times recall}{\beta^2 \cdot precision + recall} \quad (19d)$$

In the formula for the F_β score, precision and recall are evenly treated if $\beta = 1$, known as the F_1 score. Otherwise, the formula favours precision if $\beta > 1$ [31].

5.3 | Dice index

The Dice similarity index (also named as Sørensen-Dice coefficient) normalises the length of the vector labels $\hat{y}$ and ground truth y, and divides them by the total number of non-zero entries. Multiplication by a factor of 2 scales the measurement range between [0, 1] with 1 indicating label vectors identical to the ground truth [32]. It is expressed as:

$$Dice^{(c)} = 2 \times \frac{\left|\hat{A} \cap A\right|}{|\hat{A}| + |A|} = \frac{2tp}{2tp + fp + fn} \quad (20)$$

5.4 | Jaccard index

The Jaccard index or Tanimoto coefficient defines the intersection divided by the union of two label vectors.

$$Jac^{(c)} = \frac{\left|\hat{A} \cap A\right|}{\hat{A}| + |A| - \left|\hat{A} \cap A\right|} = \frac{tp}{tp + fp + fn} \quad (21)$$

It should be noted that the denominator denotes the union as $|\hat{A}| + |A| - \left|\hat{A} \cap A\right| = \left|\hat{A} \cup A\right|$.

5.5 | Consecutive block detection (CBD)

This proposed metric considers and penalises interruptions and misalignments in the sequence of predicted samples, $\hat{y}_p$, with respect to the corresponding ground truth labels, y_p. To the best of our knowledge, this aspect is not always well considered in the literature when evaluating radar-based HAR for continuous activities.

5.5.1 | Unweighted consecutive block detection (CBD)

Firstly, the individual ground truth blocks and the prediction blocks are counted as shown for the ground truth in Equation (22) and the predictions in Equation (23), respectively, as:

$$s(y_p) = \frac{1}{2} \sum_{p=2}^{P-1} \sqrt{(y_p - y_{p-1})^2} \qquad (22)$$

and

$$\hat{s}(\hat{y}_p) = \frac{1}{2} \sum_{p=2}^{P-1} \sqrt{\left(\hat{y}(\hat{y}|y_p=1)_p - \hat{y}(\hat{y}|y_p=1)_{p-1}\right)^2} \qquad (23)$$

with the counter index, $(\cdot)_p$, in the sequence of a total length, P. The ratio of blocks, as shown in Figure 8, is computed as:

$$Ed^{(c)} = \frac{s(y)}{\hat{s}(\hat{y})} \qquad (24)$$

with the range between [0, 1], where 1 indicates the same number of blocks found within the ground truth sequence of a class and the prediction. It should be noted that block length differences are not considered in Equations (22)–(24), and this can affect the result with a solution provided in the following Section 5.5.2.

5.5.2 | Weighted consecutive block detection (CBD)

Due to the aforementioned concern of the block length differences, a corresponding penalty factor can be computed as:

$$w = \sqrt{\frac{\left|\hat{A} \cap A\right|}{|A|}} \qquad (25)$$

with the numerator indicating the intersection between the ground truth and the prediction, $\left|\hat{A} \cap A\right|$, over the ground truth, $|A|$. The non-linearity impact of the weight, $w = \sqrt{(\cdot)}$, is introduced to minimise penalisation on small misalignments. The weighted Consecutive Block Detection (CBD) is then computed by combining Equations (24) and (25):

$$Ed_w^{(c)} = Ed \cdot w = \frac{s(y)}{\hat{s}(\hat{y})} \cdot \sqrt{\frac{\left|\hat{A} \cap A\right|}{|A|}} \qquad (26)$$

5.6 | Intersection-Over-Union (IoU)

IoU is another metric that penalises interruptions and misalignments in the sequences of predictions. It is a known technique for evaluating camera-based object detection algorithms and is under certain conditions equivalent to the Jaccard
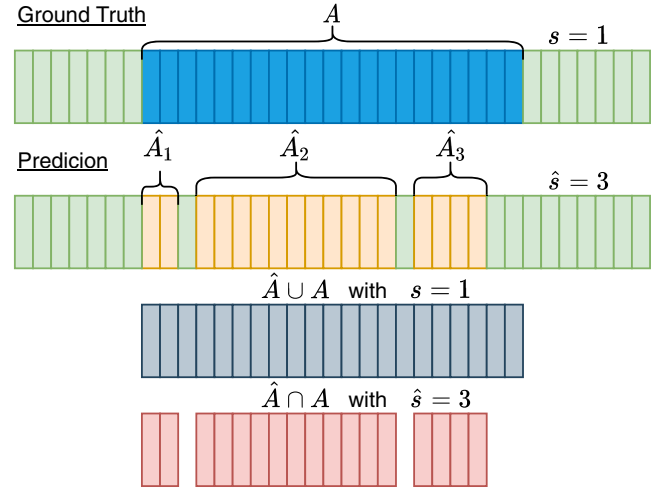
FIGURE 8 The intersection and union sequences are demonstrated and are used for CBD and IoU, as well as, the penalisation term of IoU as, $H(\hat{s}, s) = 3/4$, for this example

index. It defines the similarity on the bounding boxes [33], which are generally uninterrupted entities in vision-based detection methods. A modified expression can account for interruptions in vectors of labels as:

$$H(\hat{s}, s) = 1 - \left(\frac{2 \cdot \hat{s}}{\hat{s} + s} - 1\right)^2 \qquad (27)$$

with s and $\hat{s}$ the concatenated sequence blocks for ground truth and predictions, respectively:

$$\begin{aligned} \text{IoU}^{(c)} &= Jac \cdot H(\hat{s}, s) \\ &= \left(\frac{\hat{A} \cap A}{\hat{A} \cup A}\right) \cdot \left(1 - \left(\frac{2 \cdot \hat{s}}{\hat{s} + s} - 1\right)^2\right) \end{aligned} \qquad (28)$$

Equation (28) penalises interrupted sequences even if the predictions are broadly correct and aligned with the ground truth [34].

5.7 | Correlation index or Matthews Correlation Coefficient (MCC)

The correlation index or Matthews Correlation Coefficient (MCC) is less commonly used for classification performance assessment. It is based on the Pearson's Linear Correlation Coefficient, typically used to find linear similarities between vectors. This can also be used for sequence classification as:

$$R^{(c)}(\hat{y}, y) = \frac{\sum_{p=1}^{P} (y_p - \bar{y})(\hat{y}_p - \bar{\hat{y}})}{\sqrt{\sum_{p=1}^{P} (y_p - \bar{y})^2 \sum_{p=1}^{P} (\hat{y}_p - \bar{\hat{y}})^2}}, R \in \mathbb{R}; [-1, 1] \qquad (29)$$

with $\bar{\hat{y}}$ and $\bar{y}$ the means of the ground truth and prediction vector, respectively. Alternatively, the equation can be expressed as:

$$R^{(c)} = \frac{tp \cdot tn + fp \cdot fn}{\sqrt{(tp+fp) \cdot (tp+fn) \cdot (tn+fp) \cdot (tn+fn)}} \quad (30)$$

and is known as MCC [35]. It should be noted that $R(\hat{y}, y) = -1$ is equivalent to perfectly misclassified and $R(\hat{y}, y) = 1$ to perfectly classified sequences, respectively, whereas $R(\hat{y}, y) = 0$ is the expected value from an unbiased 'coin tossing classifier' for a balanced dataset.

6 | HYPERPARAMETERS OPTIMISATION

The following Recurrent Neural Networks (RNN) are considered as classifiers in this work:

- GRU   Gated Recurrent Unit
- Bi-GRU   Bidirectional Gated Recurrent Unit
- LSTM   Long Short-Term Memory
- Bi-LSTM   Bidirectional Long Short-Term Memory

Their key hyperparameters are optimised using the experiment manager of MATLAB with Bayesian optimisation, as this is key to boost performances in classification tasks [36]. The following hyperparameters are considered: number of epochs, hidden units, mini batch size, initial Learning Rate (LR), LR drop factor, and the L2 regularisation factor [37], with the range over optimisation performed and the determined optimal parameter for each network presented in Table 4. The optimisation algorithm ran over a cycle of 50 trials for each RNN network. Hyperparameter optimisation is evaluated using the following metric:

$$\max_{\forall_{\text{Opti.Parameter}}} \left(\frac{F_{1\,macro}}{(1 + std(F_{1\,macro}))^2} \right) \quad (31)$$

with $F_{1\,macro}$ being the macro F_1 score, that is the mean of the F_1 score for the individual classes. The denominator, $(1 + std(F_{1\,macro}))^2$, ensures that the fluctuation across the classes is minimised, while the numerator strives towards a high macro F_1 score. Furthermore, the three optimisers Stochastic Gradient Descent Momentum (SGDM), Root Mean Square Propagation (RMSProp), and Adaptive Moment Estimation (ADAM) were tested, with the best results achieved using ADAM across all networks.

7 | EXPERIMENTAL RESULTS

In this section, the classification results for the four selected RNNs and proposed data fusion methods for multiple radar nodes are reported.

7.1 | RNN results utilising sensor fusion approaches

The section presents the results achieved by using the previously discussed radar fusion methods, namely: signal fusion, feature fusion with all nodes, weighted radar selection, orthogonal radar fusion of node {#2,#4} and of node {#1,#5, #3}. These results are compared with those obtained when using only one single radar classification with a Bi-LSTM classifier, implemented with the default parameters presented in Table 4. The results are reported in Table 5 for L1PO test and visualised in Figure 9.

For the different metrics, the diversity of the shown fusion methods highlights superior performances of one method over others by analysing the standard deviation across the tested methods. In this regard, the results show higher dispersion (standard deviation) for metrics other than the more conventional accuracy or F1 score, such as the Dice, Jaccard, and Correlation (MCC) indexes as block-based metrics such as CBD and IoU.

Across these metrics, the best suitable fusion method is the incoherent signal fusion, followed by feature fusion and orthogonal radar fusion of nodes {#1,#5,#3}, which show almost equivalent performance. The weighted radar selection and orthogonal radar fusion of nodes {#2,#4} perform slightly worse than the other fusion methods. All the presented radar data fusion approaches improve performances in contrast to single radar classification across all metrics, as also visible by inspecting Figure 9.

Focussing on the block-based evaluation metrics, such as the CBD, weighted CBD, and the IoU, the low-level signal fusion approach appears to be clearly the best performing approach with more than +7% performance gain over the mean across all methods by analysing IoU and even more than +8% by evaluating the weighted CBD. Performance-wise, feature fusion approaches follow with just about +1% better performance versus the mean across all fusion methods. Specifically, feature fusion, weighted radar selection, and orthogonal radar fusion of nodes {#1,#5,#3} achieve an average IoU of about 0.05 (50%), whereas single radar classification achieves only 0.436.

7.2 | RNN results with hyperparameter tuning

The section presents performance results achieved with different RNNs as classifiers when incoherent signal fusion is applied, as

T A B L E 4 Range of hyperparameters and their best performing values after Bayesian optimisation

Descr.	Range	Bi-LSTM	LSTM	Bi-GRU	GRU	Default
Optimiser	ADAM		Best performing: ADAM			–/–
	RMSprop					
	SGDM					
Epochs	[10, 50]		Flatten growth after 45			–/–
Hidden unit	[3, 200]	168	60	199	189	–/–
Mini batch	[32, 128]		Converging → 128			128
Initial LR	$[10^{-4}, 0.1]$	0.0076	0.0029	0.0015	0.0025	0.0010
LR drop fac.	[0.1, 0.5]	0.2484	0.1361	0.1778	0.1039	0.1000
L2 reg.	$[10^{-7}, 1]$	0.0001	0.0001	0.0003	0.0039	0.0001

Abbreviations: ADAM, Adaptive Moment Estimation; Bi-GRU, Bidirectional Gated Recurrent Unit; Bi-LSTM, Bidirectional Long Short-Term Memory; RMSProp, Root Mean Square Propagation; SGDM, Stochastic Gradient Descent Momentum.

T A B L E 5 Classification performance metrics for the proposed radar data fusion methods using a Bi-LSTM classifier with the default hyperparameters from Table 4

Method/Metric	Accuracy	F1 score	TPR	TNR	Dice	Jaccard	CBD	Weighted CBD	IoU	Corr.
Signal fusion Bi-LSTM	0.933	0.84	0.693	0.95	0.722	0.583	0.829	0.686	0.566	0.68
Feature fusion Bi-LSTM	0.924	0.814	0.63	0.941	0.669	0.526	0.774	0.615	0.5	0.625
Weighted radar Bi-LSTM s.t. $\mathbf{w}(\eta)$	0.92	0.803	0.623	0.94	0.654	0.51	0.725	0.573	0.463	0.604
Weighted radar Bi-LSTM s.t. $\mathbf{w}(\Upsilon)$	0.916	0.793	0.61	0.938	0.634	0.493	0.746	0.589	0.477	0.584
Weighted radar Bi-LSTM s.t. $\mathbf{w}(\cos(\phi_{nx}))$	0.887	0.719	0.5	0.914	0.514	0.383	0.582	0.42	0.293	0.436
Orthogonal radar one-fifth&3 Bi-LSTM	0.925	0.815	0.625	0.942	0.665	0.523	0.782	0.618	0.5	0.626
Orthogonal radar 2&4 Bi-LSTM	0.919	0.796	0.603	0.939	0.635	0.495	0.729	0.569	0.463	0.589
Single radar average Bi-LSTM	0.909	0.773	0.566	0.93	0.599	0.456	0.687	0.521	0.436	0.543
Mean across all fusion methods	0.922	0.807	0.623	0.94	0.657	0.516	0.754	0.597	0.488	0.611
Standard deviation across all fusion methods	0.008	0.023	0.042	0.006	0.041	0.042	0.051	0.056	0.045	0.045

Note: For comparison, the performance for average classification using only 1 single radar is shown. The bold methods, 'signal-', 'feature-', 'weighted radar s.t. $\mathbf{w}(\eta)$', both 'orthogonal fusion' concepts, and the 'single radar average' are used to compute the mean and the standard deviation shown in the last two rows of this table.

Abbreviations: Bi-LSTM, Bidirectional Long Short-Term Memory; CBD, Consecutive block detection; IoU, Intersection Over Union; TNR, true negative rate; TPR, true positive rate.

this was shown to be the best fusion approach. Table 6 shows the results by using default parameters in the rows indicated as [·]*, whereas the bold rows marked as [·]$^{\#}$ show the attained results after hyperparameter tuning. In all cases, the L1PO approach meant that one test sequence was excluded while training the classifier, and this is repeated and averaged across all participants. Moreover, these results are also summarised in the bar graphs in Figure 10, focussing on performance improvements due to hyperparameter tuning, as shown in Figure 10c.

The accuracy metric provides again inconclusive evaluation for the macro results across the tested network architectures, which is indicated in low standard deviation of around 0.012; thus the performance improvement after hyperparameter tuning is not too evident. In contrast, for the F_1 score about 0.036 (3%) is obtained, specifically for the Bi-GRU network that gained in performance with the optimised hyperparameters from Table 4. The next metrics, the Dice and Jaccard index, provide an even better evidence that the hyperparameter tuning

of the RNN yields to performance improvement of approximately 5% for the Bi-GRU, whereas the Bi-LSTM dropped slightly. It should be noted that the little decrease in performance is only given for the sample based evaluation metrics, such as accuracy, F_1 score, Jaccard, Dice, or correlation (MCC) indices and can be neglected. Instead, the block-based evaluation metrics as CBD, weighted CBD, and IoU provide a more reliable evaluation of sequence detection since interruptions and outliers towards other classes will be taken into account. The results achieved after hyperparameter tuning have increased compared to the default values for all network architectures tested, with the most significant improvement for the Bi-GRU. Nonetheless, the Bi-GRU and the Bi-LSTM provide nearly comparable results, as shown for the sequence evaluation metrics, that is, IoU or both CBD metrics. Even more, the weighted CBD and IoU provide the highest standard deviation indicating superior evaluation capabilities compared to, that is, simply accuracy evaluation.
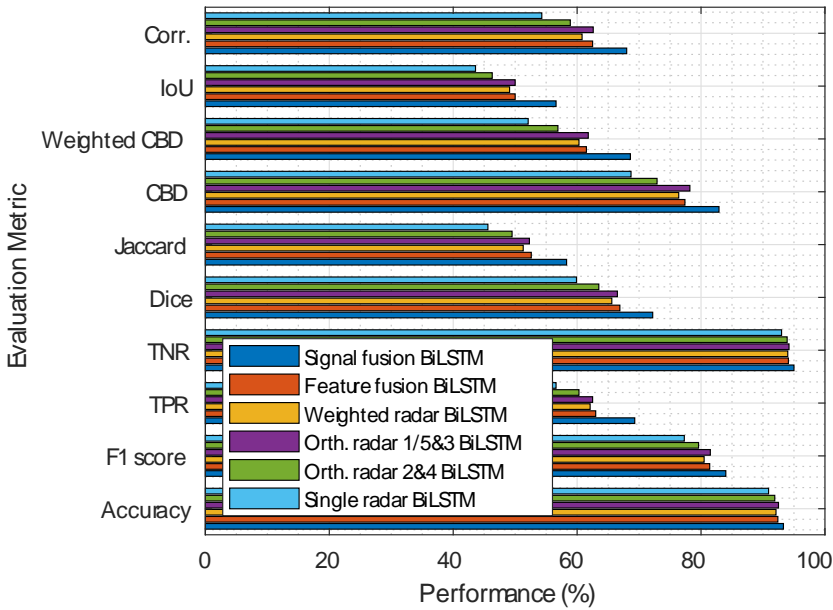
FIGURE 9 Overview of classification performance metrics for all investigated radar data fusion methods in L1PO test with Bi-LSTM network with default hyperparameters as classifier

TABLE 6 Classification performance metrics for different considered RNN classifiers using signal level fusion

Method/Metric	Accuracy	F1 score	TPR	TNR	Dice	Jaccard	CBD	Weighted CBD	IoU	Corr.
Signal fusion GRU *	(0.91)	(0.77)	(0.579)	(0.931)	(0.595)	(0.463)	(0.7)	(0.531)	(0.365)	(0.537)
Signal fusion GRU#	**0.909**	**0.778**	**0.59**	**0.932**	**0.612**	**0.473**	**0.699**	**0.539**	**0.406**	**0.553**
Signal fusion LSTM*	(0.91)	(0.772)	(0.58)	(0.932)	(0.595)	(0.462)	(0.713)	(0.538)	(0.379)	(0.539)
Signal fusion LSTM#	**0.91**	**0.769**	**0.58**	**0.931**	**0.592**	**0.464**	**0.709**	**0.55**	**0.395**	**0.534**
Signal fusion Bi-GRU*	(0.924)	(0.817)	(0.656)	(0.944)	(0.681)	(0.539)	(0.773)	(0.629)	(0.515)	(0.632)
Signal fusion Bi-GRU#	**0.933**	**0.844**	**0.705**	**0.95**	**0.731**	**0.592**	**0.826**	**0.695**	**0.579**	**0.688**
Signal fusion Bi-LSTM*	(0.933)	(0.843)	(0.708)	(0.951)	(0.731)	(0.592)	(0.823)	(0.69)	(0.562)	(0.686)
Signal fusion Bi-LSTM#	**0.931**	**0.836**	**0.697**	**0.949**	**0.718**	**0.577**	**0.829**	**0.692**	**0.569**	**0.671**
Single radar average Bi-LSTM*	(0.909)	(0.773)	(0.566)	(0.93)	(0.599)	(0.456)	(0.687)	(0.521)	(0.436)	(0.543)
Single radar average Bi-LSTM#	**0.909**	**0.773**	**0.566**	**0.93**	**0.599**	**0.456**	**0.687**	**0.521**	**0.436**	**0.543**
Mean across all*	(0.917)	(0.795)	(0.618)	(0.938)	(0.64)	(0.502)	(0.739)	(0.582)	(0.451)	(0.588)
Mean across all#	**0.92**	**0.806**	**0.638**	**0.94**	**0.661**	**0.522**	**0.76**	**0.611**	**0.488**	**0.609**
Standard deviation across all*	(0.011)	(0.033)	(0.062)	(0.009)	(0.063)	(0.061)	(0.057)	(0.074)	(0.085)	(0.068)
Standard deviation across all#	**0.012**	**0.036**	**0.066**	**0.01**	**0.066**	**0.064**	**0.068**	**0.083**	**0.085**	**0.073**

Note: Each network's results are compared before [·]*, and after [·]# hyperparameter tuning. For comparison, the performance for average classification using only 1 single radar is shown. The mean and standard deviation are shown in the last four rows of this table across all architectures.

Abbreviations: CBD, Consecutive block detection; IoU, Intersection Over Union; RNN, Recurrent Neural Network; TNR, true negative rate; TPR, true positive rate.

7.3 | Class evaluation using spider diagram

The previous Sections 7.1 and 7.2 discussed the most suitable multi-sensor fusion methods and performance improvements by using various RNNs with optimised hyperparameters. However, the discussion and conclusions were drawn on the 'macro' results across all classes of interest. In this section, class specific results ('micro' results) for the individual classes are discussed for incoherent signal fusion using uni- and bidirectional RNN architectures, namely, the GRU, LSTM, Bi-

GRU and Bi-LSTM. These results are reported in Figure 11 and discussed in this section for each considered performance metric.

7.3.1 | Accuracy

Accuracy appears to be very high for all considered classifiers (macro accuracy >90%). However, it should be noted that accuracy as a single evaluation does not capture the

(a)

(b)

(c)
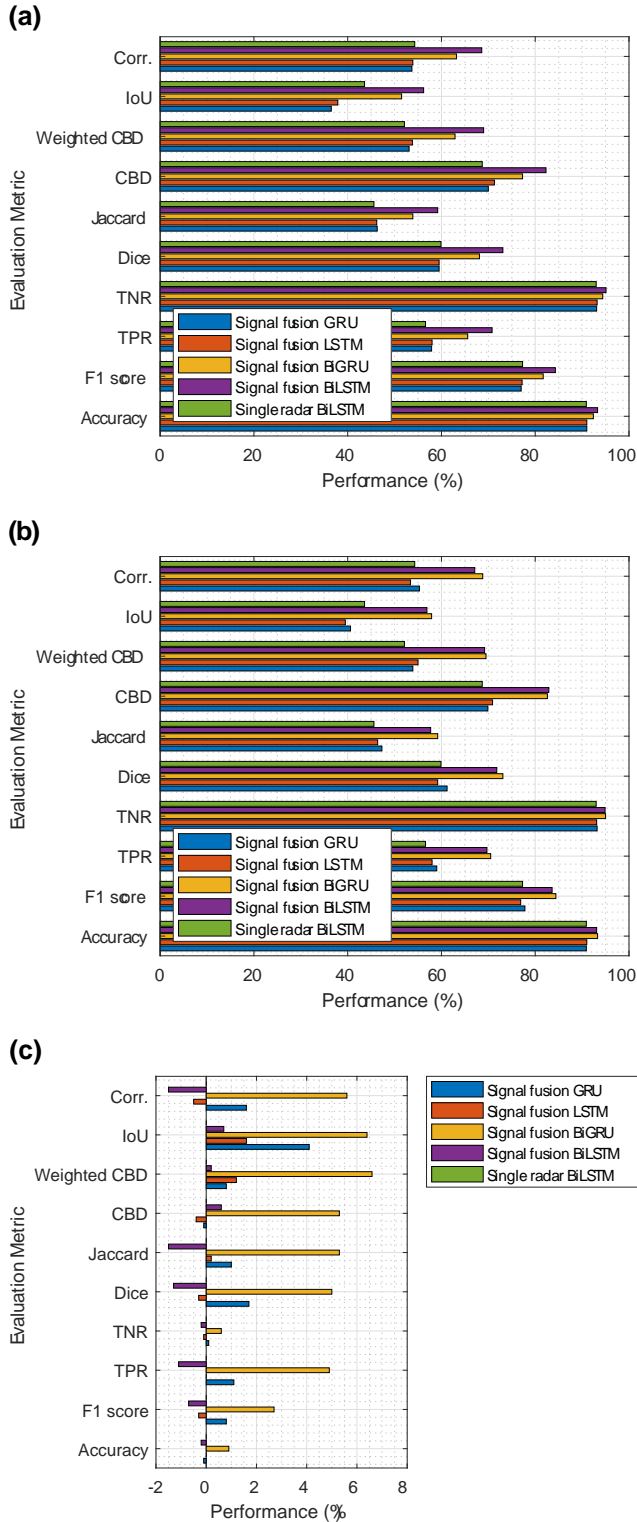
FIGURE 10 Overview of performance classification metrics for different RNNs without and with hyperparameter tuning (from Table 6). Signal level fusion is considered in L1PO test. (a) Performance metrics without hyperparameter tuning. (b) Performance metrics with hyperparameter tuning. (c) Performance improvement (positive values refer to performance improvements after hyperparameter tuning)

performances of datasets with class imbalances (e.g., fewer samples of 'falling' compared to other classes). By visual inspection of the other metrics apart from accuracy, the GRU classifier in Figure 11a suffers in detecting the falling class in the lower left corner of each spider diagram, whereas both bidirectional classifiers, Bi-GRU (Figure 11b) and Bi-LSTM (Figure 11d), outperform their unidirectional counterparts. This will be widely overseen using just the accuracy metric.

7.3.2 | F_1 score, TPR, TNR

Evaluating true positive rate (TPR) (sensitivity or recall) and true negative rate (TNR) (specificity) on their own is less effective than using the F_1 score, as this can provide a better global view on performances for each specific class. An average of the F_1 score across all classes, macro F_1 score, is also possible. For this case study, the performance differences between individual classes increase to approximately 12% for signal fusion using Bi-LSTM, specifically referring to the 'translation' (91.6%) and 'standing up from falling' (78.9%) activity, as shown in Figure 11d. However, the drastic difference can be seen for the unidirectional RNNs, for example, the LSTM where the differences between 'translation' and 'standing up from falling' rises to more than 20%.

7.3.3 | Dice index

The Dice index is a more rigorous metric than the prior shown metrics of accuracy or F1 score. Here, for example, 'standing up from falling' degrades to 58.8% (F1 score: 78.9%) and 'translation' to 91.7% (F1 score: 91.6%) for the Bi-LSTM. Nonetheless, the bidirectional classifiers perform almost equally high across the individual classes, whereas, LSTM (Figure 11c) shows a break down by detecting 'falling', which, in general, will be overseen by using the previous more conventional metrics.

7.3.4 | Jaccard index

The Jaccard index is related to the Dice index, with performance always lower than the Dice index except at their extrema [19]. In fact, this metric will report even lower performances, that is, 'standing up from falling' degrades to 41.6% (Dice index: 58.8%). In fact, the metric is the most rigorous evaluation method for individual sample detection, apart from block sequence based detection, such as the IoU.

7.3.5 | Consecutive block detection (CBD)

The CBD operates differently than the previously shown metrics. Here, interruption ratios of prediction label blocks
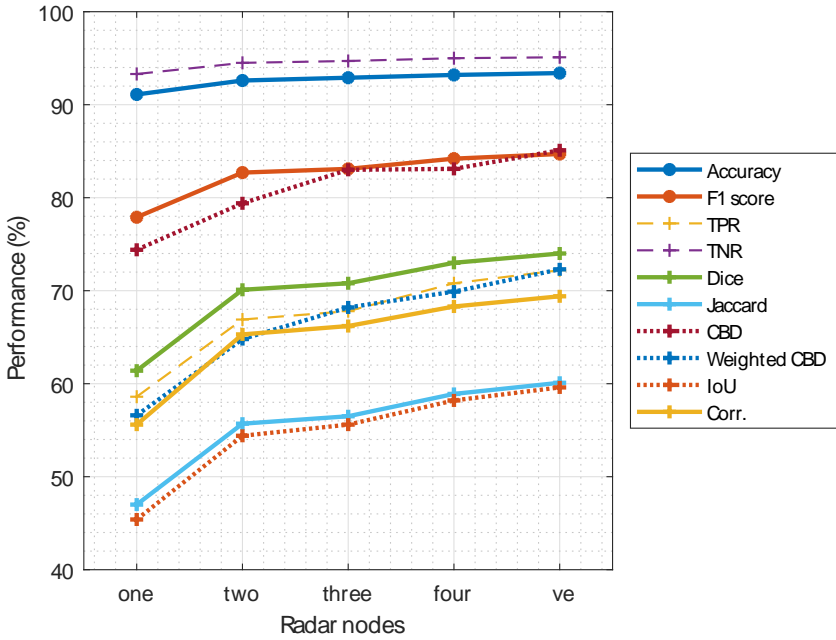
FIGURE 11 Performance metrics as a function of selected number of radar nodes (note that incoherent signal fusion is used when two or more nodes are selected). Training/testing is performed by using L1PO and Bi-LSTM classifier

to ground truth label blocks have an impact, see Equation (24). In a particular example drawn, the activities 'stationary' and 'in-place' provide the best and worst classification results, respectively, for incoherent fusion using the Bi-LSTM classifier (see Figure 11d) with 88.0% and 78.3%. Furthermore, both bidirectional RNNs (Bi-GRU and Bi-LSTM) show good results that are equally distributed across all classes.

However, the simple CBD defined in Section 5.5.1 accounts for the number of detected blocks only, while differences in block length are neglected. The weighted CBD instead, as defined in Section 5.5.2, considers the detection length differences of the predictions versus the ground truth labels. Specifically (see Figure 11d for Bi-LSTM), the best and worst class become the activity 'translation' and 'falling' with 83.5% and 56.6%, respectively. Correspondingly, the bidirectional RNNs outperform unidirectional RNNs with significant improvement provided for minority classes as 'falling'.

7.3.6 | Intersection-Over-Union (IoU)

The IoU metric is an alternative metric accounting for detected block ratio and the block length differences. The IoU is the most extreme evaluation metric for our dataset since it is a product of the Jaccard index (hard metric on its own) multiplied with a block detection term [34]. Thus the activity 'standing up from falling' degrades to 41.1% (Jaccard index: 41.6%), and the 'translation' activity to 82.4% (Jaccard index: 84.7%) as the best class. Likewise, using unidirectional classifiers (LSTM and GRU) has adverse effects specifically on minority classes as 'standing up from falling', 'falling', and 'inplace' activities.

7.3.7 | Correlation index or MCC

The MCC is rather challenging to compare with the previously introduced metrics due to its diverse definition. An advantage of this metric (MCC) is a distinct indication when classifiers provide outputs resulting in $R(\hat{y}, y) < 0$. Such results immediately indicate a mismatch between the ground truth and prediction samples. As a side note, it will be mentioned, a coin tossing classifier would converge towards the limit $R(\hat{y}, y) = 0$ for a balanced dataset. For the presented dataset, the activities 'falling' and 'translation' provide the worst and best results with 57.7% and 83.1%, respectively. Similarly to the metrics introduced before, the unidirectional classifiers (LSTM and GRU) suffer for minority classes as previously inspected, which can be seen for the macro results in Figure 10 and for the micro results in Figure 11.

7.4 | Performance w.r.t. radar nodes

A Monte Carlo simulation was set up to investigate the effect on the proposed classification performance metrics of selecting a different number of radar nodes. The Monte Carlo approach selects different nodes out of the possible combinations of 2, 3, and 4 nodes respectively, but in all cases the verification is performed with the L1P0 approach. As shown in Figure 12, the lowest performance for each metric is achieved by using just one single node. This is related to the degraded performance of one single node for unfavourable trajectories of the target, and is compensated by considering more nodes, which leads to an increase of all metrics.

It is important to notice that such increase as a function of number of nodes is less pronounced for sample-based metrics
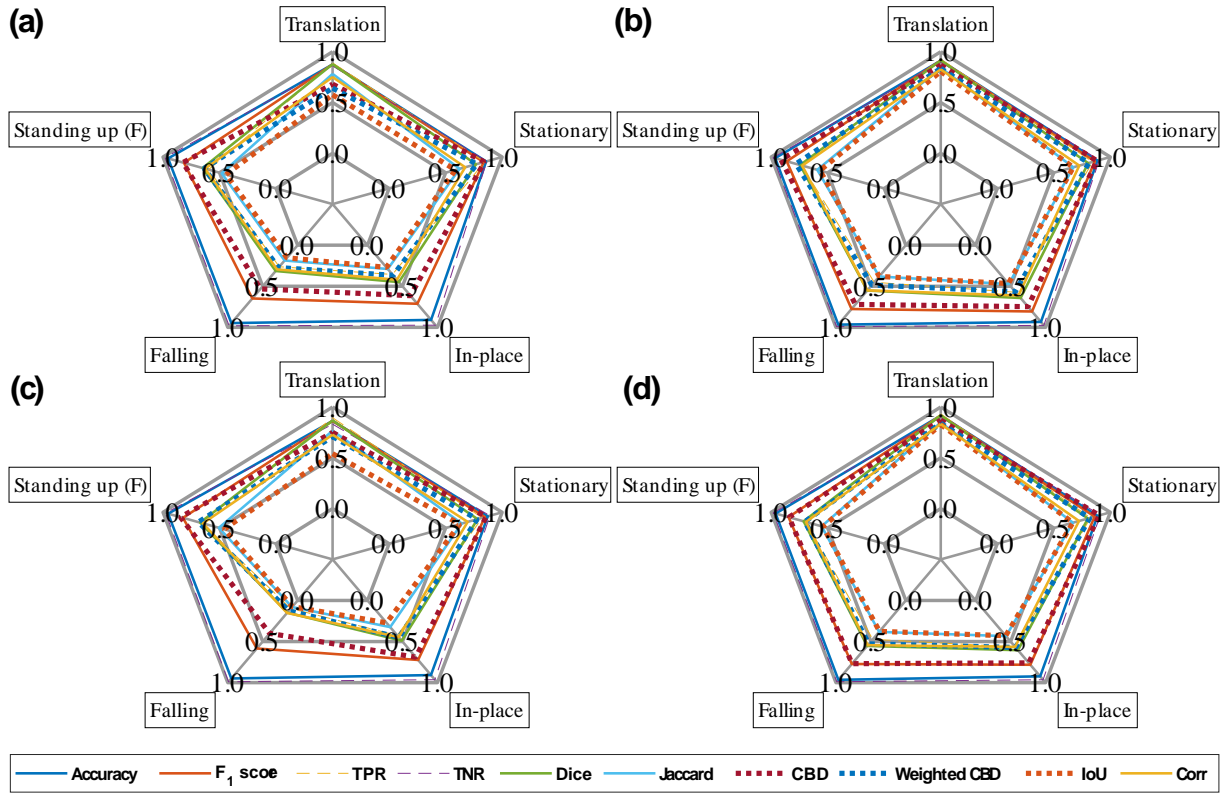
FIGURE 12 Spider graphs with performance metrics addressing the continuous human activity sequence evaluation per class. (a) GRU. (b) Bi-GRU. (c) LSTM. (d) Bi-LSTM

such as accuracy and F1-score (i.e., the curve in Figure 12 is flatter), but more evident when block sequence-based metrics such as the CBD, weighted CBD, or IoU are used. In this latter case the increase in performances with the number of radar nodes is more noticeable. From this analysis it appears that a higher number of radar nodes achieves better performances when using incoherent signal fusion and the Bi-LSTM classifier, specifically for the metrics proposed for continuous sequential activities.

7.5 | Discussion overview on performance metrics

In the analysis of the results, pros and cons of each evaluation metric were discussed when applied to the different RNN architectures and radar data fusion methods. Some considerations from this initial analysis follow:

■ *For balanced (equally-distributed) data evaluation*: The conventional accuracy metric provides satisfactory results, even if it does not describe where mistakes (e.g., missed detections or false alarms) occur for a given class. For that, precision/recall or their combination into F_1 score are more suitable.
■ *For imbalanced (skewed) data evaluation*: The F_1 score becomes a more sophisticated metric than plain accuracy and is widely used. Its importance becomes crucial as accuracy

can significantly overestimate performances, as seen in our case study. The same applies to the Dice and Jaccard indices. Both, Dice and Jaccard, are the most drastic sample-based metrics as seen in the results from the considered dataset. Also, the correlation index or MCC accounts for imbalances in the dataset and it is widely used in the medical domain [34].
■ *For evaluating continuous sequences of activities*: The prior metrics suffer by evaluating continuous sequences of activities with unconstrained and seamless transitions. Therefore, the proposed CBD is preferable for such cases with its modification as weighted CBD, and the IoU. These metrics can account for outliers (i.e., misalignments and interruptions) in the prediction label vector and are well suited for HAR based on continuous sequences. When used at the output of RNN classifiers, they can directly assess their sequential output predictions and penalise instabilities/interruptions that propagate errors within the network's memory cells.

8 | CONCLUSION

The paper proposed a variety of approaches for data fusion in a network of five distributed radar sensors in the context of human activities classification. The implementation of signal level fusion applied on Range-Time (RT) maps has been researched and is compared to state-of-the-art methods. Within the fusion process, the selection of radar nodes based

on a weighting function that accounts for the target location and velocity/trajectory has also been investigated. RNN-based classification algorithms, namely GRU, LSTM, Bi-GRU, and Bi-LSTM, were used to process the resulting μD spectrograms derived from the fusion process.

The proposed techniques are evaluated on an experimental dataset with 15 participants and 9 activities, combined into 5 macro-classes. Notably, the dataset contains continuous sequences of activities performed in random locations and with arbitrary and unconstrained trajectories and unfavourable aspect angles to the radar sensors. New evaluation metrics are proposed and compared to account for the specific nature of continuous activities in radar-based HAR, such as the presence of misalignments and interruptions (e.g., outliers).

The results show the need for metrics other than plain accuracy or precision/recall when evaluating continuous HAR, especially using recurrent networks for classification assessing performance at a fine scale, that is, in continuous sequences of time bins analysed. Specifically, evaluation metrics that account for outliers in the prediction vector (i.e., misalignments, interruptions, and fluctuations) such as weighted CBD and IoU appear to provide a more comprehensive performance evaluation than simple accuracy. For example, while IoU shows around 20% difference between imperfect and reasonable performing classifiers, conventional accuracy evaluation gives only 2% discrepancy, hence a too coarse assessment. It is demonstrated that classifiers with bidirectional capabilities provide superior performance, especially for classes with few samples in the imbalanced dataset used for evaluation. This imbalance is to some extent typical in continuous, realistic activities. For example, there will be more walking samples than in-place activities, whereas critical activities (e.g., falls) will be generally rare. For the radar data fusion, the incoherent signal level fusion of the RT data from each node appeared to outperform other methods, with the best results provided by using the full set of radars in the used network.

Future work will extend the proposed techniques and assess the evaluation metrics in more realistic home environments and scenarios, such as those including multiple targets (e.g., couple of people, or person plus pet). This could be approached by implementing a multi-target tracker with modified hypotheses for a dynamic radar selection in such scenarios of multiple targets in home environments. Moreover, the usage of diverse radar sensors in the network could be considered (e.g., adding V/W-band MIMO nodes).

CONFLICT OF INTEREST
The authors declare no conflicts of interest to report for this paper.

DATA AVAILABILITY STATEMENT
The dataset is available under the DOI: https://doi.org/10.4121/16691500.v2 at the reference: Guendel, Ronny Gerhard; Unterhorst, Matteo; Fioranelli, Francesco; Yarovoy, Alexander (2021): Dataset of continuous human activities performed in arbitrary directions collected with a distributed radar network of five nodes. 4TU.ResearchData. Dataset. https://doi.org/10.4121/16691500.v2.

ORCID
Ronny G. Guendel https://orcid.org/0000-0002-7824-4318
Francesco Fioranelli https://orcid.org/0000-0001-8254-8093

REFERENCES

1. Fu, B., et al.: Sensing technology for human activity recognition: a comprehensive survey. IEEE Access. 8, 83791–83820 (2020)
2. Shah, S.A., Fioranelli, F.: RF sensing technologies for assisted daily living in healthcare: a comprehensive review. IEEE Aerosp. Electron. Syst. Mag. 34(11), 26–44 (2019)
3. Gurbuz, S.Z., Amin, M.G.: Radar-based human-motion recognition with deep learning: promising applications for indoor monitoring. IEEE Signal Process. Mag. 36(4), 16–28 (2019)
4. Gorji, A., et al.: A multi-radar architecture for human activity recognition in indoor kitchen environments. In: 2021 IEEE Radar Conference (RadarConf21), pp. 1–6 (2021)
5. Yang, Y., et al.: Omnidirectional motion classification with monostatic radar system using micro-Doppler signatures. IEEE Trans Geosci Remote Sens. 58(5), 3574–3587 (2020)
6. Li, H., et al.: Sequential human gait classification with distributed radar sensor fusion. IEEE Sens. J. 21(6), 7590–7603 (2021)
7. Khan, U.M., et al.: A deep learning framework using passive WiFi sensing for respiration monitoring. In: GLOBECOM 2017 - 2017 IEEE Global Communications Conference, pp. 1–6 (2017)
8. Alnujaim, I., et al.: Synthesis of micro-Doppler signatures of human activities from different aspect angles using generative adversarial networks. IEEE Access. 9, 46422–46429 (2021)
9. Amin, M.G., Guendel, R.G.: Radar classifications of consecutive and contiguous human gross-motor activities. IET Radar Sonar Navig. 14(9), 1417–1429 (2020)
10. Zhu, S., et al.: Continuous human activity recognition with distributed radar sensor networks and CNN-RNN architectures. IEEE Transactions on Geoscience and Remote Sensing (Submitted to IEEE TGRS), 1–12 (2021)
11. Pham, M., Yang, D., Sheng, W.: A sensor fusion approach to indoor human localization based on environmental and wearable sensors. IEEE Trans. Autom. Sci. Eng. 16(1), 339–350 (2019)
12. Ritchie, M., et al.: Multistatic micro-Doppler radar feature extraction for classification of unloaded/loaded micro-drones. IET Radar Sonar Navig. 11(1), 116–124 (2017)
13. Clemente, C., et al. (eds.): Radar Countermeasures for Unmanned Aerial Vehicles. Radar, Sonar and Navigation. Institution of Engineering and Technology (2021)
14. Fioranelli, F., Ritchie, M., Griffiths, H.: Centroid features for classification of armed/unarmed multiple personnel using multistatic human micro-Doppler. IET Radar Sonar Navig. 10(8), 1702–1710 (2016)
15. Fioranelli, F., Ritchie, M., Griffiths, H.: Classification of unarmed/armed personnel using the netrad multistatic radar for micro-Doppler and singular value decomposition features. IEEE Geosci. Remote Sens. Lett. 12(9), 1933–1937 (2015)
16. Zhou, J., Shi, J.: RFID localization algorithms and applications-a review. J. Intell. Manuf. 20, 695–707 (2009)
17. Wang, M., Zhang, Y.D., Cui, G.: Human motion recognition exploiting radar with stacked recurrent neural network. Digit. Signal Process. 87, 125–131 (2019)
18. Ding, C., et al.: Continuous human motion recognition with a dynamic range-Doppler trajectory method based on FMCW radar. IEEE Trans. Geosci. Remote Sens. 57(9), 6821–6831 (2019)
19. Guendel, R.G., Fioranelli, F., Yarovoy, A.: Evaluation metrics for continuous human activity classification using distributed radar networks.

In: IEEE Radar Conference 2022 (RadarConf22), pp. 6. New York (2022)

20. Guendel, R.G., et al. Dataset of continuous human activities performed in arbitrary directions collected with a distributed radar network of five nodes. 4TU.ResearchData, (2021) [Online]. https://doi.org/10.4121/16691500. v2. https://data.4tu.nl/articles/dataset/Dataset_of_continuous_human_activities_performed_in_arbitrary_directions_collected_with_a_distributed_radar_network_of_five_nodes/16691500/2

21. Erol, B., Gurbuz, S.Z., Amin, M.G.: Motion classification using kinematically sifted ACGAN-synthesized radar micro-Doppler signatures. IEEE Trans. Aerosp. Electron Syst. 56(4), 3197–3213 (2020)

22. Guendel, R.G., et al.: Continuous human activity recognition for arbitrary directions with distributed radars. In: 2021 IEEE Radar Conference (RadarConf21), pp. 6 (2021)

23. Belhadi, Z., Fergani, L.: Fingerprinting methods for RFID tag indoor localization. In: 2014 International Conference on Multimedia Computing and Systems (ICMCS), pp. 717–722 (2014)

24. Tran Quang, V., Ngo Quynh, T., Jo, M.: A lateration-localizing algorithm for energy-efficient target tracking in wireless sensor networks. Ad-Hoc Sens. Wirel. Netw. 34(1-4), 191–220 (2016)

25. Brookner, E.: Tracking and Kalman Filtering Made Easy. Wiley (1998)

26. Guendel, R.G., Fioranelli, F., Yarovoy, A.: Phase-based classification for arm gesture and gross-motor activities using histogram of oriented gradients. IEEE Sens. J. 21(6), 7918–7927 (2021)

27. Petroff, A.: A practical, high performance ultra-wideband radar platform. In: 2012 IEEE Radar Conference, pp. 880–884 (2012)

28. Li, H., et al.: Bi-LSTM network for multimodal continuous human activity recognition and fall detection. IEEE Sens. J. 20(3), 1191–1201 (2020)

29. Holm, W.A. (ed.): Principles of Modern Radar: Basic principles. Radar, Sonar and Navigation. Institution of Engineering and Technology (2010)

30. Watt, J., Borhani, R., Katsaggelos, A.K.: Machine Learning Refined: Foundations, Algorithms, and Applications, 2nd edn. Cambridge University Press (2020)

31. Sokolova, M., Japkowicz, N., Szpakowicz, S.: Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation. In: Sattar, A., Kang, B.h. (eds.) AI 2006: Advances in Artificial Intelligence, pp. 1015–1021. Springer Berlin Heidelberg, Berlin (2006)

32. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. The MIT Press, Cambridge (1999)

33. Rezatofighi, H., et al.: Generalized intersection over union: a metric and a loss for bounding box regression. IEEE Computer Society. 658–666 (2019)

34. Chen, P., et al.: Multi-view real-time human motion recognition based on ensemble learning. IEEE Sens. J. 21(18), 20335–20347 (2021)

35. Chicco, D., Jurman, G.: The advantages of the Matthews correlation coefficient (MCC) over f1 score and accuracy in binary classification evaluation. BMC Genom. 21(1), 6 (2020)

36. Agrawal, T.: Hyperparameter Optimization in Machine Learning: Make Your Machine Learning and Deep Learning Models More Efficient. Apress (2020)

37. Snoek, J., Larochelle, H., Adams, R.P.: Practical Bayesian optimization of machine learning algorithms. In: Pereira, F., et al. (eds.) Advances in Neural Information Processing Systems, vol. 25, pp. 12. Curran Associates, Inc. (2012)