MULTIPLE TARGET TRACKING AND HUMAN ACTIVITY RECOGNITION BASED ON THE IR-UWB RADAR SENSOR NETWORKS

SIMIN ZHU

MULTIPLE TARGET TRACKING AND HUMAN ACTIVITY RECOGNITION BASED ON THE IR-UWB RADAR SENSOR NETWORKS

DISSERTATION

to obtain the degree of Master of Science
in Electrical Engineering
at Delft University of Technology,
to be defended publicly on Wednesday October 13, 2021 at 01:00 PM

by

SIMIN ZHU

born in Changsha, China.

This thesis has been approved by the

Supervisor:	Prof. DSc. A. Yarovoy
Daily supervisor:	Dr. F. Fioranelli

Thesis committee:

Prof. DSc. A. Yarovoy,	MS3 TU Delft
Dr. F. Fioranelli,	MS3 TU Delft
Dr. J. Dauwels	CAS TU Delft

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

We all have two lives,
the second begins when we realize we only have one.

Confucius

ABSTRACT

The Objective

Integrating the multiple target tracking (MTT) system with the human activity recognition (HAR) system is the ultimate goal for many radar-based applications. For example, in indoor monitoring scenario, it is important to know the target's position as well as the related activities performed by that target. However, the literature often treats the joint tracking and recognition problem independently due to its complexity. As a consequence, the dependencies and requirements between the tracking and recognition system are neglected. The main objective of this thesis work is to build two connectable systems for tracking and classifying human activities with radar sensors and address the problems caused by the mutual requirements through system designs.

The Mutual Requirements

Conventionally, most multitarget tracking systems only focus on tracking point-like targets. However, an extended target like human beings may occupy several range bins in recognition tasks due to their close distance to the radar sensor. Moreover, the characteristics of human activity exhibit temporal dependencies. To exploit this for classification, the tracking system is required to be able to associate the extracted activity features across time. Not to mention the details of how to extract these features when multiple targets are presented.

As for the recognition tasks, traditional systems tend to simplify the problem by constraining targets' moving trajectories. Moreover, targets are often required to perform different activities independently so that the training dataset can be separated easily in post-processing. However, in multitarget tracking applications, targets are allowed to move freely inside the measurement area. Needless to say that the performed activities may be continuous with seamless inter-activity transitions.

The Methods

To address these requirements, this thesis work proposed an MTT system and a HAR system based on a distributed IR-UWB radar sensor network (RSN).

The proposed MTT system uses a decentralized tracking architecture. Due to the introduction of the detection fusion center, it is able to fuse a target's detection information from different radar nodes and track multiple extended targets in the Cartesian plane. Besides developing the main functionality, two critical problems are are investigated and addressed by the proposed solutions. More specifically, the first problem relates to the measurement merging effect due to the use of clustering algorithm, and the second

problem relates to the false alarms introduced by the fusion center due to the association uncertainty. It has been noticed that these two problems may influence the tracking stability and recognition continuity.

The proposed HAR system is built using deep learning tools. To extract the spatial and temporal feature patterns from the input data, the proposed system is composed of a hybrid neural network architecture that uses the convolutional neural network (CNN) and recurrent neural network (RNN) simultaneously. The main advantage of the proposed recognition system over the others is it provides an end-to-end solution for data fusion and human activity classification. Moreover, it handles the recognition problem under a more realistic experimental setting as targets can have arbitrary moving directions and unconstrained inter-activity transitions.

The Results

For the proposed MTT system, the evaluation process is done based on simulated radar data. The result shows that not only the proposed system can track multiple extended targets but also mitigate the target merging problem and suppress the introduced false alarms. Moreover, the proposed system has been applied to process experimental radar data to extract the Doppler information from a moving target. The output of the MTT system has exactly the same format as the input data of the proposed HAR system, which enables a direct connection between the tracking and classification system.

The proposed HAR system is evaluated using experimental radar data collected from 14 participants performing nine types of activities in a series of unconstrained trajectories. The result shows that the proposed system is able to achieve a maximal classification accuracy of 89.88% on an unseen target for nine-class classification. Moreover, due to the combination of hybrid neural network architecture and the weight sharing technique, the proposed system has a more light-weighted neural network compared to its counterparts in the literature.

Due to the limited time, a thorough investigation of the combination of the proposed MTT and HAR system is left for future investigation. Nevertheless, this work provides a foundation for their combination and shows improvements in both tracking and classification compared to the state-of-the-art.

CONTENT

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

ADF Assumed Density Filtering. 16

ADLs Activities of Daily Living. 71, 77

Bi-GRU Bidirectional Gated Recurrent Unit. 90

Bi-LSTM Bidirectional Long-short Term Memory. 90

Bi-RNN Bidirectional Recurrent Neural Network. 73, 90

BN Batch Normalization. 84

CA-CFAR Cell-averaging Constant False Alarm Rate. 13

CNN Convolutional Neural Network. viii, 73, 80, 95

CTC Connectionist Temporal Classification. 113

DBSCAN Density-based Spatial Clustering of Applications with Noise. 16, 30, 44

EC European Commission. 9

EKF Extended Kalman Filter. 17

EM Expectation-maximization. 16, 30, 44

FCC Federal Communications Commission. 9

FCNN Fully Connected Neural Network. 80, 101

FFT Fast Fourier Transform. 82

FIR Finite Impulse Response. 12

FISST Finite Set Statistic. 18

FoV Field of View. 10

GGIW Gamma Gaussian Inverse-Wishart. 35

GNN Global Nearest Neighbor. 17

GOSPA Generalized Optimal Sub-pattern Assignment. 5, 20, 40, 43

GPS Global Positioning System. 3

GRU Gated Recurrent Unit. 88, 95

HAR Human Activity Recognition. vii, 4, 71, 77, 95, 117

IIR Infinite iImpulse Response. 12

IMU Inertial Measurement Unit. 3

IR Impulse Radio. 3, 77

JPDAF Joint Probabilistic Data-association Filtering. 17

KF Kalman Filter. 16

L1PO Leave-One-Person-Out. 74, 92, 95

LS Least-square. 14, 31, 51, 66

LSTM Long-short Term Memory. 87

MBM Multi-Bernoulli Mixture. 18

MBPP Multi-Bernoulli Point Process. 23, 35

MDS Micro-Doppler Signature. 72

MHT Multiple Hypothesis Tracking. 17

MTI Moving Target Indication. 12

MTT Multiple Target Tracking. vii, 4, 9, 21, 43, 65, 117

OMAT Optimal MAss Transfer. 40

OS-CFAR Ordered-statistic Constant False Alarm Rate. 13, 59

OSPA Optimal Sub-Pattern Assignment. 40

PDF Probability Density Function. 23

PHD Probability Hypothesis Density. 18, 37

PMBM Poisson Multi-Bernoulli Mixture. 18, 35

PPP Poisson Point Process. 23, 35, 44

ReLU Rectified Linear Activation Function. 85

RFS Random Finite Set. 18

RMS Root Mean Square. 56

RMSE Root Mean Square Error. 10, 40

RNN Recurrent Neural Network. viii, 73, 80, 95

RSN Radar Sensor Network. vii, 5, 11, 21, 43, 65, 74, 77, 98, 118

RX Receiver. 11

SMC Sequential Monte Carlo. 17

SNR Signal-to-noise Ratio. 10

STFT Short-time Fourier Transform. 38, 72, 80

SVM Support Vector Machine. 72

TX Ttransmitter. 11

UKF Unscented Kalman Filter. 17

UWB Ultra-wideband. 3, 77

I

INTRODUCTION

1

OVERVIEW OF THE THESIS

This chapter serves as an introduction. Section 1.1 presents the background stories and motivations. Section 1.2 concludes the main contributions. In Section 1.3, the structure of this thesis work is provided.

1.1. BACKGROUND AND MOTIVATION

Over recent decades, radar systems have become increasingly attractive to many fields of life. For indoor monitoring applications, the radar system can be used to track a target's location [1], monitor its vital signs [2], and classify the performed activities [3]. Compared to other sensing systems based on sensors such as the inertial measurement unit (IMU) and global positioning system (GPS), the radar sensor does not require installing any devices on the target. Moreover, when compared with other non-contact sensors, for example, the infrared and LiDAR sensor, the radar sensor is more cost-effective for massive distributions and robust against various weather, temperature, and light conditions. Last but not least, the radar sensor can help alleviate the user's potential privacy concerns. This feature is extremely important for deploying the sensing system in a sensitive environment like the washroom and bedroom.

This thesis work focuses on monitoring human activity for indoor applications. Among all the possible radar sensors, the monostatic impulse radio (IR) ultra-wideband (UWB) [4] radar is used. The IR-UWB radar is well-suited for indoor monitoring. Thanks to the large operational bandwidth, the IR-UWB radar can provide extraordinarily high range resolution and localization capability. Furthermore, it is robust against the multipath and fading effect, and it offers high data rates over short distances. Moreover, by coherently processing the range bin that contains the target [5], it is possible to use the IR-UWB radar to extract the target's micro-Doppler signatures [6]. Besides, considering

aspects related to radar system deployment, the IR-UWB radar has the advantage of low power consumption, compact installation size, and affordable prices.

Regarding the IR-UWB radar-based human activity monitoring, there are mainly two research directions in the literature. One investigates the problems in multiple target tracking (MTT) [7, 8], and the other focuses on addressing the difficulties for human activity recognition (HAR) [9, 10]. It is evident that joint tracking and activity recognition is the ultimate goal for indoor monitoring applications. Frequently, studying one problem but leaving another aside may lead to unrealistic experimental settings due to the neglected mutual requirements.

Therefore, the main objective of this thesis work is to build not only two connectable systems, one for multitarget tracking and another for activity classification, but also address the problems caused by the mutual requirements through system designs. This work serves as a solid foundation for further integration between the tracking and classification pipeline, whose thorough exploration is left for future work.

To have a clear image of the motivations behind the system design, the following summarizes the recognized requirements and the related research questions for each system:

1. **The Multiple Target Tracking System**
   Most MTT systems focus on tracking point-like targets [11–13]. However, targets in recognition tasks are often close to the radar sensor, which leads to the so-called extended target tracking [14]. Besides, conventional MTT tasks only estimate the current location of the presented targets. For recognition tasks based on Doppler signatures, it requires the MTT system to be able to associate all the history estimates of every target. Lastly, the output of the MTT system should have the same format as the input of the HAR system. This requires adding an additional feature extraction block on the top of the MTT system [15].

2. **The Human Activity Recognition System**
   Traditional HAR systems investigate the recognition tasks in a constrained fashion. For example, the moving direction of the presented target is limited [16], or the target is only allowed to perform different activities independently [17]. However, this is not true for MTT tasks, in which targets are allowed to move freely inside the measurement area. Needless to say that the performed activities are continuous with natural inter-activity transitions. Therefore, it is required that the HAR system to be able to handle the unfavorable aspect angle cause by arbitrary moving trajectories [18] and classify continuous human activities [19].

1.2. MAIN CONTRIBUTIONS

Considering all the abovementioned requirements, this thesis proposed an MTT system and a HAR system for joint tracking and activity classification. The proposed systems are

built based on a distributed IR-UWB radar sensor network (RSN) [20]. The RSN can provide a multi-perspective view on the presented targets, which helps improve the tracking robustness [21] and classification accuracy [22]. As a broad summary, the following contributions are achieved in the proposed works:

1. The proposed MTT system is capable of fusing the detection information from different radar nodes and tracking multiple extended targets. Moreover, it can extract the Doppler signature from the moving target. Besides, the output of the MTT system has the same format as the input of the HAR system, which enables integration between the tracking and classification system.

2. A simulator based on the distributed IR-UWB RSN is developed to generate the multitarget radar data for testing the functionality of the MTT system. The performance of the MTT system is measured by the generalized optimal sub-pattern assignment (GOSPA) [23] metric.

3. Two problems that were less explored in the RSN tracking literature are recognized and addressed in this work. The first problem relates to the measurement merging effect due to the use of the clustering algorithm for extended target tracking. The second problem is caused by the association uncertainties in the detection fusion center.

4. The proposed HAR system provides an end-to-end solution for data fusion and activity classification. It can automatically extract the spatial-temporal features from the input data and classify continuous human activities.

5. The performance of the HAR system is measured using experimental radar data sampled from 14 participants. To use the dataset efficiently, the leave-one-person-out method and the K-fold cross-validation method are implemented.

6. To take advantage of the RSN, three neural network-based data fusion methods are proposed. Among them, the halfway fusion (or feature fusion) method shows the most promising performance in terms of classification accuracy and model complexity.

7. To the best of my knowledge, this is the first work that investigates the feasibility and addresses the problems in joint tracking and activity recognition for radar-based indoor monitoring. Thus, part of this work is expected to contribute to a journal paper in IEEE Sensors.

1.3. THESIS OUTLINE

The rest of the thesis is structured as the follows, with two parts for the investigation of the MTT and HAR system separately:

1. **Part II: Multiple Target Tracking System**
   This part presents the proposed multiple target tracking system. It contains four chapters:

   (a) **Chapter 1**: This chapter provides the information of research challenges, recent advances in the literature, and an outline of the main contributions.

   (b) **Chapter 2**: This chapter reveals the design details of the proposed tracking system. Specifically, it discusses the functionality of all the signal processing components used in the proposed system.

   (c) **Chapter 3**: This chapter presents the evaluation result of the proposed system based on simulated radar data.

   (d) **Chapter 4**: This chapter gives an overview of the conclusions and possible future directions for further investigation.

2. **Part III: Human Activity Recognition System**
   This part introduces the proposed human activity recognition system. Similar to the previous part, it also contains four chapters:

   (a) **Chapter 1**: This chapter points out the related works which help the audience have a better understanding of the existing challenges and motivations behind the human activity classification tasks.

   (b) **Chapter 2**: This chapter clarifies the design details of the human activity recognition system. Specifically, it discusses the functionality of all the types of neural networks used in the proposed system.

   (c) **Chapter 3**: This chapter reports the evaluation results of the proposed system based on experimental radar dataset sample from 14 participants.

   (d) **Chapter 4**: This chapter highlights the conclusions and future improvements with regard to the proposed recognition system.

3. **Part IV: Closing Remarks**
   This part serves as an overview for the whole thesis work. It contains two chapters:

   (a) **Chapter 1**: This chapter provides an overall summary of the contributions made in this thesis work. Moreover, interesting future directions for joint tracking and classification are presented.

   (b) **Chapter 2**: This chapter presents the acknowledgment.

II

Part II - Multiple Target Tracking System

1

Related Work

In this chapter, the related work with regards to the IR-UWB radar-based multiple target tracking (MTT) system is presented. Specifically, Section 1.1 discusses the research challenges in the tracking system. Recent advances that address these challenges are studied in Section 1.2. Finally, Section 1.3 summarizes the main contributions in the proposed MTT system.

1.1. Research Challenges

Over the past decades, IR-UWB radar systems have gained massive attention in applications such as human detection [24, 25], human breathing and heartbeat detection [26, 27], and through-wall detection [28, 29].

Despite the previous efforts, indoor human activity monitoring [30] remains a demanding task as it requires multidisciplinary knowledge like target detection and clutter suppression [31, 32], MTT [33], and activity classification [9].

In this part of the thesis, the main objective is to investigate the problems in target detection, clutter suppression, and MTT using the IR-UWB radars. However, it is also important to remember that the proposed system should be compatible with the proposed classifier. Having said that, the main research challenges are presented as follows:

1. **Tracking Robustness**
   Robust tracking of multiple targets in a complex environment faces several critical issues. First of all, the power spectral density of the transmitted waveform must comply with the power mask imposed by, e.g., the Federal Communications Commission (FCC) in the USA or the European Commission (EC) in the Europe. As a

consequence, targets at distance are hard to be detected due to the low signal-to-noise ratio (SNR). Other than that, the shadowing effect also influences the target's detectability. In indoor environment, shadowing effect may happen between different targets (a.k.a mutual shadowing) or between the target and furniture (a.k.a occlusion).

2. **High False Alarm Rate**
   False alarms can exist almost at every signal processing component in the whole signal processing pipeline. False alarm happens when a processing component falsely reports a clutter or noise as a target. In addition, a signal processing component can also produce false alarms due to internal causes, e.g., the detection fusion center in the sensor network [34]. False alarms can significantly increase the computational costs of the tracking and classification system. However, no algorithm can accurately reject all unwanted signals and only report the actual target echoes. Thus, false alarm suppression mechanisms in the MTT system are often related to complicated system designs and stepwise reduction strategies.

3. **Extended Target**
   Most MTT algorithms are designed for tracking point targets. However, due to the high-range resolution of the UWB radar, a target appears to be extended as it can occupy several range bins. Tracking all the detection points is impractical since the computational capacity is limited. In addition, tracking the extended target becomes problematic if multiple targets are presented and closely spaced. In that case, the detections that originated from different targets may be overlapped, and it is unknown how to separate the merged measurements. In the UWB radar-based sensor network, the target merging problem has a much higher incidence rate in the 1D plane than the Cartesian plane. This is because even spatially well-separated targets can be merged in the range dimension as long as they have equal distance to the same sensor.

4. **Tracking Multiple Targets**
   Tracking multiple human targets is the main function of the proposed MTT system. However, it is a demanding task due to the following facts:

   (a) *Imperfect Measurement*: The radar measurement is impaired by noise and clutter. Thus, it requires the tracker is robust against system and measurement disturbance.

   (b) *Target Uncertainty*: The origin of the target is unknown. Moreover, the number of targets inside the radar field of view (FoV) is changing over time.

   (c) *Data Association*: The detection-to-track association is unknown. Maintaining all the hypotheses will lead to the number of hypotheses going up drastically over time.

   (d) *State Estimation*: Estimating the hidden state of a random process using the Bayesian estimator under the root mean square error (RMSE) criterion requires us to find a way to represent and evaluate a multidimensional posterior probability density function analytically. This is especially hard if the

density function is non-Gaussian and the hidden state is multivariate (a.k.a curse of dimensionality).

(e) *Feature Extraction*: The joint tracking and activity recognition task requires the proposed MTT system to output not only the estimated kinematic states of the target but also the extracted micro-Doppler signatures. In the case when a radar sensor network (RSN) is used, to extract the Doppler signatures, the central 2D tracker needs to know the target's 1D location at each radar node. Moreover, exploring the temporal dependencies of the human activity in the extracted Doppler signatures requires the multitarget tracker to be able to associate all the previously extracted features for every detected target.

(f) *System Evaluation*: Conventional evaluation metrics penalize the tracking performance on the localization error. However, the missed detections and false alarms are as crucial as the tracking accuracy. This is because the missed detection can lead to discontinuities in the classifier, while the false alarm may waste the limited computational resource.

1.2. RECENT ADVANCES

In this section, the existing solutions proposed to solve the abovementioned challenges are studied.

1.2.1. SOLUTIONS FOR ROBUST TRACKING

Chang et al. [8] presented one of the first signal processing frameworks for tracking human targets using the IR-UWB radar. Since the applied radar works in a monostatic mode, i.e., it contains one transmitter (TX) and one receiver (RX), the proposed framework can only track targets in the 1D plane. Two experiments were conducted to test the system performance for tracking one and two human targets, respectively. Despite the fact that the experimental setups were idealized, e.g., targets are well-separated in range and azimuth angle, the work verified the feasibility of using IR-UWB radar for tracking human targets.

Later work [35] improved the previous work by using a more advanced tracking algorithm to track a variable number of human (and non-human) targets. However, both works only focus on tracking targets in the range domain (i.e., 1D tracking). To localize targets in the 2D plane (or the Cartesian plane), Nguyen et al. [36] presented a tracking pipeline that uses two monostatic IR-UWB radar sensors. The experiment was conducted in an indoor environment which makes the tracking problem more challenging. The result indicates that the detection and tracking performance can be further improved by increasing the number of radar nodes in the RSN.

An RSN usually contains multiple radar sensors, and these radars are deployed according to a designed topology. Paolini et al. [20] investigated the impact of radar deployment geometry and the number of radar nodes on the area coverage, localization

accuracy, and the necessary transmission power through simulations. The result shows that using an RSN is advantageous for improving the localization accuracy and system robustness. Moreover, the result indicates that the localization precision is maximized if the radar nodes are uniformly placed on a circumference concentric with the measurement area.

Another advantage of using an RSN to track multiple targets is its robustness against the mutual shadowing effect [21]. The mutual shadowing effect may happen when one target is located at the place close to the radar antennas. In that case, only a negligible part of the transmitted power can propagate through the nearby target to other targets behind. Depending on the magnitude of the signal attenuation, this effect can be categorized as partial shadowing or total shadowing. For the partial shadowing effect, a solution that improves the target's detectability can be found in [37]. For the total shadowing effect, as suggested by the author in [21], it can be efficiently mitigated by using an RSN.

In summary, the RSN is the suggested solution from the literature that can significantly improve the system robustness for MTT. Moreover, due to the use of multiple radar nodes, it also enhances the system's fault tolerance against sensor failure.

1.2.2. SOLUTIONS FOR FALSE ALARM SUPPRESSION

Due to the ubiquitous nature of the noise, a false alarm may occur at any point in the MTT system. Therefore, the strategies for false alarm suppression often require a complicated system design. In the following, the common solutions for false alarm suppression from the radar data end to the tracker end are reviewed:

1. **Removing Static Clutters from The Raw Data**
   The raw radar data contains not only the reflected signals from the target but also the noise, multipath, and clutter.

   One popular approach to removing irrelevant echoes in the raw radar data is the empty room method [32]. To suppress the clutter, this method subtracts the received signal with the data sequence, which is pre-recorded when no target is presented (i.e., in an empty room). Although this method is efficient and straightforward, it has two major defects. First, the empty room data is not always available, and it has to be updated regularly. Second, the false alarm may occur if a target obscures a clutter echo presented in the pre-recorded sequence [34].

   Other techniques like the moving target indication (MTI) can also be used to suppress the static clutters. Ash et al. [38] introduced three MTI solutions, including: (1) the background subtraction method, (2) the finite impulse response (FIR) filtering method, and (3) the infinite impulse response (IIR) filtering method.

2. **Reducing False Alarms using Adaptive Thresholding**
   Since the previous clutter suppression algorithms can not remove the noise and clutter perfectly, the received signal still contains residual clutters. Other than

that, some methods (e.g., the empty room method) may create additional clutters [34]. Therefore, the adaptive thresholding method is usually implemented after the static clutter rejection step.

The goal of the thresholding methods is to detect the potential targets presented in the received echoes. However, since the distribution of the target's measurement is unknown, and the noise process is not stationary, which may vary over range, time, and azimuth angle, it is necessary to adaptively estimate the threshold for every range cell under test.

Two well-known adaptive thresholding techniques are the cell-averaging constant false alarm rate (CA-CFAR) and the ordered-statistic constant false alarm rate (OS-CFAR) [39]. These two methods differ by the way they estimate the power level of the noise. In summary, the CA-CFAR has a low computation complexity, but it is not suitable for handling the scenarios when multiple targets are closely spaced because of the masking effect [40]. In contrast, the OS-CFAR can provide more robust detection results compared to the CA-CFAR in multitarget scenarios. However, the OS-CFAR is computationally expensive due to the sorting procedure.

Rather than only focusing on a constant false alarm rate, the works in [31, 41] emphasized the importance of considering both the false alarm rate and the miss detection rate while calculating the threshold. This is because targets at distance often show low SNR in the UWB radar-based tracking tasks, and a constant false alarm rate might lead to a high miss detection rate.

3. **Reducing False Alarms via Architecture Design**
In addition to the previously mentioned techniques, the number of false alarms can be further reduced through the architecture design in the tracking framework.

Conventional MTT pipeline uses a centralized tracking framework. As illustrated in [34], each receiving channel processes the raw radar data independently and generates a set of 1D detection points. Then, the fusion center [42] fuses the detections, which transforms the 1D range information to the 2D plane. Finally, these 2D detections are associated to different tracks in the object tracking filter.

In the centralized tracking framework, various false alarm reduction techniques can be used. For example, Valmori et al. [43] added a weight-based thresholding mechanism into the 1D and 2D detection clustering algorithm to remove the clusters with less number of detections (or low weights). This method uses the fact that the cluster associated with a human target usually has higher weights than those associated with noise or clutter.

In contrast to the conventional pipeline, He et al. [7] consider using a decentralized tracking architecture. The decentralized architecture adds a 1D tracker to each receiving processing channel. Each 1D tracker tracks the targets in the range domain, and it acts as a clutter filter.

For the advantages, first, the false alarm reduction techniques designed for the centralized tracking framework can still be used in the decentralized framework. Moreover, the decentralized tracking architecture is more robust against miss detection due to the two-stage tracking. It has been demonstrated in [44] that the

decentralized signal processing pipeline can significantly reduce clutter and multipath and achieves lower tracking error comparing to the centralized method.

4. **Reducing Introduced False Alarms**
Indubitably, the external/internal noise, multipath, and clutter may lead to false alarms. However, the internal processing unit can also produce false alarms, for example, the detection fusion center used in an RSN.

The main objective of the fusion center is to combine the 1D detections from different radar nodes and generate 2D detections using the trilateration algorithm [45]. Thus, the fusion center is a necessary component for tracking targets in the Cartesian plane. However, due to the detection-to-detection association uncertainties, the fusion center can generate three types of false alarms [34], including:

(a) *Type-1 False Alarm*: False alarms generated by combining residual noise or clutter in the set of 1D detections from different radar nodes, assuming no target is presented.

(b) *Type-2 False Alarm*: False alarms generated by combining detections of residual noise or clutter with the detections of targets.

(c) *Type-3 False Alarm*: False alarms generated by wrongly associating the detections with different targets.

There are two types of solutions in the literature used to mitigate the high false alarm rate, differing by the number of RXs each radar node has.

For each radar node equipped with one transmitting and two receiving antennas, Jovanoska et al. [46] proposed a target localization method that can analytically calculate a target's 2D position. To handle the detection-to-detection association uncertainties, the author defined an intersection threshold. Specifically, detections from the two RXs are associated if the absolute value of the range difference between the two detections satisfies the specified threshold. This method works because the two receiving antennas are closely placed for each radar node. Thus, the detections originated from the same target will appear at a similar range bin in both receivers. The result shows that the proposed method can not only reduce the computational cost but also helps mitigate the false alarm problem introduced by the fusion center.

For each radar node equipped with one transmitting and one receiving antenna, there is no direct temporal correlation that can be exploited. Chiani et al. [34] proposed a false alarm reduction technique that exploits the residual error, generated by the least-square (LS)-based trilateration algorithm, to decrease the false alarm rate. Specifically, every 2D detection has an associated LS error, and the detection is removed if the associated error exceeds a pre-defined threshold. Results have shown that the proposed false alarm reduction method can effectively reduce the false alarms generated by the fusion center. However, it is hard to set an ideal threshold, and a weak target may be filtered out using the proposed method [47].

In summary, it is challenging for an RSN to suppress the introduced false alarm if each radar node is equipped with only one TX and one RX. However, comparing

to the two-RX RSN, the one-RX RSN is more cost-effective and computationally efficient.

1.2.3. SOLUTIONS FOR HANDLING EXTENDED TARGET

For tracking extended targets, the detection partitioning method is usually used to avoid tracking all the detection points. This method separates the detection points into multiple sets according to a pre-defined data division scheme. Each scheme represents a global hypothesis of partitioning the detection points. For partitioning N detection points into a maximum K sets, there are K^N different global hypotheses can be used at each time step.

Therefore, to achieve the optimal detection partitioning, the likelihood of each global hypothesis should be considered in the MTT system [48]. However, the combinatorial partitioning of the detections is often computationally intractable. Therefore, the clustering algorithm, which is a machine learning technique used for unsupervised learning, can be applied as a heuristic to identify the most likely global partitioning hypothesis for the multitarget tracker [14, 49, 50].

One of the most widely used clustering algorithms is the K-means [51] algorithm. It is a centroid-based clustering algorithm. The value K is a pre-defined parameter representing the number of clusters you want to identify in your data. Given a set of detection points, the K-means algorithm first randomly selects K detection points, each of which represents the center point of a cluster. Then, each center point assigns labels to its nearby detection points to form a cluster. Lastly, the position of the center point is recalculated based on the members in the same cluster. The last two steps iterate until there are no significant changes in the positions of the K center points.

The K-means algorithm has a simple implementation, low computational cost, and it works well for the ball-shaped data distribution [52]. However, one disadvantage of the K-means algorithm is that the clustering performance is susceptible to the random initialization of the starting center point. Specifically, the K-means algorithm may have a high chance to converge to a local optimum during the iterations.

To address this issue, Arthur et al. proposed the K-means++ [53] algorithm. The main difference between the K-means++ and K-means algorithm is the initialization step. Instead of randomly selecting the initial cluster center, the K-means++ uses a careful seeding method that chooses K well-separated detections as the center points. The results have shown that the K-means++ algorithm converges faster and achieves better clustering performance than the original K-means algorithm.

The K-means and K-means++ algorithms use a hard assignment mechanism while separating the data, i.e., each detection point is only associated with one particular cluster. However, when two clusters are spatially close (e.g., the target merging scenario), it is not clear which detection point belongs to which cluster. In that case, the clustering algorithm may output an inaccurate estimation of the cluster center due to the hard

assignment of the detection points.

Dempster et al. [54] proposed a soft clustering algorithm, the expectation maximization (EM) algorithm, based on a Gaussian mixture model. The EM algorithm assumes the distribution of each cluster follows a Gaussian distribution. Thus, rather than assigning a specific cluster label to the detection point, the EM algorithm computes the probability of each detection point belonging to a particular cluster. Then, the cluster's center (or mean value) and shape (or standard deviation) are calculated based on the detection points and associated probability weight. It has been shown that the EM algorithm is more flexible in the shape of each cluster and achieves better clustering performance in target merging scenarios.

However, the above-discussed clustering algorithms require prior knowledge of knowing the exact number of the presented targets in the measurement area. This condition makes these clustering algorithms hard to be applied to general tracking applications since the number of presented targets is often unknown and time-varying.

Ester et al. proposed the density-based spatial clustering of applications with noise (DBSCAN) [55] algorithm. Unlike the previous algorithms, the DBSCAN algorithm does not require specifying the number of presented targets. Thus, it is more suitable for radar-based applications. Moreover, the DBSCAN algorithm can detect outliers automatically. Comparing to other clustering algorithms, the DBSCAN algorithm works well for identifying arbitrarily shaped clusters. However, the DBSCAN algorithm performs poorly when clusters have varying densities [56]. For example, when using an FMCW radar to track targets in the range-azimuth plane, due to the non-equidistant sampling density, the density of the detection points of a target may vary as the target's spatial position changes.

A summary of the abovementioned clustering algorithms is presented in Table 1.1.

1.2.4. SOLUTIONS FOR TRACKING MULTIPLE TARGETS

There are mainly two steps in MTT problems. The first step solves the detection-to-track association problem, where the associations between the existing tracks and the new detections are established. The second step handles the filtering problem, where the hidden states of each track are estimated based on the assigned data association hypothesis.

For the filtering problem, the Kalman filter [57] (KF)-based framework is the most common choice. The KF uses the assumed density filtering (ADF) technique. It assumes the system and measurement models are linear, the initial prior is Gaussian distributed, and the system disturbance and measurement noise are additive Gaussian. The KF utilizes the Gaussian distribution to parameterize the posterior density function. It provides a closed-form solution to the recursive Bayesian filtering [58].

However, suppose the system model and measurement model are nonlinear. In that

Techniques	Main advantage	Main disadvantage
K-means [51]	Simple implementation and low complexity.	Requires prior knowledge, sensitive to cluster initialization and outliers, hard assignment.
K-means++ [53]	Simple implementation, low complexity, and converges better and faster than the K-means algorithm.	Sensitive to the outliers, requires prior knowledge, hard assignment.
EM algorithm [54]	Soft assignment, robust to outliers, flexible cluster shape, simple implementation.	Requires prior knowledge, sensitive to cluster initialization.
DBSCAN [55]	Robust to outliers and cluster initialization, not require prior knowledge, works well for arbitrarily-shaped clusters.	Complicated implementation, not good for clusters with varying detection densities.

Table 1.1: A summary of the discussed clustering algorithms that can be used for partitioning the detection points.

case, one can consider using the extended Kalman filter (EKF) [59], which is based on the Taylor series approximation of the nonlinear model, or the unscented Kalman filter (UKF) [60], which is based on the unscented transformation. For the system that is neither linear nor Gaussian, the family of sequential Monte Carlo (SMC) [61] method, which is based on the random number generator and importance sampling, can be a powerful alternative.

For the detection-to-track association problem, it is optimal to keep track of all the possible detection-to-track hypotheses. However, this is not a practical solution due to the limited computation and storage resources. Therefore, many off-the-shelf solvers such as the Hungarian algorithm [62], auction algorithm [63], and Murty's algorithm [64] have been proposed to solve the optimization problem and search for the best data association scheme.

Most MTT algorithms follow the same two-step procedure mentioned above. However, these algorithms differ by the way they approximate the multitarget posteriors.

Conventional MTT algorithms such as the global nearest neighbor (GNN) [11], joint probabilistic data-association filtering (JPDAF) [12], and multiple hypothesis tracking (MHT) [13] have been widely used for UWB radar-based MTT [44, 65, 66].

The GNN algorithm is a simple greedy algorithm that approximates the multitarget posterior density using the most likely association hypothesis for each update. On the contrary, the JPDAF algorithm merges all the marginal posterior densities into a single Gaussian distribution. In this case, the hypothesis with a small likelihood will be preserved. The MHT algorithm is more advanced compared with the GNN and JPDAF al-

gorithm. It uses the Gaussian mixture distribution to propagate the uncertainties of different data association hypotheses, where each mixture distribution represents a global association hypothesis.

Although the conventional algorithm works well in MTT problems, the system uncertainties, e.g., track death, track birth, spawning, missed detections, and clutters, are not well-defined in the filtering process [67]. Therefore, a separate track management system [68] is usually required for track initialization, deletion, and clutter adaptation.

To overcome these disadvantages, MTT algorithms based on the random finite set (RFS) [69] theory have been developed. As defined in [70], the RFS-based algorithm describes the potential targets and received detections as RFSs rather than random vectors. It gives a unified framework to model all aspects of the MTT problem. Moreover, the finite set statistic (FISST) [71] provides an elegant formulation for the Bayesian recursion.

The probability hypothesis density (PHD) filter [72] is the first proposed and still widely used multitarget tracker based on the RFS theory. The PHD filter has a straightforward implementation and low computational cost. Moreover, it can handle the appearing and disappearing natures of the target. Due to its advantages, several variants of the PHD filter have been proposed [73–75]. Comparing with the conventional MTT algorithms, it has been shown in [76] that the PHD filter outperforms the MHT filter under high-clutter scenarios. However, the PHD filter is a suboptimal multitarget Bayesian filter as it only propagates the first-order statistical moment of the multitarget posterior.

To derive the exact closed-form solution for the true posterior density, the multitarget conjugate priors have been introduced [77]. Two well-known conjugate priors for the multitarget Bayesian filter are the multi-Bernoulli mixture (MBM) [78] and the Poisson multi-Bernoulli mixture (PMBM) [79]. The main distinction between the MBM filter and the PMBM filter lies in the used model for object birth. The authors in [80] have compared the performance of the PMBM filter and the MBM filter through simulations. The result shows that the PMBM filter outperforms the MBM filter in terms of efficiency and estimation error. Implementation and derivation of the MBM filter and the PMBM filter is referred to [78, 81].

A short summary of the discussed MTT algorithms is presented in Table 1.2.

1.3. MAIN CONTRIBUTIONS

Based on the research challenges and the recent solutions, the main contributions of the proposed MTT system can be summarized as follows:

1. **Multiple Extended Target Tracking and Feature Extraction**
   In this thesis work, an MTT system is proposed. The proposed system uses an IR-UWB radar-based sensor network to mitigate the shadowing effect and reduce the tracking error. To optimize the localization precision, the radar nodes in the pro-

Techniques	Main advantage	Main disadvantage
GNN [11]	Simple implementation and low computational cost.	The most likely association hypothesis is not guaranteed to be the optimum.
JPDAF [12]	More robust than GNN in low SNR scenarios.	Poor performance in complicated tracking scenarios.
MHT [13]	Good performance in low SNR and challenging tracking scenarios.	Hard to implement, high computational cost.
PHD filter [73]	Simple implementation, low computational cost, and good estimation performance.	Inaccurate cardinality estimation when tracking a large number of targets [75].
MBM filter [78]	Better tracking performance than the PHD filter [82].	High computational cost.
PMBM filter [81]	Lower computational cost and better tracking performance than the MBM filter [80].	Complicated implementation, need to use heuristic for labelling the target

Table 1.2: A summary of the discussed MTT algorithms.

posed RSN are uniformly placed on a circumference concentric with the measurement area. The proposed system is evaluated through both simulations and experimental radar data. The result shows that the proposed system can track multiple extended targets and extract their micro-Doppler information for the proposed classifier. To the best of our knowledge, most IR-UWB radar-based MTT systems (e.g., [7, 34, 83]) only focus on tracking multiple targets. This is the first work that considers how the feature extraction procedure can be conducted while tracking, it paved the way for joint tracking and classification using UWB radars.

2. **False Alarm Reduction**
   One of the major issues in the conventional MTT system is the high false alarm rate. To suppress the false alarms progressively, a decentralized signal processing architecture is used in the proposed MTT system. Moreover, a tracking-aided fusion center is proposed to improve the conventional approach [34, 43] in reducing the introduced false alarms. Except the performance advantage, the proposed fusion center does not require a pre-defined global threshold, hence it is more robust in different tracking scenarios.

3. **Target Merging Problem**
   Another significant issue that was rarely explored in the literature of the UWB radar-based tracking is the extended target merging problem. The target merging problem can lead to a high miss detection rate during tracking. Moreover, it is much easier to happen if each radar sensor can only offer 1D information of the target. To address this issue, the subpartitioning method [50, 84] is used. This method was originally used for handling laser range sensor-based tracking prob-

lems. In this thesis work, the subpartitioning method is extended and applied to the decentralized tracking architecture to prevent the 1D and 2D target merging problems.

4. **Evaluation Metric**
Conventionally, localization error is the mainstream evaluation metric used to evaluate the MTT system. However, penalizing the system performance on the missed targets and the inaccurate estimates has equal importance as calculating the localization error. To assess the MTT system from different aspects, the generalized optimal sub-pattern assignment (GOSPA) [23] metric is used. The GOSPA metric decomposes the measured total errors into three parts, i.e., localization error, miss detection error, and false alarm error. It gives an intuitive way for us to analyze error sources of the MTT system. Moreover, it can be used to compare the proposed algorithm with the conventional methods.

2

METHODOLOGY

This chapter presents the design details of the proposed multiple target tracking (MTT) system. Specifically, Section 2.1 introduces the simulation setup for generating the multitarget detection data. Then, the signal processing components used in the proposed tracking system are discussed in Section 2.2. After that, Section 2.3 provides the information of the evaluation metrics. Finally, a summary is presented in Section 2.4.

2.1. SIMULATION SETUP

Ideally, the proposed MTT system should be evaluated using experimental radar data. However, acquiring radar data with multiple targets is expensive. Moreover, it is hard to control system variables and measure the system performance analytically using real radar data.

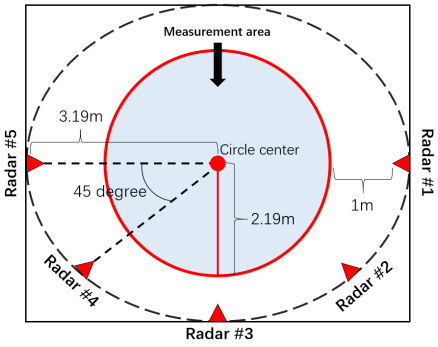
Therefore, the proposed MTT system is mainly evaluated through simulations. The following sections present the details of the simulation setup, including: (1) the geometric settings, (2) the multitarget model, and (3) the simulation limitations. Lastly, an example of the simulated data is provided.

2.1.1. GEOMETRIC SETTINGS

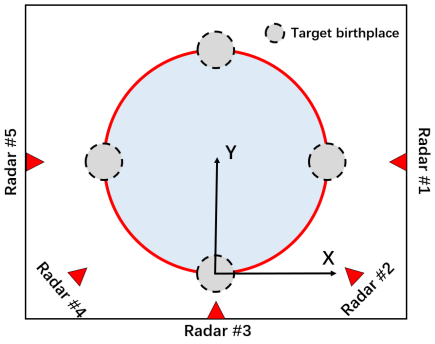
Figure 2.1a shows the layout of the simulated radar sensor network (RSN). Two concentric circles are constructed. The outer circle (dashed black circle) is used to arrange the five identical IR-UWB radar sensors. It has a radius of 3.19m, on which the radar sensors are placed 45° apart. The inner circle (solid red circle) has a radius of 2.19m. Inside the inner circle is the measurement area in which the targets can move freely.

Figure 2.1b illustrates the position of the four birthplaces. The birthplace is the entrance for the newborn targets to enter the measurement area. Inside the birthplace, the initial 2D position of a newborn target follows a Gaussian distribution. In fact, the number of birthplaces can be arbitrary. For example, one can choose as many birthplaces as possible to cover the entire measurement area. However, four is sufficient to prove that the target's origin is uncertain to the MTT system.

For the target's disappearance, a target can leave the measurement area from all directions, i.e., there is no wall on the verge of the measurement area. However, once a target is gone, it cannot be detected by the radar sensors anymore.



(a) The layout of the simulated radar sensor network. Five IR-UWB radar sensors are placed on the circumference of the dashed black circle.



(b) An illustration of the target's birth area (marked in gray), from which the newborn targets can enter the measurement area (marked in blue).

Figure 2.1: The geometric settings of the simulated radar sensor network.

2.1.2. MULTITARGET MODEL

The main goal of the multitarget model is to simulate the uncertainties in multitarget tracking. More specifically, these uncertainties include:

1. **Cardinality Uncertainty**
   The number of targets inside the measurement area is unknown and changing over time.

2. **Motion Uncertainty**
   The target's moving trajectory is unpredictable, and the target's hidden states are unobservable.

3. **Measurement Uncertainty**
   During the measurement, the target can be miss detected, and false detections can appear.

4. **Association Uncertainty**

It is unclear which detection belongs to which target, which detection is a false alarm, and which detections should be associated together in the fusion center.

To express these uncertainties, the proposed multitarget model consists of four fundamental modules:

1. **Target Birth Module**
   The target birth module describes the process of generating new targets. It is characterized by a multi-Bernoulli point process (MBPP). The MBPP is parameterized by two parameters, the success probability s_l and the probability density function (PDF) $p_l(x)$, where l represents the index of the birthplace. While generating the new targets, the birth module also initializes the hidden states of the newborn targets, e.g., the target's initial velocity, shape, birth time, and identity label. In the following sections, the set of newborn targets is denoted by B_i, where the notation i indicates the system time.

2. **Target Motion Module**
   The target motion module describes how the hidden states of the target evolve over time. At time i, the motion module updates the set of hidden states H_i given the previous state H_{i-1}. The H_i is also an RFS containing the state information of all the existing targets in the measurement area. To model a more realistic tracking scenario, the designed motion module conducts a nonlinear transformation on the target's kinematic states. More specifically, the target's acceleration follows a Gaussian distribution, and the acceleration value changes during the target's movement. However, to prevent the target from having an unrealistic moving speed, the target's velocities in the X and Y dimension are clipped (dimension direction indicated in Figure 2.1b).

3. **Target Detection Module**
   The target detection module describes the relationship between the target's hidden states and the received target measurements. It takes the miss detection rate P_D and state set H_i as its input. Assuming the positions of the five radar nodes are known, if a target is detected, the target detection module reads the target's information from H_i and directly calculates the target's distance to the five radar sensors. The output of the target detection module is a set of detection points T_k, where k indicates the index of the radar channel ($k \in [1,5]$).

4. **Clutter Detection Module**
   The clutter detection module describes the process of generating the clutter measurements. Given the high range resolution of the IR-UWB radar, a homogeneous Poisson point process (PPP) is used to characterize the clutter generation process. The homogeneous PPP is parameterized by the Poisson rate λ, which represents the expected number of clutters in the measurement area. In the clutter detection module, the Poisson rate is assigned to a constant value , and the generated clutters are uniformly distributed in the measurement area. In the following, the set of clutter measurements is denoted as C_k.

The final measurement set $M_{k,i}$ is the union of set C_k and T_k at time i after random shuffling. To have a clear image of how these four modules cooperate, Algorithm 1 shows the pseudocode that illustrates the generation process of the measurement set.

2.1.3. SIMULATION LIMITATIONS

It is always desired to have a model that can accurately reflect all the aspects in the experimental radar tracking scenario. However, the simulation accuracy is often a trade-off between model complexity and limited computational power. Since the main objective of the proposed system is to investigate the target merging problem and the introduced false alarm problem, the following limitations exist in the proposed multitarget model:

1. **Mutual Shadowing and Occlusion**
   In this simulation, the mutual shadowing effect and occlusion effect are not modeled. From the measurement point of view, these two effects can influence the target's detectability. However, the miss detection rate P_D is set as a constant value, which means once the target is detected, it can generate measurements at all radar receivers. Although assigning the miss detection rate a more informative value is possible, e.g., making the P_D as a function of the target's position, it will increase the computational complexity in the simulation and tracking stage. Therefore, investigating these two effects is outside the scope of this thesis.

2. **Homogeneous Clutter**
   The main objective of the clutter detection module is to increase the association uncertainties in the fusion center and the multitarget tracker. Therefore, specific types of clutter, e.g., multipath echo or stationary clutter, are not modeled. Instead, all clutter measurements belongs to the same type as they are extended in range and uniformly distributed in the measurement area.

3. **Radar Measurement**
   The received measurement set is assumed to be preprocessed by a clutter reduction technique and a detector. Thus, in the fast time dimension, each range bin either has a "0" value representing a negative detection result, or a "1" value representing a positive detection result.

2.1.4. EXAMPLES OF SIMULATED DATA

Figure 2.2 shows an example of the range-time plot of simulated data from the multitarget model. Based on it, the following observations can be made:

1. The number of targets varies during the radar measurement. In this example, five targets had entered the measurement area. However, it is hard to recognize at which time a target was born or left.

Algorithm 1 Pseudocode of the multitarget model for generating simulated measurements

 # Global parameter initialization
$H_i \leftarrow []$ ▷ The ground truth of the target's hidden states
$M_{k,i} \leftarrow []$ ▷ The measurement set
$T_s \leftarrow 0.26$ ▷ Pulse repetition interval
$N \leftarrow 100$ ▷ Number of frames
$K \leftarrow 5$ ▷ Number of radar nodes
$P_D \leftarrow 0.8$ ▷ Probability of detection
$\lambda \leftarrow 1$ ▷ Poisson rate

 # Starting data generation
for $i = 1, 2, \ldots, N$ **do**

    # Running the target motion module
    if $\{(|H_{i-1}| \neq 0)$ && $(i \neq 1)\}$ **then**
        $H_i = \text{Target\_Motion\_Module}(H_{i-1}, T_S)$
    end if

    # Running the target birth module
    $B_i = \text{Target\_Birth\_Module}(i)$
    $H_i = H_i \cup B_i$

    # Running the target detection module
    $T_k \leftarrow []$
    if $|H_i| \neq 0$ **then**
        for $k = 1, 2, \ldots, K$ **do**
            $T_k = \text{Target\_Detection\_Module}(H_i, P_D)$
        end for
    end if

    # Running the clutter detection module
    $C_k \leftarrow []$
    for $k = 1, 2, \ldots, K$ **do**
        $C_k = \text{Clutter\_Detection\_Module}(\lambda)$
    end for

    # Generating the measurement set
    for $k = 1, 2, \ldots, K$ **do**
        $M_{k,i} = \text{Shuffle}(T_k \cup C_k)$
    end for

end for

2. The target's motion is not linear, and the target's kinematic state changes over time. As a result of random initialization, each target has a different moving trajectory.
3. The measurements are impaired with miss detection, false alarm, and target merging problem. Although the target's measurement shows a strong temporal dependency in the range-time plot, it is hard to tell the detection source by just looking at the plot from a specific timestamp.
4. The uncertainty in the detection-to-detection association exists. For example, there were two targets presented at time 5s, but it is unclear which detections from the five radar channels were originated from the same target. However, knowing the detection-to-detection association is important for the trilateration localization method.

2.2. PROPOSED TRACKING SYSTEM

In this section, the design details of the proposed MTT system are presented. The main goal of the proposed system is to conduct multitarget tracking and feature extraction. Additionally, two research questions have been investigated, i.e., the target merging problem in 1D and 2D tracking and the introduced false alarm problem caused by the association uncertainties in the detection fusion center.

2.2.1. SYSTEM OVERVIEW

Figure 2.3 presents an overview of the proposed MTT system. As you observed, the system uses a decentralized tracking architecture [7]. The simulated radar data is first grouped and tracked in the range plane before the 2D detections are generated and being tracked. As shown in [44], comparing to the centralized tracking, the decentralized tracking architecture is more robust against clutter and achieves better performance in terms of tracking error.

To construct the decentralized tracking architecture, the MTT system consists of four signal processing components:

1. **Tracking-aided Clustering (1D/2D)**
   The tracking-aided clustering module is used to select the most likely detection partitioning hypothesis for grouping the set of 1D and 2D detection points. Besides the measurement partitioning functionality, it also mitigates the target merging problem by using the subpartitioning technique [84].
2. **Tracking-aided Detection Fusion Center**
   The tracking-aided detection fusion center is responsible for transforming the target's 1D position into the Cartesian plane. Moreover, it mitigates the introduced false alarm problem by exploiting the predicted state information of the targets from the multitarget tracker.

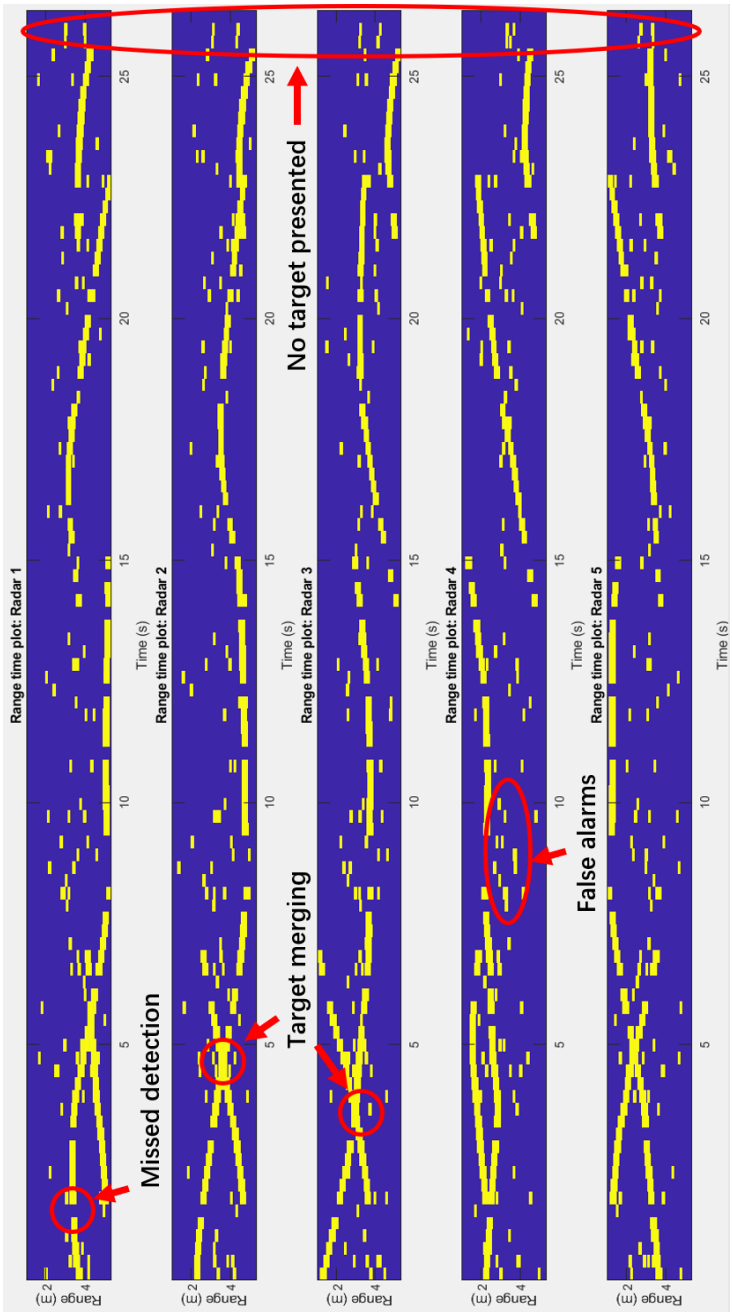


Figure 2.2: An example of the range-time plot of simulated data from the five radar channels. During the measurement time, five targets entered the measurement area. The global parameters are chosen as $T_s = 0.26$, $N = 100$, $\lambda = 1$, $P_D = 0.8$.

3. **Multitarget Tracker (1D and 2D)**
   The multitarget tracker is the core of the proposed MTT system. It is responsible for tracking the targets in the 1D and 2D planes. Moreover, it propagates the target's information to other signal processing components to improve their performance. Furthermore, the multitarget tracker can also help reduce the 1D and 2D clutters.

4. **Feature Extraction Module**
   The feature extraction module is the bridge connecting the proposed tracking and classification system. It extracts the target's micro-Doppler signatures using the position information provided by the 2D and 1D tracker. The output of the feature extraction module is a sliding-window spectrogram that matches the input format of the proposed classifier. Although the simulated radar data does not contain any Doppler information, the procedure is the same when the MTT system is applied to process experimental radar measurement.

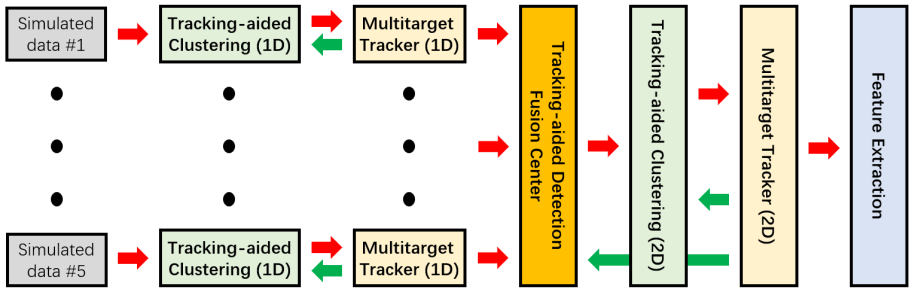


Figure 2.3: An overview of the proposed multiple target tracking system. The input data to the system is the simulated radar data from channel #1 to channel #5. The arrows denote the information flow. Red arrow means forward propagation, while green arrow means backward propagation.

2.2.2. TRACKING-AIDED CLUSTERING

Conventionally, most multiple target tracking algorithms are built based on the assumption that a target can at most generate one measurement. For radar-based applications, it might be true if the distance between the radar and target is sufficiently large. However, for applications like indoor monitoring, especially with the use of a high range resolution radar sensor, a target can often occupy multiple resolution cells if it is detected.

To continue using the conventional tracking algorithms, the clustering algorithm is introduced to the radar signal processing pipeline for selecting the most likely measurement partitioning scheme [34, 43].

However, the clustering algorithm is not guaranteed to provide an optimal measurement partitioning scheme. For example, when several extended targets are spatially close, the clustering algorithm may wrongly partition the measurements of these targets

into the same group. It is important to note that the target merging problem happens when the clustering algorithm is used. In other words, if all combinatorial partitioning schemes are considered, there will be one partitioning hypothesis in which two targets are correctly separated.

For the proposed MTT system, the target merging problem can happen in both 1D and 2D planes. Figure 2.4 shows an example of the target merging problem in the 1D plane. Comparing to 2D target merging, the 1D merging problem has a higher incidence rate, and the situation deteriorates drastically as the number of targets increases or the size of the measurement area reduces. This is because two targets can be merged in the 1D plane as long as they have equal distance to the radar sensor.

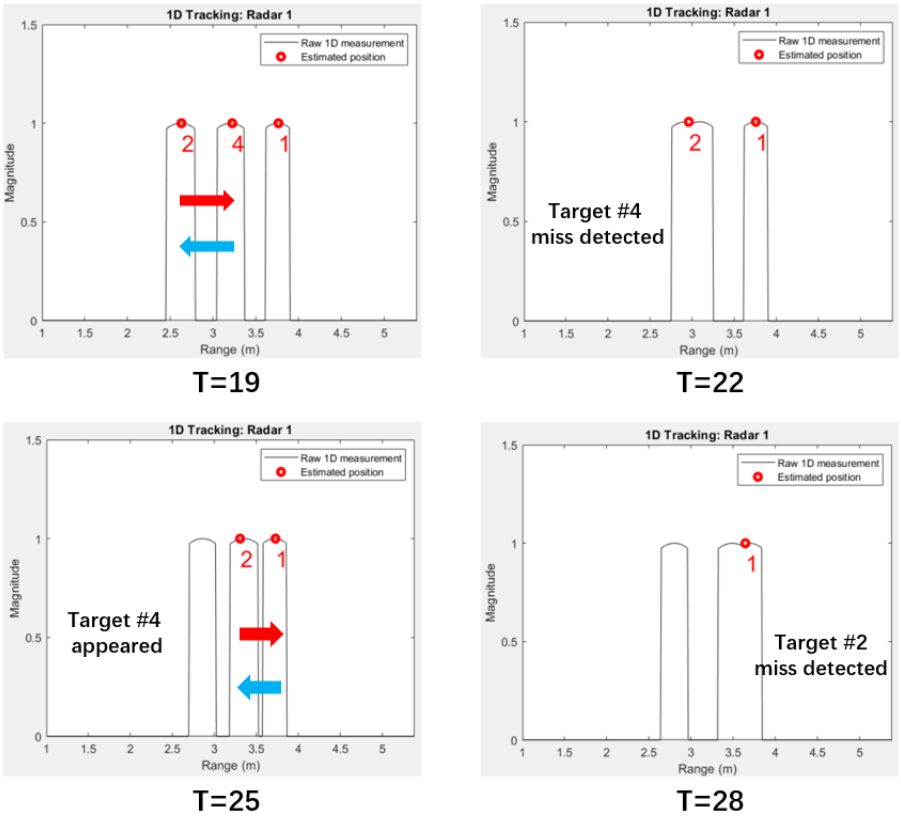


Figure 2.4: An illustration of the target merging problem in the 1D plane. This example uses simulated data from radar #1. As you see, at time 19, there were three targets presented. At time 22, target #4 and target #2 merged which led to a missed detection of target #4. At time 25, target #4 and target #2 were separated, however, due to consecutive miss detections, target #4 was deregistered from the track table. At time 28, the target merging problem happened again.

However, the target merging problem has rarely been explored in the UWB radar-

based tracking system. Solving this problem is essential for investigating the joint multitarget tracking and classification because:

1. The missed detections caused by the target merging effect may influence the continuity in target tracking and activity monitoring.
2. Knowing targets are merged in one of the sensors can be used as a piece of prior knowledge to guide the feature extraction module to extract the reliable information only.

A tracking-aided clustering module is constructed to address the extended-target tracking problem and the target merging problem. The proposed clustering module is similar to the sub-partitioning method proposed in [50, 84], in which two clustering algorithms are used in a row. In the rest part of this section, the architecture design of the proposed clustering module for 1D clustering is presented. For measurement clustering in the 2D plane, despite the difference in the data format, the methodologies are the same.

As illustrated in Figure 2.5, the proposed module consists of two data clustering algorithms. They are placed in sequential order. The first clustering algorithm is responsible for separating the measurements into multiple groups and detecting the wrongly merged target. Considering the number of presented targets is unknown, the density-based spatial clustering of applications with noise (DBSCAN) algorithm [55] is used. To detect the wrongly merged target, the DBSCAN algorithm takes the predicted state information of the registered targets $\hat{X}_{k,i|i-1}$ and the measurement set $M_{k,i}$ as its input. Given the clustering result, if there is more than one target in set $\hat{X}_{k,i|i-1}$ is associated with the same set of measurements in set $M_{k,i}$, these associated targets are marked as the merged targets.

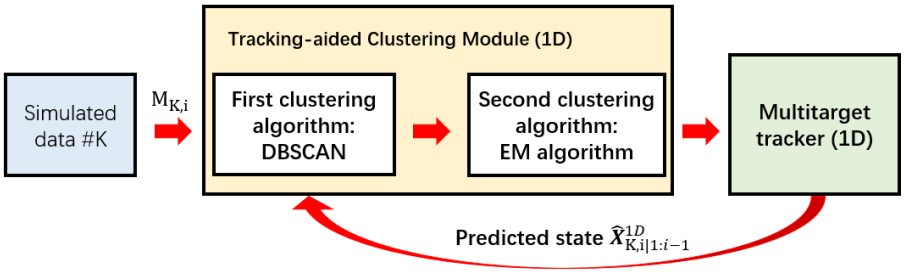


Figure 2.5: A schematic diagram of the tracking-aided clustering module for 1D measurement partitioning. The module takes the measurement set $M_{K,i}$ from the K-th radar channel and the predicted state information of the registered targets $\hat{X}_{K,i|i-1}$ as its input.

To separate the measurements that were merged in the first step, the expectation maximization (EM) algorithm [54] with Gaussian mixtures is used as the second clustering algorithm. The Gaussian parameters of the EM algorithm are initialized with the

predicted positions, shape extends, and number of measurements of the merged targets to improve the convergence speed and clustering accuracy.

The final output of the clustering module contains multiple groups of measurements that were well-separated based on the predicted target information. Before sending them to the multitarget tracker, the center point of each group is calculated. Moreover, each group is assigned a label according to the source of the measurements. This label information is helpful for the feature extraction module.

Despite the advantages of solving the target merging problem, it is important to note that the performance of the proposed clustering module is highly dependent on the prediction accuracy of the multitarget tracker. In other words, the performance deteriorates if the target maneuvers quickly. However, for indoor monitoring using IR-UWB radar, the performance is less influenced due to the sensor's high scanning rate and human's limited maneuvering capability.

2.2.3. TRACKING-AIDED FUSION CENTER

Target localization is a challenging task in the IR-UWB radar-based RSN. This is because the UWB radar can only provide the range information of the presented target in the measurement area. Thus, to localize a target in the Cartesian plane, the detection fusion center [43] is introduced.

The detection fusion center uses the trilateration technique [45] to calculate a target's 2D position. However, the prerequisite of using the trilateration technique requires finding all the measurements originated by the same target from different radar channels. This step is called data association.

The data association is a trivial task if only one target is presented in the measurement area. However, in multitarget scenarios, especially when the measurements contain false alarms and missed detections, the uncertainty in finding a set of correct data associations for each presented target is extremely high.

Since the origins of the measurements are unknown to the fusion center, conventionally, all the combinatorial data associations are considered as possible association hypotheses. However, as illustrated in Figure 2.6, this will lead to a large number of introduced false alarms in the 2D plane. Although many false alarms can be eliminated by the subsequent signal processing units such as the 2D clustering and 2D tracking, it is beneficial to find a mechanism to prevent generating these false alarms at the beginning.

Traditional false alarm suppression method for the detection fusion center involves the evaluation of the least-square (LS) error. As detailed in [34], during the LS-based trilateration process, every calculated 2D position is associated with a residual error. A thresholding method is implemented afterward to remove the 2D position with a residual error above a pre-defined threshold.

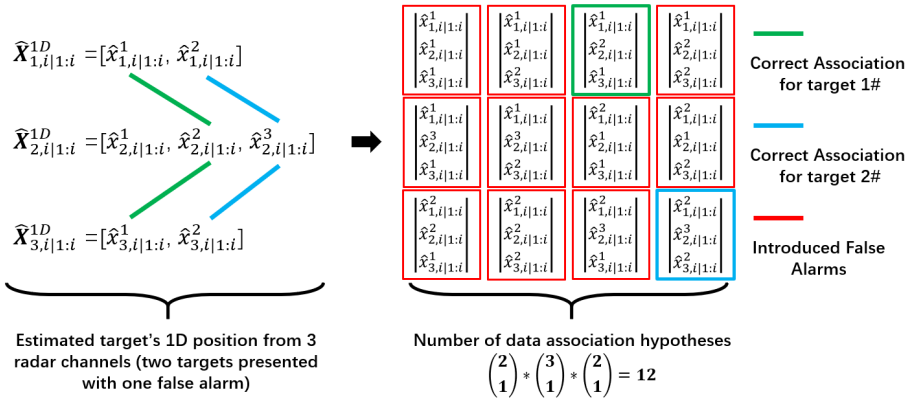


Figure 2.6: A simple illustration of the data association process, where three radar sensors are used to capture the range information of two moving targets. For each radar channel, the estimated target's 1D position at time i contains two elements, except radar channel #2, in which one additional element is generated due to a clutter measurement. Although only two targets are presented, due to the data association uncertainties, there are 12 association hypotheses, which will lead to 12 detection points in the 2D plane.

Although the conventional approach works in general, its performance is highly dependent on the predefined threshold, which controls the trade-off between the false alarm rate and missed detection rate. Moreover, achieving a good suppression performance requires the RSN to have at least four radar nodes, and the performance improves as the number of nodes increases. Another disadvantage of the conventional method is its robustness against the estimation error and closely-spaced targets. For example, when targets are spatially close, the residual error generated by an incorrect data association may be too small to pass the threshold.

To improve the defects in the conventional method, a tracking-aided detection fusion center is proposed. As illustrated in Figure 2.7, the proposed fusion center takes the estimated target's 1D range information as its input, and outputs a set of 2D detection points for the 2D clustering module. To reduce the data association uncertainty, it uses the predicted 2D positions of the registered targets as prior knowledge. More specifically, for every tracked target, its predicted 2D position is used to calculate the corresponding 1D distance to each radar node, and these distances are used to associate the estimated ranges from the 1D tracker.

Comparing to the conventional method, the proposed false alarm suppression scheme directly reduces the data association uncertainty. It is more flexible in choosing the trilateration technique since it does not rely on a specific type of trilateration approach. Moreover, the performance of the proposed method is more robust as it does not rely on setting a suitable threshold value. Furthermore, it is also possible to combine the proposed method with the conventional method or others. This is because the proposed method is only applied to reduce the data association uncertainty. Other methods based on the residual error or track characteristics are still applicable.

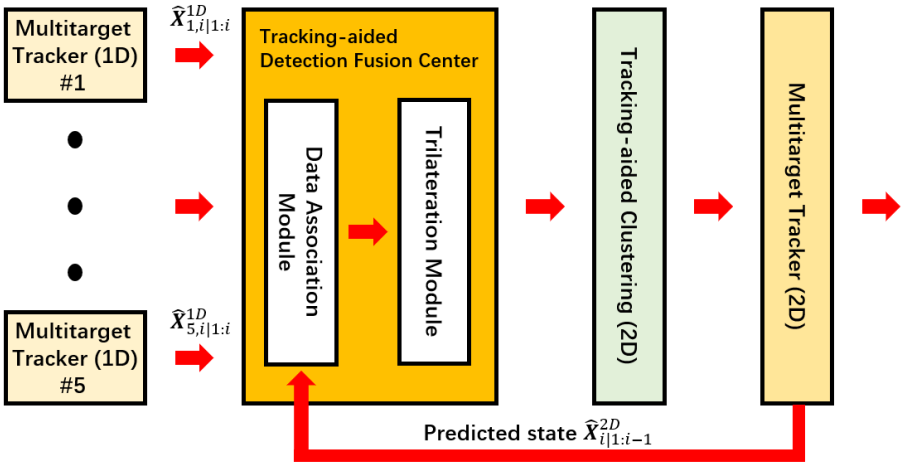


Figure 2.7: A schematic diagram of the tracking-aided detection fusion center. The fusion center is responsible for transforming the range information collected from each radar channel to the Cartesian plane.

2.2.4. MULTITARGET TRACKER

The multitarget tracker is the most important component in the proposed MTT system. As shown in Figure 2.3, due to the use of a decentralized processing architecture, two multitarget trackers are applied to estimate the target's kinematic states at a different dimension. As the name suggests, the multitarget tracker 1D is used to update the target's range information, whereas the target's location in the Cartesian plane is estimated by the multitarget tracker 2D. Other than the tracking domain difference, the multitarget tracker 1D and 2D follows the same tracking procedures. Thus, in the remaining part of this section, the challenges of tracking multiple targets and the specific tracker design are discussed assuming tracking targets in the 2D plane.

Figure 2.8 shows an example of tracking multiple targets in the 2D plane. The ground truth position of the presented target is represented as a green dot, whereas the raw 2D measurements generated by the detection fusion center are denoted as gray asterisks. At time 1, two targets entered the measurement area from two different birthplaces, each of which has a trajectory marked as the dashed curve. Then, target #1 left the area at time 34. At time 41, there is no target presented in the measurement area as no 2D measurement was received. After a short period, target #3 entered the area at time 47.

Thanks to the decentralized tracking architecture and the proposed signal processing units, there is less clutter and uncertainty in the 2D target tracking. Nevertheless, there are some issues that were still left to be addressed:

1. **Unknown target birth**
   The simulated data models the target birth as a multi-Bernoulli point process.

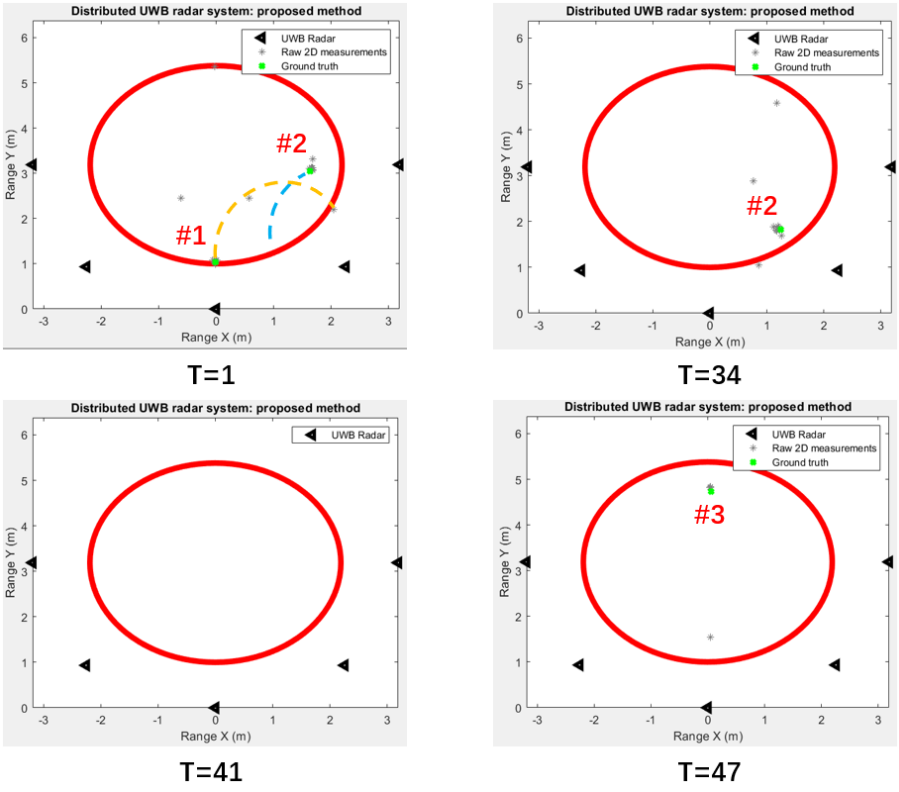


Figure 2.8: An example of tracking multiple targets in the 2D plane using simulated data. In the above figure, targets are represented as green dots, whereas the raw 2D measurements generated by the detection fusion center are denoted as gray asterisks. At time 1, two targets entered the measurement area. The ground truth trajectory of each target is denoted as a dashed curve.

That is to say, at every time step, the number of newborn targets as well as their 2D locations are unknown to the multitarget tracker.

2. **Arbitrary moving trajectory**
   As shown in Figure 2.8, the target's moving direction is arbitrary, and only the current position information can be measured. To track multiple targets accurately, it is necessary to estimate their kinematic states (e.g., velocity) over time.

3. **Association uncertainty**
   Although most clutters have been suppressed by the previous processing steps, the origin of each measurement is still uncertain to the multitarget tracker. This is because a measurement may originate from a residual clutter, a detected target, or a newborn target.

4. **Forming trajectory**

Conventional multitarget tracking task only requires updating the target's state information at every time step. For joint tracking and classification tasks, to collect a target's Doppler information over time, it is necessary to form a trajectory for every registered track.

5. **Estimating shape attributes**
Since the target being tracked has an extended shape, the clustering algorithm outputs not only the center point of each cluster but also the partitioned raw 2D measurements. It can be assumed that all 2D measurements in one cluster have the same source, e.g., clutter or target. Thus, it is possible to estimate the target's shape attributes based on the set of measurements. Moreover, it has been shown in [50] that incorporating the estimated shape attributes and the clustering algorithm can achieve better measurement partitioning performance.

Considering all the aspects mentioned above, it is evident that the traditional multitarget trackers [11–13], which are often used to track a known number of point targets, are insufficient to handle these challenges. Therefore, in this thesis work, the gamma Gaussian inverse-Wishart (GGIW) Poisson multi-Bernoulli mixture (PMBM) filter [85] is used for tracking multiple extended targets. Since the GGIW-PMBM filter has already been formally derived, to avoid repetition, the reader is referred to [81, 85–87] for information regarding the implementation and derivation details. Instead, the following will discuss how the challenges are handled by this filter.

Figure 2.9 shows the schematic of the PMBM components in the multitarget tracker. As you see, the Poisson point processes (PPP) and the multi-Bernoulli point process (MBPP) are used in parallel.

The PPP is responsible for characterizing the undetected target (e.g., newborn target or target blocked due to occlusion) and the clutter measurement. It is parameterized by an intensity function. To model the uncertainty in the target birth, at every time step, a mixture of Gaussian distributions is added into the intensity function. Each Gaussian component is parameterized by a given mean and variance value expressing our belief of the potential birthplace. Since the shadowing effect is not modeled in the simulation and a target can not be generated inside the measurement area, the Gaussian components are normally placed on the edge of the measurement area. Other than the necessary Gaussian parameters, each Gaussian component has a corresponding weight representing the expected number of targets generated by that Gaussian component.

The PPP also models the clutter measurements. In this thesis work, a uniform distribution is used to characterize the spatial distribution of the clutter measurement. Similar to the mixture of Gaussian distributions, the expected number of clutter generated by the PPP is described by a weight value. Finally, the integral of the intensity function in the PPP indicates the sum of the expected number of newborn targets and the clutter measurements.

The MBPP is used to model the set of detected targets. In the MBPP, each registered target (or Bernoulli component) is parameterized by an existence probability (or suc-

cess probability) and a spatial distribution. The existence probability represents our confidence in the target's existence, whereas the spatial distribution provides the target's location information. In this work, the spatial distribution of each Bernoulli component is characterized by the Gaussian distribution. To handle the uncertainty in the measurement-to-track association, a mixture of multi-Bernoulli distributions is used, where each mixture represents a global data association hypothesis.

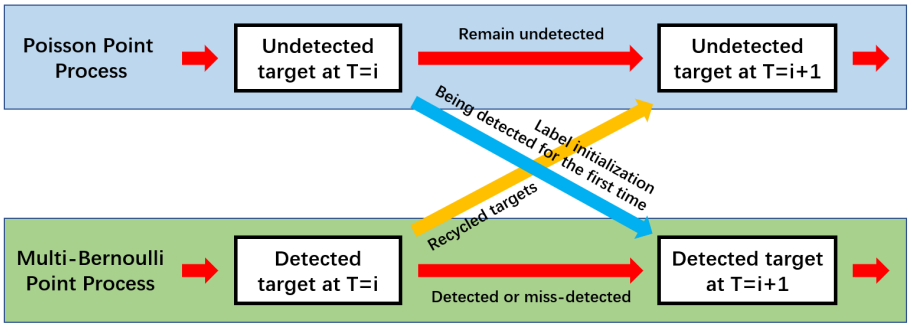


Figure 2.9: An illustration of the PMBM components in the multitarget tracker. The undetected targets and clutters are model as a Poisson point process, whereas the detected targets are model as a multi-Bernoulli point process. For the multi-Bernoulli point process, the mixture representation is used to convey different data association hypotheses for the tracked targets.

Given a set of partitioned measurements at the time i, the cost matrix is calculated first based on the predicted information of the detected and undetected targets. Then, the off-the-shelf combinatorial optimization algorithm (e.g., [62–64]) can be used to find the optimal global data association hypothesis. In this work, the Jonker-Volgenant algorithm [88] is used, which reduces the tracking complexity to track only the most probable hypothesis. For a given data association hypothesis, a measurement can be associated either with a detected target in the set governed by the MBPP or with an undetected target or clutter in the set governed by the PPP.

If a measurement is associated with a detected target, the target's kinematic states as well as the shape attributes will be updated based on the assigned measurements. However, it is possible that after the data association procedure, there are previously detected targets assigned with an empty set of measurements. In that case, such a target is regarded as miss detected, and its states will be updated using the predicted values. In case of a consecutively missed detection happened to a registered target, the target's density will be removed from the MBPP and added to the intensity function of the PPP. This step is called target recycling. Compared to just discarding the miss detected targets, the recycling mechanism helps preserve the information of the lost target and improves the tracking efficiency.

Suppose a measurement is associated with an undetected target. In that case, not only the target's kinematic states and the shape attributes will be updated, but also the density of that undetected target will be added to the MBPP. It is important to note that

while adding a new Bernoulli component for the target being detected for the first time, an identity label is also initialized. However, comparing to the formally derived labeled multi-Bernoulli filter [89], the aforementioned labeling method serves only as a heuristic to mend the disadvantages of using the PPP to describe the behavior of the undetected target.

For the undetected target remains undetected, its state information will be updated using the predicted value. Moreover, the weight is multiplied by a constant factor s ($0 < s < 1$) after every iteration to reduce the expected number of generated targets related to that Gaussian distribution. Since after every term, there will be a new mixture of Gaussian distributions and the recycled densities added into the intensity function of the PPP, it is necessary to conduct the pruning, merging, and capping procedures to reduce the computational complexity of the multitarget tracker.

Thanks to the PMBM multitarget conjugate prior, the filter has an explicit model to describe different uncertainties in multiple target tracking. Using the PPP to model the set of undetected targets is advantageous in terms of computational efficiency. This is because the data association step is not required during the Bayesian prediction and update steps. However, one drawback of such modeling is all elements generated by the PPP are independent and identically distributed. As a consequence, using the labeled random finite set for the PPP is problematic because a set of measurements may generates several targets with the same identity label.

For the set of detected targets, the MBPP can give an accurate cardinality estimation. It also shows a better tracking performance comparing to the probability hypothesis density filter (PHD) [82]. However, it has been noted that the multi-Bernoulli part often has high computational costs due to the explicit data association and a mixture of global hypotheses. This problem is mitigated using the measurement-driven mechanism in the PMBM filter.

To estimate the target's shape attributes through the Bayesian recursion, the GGIW density is the conjugate prior for the single extended target tracking [85]. It assumes the number of detections generated by one target follows a Poisson distribution. Therefore, it is logical to use the gamma distribution as the prior density to estimate the rate of the Poisson distribution. For the target's extend, it is characterized using the random matrix model. That is to say, the set of measurements follows a Gaussian distribution around the target's center. Thus, the Gaussian-inverse-Wishart distribution, which is conjugate to the multivariate Gaussian distribution, is used.

In this thesis work, the estimated state information is used to guide the feature extraction process, and the estimated shape attributes are propagated to the clustering module for improving the subpartitioning accuracy. Although there are many possibilities in using these estimated results (e.g., target differentiation based on measurement density), and many problems are left undiscussed (e.g., label switching, track discontinuity, and target spawning), limited by the scope of this work, these aspects are expected to be found in the multitarget tracking literature.

2.2.5. FEATURE EXTRACTION

Although the simulated data are pure detection points that do not contain any Doppler information, it is necessary to discuss the feature extraction process given its importance in the proposed signal processing pipeline and the ultimate goal of this thesis work. However, extracting the motion characteristics from multiple moving targets is not always straightforward since a missed detection can happen at any time during the target's movement.

In this work, once the miss detection occurs to a target, it is assumed that all radar sensors lost the detection of that target. The direct consequence is no Doppler information can be extracted from that target anymore. However, more frequently, the miss detection effect tends to occur in one of the radar sensors due to, for example, the shadowing effect. In that case, the Doppler information from other radar sensors may still be available and can be used for activity classification.

To address the partial miss detection, the set of measurements is labeled. That is to say, every element inside the set of measurements has a specific label indicating its origin (i.e., the radar index). The label information is propagated through the signal processing pipeline from the input of the 1D Clustering to the output of the 2D multitarget tracking. Therefore, when a target is detected, it is immediately known for the feature extraction process that which measurements from which radar sensors had made the measurement-to-track association.

Then, according to the target's identity label, the range bins that contain the target's movement are collected across the slow time for every tracked target. As illustrated in Part III Chapter 2.2.2, the input data for the proposed human activity recognition system are small chunks of spectrograms generated by using the short-time Fourier transform (STFT). In this work, a sliding window is applied to the generated spectrogram. For every time interval, the oldest Doppler information is removed, and the newest is calculated and added into the spectrogram. Finally, the updated spectrograms from all radar channels are concatenated to form a data cube and sent to the classifier.

2.3. EVALUATION METHOD

Although the evaluation step is not part of the proposed MTT system, it is still an indispensable process during system development. In the evaluation step, a specific evaluation metric is selected to measure the system error given the ground truth. The evaluation metric provides a quantitative value that indicates the performance of the proposed system. More importantly, the evaluation results can be used to compare different tracking systems or system performances under various signal processing components.

The output of the proposed MTT system at every time step is a set of 2D vectors. Each element indicates the estimated position of the potential targets in the Cartesian plane. Similarly, the system ground truth is also a set of 2D vectors, each of which provides the

actual locations of the presented targets in the measurement area. Therefore, the goal of the evaluation metric is to measure the "difference" between the set of estimates and ground truths.

Figure 2.10 shows an example of the performance evaluation scenario for tracking multiple targets, where the ground truths are denoted as green dots, and the tracker estimates are marked as red dots. As demonstrated, there are several issues that need to be addressed by the evaluation metric:

1. The evaluation metric should be able to reflect the total localization error. For example, the distance between the estimate #1 and ground truth #3.

2. The evaluation metric should have a specific cut-off value that defines the relationship between an estimate and a ground truth. For example, when the distance between the estimate #2 and the ground truth #1 exceeds the predefined cut-off point, the distance of theses two points will not be aggregated into the localization error anymore.

3. The evaluation metric should be able to penalize the missed detections and false alarms (e.g., the ground truth #2 and the estimate #3). It is important to note that measuring the system performance on these two aspects is extremely important since the missed detection may influence the temporal continuity of the extracted Doppler signatures, and the false alarm may waste the limited computational resource.

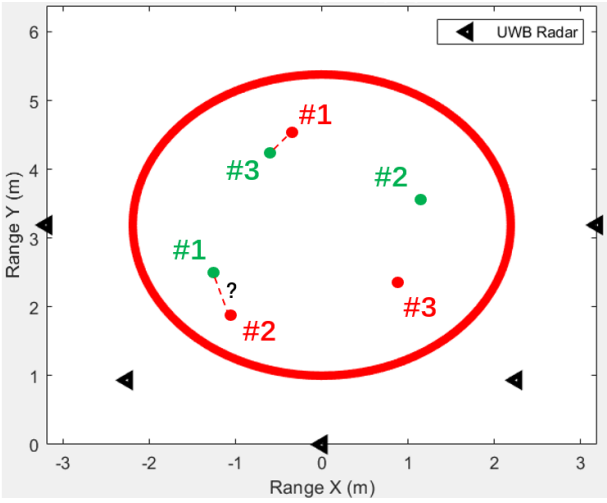


Figure 2.10: An example of the performance evaluation scenario for tracking multiple targets. The set of estimated positions are denoted as red dots, whereas the ground truths are denoted as green dots.

The optimal sub-pattern assignment (OSPA) [90] metric is the most popular evaluation metric for multitarget tracking problems. It is a mathematically and intuitively consistent metric that outperforms the traditional evaluation metrics such as the Hausdorff metric and the optimal mass transfer (OMAT) [91] metric. However, the OSPA metric cannot assess the errors caused by the false and missed targets. Thus, the generalized optimal sub-pattern assignment (GOSPA) [23] metric, which successfully solved the aforementioned issues, is used to evaluate the proposed MTT system.

For a specific selection of metric parameter (discussed in [23] Part II, Section B), the GOSPA metric can be expressed as the following [23]:

$$d_p^{(c,2)}(\mathbf{X},\mathbf{Y}) = \left[\min_{\gamma\in\Gamma} \Big(\sum_{(i,j)\in\gamma} d\left(x_i, y_j\right)^p + \frac{c^p}{2}\left(|\mathbf{X}| - |\gamma| + |\mathbf{Y}| - |\gamma|\right)\Big) \right]^{1/p} \tag{2.1}$$

Where $\mathbf{X}$ is the set of estimates, $\mathbf{Y}$ is the set of ground truths, γ is the global association hypothesis that assigns elements from $\mathbf{X}$ to $\mathbf{Y}$, Γ is a set that contains all possible global association hypotheses, c and p are two parameters of the GOSPA metric, and $d(x_i, y_j)$ calculates the distance between the element x_i in the estimate set and y_j in the ground truth set.

The parameter c and p controls the cut-off point and the metric's sensitivity to outliers, respectively. In this work, c is set to 0.5, which means the maximum forgivable localization error is 0.5 meter. To make the GOSPA metric analogous to the root mean square error (RMSE), the p is set to 2, and the $d(\cdot,\cdot)$ uses the Euclidean distance [23].

Given the set $\mathbf{X}$ and $\mathbf{Y}$, to calculate the GOSPA error, a cost matrix is formulated first. The cost matrix is similar to the one in the multitarget tracker. Each value in the cost matrix represents the cost for assigning an element in the set $\mathbf{X}$ to an element in the set $\mathbf{Y}$ or to miss-cardinality cost c. Then, the optimal assignment scheme γ is calculated by using the combinatorial optimization algorithm (e.g., the Hungarian method).

Each element in γ indicates a specific association between an estimate x_i and a ground truth y_i. The cardinality of the set γ (denoted as $|\gamma|$) shows the total number of established associations. Thus, the cardinality of the set γ should satisfy:

$$|\gamma| \leq min(|X|, |Y|) \tag{2.2}$$

Finally, it is obvious to see that the total GOSPA error $d_p^{(c,2)}(\mathbf{X},\mathbf{Y})$ can be decomposed into three error sources:

1. The localization error $\sum_{(i,j)\in\gamma} d\left(x_i, y_j\right)^p$.
2. The false alarm error $\frac{c^p}{2}(|X| - |\gamma|)$.
3. The miss detection error $\frac{c^p}{2}(|Y| - |\gamma|)$.

Therefore, with the use of the GOSPA metric, the performance of the proposed MTT system can be intuitively observed in terms of localization accuracy and missed and false targets. Moreover, the GOSPA metric can be extended to measure the distance between two RFSs (e.g., the mean GOSPA and root mean square GOSPA) [23].

2.4. SUMMARY

In this chapter, the details of the proposed MTT system are presented.

To analyze the system performance under different tracking scenarios, a simulated multitarget model is designed based on a distributed radar sensor network with five identical IR-UWB radar sensors. The proposed multitarget model contains four fundamental modules that encompass all the tracking uncertainties discussed in Section 2.1.2. Although with some limitations, the proposed model is regarded to be able to reflect the important aspects of multitarget tracking. An example of the simulated data is shown in Section 2.1.4.

Based on the simulated data, the proposed tracking system is presented in Section 2.2. The proposed tracking system uses the popular decentralized tracking architecture to track multiple targets in the 1D and 2D plane. Inside the tracking system, four signal processing components are designed: (1) the tracking-aided clustering (1D/2D), (2) the tracking-aided detection fusion center, (3) the multitarget tracker (1D/2D), and (4) the feature extraction module.

The design details of the four processing components are presented in Section 2.2.2, 2.2.3, 2.2.4, and 2.2.5, respectively. It is important to note that the main objective of these components is to address the research challenges mentioned in Part II, Chapter 1.

Finally, to be able to validate the proposed MTT system quantitatively, the evaluation method is discussed in Section 2.3. In this work, the GOSPA [23] metric is selected to measure the system performance on three different domains, i.e., the localization error, miss detection error, and false alarm error.

3

RESULTS

In this chapter, the evaluation result of the proposed multiple target tracking (MTT) system is presented.

As detailed in Part II, Section 2, the proposed system contains four fundamental signal processing components. In the following sections, the performance of the tracking-aided clustering module will be presented first in Section 3.1. Section 3.2 shows the evaluation results of the proposed detection fusion center. After that, in Section 3.3, the overall system performance for tracking multiple targets is measured using the generalized optimal sub-pattern assignment (GOSPA) metric. Lastly, the feature extraction module is tested in Section 3.4 using the real radar data.

3.1. TRACKING-AIDED CLUSTERING

In this section, the performance of the proposed tracking-aided clustering module is presented.

The main objective of the clustering module is to select the most probable measurement partitioning scheme so that the computational complexity can be reduced. Moreover, the clustering module is dedicated to solve the measurement merging problem that happens to closely spaced targets. As demonstrated from the previous chapter, in the IR-UWB radar-based radar sensor network (RSN), the merging problem has a higher incidence rate in the 1D plane than the Cartesian plane.

Thus, to address this problem, the proposed clustering module uses the subpartitioning technique [50, 84]. In the following sub-sections, the clustering module will be first tested based on different tracking scenarios. Then, the performance of the clustering module is evaluated in the proposed MTT system.

3.1.1. PERFORMANCE ON PREDEFINED TRAJECTORIES

To validate the performance of the proposed clustering algorithm independently, three target tracking scenarios are simulated. It is important to note that simulated dataset has a different simulation setting than the proposed RSN. In this case, no detection fusion center and distributed RSN is used. Instead, the received radar measurements are directly transformed into the 2D plane. The set of measurements of each detected target is modeled as a Poisson point process (PPP), in which the number of the expected detections is controlled by the Poisson rate, and the distribution of these measurements follows a Gaussian spatial density.

In the following experiments, the performance of the proposed clustering module is compared to the conventional clustering module. The proposed module is referred to as the tracking-aided clustering module, in which the density-based spatial clustering of applications with noise (DBSCAN) [55] algorithm is used as the main clustering component and the expectation-maximization (EM) [54] algorithm is added as a sub-partitioning technique [84]. In contrast, the conventional clustering module is referred to as the clustering process with only the DBSCAN method used.

Figure 3.1 shows the performance comparison between the proposed and conventional clustering module evaluated on the predefined tracking scenario A. As was shown, two targets were presented on the left side in the measurement area, and they were well separated in space at the beginning but gradually moving toward each other. After several time steps, the measurements generated by these two targets were merged, and they kept the close distance while walking before they finally separated.

The consequence of the target merging phenomenon is evident. As shown in Figure 3.1b, the target #A immediately lost after the conventional clustering module wrongly partitioned the measurements originated by two targets into one group. In contrast, Figure 3.1a shows that the proposed clustering module is able to separate the merged measurements and maintain the tracks for both targets. With the proposed method, each established track has a set of measurements to associate. Moreover, the proposed clustering module can also handle the case when two closely-spaced targets start separating.

Two more complicated tracking scenarios are presented. Figure 3.2 shows the case when two targets are merged but they keep maneuvering. This is a typical case in tracking multiple humans as people may walk together. Figure 3.3 shows a more challenging tracking scenario, in which three targets were presented and merged during their movement. As you can observe, the multitarget tracker can still preserve the tracks for each target when the proposed clustering module is used.

However, it is important to note that the track switching problem is not considered in this work. Although the ground truth trajectory of each target is known, the estimated track could end up with a different trajectory due to the association uncertainties that existed when merged targets started to separate. For example, in Figure 3.1a, the estimated trajectory of target #B starting from the bottom-left may end at upper-right.

In summary, it has been shown in [50, 84] and this work that implementing the sub-partitioning technique is an effective way to address the measurement merging problem. In the following sub-section, the proposed clustering module will be added to the proposed MTT system to solve the 1D and 2D target merging problem.

3.1.2. PERFORMANCE ON PROPOSED TRACKING SYSTEM

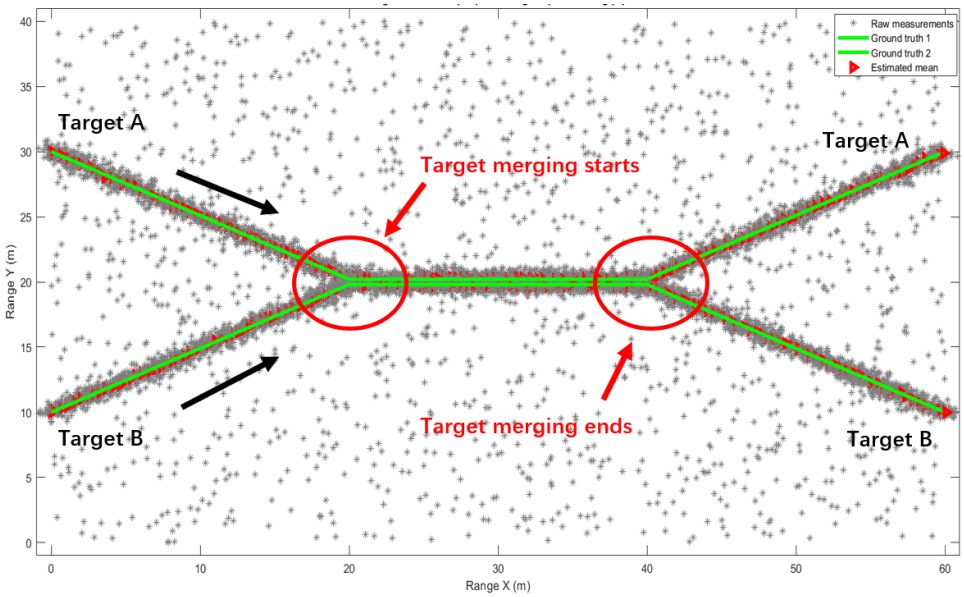
The previous sub-section verified the feasibility of the proposed clustering module. In this section, the clustering module is added to the signal processing pipeline for tracking a random number of targets using the IR-UWB radar-based RSN. Since simulated targets have extended shape but the IR-UWB radar can only measure the range information of the target, both the 1D and 2D clustering are required in the processing pipeline.

Figure 3.4 shows three snapshots of simulated multitarget data. The data is processed by the conventional clustering module. As shown by Figure 3.4a, two targets were presented in the measurement area at time 89. Although the two targets were neither close to each other nor moving toward each other in the Cartesian plane, their 1D measurements sampled by the radar #3 are fairly close and tend to be merged.

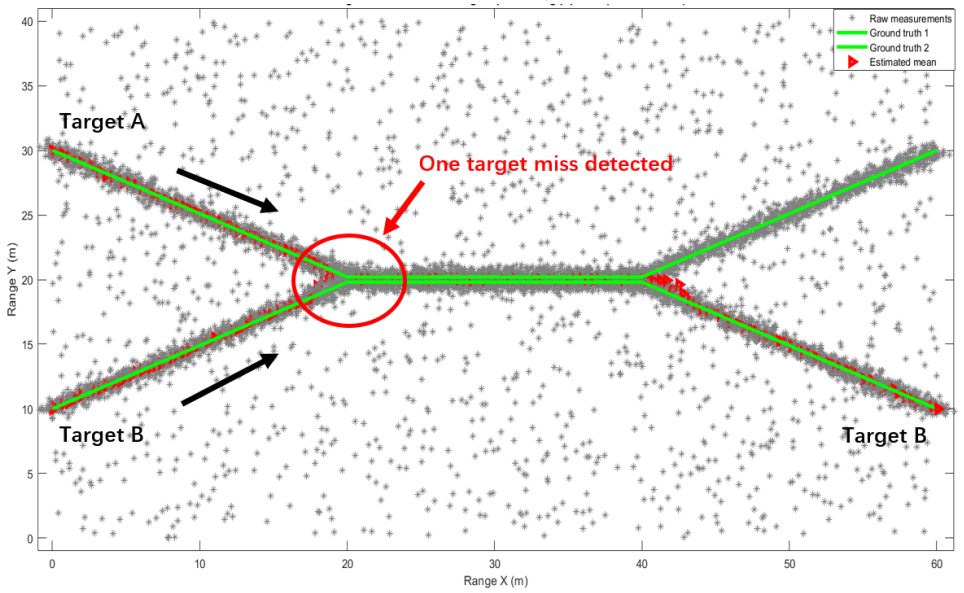
At time 92 (shown in Figure 3.4b), the measurements of the target #3 and #4 were merged, which leads to a miss detection of target #3. Although the two targets were separated again at time 94, the target #3 remains undetected (shown in Figure 3.4c). This is because the target #3 had already been removed due to several consecutively missed detections.

Based on the above observations and the performance of the conventional clustering algorithm, it is extremely important for radar-based multitarget tracking applications to solve the target merging problem, especially when the applied radar sensor can only measure the range information. As has been seen, even targets are not close in the 2D plane, as long as they have equal distance to the same radar sensor, the sampled 1D measurements may be merged. More importantly, if the number of presented targets increases, the problem may become more evident.

Figure 3.5 shows the performance of the proposed clustering module evaluated under the same simulated multitarget data. As it shows the tracks of the two targets were preserved when the 1D measurements merged at time 92. Due to the sub-partitioning technique, the set of merged measurements were further partitioned into two groups, each of which is used to associate one of the established tracks. Moreover, thanks to the soft assignment feature of the EM algorithm, a measurement can be assigned to both groups. Comparing to the clustering algorithms with hard data assignment, for example, the K-means [51] and K-means++ [53] algorithm, the proposed method can lead to a more accurate estimation of the center of mass of the partitioned groups.

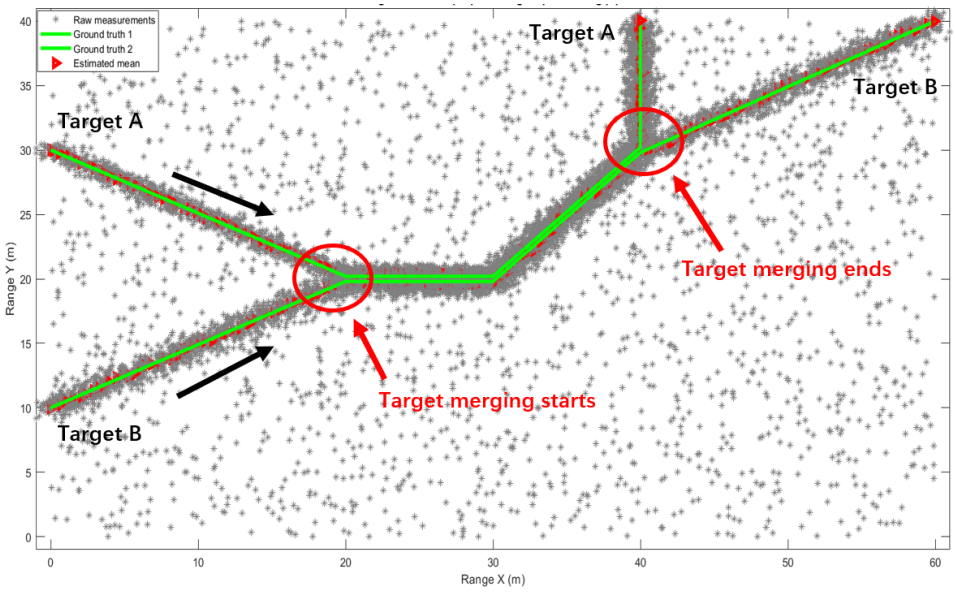


(a) The proposed clustering module.

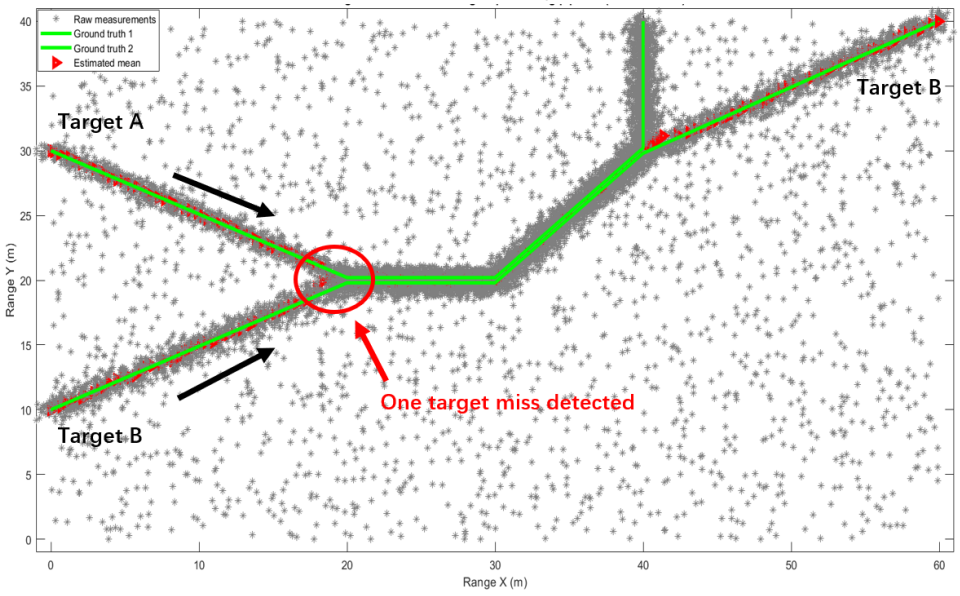


(b) The conventional clustering module.

Figure 3.1: Tracking scenario A: two targets are presented in the measurement area moving from the left to the right (moving direction marked as dark arrow). In the above plots, the radar and clutter measurements are denoted as gray asterisks, the estimates provided by the multitarget tracker are denoted as red triangle, and the ground truths are denoted as green lines.

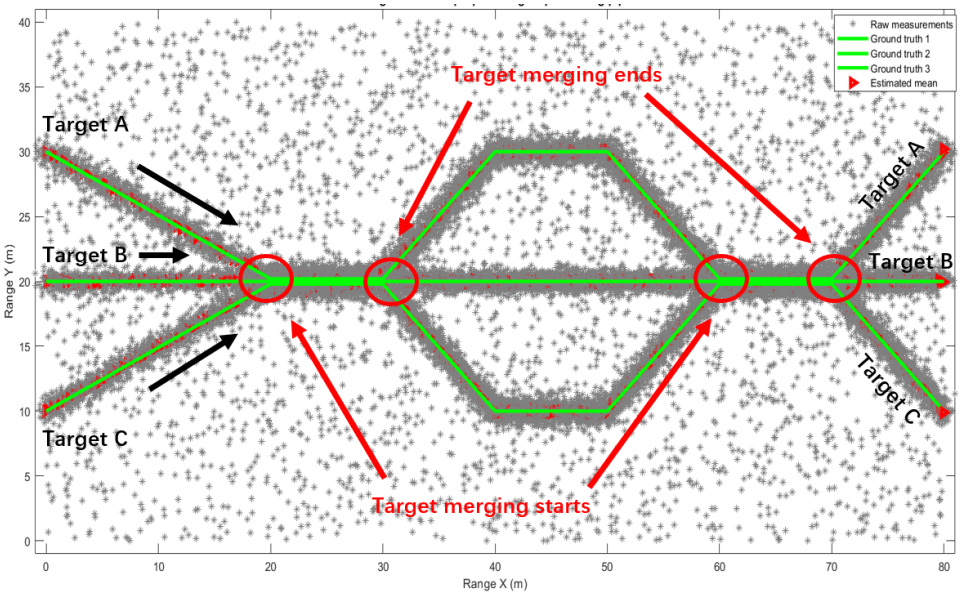


(a) The proposed clustering module.

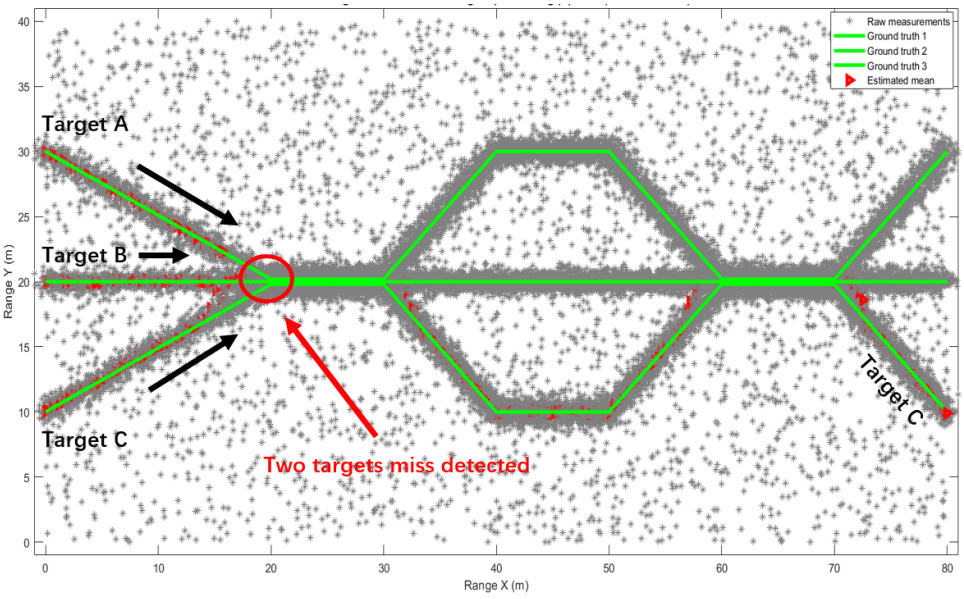


(b) The conventional clustering module.

Figure 3.2: Tracking scenario B: two targets are presented in the measurement area moving from the left to the top (moving direction marked as dark arrow). In the above plots, the radar and clutter measurements are denoted as gray asterisks, the estimates provided by the multitarget tracker are denoted as red triangle, and the ground truths are denoted as green lines.

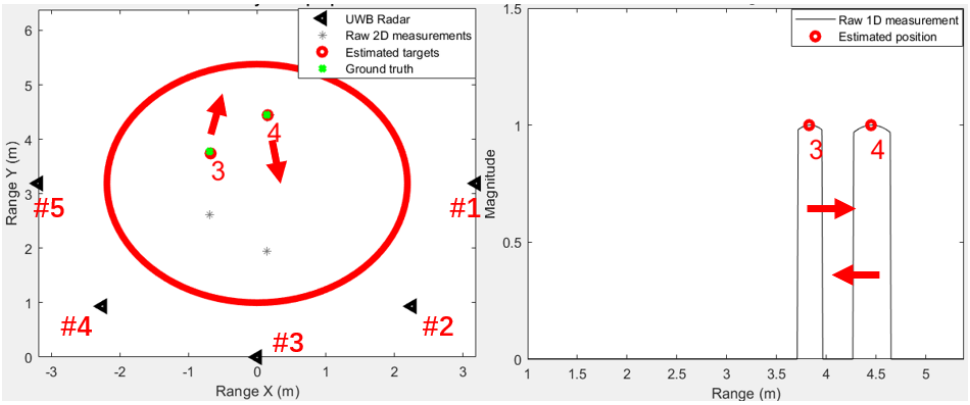


(a) The proposed clustering module.

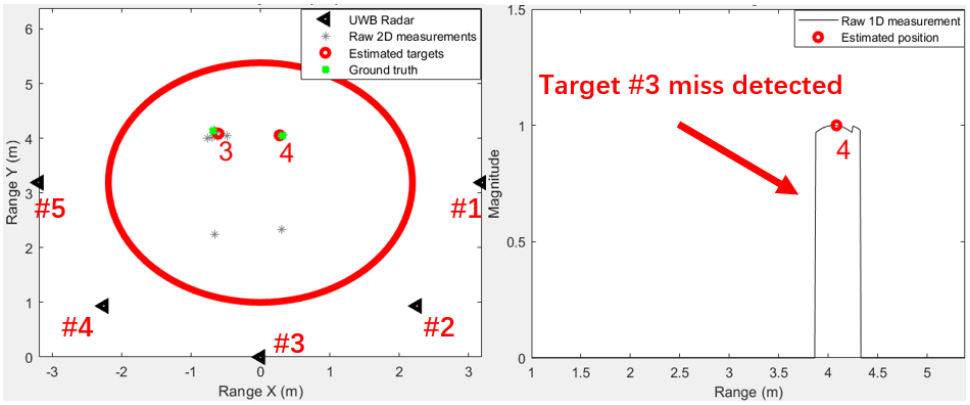


(b) The conventional clustering module.

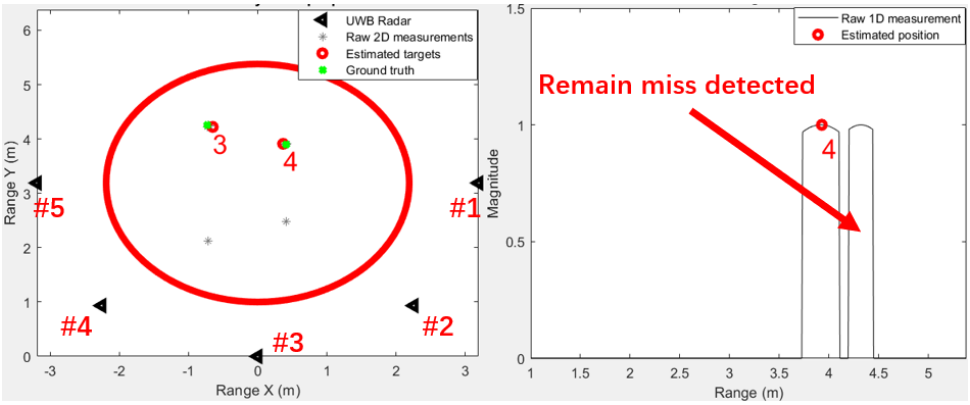
Figure 3.3: Tracking scenario C: three targets are presented in the measurement area moving from the left to the right (moving direction marked as dark arrow). In the above plots, the radar and clutter measurements are denoted as gray asterisks, the estimates provided by the multitarget tracker are denoted as red triangle, and the ground truths are denoted as green lines.



(a) $T = 89$

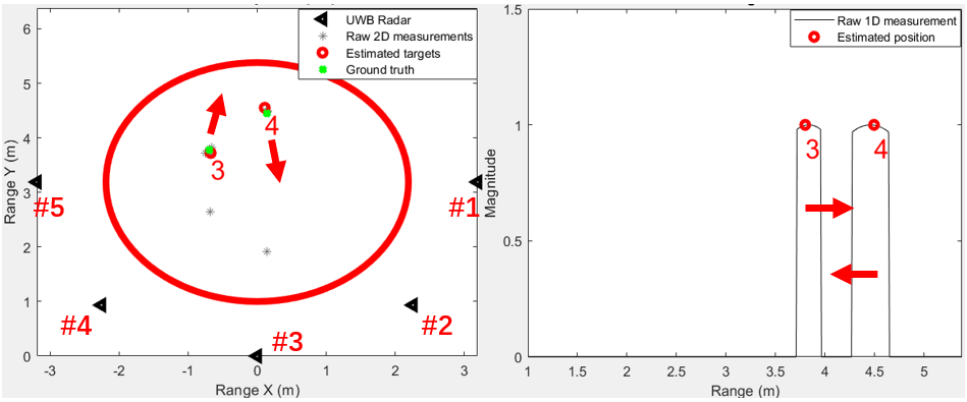


(b) $T = 92$

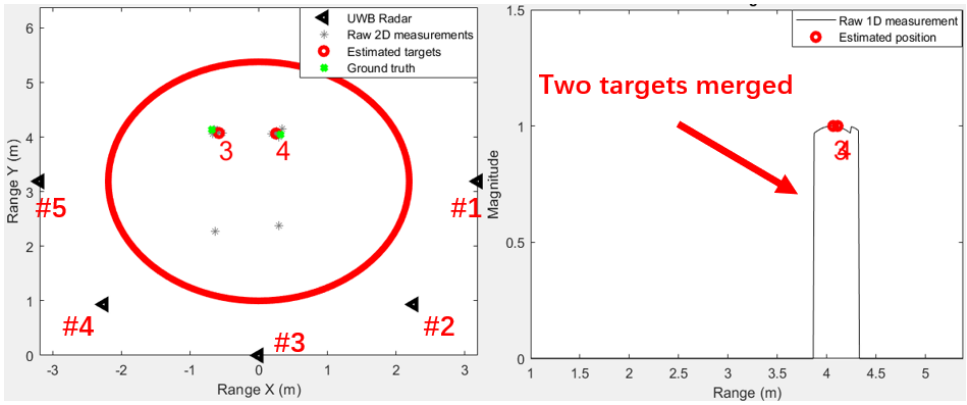


(c) $T = 94$

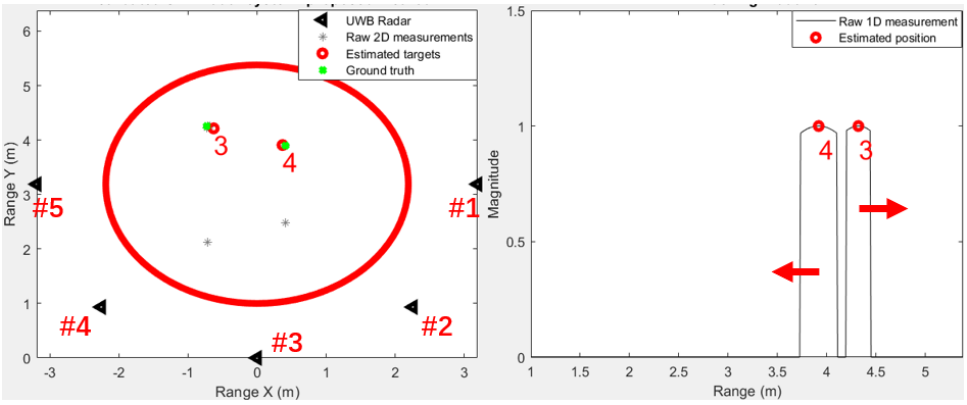
Figure 3.4: The conventional clustering module handles the 1D target merging scenario. The ground truth of the presented target is denoted as green dots, whereas the estimates are marked as red dots. The left graph shows the target's position in the Cartesian plane, and the right graph shows the target's 1D position sampled from radar #3. At time 89, two targets were presented in the measurement area, their moving directions are denoted as red arrows. The 1D measurements of the two targets merged at time 92. Finally, the two targets were separated at time 94.



(a) $T = 89$



(b) $T = 92$



(c) $T = 94$

Figure 3.5: The proposed clustering module handles the 1D target merging scenario. The ground truth of the presented target is denoted as green dots, whereas the estimates are marked as red dots. The left graph shows the target's position in the Cartesian plane, and the right graph shows the target's 1D position sampled from radar #3. At time 89, two targets were presented in the measurement area, their moving directions are denoted as red arrows. The 1D measurements of the two targets merged at time 92. Finally, the two targets were separated at time 94.

3.2. TRACKING-AIDED DETECTION FUSION CENTER

In this section, the evaluation result of the proposed detection fusion center is presented. The main objective of the detection fusion center is to transform the 1D measurements from different radar channels into the Cartesian plane. However, due to the measurement-to-measurement association uncertainty (discussed in Chapter 2.2.3), many false alarms may be generated during this process. The generated false alarms may severely influence the tracking accuracy. Moreover, they can increase the computational costs in the tracking and classification system.

Conventional approaches used to address this problem can be divided into two branches. One considers using the multistatic radar system for each radar node in the RSN [46]. In that case, the 2D locations of the presented targets can be directly calculated. Moreover, the association uncertainty is reduced by exploiting the echo's time-of-flight dependency. Another direction implements a global thresholding on the residual error generated by the least-square (LS) trilateration process [34, 43]. Compared to the previous method, it is not required for each radar node to have multiple receivers. However, this method is not able to reduce the association uncertainty.

Since the proposed MTT system uses the IR-UWB radar, in the following experiments, comparisons have been made between the proposed detection fusion center and the conventional LS-based method.

3.2.1. LEAST-SQUARE BASED METHOD

In this sub-section, the conventional detection fusion center is evaluated based on simulated multitarget data. The simulation setup is the same as detailed in Chapter 2.1. In the detection fusion center, all the measurement-to-measurement association hypotheses are considered. As the conventional method requires, the LS trilateration algorithm is used to localize the potential targets in the 2D plane.

Figure 3.6 shows three snapshots of simulated multitarget data processed by the conventional detection fusion center. The global threshold value, which is used to filter the 2D measurement with a high residual error, is set as 0.1. Since not all radar data are available during tracking, the LS trilateration uses four 1D measurements to calculate the 2D location. In this case, a target may become extended in the 2D plane if all radars had captured it. As you see, when only one target is presented in the measurement area (at time 12), no false alarm was introduced.

However, when the two targets (target #3 and #2) moved close to each other, as shown in Figure 3.6b, the combinatorial association of the 1D measurements resulted in two false alarms (false target #4 and #6) in the 2D plane. Based on the formulation of the LS trilateration method, it is understandable that the performance of the conventional method degrades when two targets are closely spaced. This is because the residual error of the incorrect association hypothesis will be small enough to pass the predefined threshold. This phenomenon can also be reflected in Figure 3.6c when two targets were

well separated. In that case, the conventional method performs well in suppressing the false alarm.

A further experiment is conducted to investigate the influence of the number of presented targets on the performance of the conventional method. As shown in Figure 3.7, three plots represent three different simulation settings, in which the maximum number of presented targets varies from one to five. Each plot shows the number of estimated targets (red line) versus the number of actual targets (black line). As demonstrated, the conventional detection fusion center provides a bad false alarm suppression performance if the number of presented targets is large.

Figure 3.8 served as another proof that shows the instability of the conventional detection fusion center. These experiments were conducted with the global threshold value set as 0.01, 0.1, and 1, respectively. Based on the three cardinality plots, it is evident that the performance of the conventional method is susceptible to the threshold value. Apparently, a small threshold value may lead to many missed detections, whereas a large threshold value may increase the number of false alarms since most residual errors can pass this threshold. However, it is hard to find the perfect threshold value due to the noise and estimation errors.

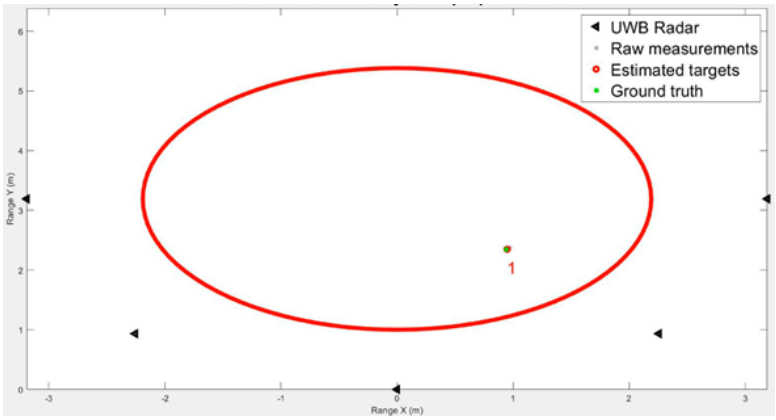
3.2.2. THE PROPOSED DETECTION FUSION CENTER

As is evident from the previous section, the performance of the conventional detection fusion center is susceptible to the predefined global threshold and the distance between different targets. Moreover, the conventional method is based on analyzing the residual error generated by the LS process, which means it does not reduce the association uncertainty directly.

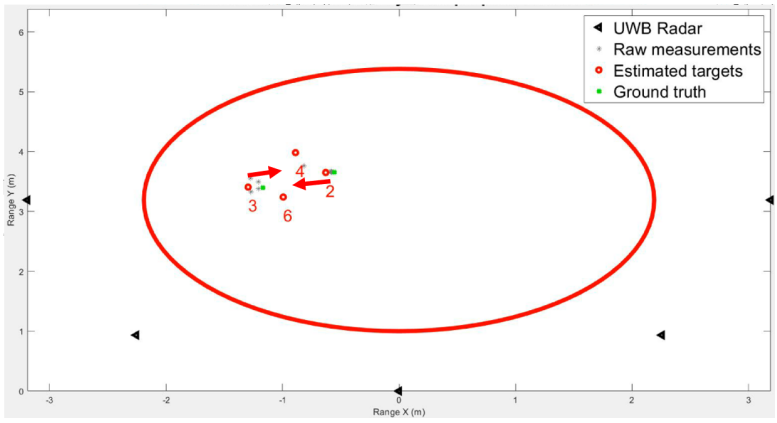
Thus, to achieve an excellent false alarm suppression performance, the conventional method often requires using an RSN with a large number of nodes. Even so, the thresholding method is not robust against the estimation error. For example, if one radar channel generates an erroneous 1D estimation, its corresponding 2D measurement may be removed due to a high residual error.

In this section, the proposed detection fusion center is evaluated using simulated multitarget data same as before. As shown in Figure 3.9, the proposed method performs well when only one target is presented (at time 12). For multitarget scenarios, when targets are well separated (e.g., at time 57), the proposed method has a similar performance as the conventional method (shown in Figure 3.6c). The performance of the conventional and proposed methods differs if the presented targets are close to each other. As shown in Figure 3.9b, the proposed method correctly estimated the actual cardinality of the ground truth set.

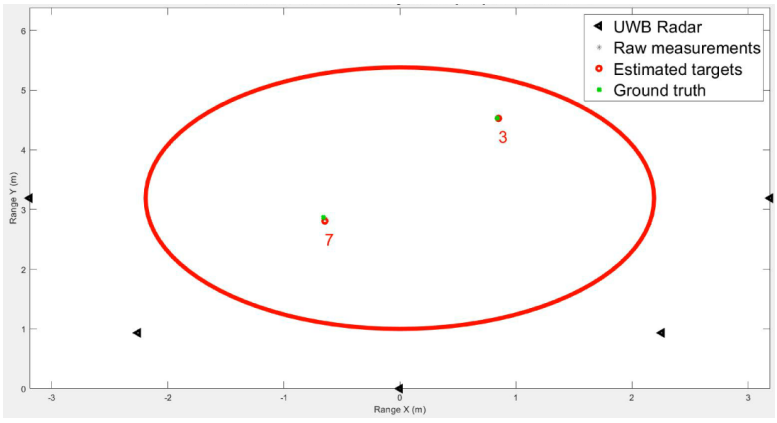
A more evident comparison between the proposed and conventional methods is presented in Figure 3.10. As you see, the proposed method outperforms the conventional method in terms of the GOSPA metric. Specifically, when multiple targets are closely



(a) $T = 12$



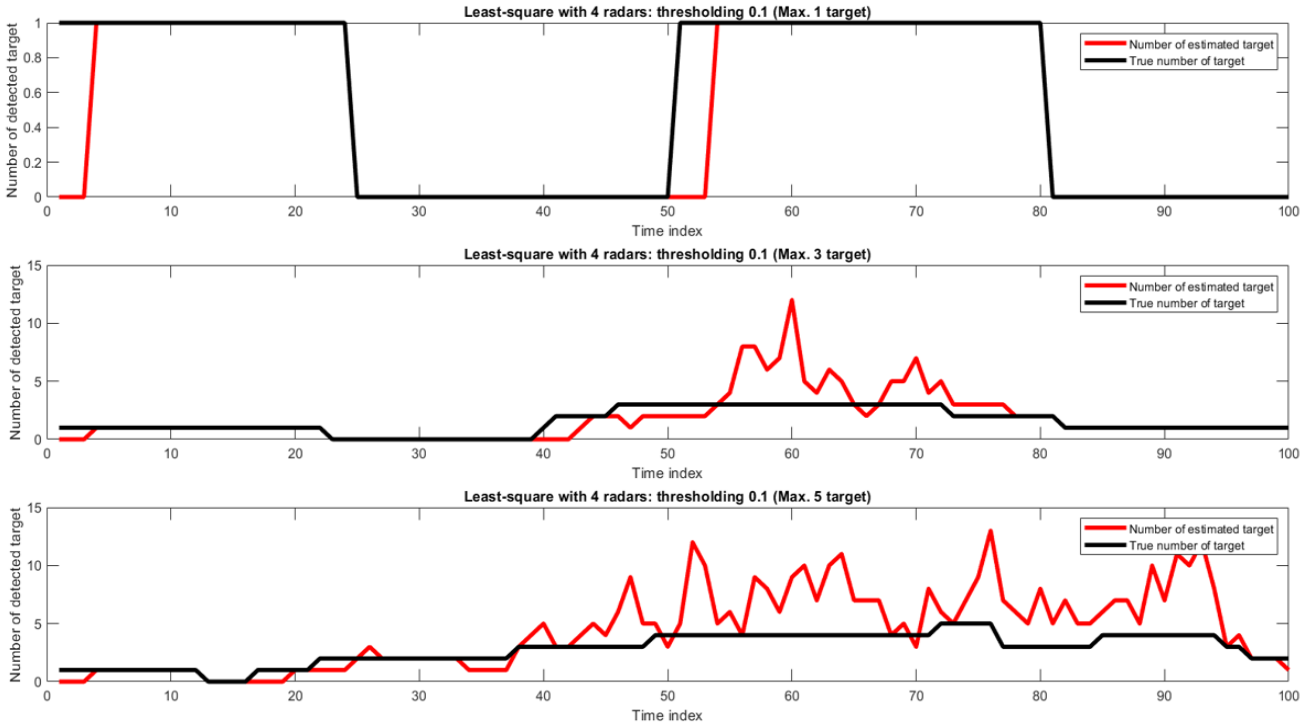
(b) $T = 42$



(c) $T = 57$

Figure 3.6: Three snapshots of simulated multitarget data processed by the conventional detection fusion center. The global threshold value is set as 0.1. The ground truths, the estimates, and the raw 2D measurements generated by the fusion center are denoted as green dots, red dots, and gray asterisks, respectively. At time 12, only one target is presented in the measurement area. At time 42 and 57, there are two targets inside the area but the distance between them is different.

Figure 3.7: The cardinality plots of three random simulations. Each simulation has a different setting, i.e., the maximum number of presented targets varied from one to five. The number of actually presented targets is denoted as a black line, and the number of estimated targets is marked as a red line. The global threshold value is set to 0.1. In other words, a 2D measurement with a residual error larger than 0.1 will be removed.



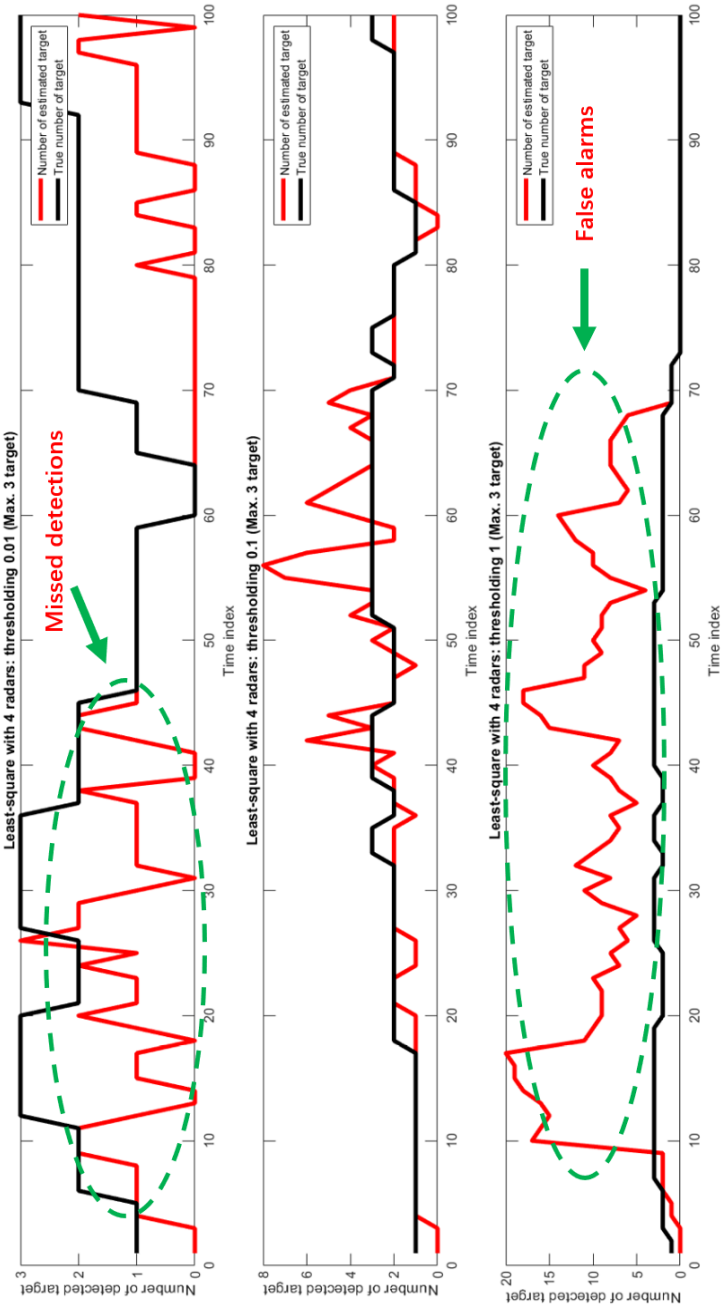


Figure 3.8: The cardinality plots of three random simulations. Each simulation has a different setting, i.e., the global threshold value varied from 0.01 to 1. The number of actually presented targets is denoted as a black line, and the number of estimated targets is marked as a red line. The maximum number of presented targets is set to three. In other words, at most, three targets can coexist in the measurement area at any time step during the simulation.

spaced, the proposed method has fewer missed detections and false alarms than the conventional. However, it is important to note that the conventional method is still acceptable in simple multitarget tracking scenarios since it can achieve similar performance as the proposed one.

One significant advantage of the proposed method is it is parameter-free. Thus, it is more robust against different tracking scenarios. Comparing to the conventional, the proposed method uses the predicted state information to reduce the association uncertainty directly. Moreover, it does not have a requirement of using a specific trilateration method. To further improve the false alarm suppression performance, it is possible to integrate the proposed method with the conventional.

3.3. MULTITARGET TRACKING

This section presents the evaluation results of the proposed MTT system for tracking a variable number of targets. Since the set of actual targets and estimates are random finite sets, to measure the average distance between them, the root mean square (RMS) GOSPA metric [23] is used.

Table 3.1 shows the system performance measured by the RMS-GOSPA metric under four experiments. Each experiment has a predefined global variable that sets the upper limit for the maximum number of targets allowed to co-exist in the measurement area. For a given experiment, the RMS-GOSPA metric applied four times to test the performance under different system settings.

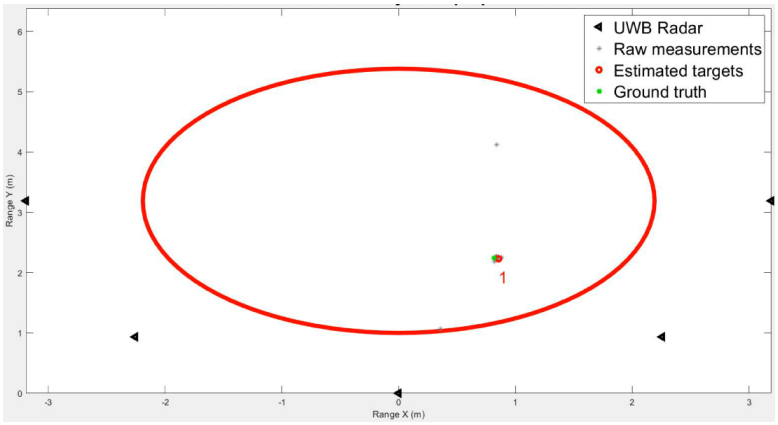
The "Proposed Clustering (ON)" means the subpartitioning technique is activated to help the system separate the merged target in the 1D and 2D planes. Similarly, the "Proposed Fusion center (ON)" means the predicted 2D target information is used to reduce the data association uncertainty before trilateration. "ALL ON" means all the proposed methods are activated, whereas "ALL OFF" indicates all the proposed methods are disabled.

Based on the experiment results, the following observations can be made:

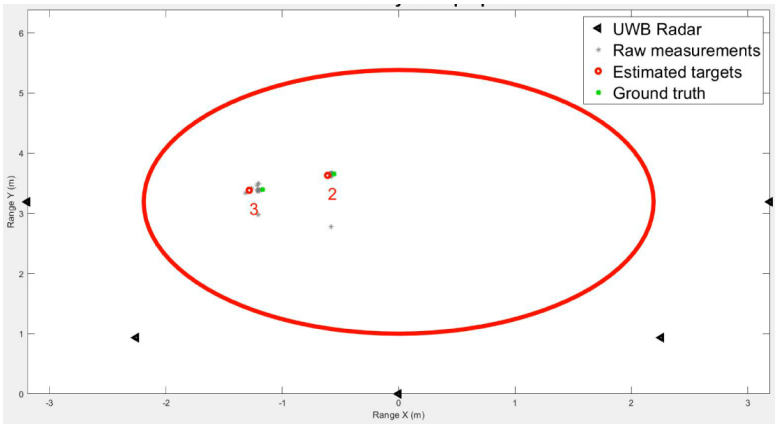
1. In the "Max 1 Target" case, the performance gain provided by the proposed modules is trivial (only 1%). This is because the uncertainties in the measurement partitioning and data association step are small since at most one target is allowed to enter the measurement area.

2. As the maximum allowable targets increase from one to four, the GOSPA error under different system settings also increased. This reflects the difficulties in the multitarget tracking have been raised.

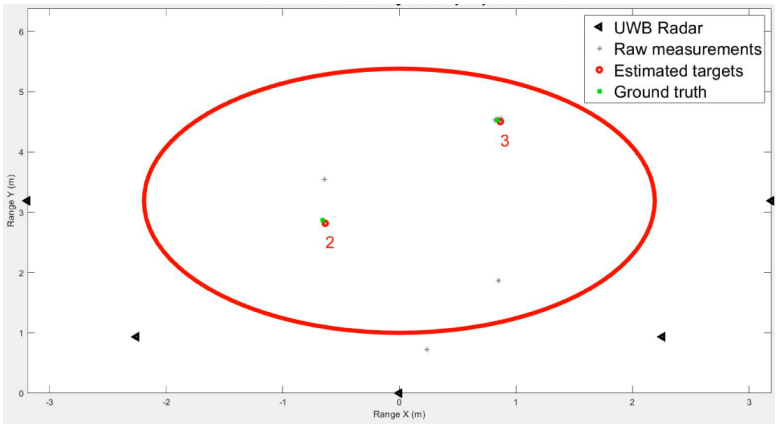
3. The system performance under the "ALL ON" setting outperforms the "ALL OFF" setting. The percentage improvement becomes significant as the maximum allowable targets increases. In the "Max 4 Target" case, the proposed modules provided



(a) $T = 12$



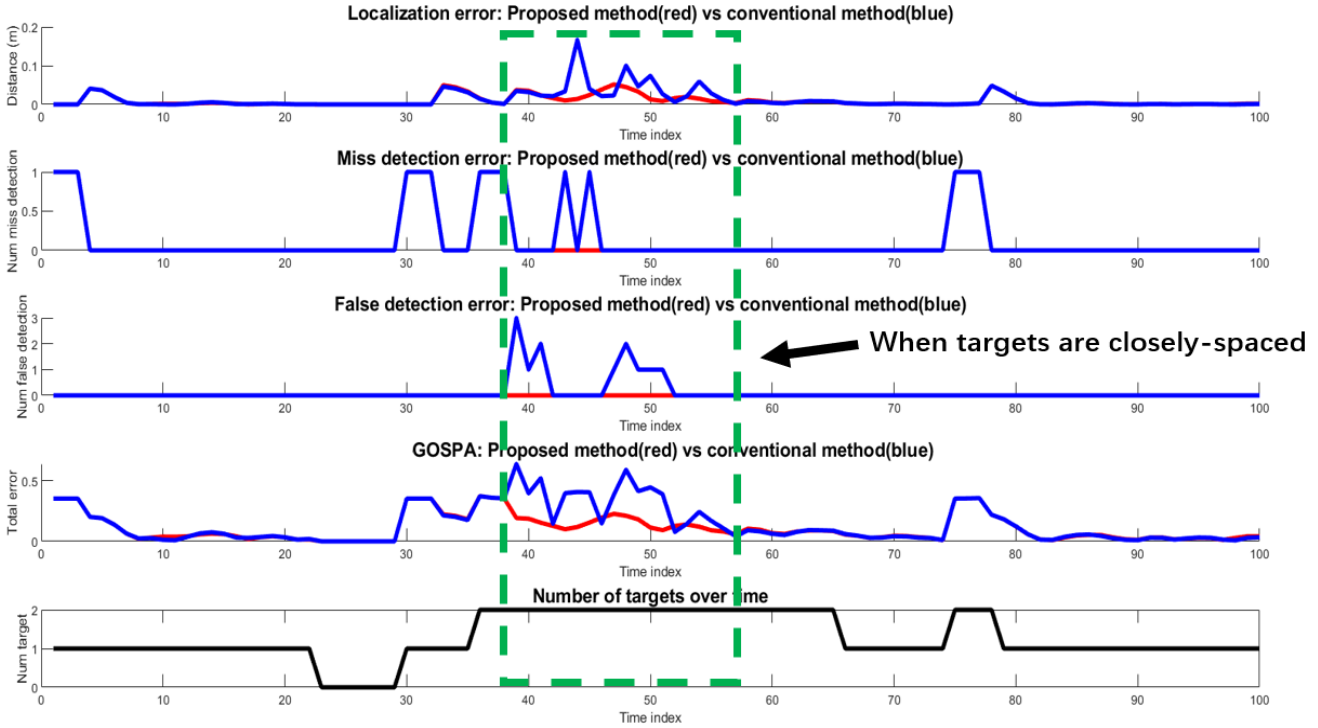
(b) $T = 42$



(c) $T = 57$

Figure 3.9: Three snapshots of simulated multitarget data processed by the proposed detection fusion center. The ground truths, the estimates, and the raw 2D measurements generated by the fusion center are denoted as green dots, red dots, and gray asterisks, respectively. At time 12, only one target is presented in the measurement area. At time 42 and 57, there are two targets inside the area but the distance between them is different.

Figure 3.10: A performance comparison between the proposed and the conventional detection fusion center. The plots from the top to the bottom shows the localization error, the number of missed detections, the number of false alarms, the evaluation result of the GOSPA metric, and the number of actual presented targets, respectively. For each plot, the proposed method is marked as red curve, and the conventional method is denoted as blue curve.



a 19.7% performance gain for the MTT system. However, it is important to note that the measured gain only represents a general performance increase. Although for a given experiment, all settings are tested on the same set of simulated data, due to the random process, it is impossible to generate the target merging problem and keep four targets in the measurement area at all time steps.

4. In the "Proposed Clustering (ON)" setting, where only the proposed clustering module is activated, the system performance deteriorated severely compared to the "ALL OFF" setting, especially in the "Max 4 Target" case. To understand this phenomenon, it is important to remember that the GOSPA metric penalizes the localization error and the missed and false targets. Since the proposed clustering module can separate the merged measurements, it will add more association uncertainties to the detection fusion center. However, the proposed fusion center is disabled under the "Proposed Clustering (ON)" setting. Thus, the performance degradation is mainly caused by the introduced false alarms. The huge leap of the GOSPA error across the four experiments (from 0.3191 to 3.1259) also reflects that the target merging problem has a high incidence rate when the number of presented targets is large.

5. In the "Proposed Fusion Center (ON)" setting, where only the proposed detection fusion center is activated, the RMS-GOSPA errors are similar to those in the "ALL OFF" setting. This is because the increased miss detection rate offsets the performance gain provided by the reduced false alarms. For example, when two targets' 1D measurements were merged, the proposed fusion center reduces the association uncertainty by assigning the merged measurement to one of the two targets, which leads to another target miss detected.

6. As shown in the results from the previous sections, each of the proposed methods works well for addressing a specific problem in MTT. However, based on the above observations, it is evident that the proposed solutions can not be applied solely. This is because the proposed clustering module separates the merged targets but increases the association uncertainties in the fusion center; The proposed fusion center reduces the association uncertainties but may cause one of the merged targets to be miss detected. Therefore, it is recommended to apply these two solutions together in the MTT system.

3.4. FEATURE EXTRACTION

In this section, the proposed MTT system is applied to process experimental radar data sampled by five IR-UWB radar sensors. A detailed description about the geometry of the radar sensors can be found in Part III, Chapter 2.1. For the radar deployment, five radar sensors were placed exactly as the used radar positions in simulated RSN. To detect the target and suppress the false alarms, the ordered-statistic constant false alarm rate (OS-CFAR) [39] is added to the signal processing pipeline.

RMS-GOSPA	Proposed Clustering (ON)	Proposed Fusion Center (ON)	All ON	All OFF	Percentage Improvement
Max 1 Target	0.3191	0.3159	0.3160	0.3191	1.0%
Max 2 Target	0.7772	0.6352	0.5621	0.6007	6.4%
Max 3 Target	2.2749	0.9084	0.7471	0.8592	13.0%
Max 4 Target	3.1259	1.1254	0.9714	1.2095	19.7%

Table 3.1: RMS-GOSPA metric averaged over 1000 realizations. The maximum number of actual targets is fixed for a given experiment, but the maximum number varies across different experiments. For the notation, the "ON" indicates that only the specified component is enabled. For example, the proposed clustering (ON) means the proposed 1D and 2D clustering module is used, and ALL ON means both the clustering module and the detection fusion center are activated. The percentage Improvement is calculated by subtracting the RMS-GOSPA value of ALL ON from ALL OFF, and then divided by the ALL OFF.

The feature extraction process has been discussed in Chapter 2.2.5. Simply speaking, if a target is successfully detected, the locations of the 1D range bins that lead to the detection are immediately known by the feature extraction module since the set of 1D and 2D measurements are labeled during the propagation through the signal processing pipeline. However, if a target is miss detected but still regarded as a reliable target, the distance between the target to the five radars will be calculated based on its updated 2D location. Finally, if a target is miss detected and removed from the set of reliable targets, no Doppler information will be extracted for that target.

Figure 3.11 shows an example of the output of the feature extraction module. Five spectrograms were generated, each of which belongs to one of the five radar channels. At a given time step, the outputs of the feature extraction module are the small spectrograms marked by the black rectangles. As time moving forward, a newly extracted Doppler vector will be added to the small spectrogram, whereas the oldest information will be removed to keep the size of the spectrogram fixed. As one may expect, the outputs of the feature extraction module have the same format as the inputs of the proposed recognition system presented in Part III, Chapter 2.2.2.

However, due to the imperfect tracking, it is important to note that there are two common problems in the extracted spectrograms. First, the red arrows in Figure 3.11 denote a global miss detection that happened to all radar channels. The miss detection may be caused by the activity of "falling on the ground" or "bending from standing" performed by the participant. In case of "falling", suppose the participant stayed on the ground for a longer period. In that case, the MTT system may already removed the missed target from the tracking list.

The green arrow denotes another problem, the local miss detection, which happens to one of the radar channels. The reasons that led to this problem can be miscellaneous, for example, occlusions or low SNR. Thanks to the RSN, even though one radar channel had a missed detection, the other radar sensors were able to capture the target's movement.

Not only the above problems will increase the difficulty of tracking multiple targets but recognizing human activity. For the first problem, it is desirable if the recognition system can utilize both the past and future information to make the predictions for the activities when global miss detection happened. For the second problem, it is important for the recognition system to learn how to select the useful information from the five spectrograms.

Figure 3.12 shows one more example of the output of the feature extraction module. The processed radar data contains only the "human walking " activity. Clearly, the extracted spectrogram is more continuous and less influenced by the miss detection.

3.5. SUMMARY

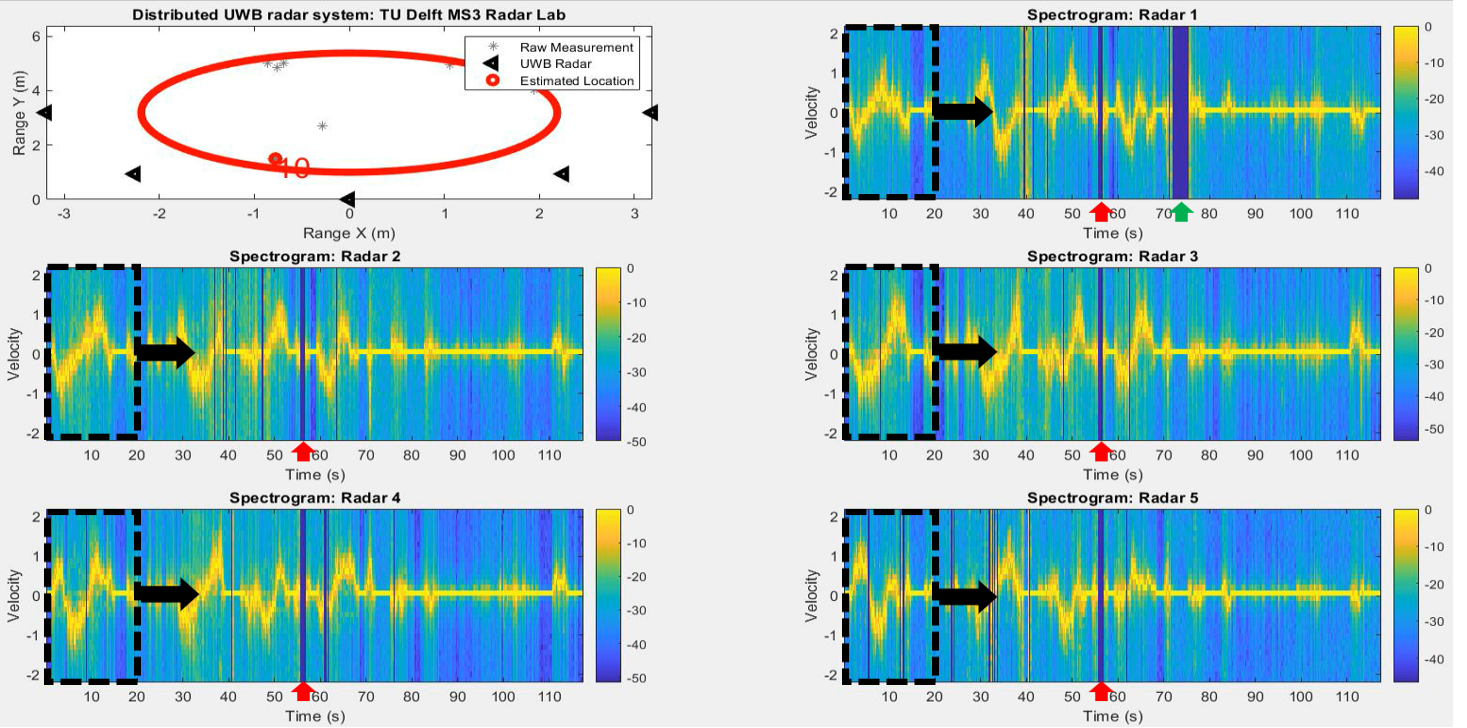
In this chapter, the evaluation results of the proposed MTT system is provided.

In Section 3.1, the validity of the proposed tracking-aided clustering module is proved. First, the proposed solution is tested independently to show its feasibility in separating merged measurements in the 2D plane. Three different target merging scenarios were simulated. The result shows that the proposed method can maintain the tracks for each of the merged targets. Then, the clustering module is tested in the suggested MTT system. The result shows its ability to separate the merged measurements. Moreover, the simulation also reflects that the 1D target merging problem may have a high incidence rate due to the use of IR-UWB radar.

Section 3.2 provides the evaluation result of the proposed tracking-aided detection fusion center. As you understand, the proposed solution can directly reduce the association uncertainties in the detection fusion center by exploiting the predicted target information from the multitarget tracker. Comparing to the conventional method based on thresholding the residual error generated by the LS trilateration process, the proposed solution is parameter-free. It shows a better performance in terms of GOSPA metric (see Figure 3.10). Moreover, since the proposed method does not require the fusion center to use a specific trilateration algorithm, combining the proposed solution with the conventional one is possible.

In Section 3.3, the overall performance of the proposed MTT system in tracking a variable number of targets is presented. Due to the use of the decentralized signal processing architecture and the advanced clustering and tracking algorithms, the MTT system is already able to track multiple targets even without the suggested modifications. To further prove the advantages of the proposed solutions, a set of experiments were

Figure 3.11: An example of the output of the feature extraction module. The proposed MTT system is applied to process a recording of real radar data that contains a set of continuous human activities. The extracted spectrograms from the five radar channels are presented.



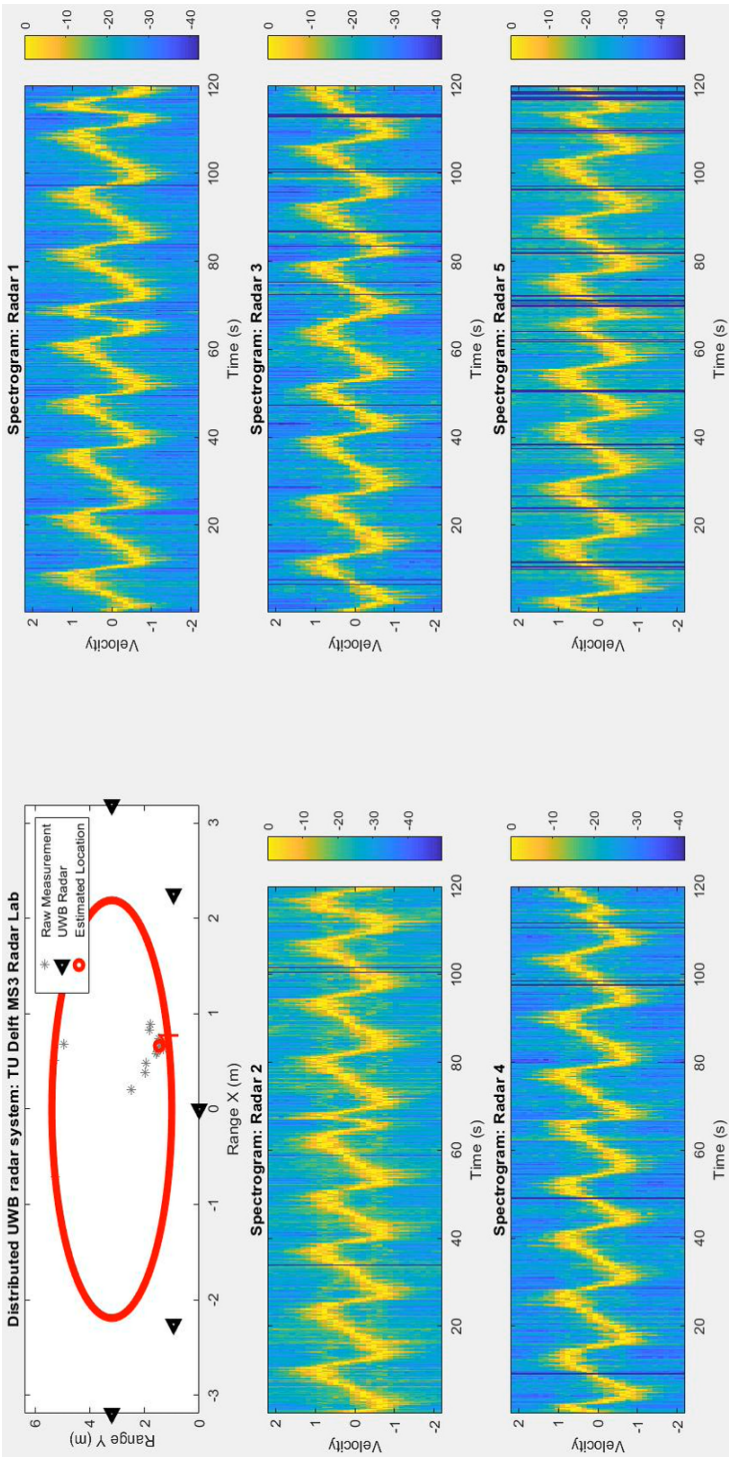


Figure 3.12: Another example of the output of the feature extraction module. The proposed MTT system is applied to process a recording of real radar data that contains the human walking activity. The extracted spectrograms from the five radar channels are presented.

conducted and the RMS-GOSPA error was measured. The result shows a maximal 19.7% reduction in RMS-GOSPA error when the suggested solutions were applied.

Finally, the output of the feature extraction module is presented in Section 3.4. The generated spectrograms have the same format as required by the proposed recognition system. In other words, it is possible to conduct a joint human tracking and activity classification. Moreover, two problems caused by global and local miss detections were observed in the spectrogram. It is obvious that these problems added up the difficulties for the recognition system.

4

CONCLUSION AND FUTURE WORK

In this chapter, a summary regarding to the proposed multiple target tracking (MTT) system is provided. More specifically, Section 4.1 concludes the contributions that were made in this work. The interesting future directions are discussed in Section 4.2.

4.1. CONCLUSION

The main contributions of this thesis work in Part II can be summarized as the followings:

1. **Multiple Extended Target Tracking and Feature Extraction**
   The proposed MTT system is able to track multiple extended targets in the Cartesian plane. It is built based on an IR-UWB radar sensor network (RSN). The RSN improves the tracking robustness by providing a multi-perspective view on the presented targets. Besides, a decentralized tracking architecture is implemented to improve tracking accuracy and reduce the number of false alarms. The results have shown its ability to track multiple simulated targets. Furthermore, the proposed system is tested on experimental radar data, and the Doppler information of the moving target has been extracted successfully. Therefore, it paved the road for the future investigation of designing a joint system for target tracking and activity classification.

2. **Two Problems and Two Solutions**
   Other than tracking targets, two problems that were rarely explored in the IR-UWB radar-based sensor network are investigated. The first problem relates to the measurement merging effect due to the use of the clustering algorithm. The corresponding solution for addressing this problem is provided. The provided solution

was originally used to handle the 2D measurement merging problem for Lidar-based tracking. However, it is evident based on the simulation result that the 1D merging problem has a higher incidence rate, and it may lead to missed detections and inaccurate estimations of the center point of a cluster.

The second problem is caused by the data association uncertainty in the detection fusion center. As was shown, the combinatorial association of the 1D measurements across different radar channels can generate many false alarms. The proposed method, which uses the predicted target information, can directly reduce the number of association hypotheses. Comparing to the conventional thresholding method, which exploits the residual error generated by the Least-square (LS) trilateration process, the proposed method is more robust and parameter-free. Moreover, as the proposed method does not require the fusion center to used a specific trilateration method, it is possible to combine the proposed method with the conventional one.

The evaluation results have shown that the proposed solutions are effective in solving these two problems separately. Moreover, the comparison between the MTT systems with and without the proposed methods is made. When at most four targets were allowed to co-exist in the measurement area, the result shows that the proposed solutions led to a 19.7% reduction in the RMS-GOSPA error compared with the conventional.

3. **Simulation Model and System Evaluation**
   A simulation model is provided in this thesis work for investigating MTT problems. The proposed model uses various models to describe different uncertainties, such as the target's birth and motion, target measurement, and clutter measurement. In this thesis work, the simulation model is used to test the performance of the proposed MTT system. However, with some modifications, it is possible to evaluate other aspects of radar-based tracking, for example, the influence of different radar deployment geometries. Regarding the system evaluation, unlike previous works which focus on the system's localization accuracy only, the GOSPA metric is used to measure the system error from three different aspects (i.e., the localization error, miss, and false targets).

4.2. FUTURE WORK

Limited by the scope of this thesis work, there are two interesting directions that could be further investigated in the future:

1. **Target Detection**
   One significant aspect that was not discussed in this work is the target detection techniques. By using simulated radar data, the detection step was skipped. However, the detection procedure directly influences the miss detection and false alarm rate of the MTT system. Conventionally, the target detection is done by using the adaptive thresholding methods such as the CA-CFAR and OS-CFAR [39]. However,

due to the characteristics of the IR-UWB radar, a constant false alarm rate might lead to a high miss detection rate for targets at distance or a high false alarm rate in the region close to the radar sensor [31].

To address this problem, there are two possible solutions. One is to utilize the high data rate feature of the IR-UWB radar to increase the detectability of the weak target, for example, using the Hough transform-based track-before-detect technique for detecting weak targets [92, 93].

Another solution could consider using tracking information to learn the measurement distribution of the target and noise. Generally, this solution can be separated into two steps. The first step is for training, in which a multitarget tracker is applied to pre-collected radar data to extract the target and noise distributions at different range intervals. The preliminary result shows that as the distance between the target and radar increases, the noise and target distribution become more overlapped. In the second step, multiple thresholds for different ranges are calculated based on a trade-off between the desired miss detection and false alarm rate. These learned thresholds can be stored in the memory and applied to detect unseen targets. Moreover, it is possible to keep updating these thresholds while tracking unseen targets.

2. **Processing Real Data**
   In this thesis work, simulated radar data is used to analytically measure the performance of the proposed MTT system. However, limited by the time and computational power, the simulation model can not perfectly reflect all the aspects in multitarget tracking. Thus, it might be interesting to deploy the proposed system to process experimental radar data.

III

PART III -HUMAN ACTIVITY RECOGNITION SYSTEM

1

Related Work

This chapter focuses on the recent progresses in radar-based human activity recognition (HAR) systems for monitoring the activities of daily living (ADLs). To have a clear picture of the motivations behind the literature and this thesis work, the research challenges that need to be solved for the recognition systems are discussed first in Section 1.1. Then, the approaches proposed in previous works to addressed these challenges are studied in Section 1.2. Finally, with a discussion over the gaps between the literature and the challenges, the main contributions of this thesis work are presented in Section 1.3.

1.1. Research Challenges

Over the past decades, radar-based HAR systems have gained massive attention in applications such as personnel recognition [94, 95], hand gesture recognition [96, 97], and fall detection [98, 99]. In terms of monitoring ADLs [100, 101], although many significant improvements have been made, it is a challenging task due to the fact that:

1. **Requirement for Feature Engineering**
   The conventional recognition systems often require additional pre-processing steps, e.g., feature extraction and selection, to generate handcrafted features from the raw radar data. However, these steps are less efficient, and prior knowledge is needed to guarantee good system performance.

2. **Challenges for Classifying Real Human Activities**
   Human movements are continuous in nature, with seamless inter-activity transitions that can happen at any time during the movement. Furthermore, translational and in-place actions such as 'walking' versus 'sitting down' will likely have variable durations. However. to reduce the complexity, conventional recognition

systems often assume that the target can only perform different activities separately and independently during the radar measurement. Obviously, this assumption is not realistic.

3. **Modeling Spatial-temporal Characteristics**
   For most HAR systems, the spectrogram, generated by implementing a short-time Fourier transform (STFT) on the raw radar data, is used as the input data for the recognition system. Conventional machine learning approaches treat the input data as images and explore the local connections in the data. However, the spectrogram contains both temporal and spatial characteristics, especially for continuous human activities, where long-term temporal dependencies may exist. It is an ongoing research on how to design the system architecture so that the two crucial characteristics of human activity can be well exploited.

4. **Varying Aspect Angles**
   The trajectories of human activities are generally arbitrary, which means the direction of the target's movement is unconstrained. Thus, the performance of the recognition system should not be influenced heavily by the angle between the target's moving direction and the radar line-of-sight.

5. **Limited Dataset**
   Acquiring training examples from the radar sensors is not as simple as creating the image or audio dataset due to the costs and required resources for data collection. For HAR, a training set usually contains thousands of examples sampled from tens of participants. Consequently, there are two research challenges. The first is how to efficiently use the limited dataset for system training and evaluation, especially to test the generalization capability of the proposed system for unknown target. The second is how to design the recognition system so that it has a low model complexity.

1.2. RECENT ADVANCES

For rigid targets such as aircraft and vehicles, the target classification is usually solved by utilizing different motion models [102] or known shape attributes [103]. For nonrigid targets, popular classification methods are based on analyzing the micro-Doppler signature (MDS) [6].

The MDS reflects the micro motions of the target's movement, it is more unique and informative than the motion and shape attributes. The MDSs have been used to differentiate different types of targets[104], or distinguish different human activities [16]. However, establishing the correct mappings between different MDSs and class labels is still a challenging task.

In the research area of HAR, Kim et al. [16] presented one of the first works that proved the feasibility of classifying different human activities based on the Doppler spectrogram. The recognition system they proposed uses a support vector machine (SVM)

[105] trained on a set of handcrafted features to classify seven types of human activities. The results have shown promising potentials of using radar sensors for HAR.

To remove the additional feature extraction and selection steps, Kim et al. [17] proposed an alternative deep learning-based recognition system. The system takes the raw Doppler spectrogram as input data and applies convolutional neural network (CNN) directly to the spectrogram without using any handcrafted features. The results indicate that CNN can automatically extract features from the input data and provide an end-to-end solution for HAR tasks.

However, CNN or other similar variants treat the input spectrogram as an optical image and perform convolution by sliding filters over it. Although CNN has shown its ability to capture the spatial-temporal correlations in the spectrogram for speech recognition problems [106], many convolutional layers are needed to help the system achieve more nonlinearities and sizeable input receptive fields [107]. Consequently, it will increase the model complexity and the requirement for the number of training data. Besides, CNN often requires the input spectrograms to have the same time duration and contain only one type of the activities.

To address these issues, Jiang et al. [108] proposed a system that uses recurrent neural network (RNN) for activity recognition. RNN takes the spectrogram as continuous time series. The results have shown that RNN can learn the temporal dependencies in the radar data and reduce the computational load. RNN was also used in work [19], where a bi-directional implementation was proposed. The bi-directional recurrent neural network (Bi-RNN) can utilize both forward and backward temporal information within the radar data for the prediction. The results have demonstrated that the Bi-RNN outperforms the uni-directional RNN in terms of classification accuracy.

Nevertheless, one significant disadvantage of RNN is the inefficiency of extracting complex feature patterns from the Doppler spectrogram. Thus, additional steps to preprocess the raw spectrogram are usually required [109, 110]. To take advantage of both CNN and RNN, the hybrid CNN-RNN architecture was proposed in [111, 112] for speech recognition tasks. Recently, the hybrid network was introduced to handle HAR problems in [113, 114]. The hybrid network uses CNN to extract the spatial information and RNN to capture the temporal dependencies in the input data, leading to a simpler network with better performance and fewer trainable parameters than using only one type of architecture [113].

Thanks to the hybrid CNN-RNN structure, the recognition system can automatically extract the spatial-temporal characteristics from the spectrogram. Moreover, there is no limitation for inter-activity transition and classifying activities with different lengths. More importantly, the requirement for acquiring a large dataset for model training is reduced. However, there is one crucial issue that limits these micro-Doppler signature-based HAR systems from being deployed to the real world. The trajectories of human activities are usually arbitrary and unconstrained. It has been shown in [115] that the system performance is highly dependent on the aspect angle, and the classification accuracy deteriorates as the aspect angle increases from 0° to 90°.

One effective solution to reduce the influence of unfavorable aspect angles is to use the distributed radar sensor network (RSN) [18]. A RSN consists of several radar nodes. According to a pre-designed geometry, these radar nodes are placed at different locations to monitor human activities simultaneously. Hence, there is a higher chance that at least one of the radar nodes can capture the human movement from a favorable aspect angle.

With the use of RSN, it becomes necessary to look into the impact of different radar geometries and information fusion methods. A simulation framework was proposed in [22] to benchmark the activity recognition performance under different radar deployment geometries. Furthermore, several data fusion strategies have been explored and compared in [100, 110]. The results indicate that using distributed radar system together with information fusion significantly outperforms the use of single radar.

1.3. MAIN CONTRIBUTIONS

Many efforts have been made to address different research challenges motioned in Section 1.1 for the radar-based HAR system. However, previous works focused only on individual problems. This thesis work made one step further to fill the gaps between the literature and the desired recognition system. The main contributions are summarized as follows:

1. **An End-to-end Solution for HAR**
   This thesis proposed a neural network-based human activity recognition system that addresses all the challenges discussed before. Furthermore, thanks to the deep learning approaches, the proposed system offers an end-to-end solution for continuous human activity recognition without constrains on target trajectories. For system evaluation, the proposed system is trained and tested on real radar data. The results have shown a significant performance improvement comparing to the previous works.

2. **Neural Network-based Data Fusion**
   A thorough hyperparameter search is conducted to find the best system configurations and model architectures. Moreover, different neural network-based fusion strategies have been investigated and implemented. The results indicate that the halfway fusion, or feature fusion in other words, gives the best result. To analyze the benefits of using the distributed RSN, the proposed system is also compared to the case when only a single radar sensor is used.

3. **Comprehensive Model Evaluation**
   To use the limited dataset more efficiently, this work proposed a model training and evaluation strategy. The strategy combines the popular K-fold cross-validation method and the leave-one-person-out (L1PO) method. Comparing to the conventional approach, the proposed evaluation method can measure the generalization capability and robustness of the recognition system and output a more accurate

validation result.

4. **Designed for Joint Tasks**
   Unlike previous works which only investigate the HAR problem and leave the target tracking problem aside, the proposed recognition system is designed in conjunction with the proposed tracking system under the same experimental and simulation setup. That is to say, the outputs and inputs to both systems are matched. Therefore, the proposed recognition system can be deployed with the proposed multi-target tracking system to conduct a joint multi-target tracking and classification.

2

METHODOLOGY

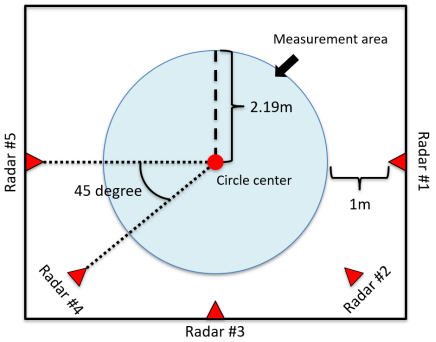
In this chapter, the proposed human activity recognition (HAR) system is presented. Section 2.1 illustrates the experimental setup for data acquisition. The collected dataset is used for model training and evaluation. Then, Section 2.2 explains the design details about the proposed recognition system. The system uses a hybrid neural network model to conduct automatic feature extraction and activity prediction. After that, the model training and evaluation strategy is given in Section 2.3. Lastly, Section 2.4 summarizes this chapter.

2.1. EXPERIMENTAL SETUP

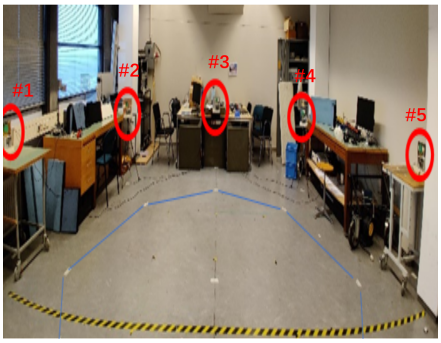
The experiment was conducted in the radar laboratory at TU Delft. As shown in Figure 2.1, the proposed radar sensor network (RSN) contains five identical impulse-radio (IR) ultra-wideband (UWB) radars working in the monostatic mode. The circular geometry was used, and each radar is placed 45° apart with regard to the center of the circle. The measurement area is also a circle with a radius of 2.19 m. The radar settings and geometry of the RSN is the same as [100].

All the human activities are conducted inside the measurement area without restrictions on the direction of the target's movement. That is to say, every target is allowed to choose their trajectory while performing a set of activities. Comparing to the previous works, e.g., [17, 116], where the target's action is either constrained in fixed trajectories or artificially separated by different types of activities, the arbitrary trajectory and continuous nature of the proposed dataset make the classification task more challenging.

To measure the system performance for human activities of daily living (ADLs), nine types of human activities are designed, including: (1) walking, (2) stationary, (3) sitting



(a) Schematic diagram of the radar deployment geometry.



(b) The image shows the locations of the radar sensors (marked by red circle) in the Radar Lab of TU Delft.

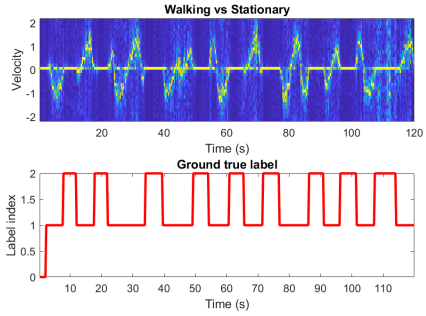
Figure 2.1: The layout of the radar sensor network, where the radar sensors are circularly placed around the measurement area.

down, (4) standing up from sitting, (5) bending from sitting, (6) bending from standing, (7) falling while walking, (8) standing up from the ground, and (9) falling while standing. The same types of activities were also used in [100]. Table 2.1 shows the size of the recorded dataset and the corresponding label for each action type. As was shown, the dataset is imbalanced. The "walking" action has the highest occurrence with 32350 recorded examples, while the "falling while walking" has the lowest occurrence with only 3618 examples. It is natural to have an imbalanced data distribution across different activity types, given the nature of the human movement. However, it makes the HAR task more challenging, especially for activities with a small dataset.

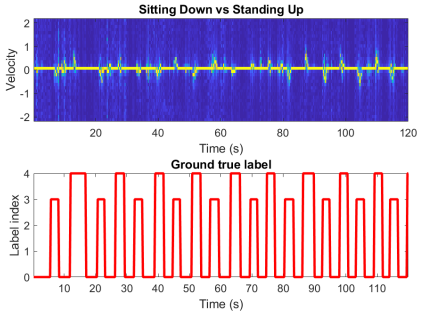
Label Index	Action Type	Number of recorded data
1	Walking	32350
2	Stationary	16114
3	Sitting Down	5580
4	Standing up from sitting	5085
5	Bending from sitting	12823
6	Bending from standing	14026
7	Falling while walking	3618
8	Standing up from the ground	12612
9	Falling while standing	5762

Table 2.1: The label index and number of recorded radar data for each action type.

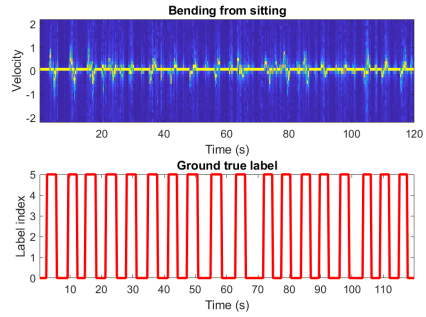
There are fourteen participants involved in the data acquisition process. During the radar recording, a combination of different activities is conducted by one of the participants and recorded by the proposed RSN. Each recording lasts for 120 seconds (or 2 minutes). For illustration, the spectrograms of the nine activities recorded by radar #3 and the labels for the ground truth are presented in Figure 2.2.



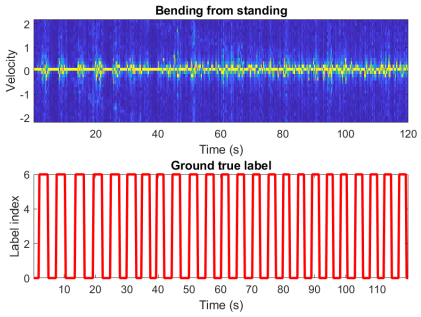
(a) Walking vs Stationary.



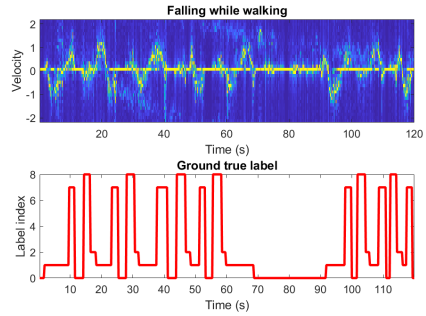
(b) Sitting Down vs Standing Up.



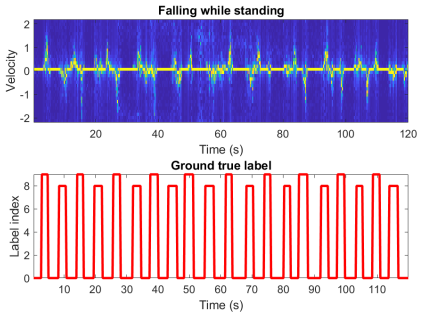
(c) Bending from sitting.



(d) Bending from standing.



(e) Falling while walking.



(f) Falling while standing.

Figure 2.2: The spectrograms of nine types of human activities recorded by radar #3. (Label "0" represents all irrelevant human activities that happened during data collection. For example, in Figure (b), after "sitting down" the target may change its position before "standing up", the time interval between its change position is labeled as irrelevant. Thus, the label "0" and the corresponding spectrograms will be discarded in late processing stages)

2.2. PROPOSED RECOGNITION SYSTEM

In this section, The proposed HAR system is presented. The system is based on a hybrid neural network model with several preprocessing steps that handle the raw radar

data. The hybrid model aims to address the challenges mentioned in Chapter 1.1. In the following sub-sections, an overall and closed look at the system design will be served.

2.2.1. SYSTEM OVERVIEW

As shown in Figure 2.3, the proposed recognition system incorporates five fundamental components, including:

1. **The Data Preprocessing Module**
   The data preprocessing module transforms the raw radar data into the input data for the proposed neural network. The preprocessing procedure contains three steps: (1) radar data alignment and label synchronization, (2) target localization, and (3) short-time Fourier transform (STFT) [117] and spectrogram segmentation. The output of the preprocessing module is the aligned spectrogram-label pairs.

2. **The Convolutional Neural Network Block**
   The convolutional neural network (CNN) block acts as a hierarchical feature extractor. It takes the spectrograms generated by the preprocessing steps as its input data. The output of the CNN block is a feature map that represents the extracted feature patterns from the input data. The main objective of the CNN block in the hybrid CNN-RNN architecture is to conduct an automatic feature extraction process to capture the local correlations in the data.

3. **Data Fusion Module**
   The data fusion module handles the feature maps generated by the five CNN blocks. It concatenates the feature maps vertically into a feature cube. Then, an element-wise max-pooling is conducted to select the most prominent feature across both time dimension and feature dimension. This makes the output of the data fusion module becomes a flat feature map again. The data fusion module aims to find the best middle-level features that can conclude the target's activity in the spectrogram.

4. **The Recurrent Neural Network Block**
   The input data to the recurrent neural network (RNN) block is the generated feature map from the previous data fusion module. The RNN block takes the input data as a set of time sequences, and it captures the long-short term temporal dependencies in these sequences. The output of the RNN block is a feature map in which each value is a high-level feature representation of the input spectrogram for a corresponding local region.

5. **The Fully Connected Neural Network Block**
   The fully connected neural network (FCNN) block is added at the top of the proposed neural network. It takes the high-level feature map from the RNN block and generates the final predictions for the input spectrogram. During model training, the main goal of the FCNN block is to learn various mapping functions between different high-level features and the final activity class.

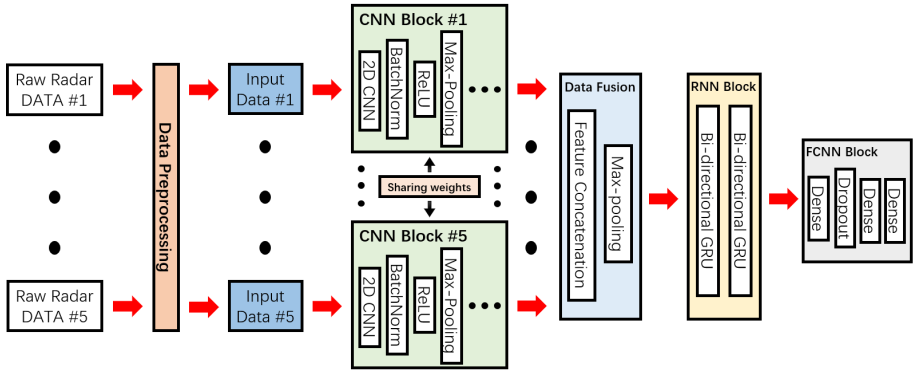


Figure 2.3: The architecture overview of the proposed recognition system. The input data on the leftmost is the raw radar data from the five radar sensors. On the rightmost is the FCNN block, which generates the final model prediction.

2.2.2. DATA PREPROCESSING

The raw radar data constitute a complex-valued matrix containing a 120 seconds recording of continuous human activities. Although applying the neural networks directly on the raw data is possible [118, 119], it will be more beneficial if several preprocessing steps are implemented beforehand due to:

1. The five IR-UWB radar sensors work in a monostatic mode in this experiment. To exploit the information from them, it is necessary first to align the radar data as well as synchronize the labels.
2. During radar measurement, the target moves freely inside the measurement area. Conventionally, knowing the target's location is the prerequisite of extracting the Doppler information.
3. The MTT system and HAR system are developed separately. However, to bridge these two systems afterward, the input data for the recognition system and the output data from the tracking system need to have the same format.

Figure 2.4 illustrates the key preprocessing steps that address the above crucial points and transfer the raw radar data into the input data.

Since the radar sensors work in a monostatic mode during data acquisition, each radar has a different start and end time. Thus, in **step #1** [1], the raw radar data from five radar sensors and the corresponding labels are aligned according to the latest acquisition start timestamp and the earliest acquisition stop timestamp.

[1] The data collection process and the step #1 in the preprocessing steps was done by PhD Ronny Guendel from the MS3 group, at TU Delft.

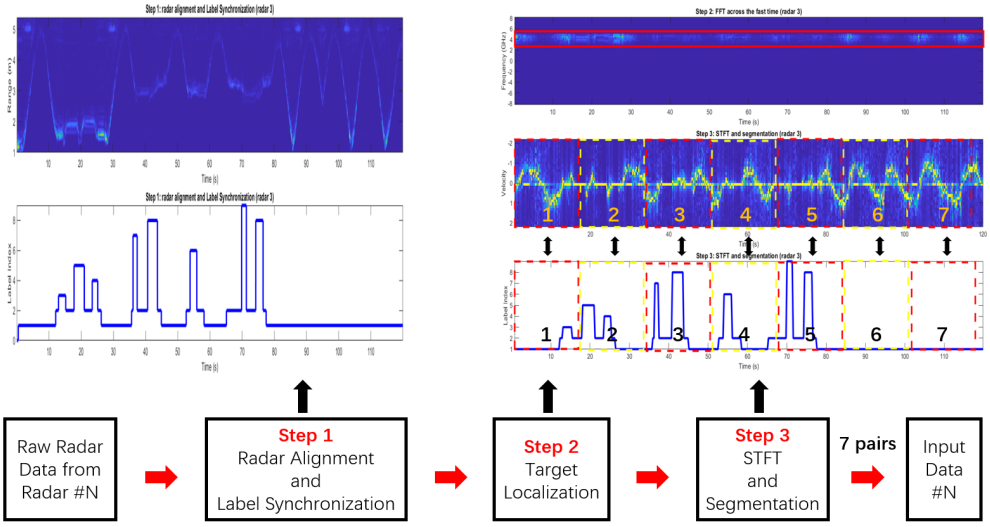


Figure 2.4: An overview of the key preprocessing steps. These steps transform the raw radar data into the input data for the proposed neural network.

Then, **step #2** [2] implements the target localization. During radar sensing, the target moves freely inside the measurement area. Conventionally, knowing the target's position is a prerequisite for extracting the Doppler information from the target. However, integrating the target tracking algorithm with the recognition system too early is not desirable, mainly because:

1. The tracking algorithm will drastically increase the system complexity and hinder the development speed. It will be more efficient to develop different systems individually.

2. Due to the challenges mentioned in Part II, Section 1.1, a simple tracking algorithm will not function well and may lose track of the target constantly. However, it is acceptable if a large dataset is available.

To avoid using a tracking algorithm, the alternative method implements the fast Fourier transform (FFT) on the raw radar data across the fast time dimension. The FFT transforms the received impulses at every scan from the time domain to the frequency domain. Thus, instead of localizing and tracking the position of the received impulse, this can be done by searching for the frequency bin with the highest averaged energy.

Although the alternative method has several disadvantages, it is more effective and efficient. This method works under the given experimental setup due to the fact that: (1)

[2] The idea is provided by PhD Ronny Guendel from the MS3 group, at TU Delft.

the center frequency and the bandwidth of the IR-UWB radar are known and constant. (2) only one person is presented inside the measurement area conducting the designed activities for each recording.

Finally, the STFT and spectrogram segmentation are implemented in **step #3**. The STFT is a popular tool used for time-frequency analysis. It applies FFT on the time series extracted from the previous step to generate spectrograms. Rather than implementing FFT on the entire signal, the STFT divides the signal into multiple fixed-length segments and performs FFT on these segments. Hence, the STFT outputs the spectral information given the segment length of time.

There are several parameters that can be adjusted in STFT, including: (1) the length of each segment, (2) the overlap ratio between successive segments, and (3) the window functions applied to the segment before FFT. In the proposed recognition system, the most crucial parameter is the time length of each segment. It balances the time-frequency resolution of the spectrogram. Moreover, it determines the minimum update rate of the MTT system.

In this thesis work, the segment length is chosen as 262.4ms (or 32 scans), there is no overlap between successive segments, and no window function is applied on the segments. It should be acknowledged that the three parameters are performance-related and can be advantageous to explore, but investigating the influence of the format of the input data is beyond the scope of this thesis.

After the STFT, the generated spectrogram and the ground truth label are further partitioned into small spectrogram-label pairs. These pairs are the input data for model training and evaluation. More importantly, they have the same format as the output data from the MTT system.

2.2.3. THE CNN BLOCK

The CNN has shown its exceptional ability for capturing spatial dependencies in computer vision and speech recognition tasks. The main objective of using the CNN architecture in the hybrid CNN-RNN model is to perform a stepwise feature extraction on the input data. The CNN can discover local features automatically. Therefore, unlike previous works [109, 110], it does not require to have handcrafted features for the RNN block. Furthermore, the CNN architecture is invariant against translations of the variations [112]. It is a crucial property for HAR tasks since the observed time-Doppler map of given the Doppler signatures of human activity may translate in both the time and frequency dimensions depending on different participants.

Figure 2.5 shows the architecture design of the proposed CNN block, Each CNN block contains three CNN module and one depth reduction module. For five input radar channels. the same CNN block is duplicated five times. The weight sharing strategy [120] is implemented on these duplicates to reduce the number of trainable parameters and relieve the overfitting problem. That is to say, the CNN blocks across different input chan-

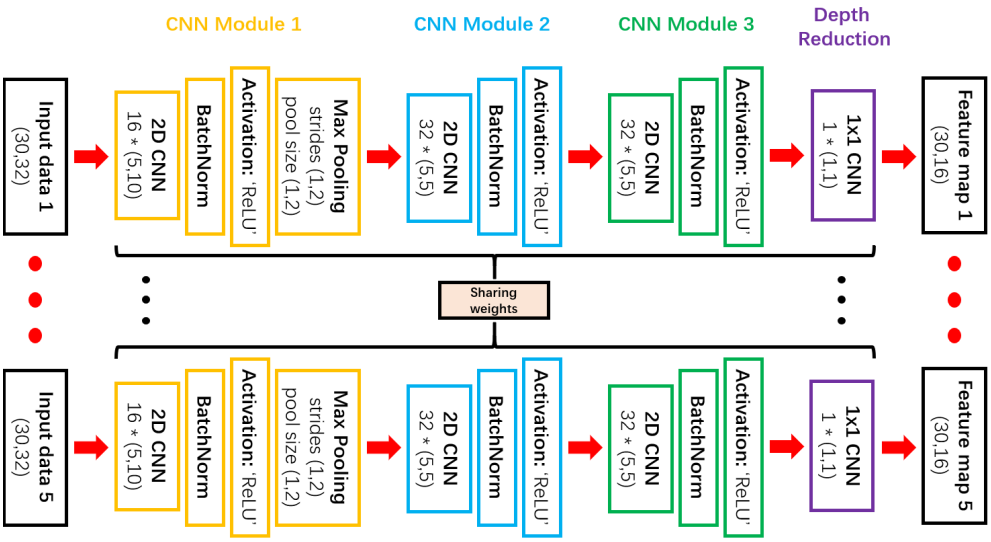


Figure 2.5: The architecture of the proposed CNN block. It consists of three CNN modules and one depth reduction module. The input data are the spectrograms from the preprocessing steps.

nels share the same weights.

For each CNN module, there are typically four major components, including: (1) the convolutional layer, (2) the Batch Normalization (BN) [121] layer, (3) the nonlinear activation layer, and (4) the pooling layer.

The convolutional layer consists of many filters, or kernels in other words. Each filter has a set of trainable weights. In the forward-propagation stage, each filter acts as a feature extractor. The filter convolves with the input data and generates a feature map. In the back-propagation stage, the filter weights will be updated in order to optimize the objective function. In our work, the rectangular filters are used as more information is stored across the frequency domain than the time domain [112]. Furthermore, the zero-padding method is used to avoid losing the boundary information and prevent shrinkage in the time dimension.

Then, the BN layer is implemented to standardize the generated feature maps from the current convolutional layer. It implements feature normalization and feature scaling on the feature maps and re-adjusts the mean and standard deviation before feeding them to the next layer. In summary, there are several advantages of adding the BN layer to the CNN module:

1. Accelerating the model training process [122].

2. Reducing the generalization error and mitigating the problem of internal covariate

shift [123].

3. Adding regularization effect to the model to reduce overfitting.
4. Increasing the model robustness to different weight initialization schemes.

After the BN layer, the non-linearity is introduced to the model by adding nonlinear activation layers. The activation layer gives the model the ability to learn any complex relationship between the input and the output. It converts the learned linear mappings into nonlinear forms for propagation in the hidden layer or prediction in the top layer [124]. In this work, the rectified linear activation function [125], or ReLU for short, is used to conduct an element-wise nonlinear transformation on the data from the BN layer.

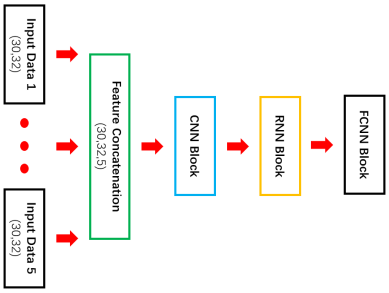
Except for the previously discussed three types of layers, the first CNN module contains one additional layer, the pooling layer. The pooling layer provides additional translational invariances to the CNN block [111]. Furthermore, it offers a cheap way fast to increase the receptive field of the CNN block. It also reduces the computational cost since the data dimension is reduced after the pooling layer. In our model design, the max-pooling [126] strategy is used. Inspired by the works in [107, 123], the max-pooling layer is added into the first CNN module and applied only in the frequency dimension to prevent losing too much information.

The output data from the last CNN module is a data cube with multiple feature maps. The depth of the data cube depends on the number of parallel filters used in the previous convolutional layer, as each filter creates one corresponding feature map. Since there are five data cubes from the five radar channels, it is computationally expansive to process all of them. Hence, the depth reduction module is added on the top of the each CNN block to reduce the dimensionality of the data cube. The depth reduction is achieved by using a 1x1 convolutional layer [127]. The 1x1 convolutional layer applies an element-wise linear projection on the data cube. In our setting, the output of the depth reduction module for each radar channel is a single feature map.

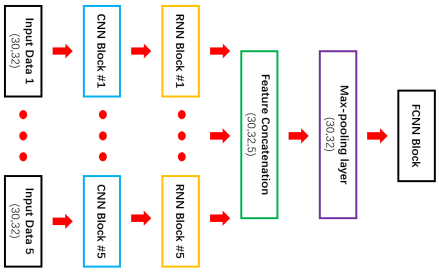
2.2.4. DATA FUSION

Conventionally, radar-based HAR problems have been mainly explored using a single radar sensor [16, 128]. It has been shown that the conventional methods suffer from the changes in the aspect angle [3]. In this thesis work, the RSN is introduced to handle the arbitrary movement directions of human activities. Therefore, the problems in the unconstrained HAR become a data fusion problem. To investigate different data fusion methods for our problem, three distinct neural network-based data fusion architectures are designed and compared. Depending on where the data sources are fused in the network, the fusion method can be separated into three categories, i.e., early fusion (or signal-level fusion), late fusion (or decision-level fusion), and halfway fusion (or feature-level fusion).

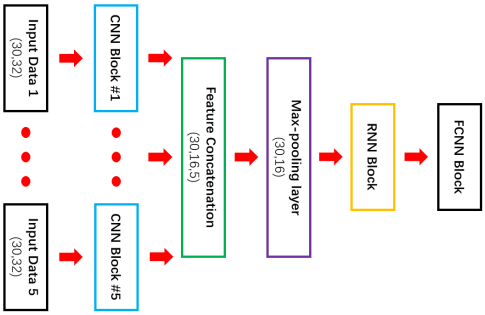
Figure 2.6a depicts the schematic for adding the early fusion method to the proposed



(a) Early fusion. The input spectrograms from five radar channels are concatenated before feeding them to the proposed neural network.



(b) Late fusion. The high-level feature maps from each RNN block are combined and sent to the FCNN block.



(c) Halfway fusion. The middle-level feature maps generated by the CNN block are concatenated and fed to the RNN block.

Figure 2.6: An illustration of the neural network-based data fusion strategies. The input data is the spectrogram generated by the preprocessing steps.

neural network. The early fusion concatenates low-level feature maps or raw sensor outputs before feeding them to the network. In our case, the low-level feature map is the input spectrogram. That is to say, the spectrograms from five radar sensors are vertically concatenated to form a cube structure. Then, during model training, the neural network learns the dependencies among different modalities or sensors. As you understand, the early fusion method is easy to implement. Moreover, it allows low-level feature interactions among various sensors. However, the early fusion method is hard to use if the input features do not have a similar size or feature representation, e.g., fusing image data with audio data.

Figure 2.6b illustrates the scheme for the late fusion method, or the decision-level fusion method in other words. Contrary to the early fusion method, the late fusion concatenates the high-level feature maps or the predictions generated by different classifiers. Thus, each sensor is processed independently. The final prediction after combining different modalities can be achieved by using the voting [120], weighted sum [100], or machine learning approaches [110]. One advantage of using the late fusion method is that it allows fusing information from heterogeneous sensors. Furthermore, it permits uni-modal pre-training. That is to say, the dataset size of different sensors does not need

to be aligned. Nevertheless, the late fusion is more time-consuming since it requires multi-stage pre-training. Moreover, it can not model the low-level feature interactions among different modalities.

Figure 2.6c shows the implementation strategy for the halfway fusion method. The halfway fusion usually starts from multiple branches. Each branch acts as a feature extractor that extracts information from a given sensor. Then, the outputs from each branch are merged and sent to late neural networks for further feature extraction and interaction. Similar to the late fusion, the halfway fusion method can fuse data from heterogeneous sensor systems. Furthermore, it provides an end-to-end fusion model without the need for multi-stage training. However, the halfway fusion method is often hard to train. During training, the prediction errors need to be propagated through different feature extractors in order to adjust their weights. Moreover, the neural network using halfway fusion method will have higher model complexity.

Table 2.2 gives an summary of the fusion methods discussed in this section.

Methods	How	Pros	Cons
Early fusion	Combining low-level features or raw signals	Simple implementation, model-free fusion	Not applicable for heterogeneous sensor systems
Late fusion	Combining high-level features or predictions	Suitable for heterogeneous sensor systems, no need for dataset alignment	Requiring multi-stage pre-training for different modalities, no low-level feature interactions
Halfway fusion	Combining intermediate feature representations	Suitable for heterogeneous sensor systems, providing an end-to-end fusion model	Higher model complexity, hard to train. unclear how and where to fuse the data [129]

Table 2.2: Comparison of the fusion methods discussed in this section.

2.2.5. THE RNN BLOCK

Although the CNN block can exploit the local correlations from the input spectrogram in both the time and frequency dimension, a very deep CNN is required to achieve a relatively large receptive field so that the long-term dependencies can be captured. However, deep CNN is hard to design and can significantly increase the number of training parameters. Therefore, the RNN architecture is introduced after the data fusion module to directly model the signal in time.

There are three types of RNNs that are studied and compared in this work, including: (1) the basic RNN, (2) the long short-term memory (LSTM) [130], and (3) the gated re-

current unit (GRU) [131]. Figure 2.7 demonstrates the architecture design of these three RNNs.

The basic RNN is similar to a single neuron but has loops across the time dimension. As shown in Figure 2.7a, at time t, the basic RNN takes the input data sequence x_t and concatenates it with the previous hidden state h_{t-1}. Then, the hidden state h_t and the prediction vector y_t are generated through a combination of linear and nonlinear transformation. The loop continues and the hidden state propagates until the last data sequence is processed. Thus, to make a prediction at a given timestamp, the basic RNN uses not only the current input but also the previous hidden state which contains the information of the past. It is a crucial property of the basic RNN since continuous human activity has temporal dependencies.

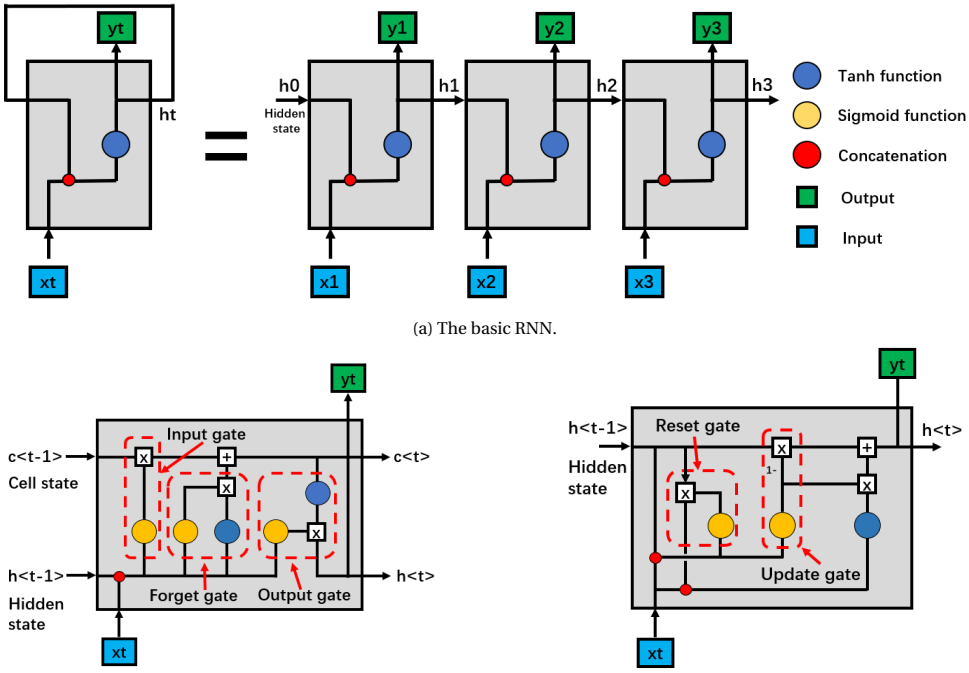
However, the basic RNN is not good at capturing long-term dependencies due to the vanishing gradient problem [132]. One variant of the basic RNN that is effective in handling long-term temporal correlations is the LSTM. Figure 2.7b shows the architecture of the LSTM. The LSTM has a similar control flow as the basic RNN. It processes the input data sequentially and propagates the information forward along the time dimension. For the differences, the LSTM introduces three types of gates, i.e., the forget gate, output gate, and the input gate, each of which can learn to keep only relevant information during the forward propagation. Moreover, the LSTM uses a separate cell state c_t to transport the relative information through the time. In this way, the vanishing gradient problem can be mitigated, and the training speed is improved.

The GRU is a variant of the LSTM, which can also relieve the vanishing gradient problem. As shown in Figure 2.7c, the GRU has a similar architecture as the LSTM. It uses the reset gate and update gate to capture the important temporal characteristics. However, instead of propagating the cell state and hidden state, the GRU only uses the hidden state to transfer the relevant information. For the advantages, since GRU has fewer gates and trainable parameters than the LSTM, it runs faster during model training. Nevertheless, both GRU and LSTM are commonly used in research as it is hard to tell which variant works better.

Apart from the different variants of RNNs, the input-output scheme for the RNN model is another important aspect that needs to be considered. As shown in Figure 2.8, there are principally four types of input-output schemes, including: (1) one-to-many, (2) many-to-one, (3) many-to-many-A, and (4) many-to-many-B.

The one-to-many scheme is often used for tasks like image captioning. It takes one input data (e.g., one image) and generates a sequence of outputs (e.g., captions). On the contrary, the many-to-one scheme takes a time series as its input and generates a single output. Applications using this scheme can be text-based sentiment classification. There are two types of many-to-many scheme depending on if the length of the input sequence and output sequence are matched. If not, like in machine translation tasks, the many-to-many-A can be used. Otherwise, the many-to-many-B is used.

For the proposed recognition system, the objective is to classify a set of continuous



(a) The basic RNN.

(b) The LSTM.

(c) The GRU.

Figure 2.7: The architecture design of the basic RNN, LSTM, and GRU.

human activities. Hence, only the two many-to-many schemes are applicable. In addition, the many-to-one scheme can be considered if each recording only contains one type of human activity. In our experimental settings, the ground truth label and the radar recording are aligned and synchronized. Therefore, it is natural to use the many-to-many-B scheme in the proposed system since every timestamp has a corresponding activity label. However, it will be advantageous also to consider the many-to-many-A scheme for the HAR tasks since a perfect activity-to-label alignment is hard to achieve.

The RNNs discussed above are unidirectional. In other words, the RNN prediction at a given timestamp is based on the current and previous system inputs. However, there are temporal dependencies in human activities for both forward and backward directions. For example, a person is conducting a continuous movement of 'walking, falling down, and standing up'. To predict the 'falling' action, not only using the Doppler signatures of 'walking' and 'falling' will help but also the 'standing up'.

Therefore, a modification on the network architecture is introduced in [133] to make the RNNs bidirectional. Figure 2.9 illustrates the structure of a bidirectional RNN with the many-to-many-B scheme. The bidirectional RNN has an additional backward recurrent layer compared to the unidirectional RNN. The forward layer takes the input sequence forward in time, and the backward layer takes the input sequence backward

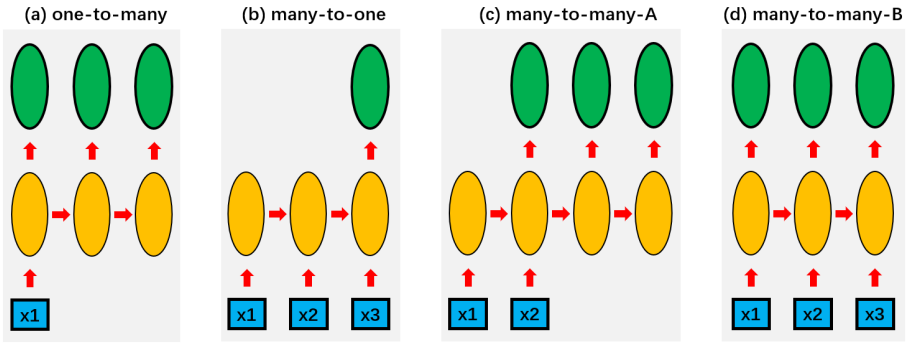


Figure 2.8: The input-output schemes for the RNN, where the input time sequences are simplified and marked in blue, the RNNs are marked in orange, and the outputs are marked in green.

in time. Then, the outputs from both layers for each timestamp are concatenated in the output layer. Hence, the bidirectional structure allows the prediction to exploit the past, present, and future information from the input data. More importantly, the bidirectional design can be applied to the basic RNN and its variants. For the RNNs discussed above, the bidirectional structure will lead to the basic bidirectional RNN (Bi-RNN), bidirectional LSTM (Bi-LSTM), and bidirectional GRU (Bi-GRU).

In this thesis work, the unidirectional RNNs and the bidirectional RNNs are investigated. Furthermore, the RNNs are stacked into multiple parallel layers to improve their capability of learning complex temporal characteristics.

2.2.6. THE FCNN BLOCK

The FCNN block usually occurs at the top of the neural network to make the final predictions. It consists of multiple fully connected layers that connect each neuron in one layer to all neurons in its neighboring layers. Due to the special network architecture, the FCNN block can learn how to combine different high-level feature representations during model training.

Figure 2.10 shows the architecture design of the FCNN block. The feature map generated by the previous RNN block is a two-dimension matrix that provides a high-level feature representation of the input data. During model training, the FCNN block learns to interpret the complex feature representations. Since the input data contains continuous human activities, the FCNN block is distributed across the time dimension. In other words, for each time step, the FCNN block takes a feature vector from the feature map and makes a prediction. For the final architecture of the FCNN block, three fully connected layers are stacked and the dropout layer [134] is applied after the first fully connected layer to reduce the overfitting problem.

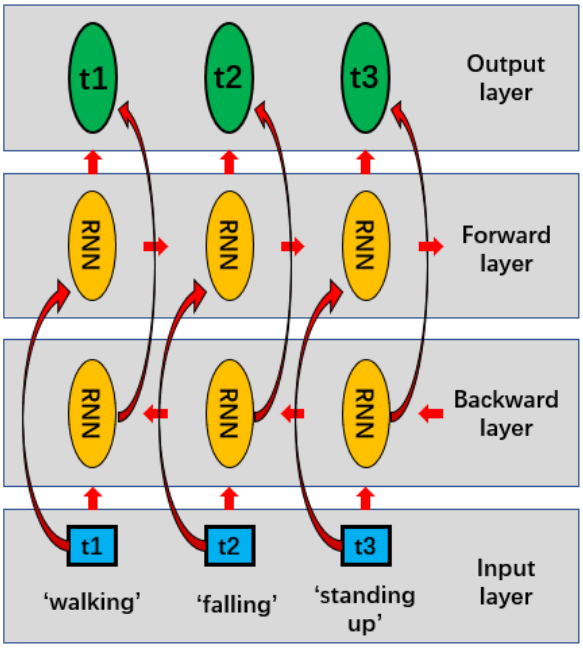


Figure 2.9: The structure of a bidirectional RNN with the many-to-many-B scheme.

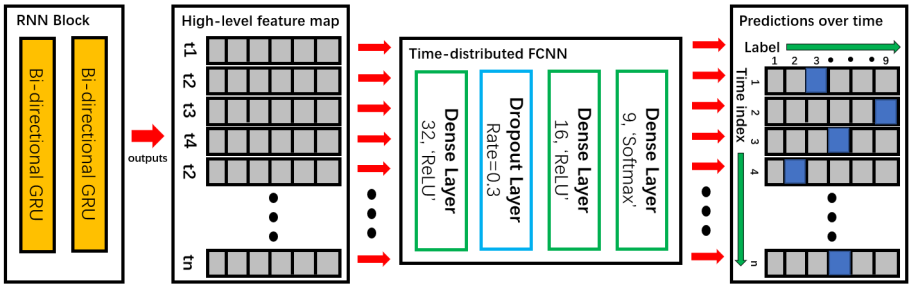


Figure 2.10: The architecture design of the FCNN block. It takes the output of the RNN block and generates predictions over time.

2.3. MODEL TRAINING AND EVALUATION

After building the deep learning model, the next step is to implement model training and evaluation using the collected dataset. The conventional training and evaluation strategy randomly divides the dataset into three subsets, i.e., the training set, validation set, and test set. As illustrated in Figure 2.11, the model is trained using the training set first. Then, the model performance is measured using the validation set. Based on

the validation results, hyperparameter tuning is used to finely adjust the model configurations. The tuning process repeats the previous training and validation steps until the validation set maximizes the model performance. Lastly, the test set, which has not been involved in the training and validation procedures, is used to evaluate the performance of the fine-tuned model.

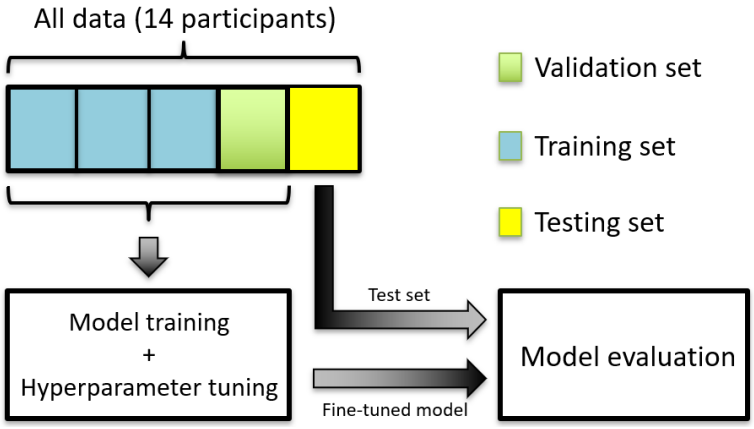


Figure 2.11: The conventional model training and evaluation strategy.

The conventional strategy is fast and easy to implement. It can accurately estimate the model performance when the collected dataset is large. Moreover, instead of dividing the dataset into the training and test set, introducing a separate validation set for hyperparameter tuning can prevent the model parameters from overfitting the test set.

Nevertheless, there are two critical issues with the conventional strategy in consequence of a small dataset:

1. During the hyperparameter tuning stage, the validation result can be unreliable and heavily influenced by how the dataset is partitioned. Furthermore, since only part of the dataset is used to validate the model performance, the validation result may have a large variance.
2. During the model evaluation stage, the conventional method can not assess the generalization capability of the proposed model to unknown participants. In our case, even though the dataset contains thousands of labeled human activities, they all sampled from fourteen participants. Therefore, it is essential to know if the recognition system can generalize well and maintain the performance when the test set contains a person the model has never seen.

In this thesis work, the leave-one-person-out (L1PO) method [19, 109, 110] and K-fold cross-validation method are used together to address these aspects. Figure 2.12 shows an overview of the proposed model training and evaluation strategy.

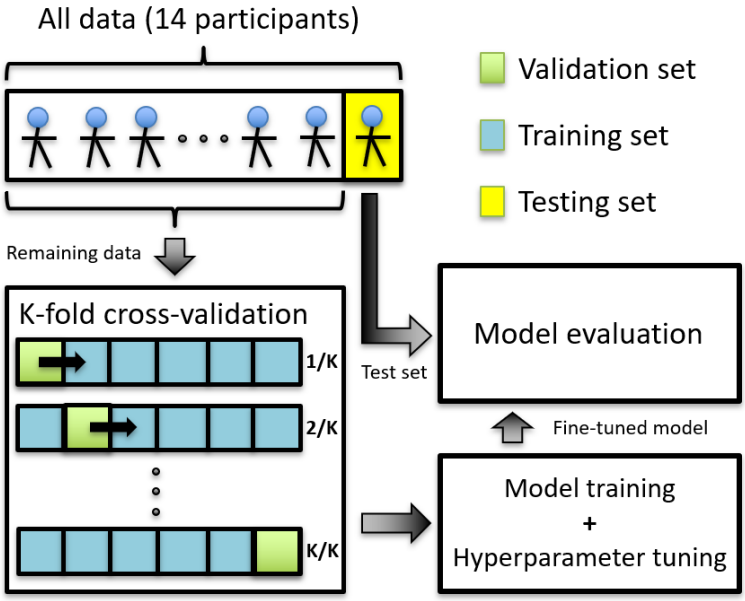


Figure 2.12: An overview of the data partitioning strategy.

The total dataset contains the human activities of fourteen participants. First, the L1PO method separates the dataset into two subsets. One is the test set, or holdout set in other contexts. It contains all radar recordings of one randomly selected participant. Another set, which has the recordings of the remaining thirteen participants, is further partitioned using the K-fold cross-validation method.

The K-fold cross-validation method randomly divides the remaining data into K folds. For each experiment, one of the K folds is used as the validation set, and the rest are used as the training set. Similar to the previous, the model is trained on the training set, and the performance is measured using the validation set.

However, the experiment repeats until all folds have been used once as the validation set. Thus, the recognition system is trained and validated K times. Then, the hyperparameter tuning is implemented based on the validation results averaged across the K experiments. Similarly, the tuning process stops until the performance of the fine-tuned model is maximized.

For model evaluation, the fine-tuned model is evaluated on the test set. To have a more accurate estimation, each participant has been left out once and used as the test data. In this way, the generalization capability and robustness of the proposed recognition system can be verified.

In summary, the proposed model training and evaluation strategy is computationally

expensive during the model development stage, since every minor tweak on the model hyperparameters will cost training and validating the model K times. In spite of that, using the K-fold cross-validation method and the L1PO method for training and evaluating neural networks is still widely accepted due to:

1. The K-fold cross-validation method exploits the collected data more efficiently, as all the data have been used once to validate the model performance. Therefore, the validation result is more accurate, especially when the available dataset is small.

2. The L1PO method can measure the generalization capability and robustness of the proposed recognition system. The test set contains the characteristics of a person's activities that the system has never seen. Thus, the evaluation results will be more realistic and informative.

2.4. SUMMARY

This chapter presents the details of the proposed HAR system, including the experimental setup, the neural network design, and the method for model training and evaluation.

The radar experiment was conducted at the radar laboratory of TU Delft with 14 participants. Five IR-UWB radar sensors are used to constitute a RSN and arranged according to a circular geometry. Nine types of human activities are designed to test the system performance for the ADLs. Unlike the previous works in which the human activity is either constrained in moving directions or performed as separated action, the designed activities are continuous and have arbitrary moving trajectories. Moreover, there are seamless inter-activity transitions during the target's movement.

To address the challenges mentioned in Chapter 1.1, a neural network-based recognition system is proposed. Thanks to the deep learning approaches, the proposed system provides an end-to-end solution for continuous HAR as well as data fusion. Moreover, the proposed system has a hybrid CNN-RNN architecture for directly modeling the spatial and temporal characteristics in the input data. Various architectures and network configurations have been investigated to improve the system performance. Furthermore, although the proposed system is developed independently, it is compatible with the proposed MTT system. That is to say, it is possible to combine the systems for joint tracking and activity classification.

Finally, the model training and system evaluation method is proposed to address the limited dataset problem. The method combines the popular K-fold cross-validation method with the L1PO method. It can use the dataset more efficiently for model training, hyperparameter tuning, and performance evaluation. More importantly, compared to the conventional strategy, the proposed method can measure the generalization capability and robustness of the proposed system. Combined with different evaluation metrics, the proposed method is applicable to evaluate other radar-based recognition systems that suffered from a limited dataset.

3

RESULTS

This Chapter presents the results of the proposed human activity recognition (HAR) system. Specifically, the system performance under various model configurations and architectures is explored first, since this helps us find the best model hyperparameters. Then, the proposed system is compared with other recognition systems from the literature. Lastly, the generalization capability and robustness of the proposed system are measured.

The radar dataset is collected based on the experimental setup presented in Chapter 2.1. Nine types of human activities are collected, including: (1) walking, (2) stationary, (3) sitting down, (4) standing up from sitting, (5) bending from sitting, (6) bending from standing, (7) falling while walking, (8) standing up from the ground, and (9) falling while standing. In total. 14 participants are involved in the dataset collection process.

For the model evaluation, the leave-one-person-out (L1PO) and K-fold cross-validation method are used. The dataset for model training and validation is divided into five subsets, i.e., K equals five. The validation accuracy is measured to compare different model settings. To get a more accurate validation result, the validation accuracy is averaged across the five subsets, and the standard deviation of the validation accuracy is calculated. Furthermore, the number of model parameters is also presented to indicate the model complexity. The reader is referred to Chapter 2.3 for more details about the model training and evaluation strategy.

For the default model setting, the hybrid CNN-RNN architecture combined with the weight sharing and halfway fusion strategy is used. The convolutional neural network (CNN) block contains three 2D CNN layers, whereas the recurrent neural network (RNN) block has two gated recurrent unit (GRU) layers with the many-to-many-B scheme and the bidirectional network structure. Unless specific notification, the default model setting applies to all the experiments conducted below.

3.1. NUMBER OF CNN LAYERS

Table 3.1 shows the system validation results for varying the number of CNN layers. There are five experiments conducted with the CNN layer increased from one to five. Each CNN layer is followed by a Batch Normalization layer and an activation layer. For the first CNN layer, a pooling layer is added to the last. Other model parameters follow the default settings.

Number of CNN	Model Parameters	Validation Accuracy	Standard deviation
1	32K	84.89%	1.72
2	45K	84.91%	1.94
3	**71K**	**87.11%**	**1.11**
4	97K	86.15%	1.06
5	122K	86.25%	1.35

Table 3.1: Comparison of models with different number of CNN layers.

Based on the results, raising the number of CNN layers from one to three will increase the validation accuracy. Moreover, the standard deviation is reduced, which means the model performance becomes more robust across different validation folds. However, further deepening the CNN architecture will not improve the model performance. The validation accuracy decreases as the number of CNN layers increases from three to five. The reasons behind these phenomena can be summarized as follows:

1. Increasing the number of CNN layers can give the model more nonlinearities to learn complicated feature patterns. Moreover, deepening the CNN will enlarge the receptive field and help the model capture the spatial-temporal features in a larger scope.

2. However, further increasing the CNN depth can overshoot the desired model complexity and lead to the overfitting problem, especially when the size of the dataset is small. A similar trend is also observed in [135], in which the hybrid CNN-RNN architecture is used for a speech recognition task based on spectrograms.

Furthermore, to investigate the influence of the RNN layer in the hybrid CNN-RNN architecture, Table 3.2 presents the evaluation results, where the number of CNN layers is varying but the number of RNN layers is set to one (the default is two). Based on the results, there are several conclusions can be made:

1. Reducing the number of RNN layers will result in a deeper CNN architecture (layers increased from three to five). This is because the RNN layer has the ability to learn the long-term temporal dependencies. However, reducing the number of RNN layers will weaken this ability, and a deeper CNN architecture is needed for

compensation. Therefore, the turning of the system performance from increasing to decreasing comes later.

2. The model complexity can be reduced with appropriate spatial and temporal modeling using the hybrid CNN-RNN architecture. In other words, it is not required to have a very deep CNN with the help of RNN. A deep CNN is often hard in design and has overfitting problems with small dataset.

Number of CNN	Model Parameters	Validation Accuracy	Standard deviation
1	13K	79.42%	1.88
2	26K	83.31%	2.56
3	52K	84.10%	1.18
4	78K	85.00%	1.51
5	**103K**	**85.81%**	**1.04**
6	129K	85.69%	1.47

Table 3.2: Comparison of models with different number of CNN layers, the number of RNN layers is set to one.

3.2. WEIGHT SHARING

The input data for the proposed recognition system comes from five identical IR-UWB radar sensors. For a given time step, the differences in the input data are related to the distance and aspect angle between the radar sensors and the target. Since the target moves freely inside the measurement area while performing activities, all radar sensors should have the same chance to capture the target's movement from different distances and aspect angles.

For the CNN block, the main objective is to extract the activity patterns so that the later neural networks can learn how to select and combine the extracted features. Therefore, it makes sense to duplicate the CNN block five times for the five radar channel and share the same weights. Table 3.3 shows the impact of the weight sharing method on the model performance and complexity. The result shows that the weight sharing method reduces model parameters from 229K to 71K. Moreover, the model using the weight sharing strategy has a higher validation accuracy. This is because the model without the weight sharing method has more capability to capture irrelevant information, instead of focusing on learning the general feature patterns.

3.3. TYPE OF DATA FUSION

Table 3.4 presents the model validation results, where the model performance is a function of different data fusion strategies. Three neural network-based data fusion methods are implemented and investigated, including: (1) early fusion, (2) halfway fusion, and

Weight Sharing	Model Parameters	Validation Accuracy	Standard deviation
With Weight Sharing	**71K**	**87.11%**	**1.11**
Without Weight Sharing	229K	83.70%	0.89

Table 3.3: Comparison of models with or without weight sharing in the CNN block.

(3) late fusion. Moreover, the performance of the three data fusion methods are compared with the case when only one radar sensor is used (denoted as 'No Fusion' in the following). Based on the results, the following conclusions can be made:

1. The halfway fusion method outperforms the early and late fusion methods in terms of validation accuracy. The results demonstrate that allowing the middle-level feature interactions is more promising than other fusion strategies. A similar conclusion has been drawn in [120], where the author compared the halfway fusion method with the late fusion method for personnel recognition and gait classification.

2. The halfway fusion method has fewer model parameters than the early fusion method. This is because the CNN block has only one objective in the halfway fusion method, which is to capture the characteristics of human activity. However, the early fusion method requires the CNN block also to handle low-level feature interactions among different radar channels.

3. All three data fusion strategies outperform the 'No Fusion' cases. The result shows that the data fusion methods can improve the system robustness against the less favorable aspect angles. Moreover, it also demonstrates that using a distributed radar sensor network (RSN) for human activity monitoring is superior to the single radar case.

4. The validation results of the 'No Fusion' cases are similar. This implies that the target's movements inside the measurement area are nearly random, as the chances for each radar node to have a good or bad aspect angle are equal.

3.4. NUMBER OF RNN LAYERS

Other than the CNN block, the RNN block is another crucial component in the proposed hybrid CNN-RNN architecture. It is responsible for capturing the long-short term temporal dependencies in the input data and generating the high-level feature maps for the fully connected layers. Therefore, to find the optimal RNN architecture, the number of RNN layers is set as a variable to measure the changes in model performance.

Fusion Method	Model Parameters	Validation Accuracy	Standard deviation
Early Fusion	74K	85.06%	1.66
Halfway Fusion	**71K**	**87.11%**	**1.11**
Late Fusion	71K	83.71%	1.11
No Fusion (use radar 1)	71K	69.65%	1.72
No Fusion (use radar 2)	71K	71.81%	2.78
No Fusion (use radar 3)	71K	72.03%	1.69
No Fusion (use radar 4)	71K	71.39%	1.09
No Fusion (use radar 5)	71K	69.44%	1.33

Table 3.4: Comparison of models with different data fusion strategies.

As shown in Table 3.5, four experiments are conducted with RNN layers increasing from one to four. As shown, the validation accuracy had a significant jump from 83.99% to 87.11% as the number of RNN layers increases from one to two. This is because more RNN layers give the model more capacity to learn complicated temporal relationships. However, the model does not gain more performance improvement by increasing the number of RNN layers further.

Previously, it was observed that the number of RNN layers could influence the depth of the CNN architecture. As demonstrated, with an appropriate number of RNN layers, the model performance is improved, and the complexity of the CNN block is reduced. Therefore, it is possible that the influence is mutual. That is to say, the depth of the CNN block can also affect the architecture of the RNN block. Moreover, it can be inferred that the hybrid CNN-RNN architecture can lead to a more light-weighted neural network model with better performance than the model using purely one type of network architecture.

Number of RNN	Model Parameters	Validation Accuracy	Standard deviation
1	52K	83.99%	1.61
2	**71K**	**87.11%**	**1.11**
3	90K	86.98%	1.02
4	108K	86.78%	1.07

Table 3.5: Comparison of models with different number of RNN blocks.

To verify the inference, the capability of the CNN block is weakened by reducing its depth from three to one. Then, the same experiment is conducted, in which the model

performance is measured under different number of RNN layers. Table 3.6 presents the validation results. As it shows, the neural network requires more RNN layers to maximize the system performance (from previous two layers increased to six). Moreover, the model complexity has also been increased (from 71K parameters to 107K parameters).

Based on the results from this section and Chapter 3.1, it can be concluded that the hybrid CNN-RNN architecture can lead to lower model complexity but better system performance compared to the case when a single type of neural network is used. It also implies that the hybrid architecture is more promising for the radar-based HAR tasks. Not only because a shallower CNN or RNN model is much easier to design and train, but also a light-weighted model is less likely to be overfitted.

Limited by the experimental setups and model loss function, the CNN or RNN architecture can not be deleted completely from the proposed model. Nevertheless, the inference can be extended that the general trend should be consistent. This is because the radar data contains both spatial and temporal characteristics. If one of the architectures is weakened, increasing the complexity of the other for compensation will be needed.

Number of RNN	Model Parameters	Validation Accuracy	Standard deviation
1	13K	78.57%	2.04
2	32K	84.47%	1.96
3	51K	85.57%	1.25
4	70K	86.29%	1.24
5	88K	86.86%	0.74
6	**107K**	**87.04%**	**0.74**
7	126K	85.77%	1.61

Table 3.6: Comparison of models with different number of RNN blocks, the number of CNN layers is set to one.

3.5. TYPE OF RNN

Due to the particular loop architecture, the basic RNN can exploit the temporal characteristics in the input data. Limited by the vanishing gradient problem, the long-term dependencies are hard to learn for the basic RNN. Moreover, for continuous human activities, both the past and future information can help improve classification accuracy. Thus, to find the best RNN for the HAR problems, six experiments are conducted in this section, each using a different type of RNNs.

Table 3.7 shows the comparison results. the basic RNN is compared with its two variants, i.e., GRU and LSTM. Then, the bidirectional architecture is applied to these RNNs, which leads to the basic Bi-RNN, Bi-GRU, and Bi-LSTM. The performance of these bidirectional RNNs is measured further. Based on the results, the following conclusions

Type of RNN	Model Parameters	Validation Accuracy	Standard deviation
Basic RNN	45K	79.15%	0.30
GRU	53K	81.68%	0.78
LSTM	56K	81.61%	0.77
Basic Bi-RNN	52K	80.15%	2.10
Bi-GRU	**71K**	**87.11%**	**1.11**
Bi-LSTM	80K	86.17%	1.11

Table 3.7: Comparison of models with different types of RNNs.

can be made:

1. In general, the bidirectional architecture outperforms the unidirectional architecture. This proves the benefits of exploiting the forward and backward temporal dependencies for HAR. However, the Basic RNN does not gain much performance improvement from this modification. This reflects from another aspect that the long-term temporal dependencies from the past and future are more important for improving the system performance.

2. The GRU and LSTM achieve similar performance in both the unidirectional and bidirectional architectures. However, the Bi-GRU is set as the default model setting. This is because the Bi-GRU has fewer model parameters and trains faster.

3. In the bidirectional architecture, the performance gap between the basic RNN and its two variants is more significant than the unidirectional. This directly demonstrates the importance of combining the long-term information from both the past and future.

3.6. DROPOUT

The main goal of supervised machine learning for multiclass classification problems is to train a neural network that can learn multiple decision boundaries. Since the training data are not perfect, label error is pervasive, and the size of the dataset is often limited. Rather than generalizing a smooth decision boundary, the neural network often tends to overfit the training data. The dropout is a popular regularization method used to mitigate the overfitting problem. In this experiment, the model performance between using and not using the dropout method is compared.

The dropout layer is introduced into the fully connected neural network (FCNN) block, specifically after the first fully connected layer. The dropout rate is set to 0.3, meaning 30% of the inputs will be randomly excluded. Table 3.8 shows the validation results. Like you see, the model with the dropout layer achieves higher validation accuracy than the model without the dropout layer. A more evident comparison can be found

Dropout Layer	Model Parameters	Validation Accuracy	Standard deviation
With Dropout Layer	**71K**	**87.11%**	**1.11**
Without Dropout Layer	71K	86.19%	1.22

Table 3.8: Comparison of models with or without dropout layer (dropout rate=0.3).

in Figure 3.1, where the model with the dropout method has a smaller gap between the training accuracy and the validation accuracy (2.79 compared to 7.32).

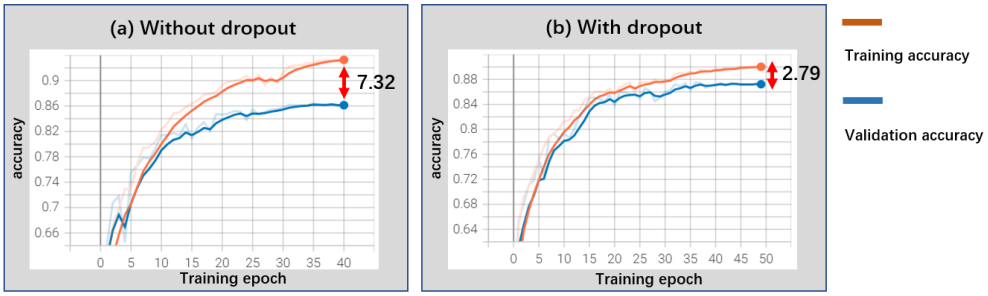


Figure 3.1: Accuracy curves for models with or without dropout layer. The dropout layer reduces the gap between the training and validation accuracy from 7.32 to 2.79.

3.7. STATE-OF-THE-ART COMPARISON

In this section, the performance of the proposed recognition system is compared with two recent works from the literature.

The first work [113] uses a hybrid CNN-RNN network for HAR. Similarly, the author uses CNN to extract the spatial features and RNN to learn the time-dependent information. The model input is the spectrogram. To evaluate the model performance, the author designed seven types of human activities, including: (1) running, (2) walking, (3) walking while holding a stick, (4) crawling, (5) boxing while moving forward, (6) boxing while standing in place, and (7) sitting still. Based on their results, the proposed model outperforms other models, e.g., CNN [136] and ResNet-18 [137], regarding classification accuracy and model complexity.

Although the proposed model in the first work has shown significant performance improvement over the previous works, several limitations were not well-addressed. First, the input spectrograms have the same size, and each of them contains only one type the designed activities. However, human movement is continuous and has seamless transitions among different activities. Second, the movement direction is constrained due to

the use of a single radar sensor for data collection.

The second work [100] is one of the first works that focuses on classifying continuous human activity with unconstrained directions. The work uses the Softmax classifier [138], which takes the handcrafted features as the input data. To explore the information from the distributed radar sensors, two fusion strategies were investigated and compared. Based on their results, there are two significant contributions. First, the author demonstrated the feasibility of using a distributed RSN to recognize continuous human activities for arbitrary moving directions. Second, the author proved the superiority of using multiple radar sensors for HAR problems.

The proposed model in this thesis work continues the previous work [100]. The same experimental setup and radar dataset are used to evaluate the system performance. Different from the work [113], this work considers classifying human activities under more realistic scenarios, i.e., unconstrained and continuous human actions. Compared to work [100], It is dedicated to provide an end-to-end solution for HAR and data fusion without the need for handcrafted feature engineering.

Table 3.9 presents comparison results between the proposed model and the two significant works from the literature. Compared to work [113], the proposed model has fewer model parameters, which is more beneficial for small dataset tasks. Although it is hard to compare the accuracy between these two models due to different experimental setups and radar datasets, they should be comparable given the nine-class classification versus seven-class. Compared to work [100], the proposed model conducts automatic feature extraction and sensor fusion, and it also shows higher classification accuracy.

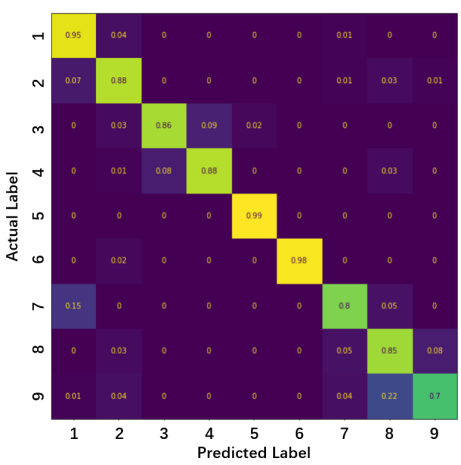
Model	Parameters	Accuracy	Method	Inputs
[113]	205K	98.28% (5-fold cross-validation)	Hybrid CNN-RNN	Raw spectrograms
[100]	-	52.11% (L1PO test)	Softmax classifier	Handcrafted features
Proposed Model	71K	89.88% (L1PO test)	Hybrid CNN-RNN	Raw spectrograms

Table 3.9: An overview of the state-of-the-art comparison for HAR.

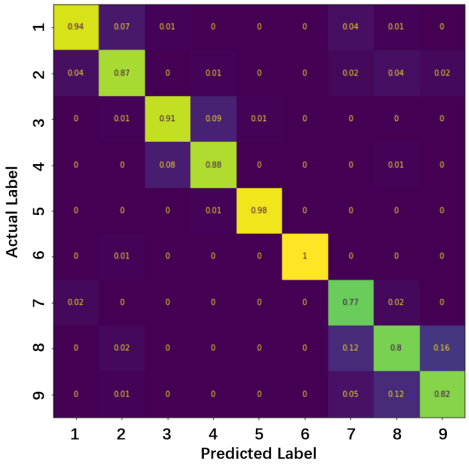
3.8. OTHER EVALUATION METRICS AND ERROR ANALYSIS

In the previous sections, accuracy is the primary evaluation metric used to compare different models and model configurations. Accuracy metric is also one of the most popular metrics used widely in the literature. However, for multiclass recognition tasks, the accuracy metric can not reflect the model performance for each class. Moreover, if the dataset is imbalanced, the accuracy metric can hide significant classification errors for classes with a few testing examples.

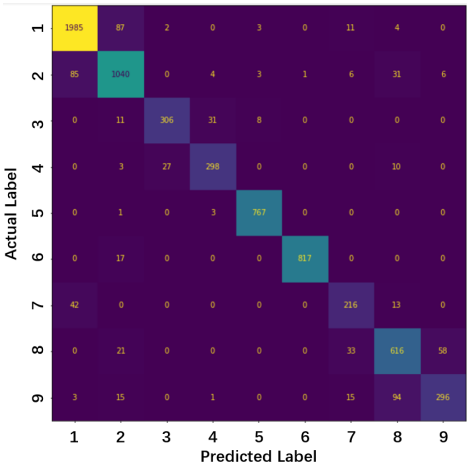
To closely look at the model from different aspects, Figure 3.2 shows the model performance tested on various evaluation metrics using the test dataset. The applied metrics include: (1) recall, (2) precision, (3) confusion matrix, (4) macro-average recall (balanced accuracy), (5) macro-average precision, and (6) macro F1-score.



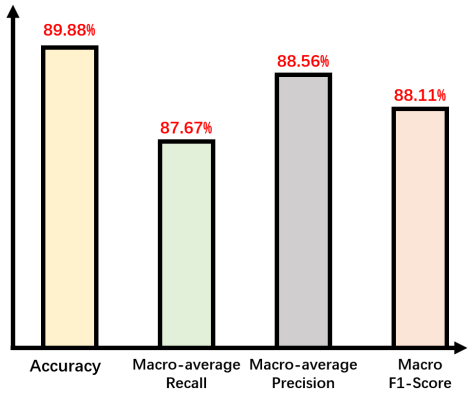
(a) Recall.



(b) Precision.



(c) Confusion matrix.



(d) Additional evaluation metrics.

Figure 3.2: Model performance on different evaluation metrics.

Figure 3.2a presents the recall metric of the proposed model. The recall index measures the model's ability to retrieve all positive examples. Therefore, it is necessary to check the model's recall index on life-threatening human activities, e.g., "falling while walking" and "falling while standing". As shown, the proposed recognition system scored a recall value of 0.8 for the "falling while walking" action and 0.7 for the "falling while standing" action.

Figure 3.2b shows the precision metric of the proposed model. The precision index tells the model's ability to make a correct prediction. In other words, it measures the quality of the model's prediction. Generally, the model must have a high precision if the cost of acting after an event happens is higher than not acting.

Figure 3.2c gives the confusion matrix of the proposed model. Each row of the confusion matrix represents the true label, while each column represents the predicted label. The confusion matrix is often used to check if the model is mislabeling two classes. Based on the confusion matrix and an inspection of the test data, there are mainly three causes behind the observed confusion patterns:

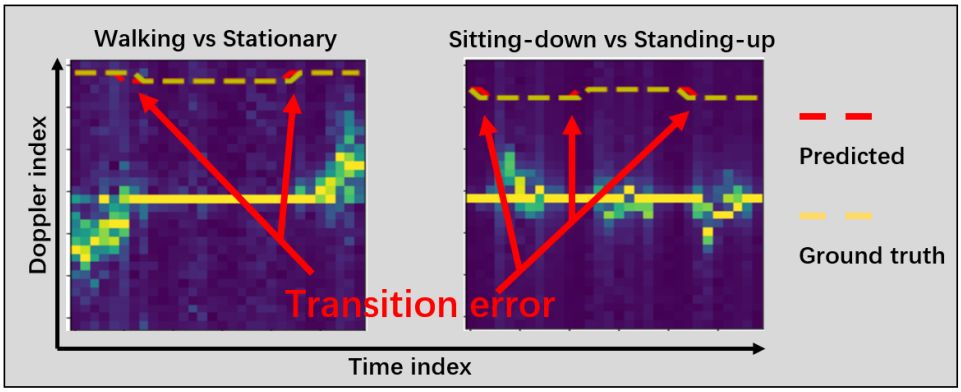
1. The most prominent error is the transition error, i.e., the target translates its activity from one class to another. However, this confusion happens not only to the recognition system but also to the activity performer, since it is hard for the performer to perfectly define the transition point of two consecutive activities. An illustration of the transition error is shown in Figure 3.3a.

2. The second type of error is the boundary error. From previous experiments, it is known that both the past and future temporal information is helpful for model prediction. However, for the predictions at the boundaries of the input spectrogram, there are not enough forward or backward features that the recognition system can use. An illustration of the boundary error is presented in Figure 3.3b.

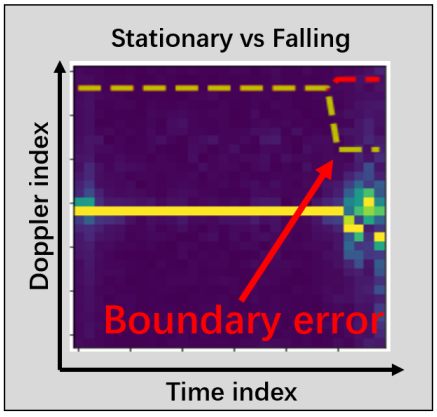
3. The third type of error is the label error. It is a typical human error that can happen at any time during the target's movement. As shown in Figure 3.3c, where the target performed the activity of "stationary first, then walking". However, the ground truth label (marked in dashed yellow line) indicates that the target was performing "walking" only, though the recognition system correctly predicted the "stationary" action (marked in dashed red line).

Finally, figure 3.2d provides three additional metrics for model evaluation, including: (1) macro-average recall, (2) macro-average precision, and (3) macro F1-score. The three additional metrics use the macro-average method to evaluate the model. The macro-average method gives equal importance to all classes, regardless of their size. Thus, they are suitable metrics when we care about the model performance for each class rather than for each example. The macro-average recall and macro-average precision reflect the averaged recall and precision performance across different classes. The macro F1-score, on the other hand, aggregates the macro-average recall and macro-average precision. It indicates the model's overall performance on all classes.

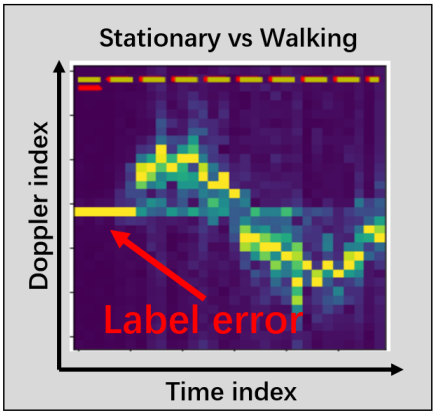
The model accuracy is plotted in the figure as a comparison to other metrics. Based on the results, even though the dataset for model training and testing is imbalanced, the proposed model is able to score well on the three additional metrics.



(a) An illustration of transition error. (Error happens when the target translate from one activity to another)



(b) An illustration of boundary error.(Error happens at the beginning or end of of the input spectrogram)



(c) An illustration of label error. (Error happens due to mislabelling of the target's activity)

Figure 3.3: Error analysis based on the test data.

3.9. GENERALIZATION CAPABILITY TEST

Given the fact that the dataset used for system development is small with only 14 participants, it is important to know the generalization capability and robustness of the proposed recognition system for unknown target. Therefore, the L1PO method is combined with the K-fold cross-validation method to measure the system performance more rigorously.

Table 3.10 shows the results of 14 experiments. For each experiment, one person out of the 14 participants is selected as the test dataset. The remaining data is used for model training and validation. According to the K-fold cross-validation method, for each experiment, the model is trained five times, and the averaged validation accuracy and test accuracy are recorded. In total, the proposed model is trained, validated, and tested 70 times, respectively. Finally, all the measured results are averaged across the 14

participants to have a more accurate performance estimation. Based on the results, the following conclusions can be made:

1. The proposed recognition system is able to generalize the different human activity patterns well from the training dataset. It achieved an average accuracy of 84.21% using the test data that is never seen to the model. This is an acceptable model accuracy considering classifying human activities of nine classes and using a strict performance measurement method.
2. The results also reflect the necessity of combining the L1PO method with the K-fold cross-validation method for model evaluation. As is shown, the test accuracy varies from person to person (ranging from 89.88% to 76.16%). Without using the proposed evaluation method, the evaluation result can be severely biased.

Participant Chosen as the Test Data	Validation Accuracy (5-fold averaged)	Standard deviation (Validation Data)	Test Accuracy (unseen participants)	Standard deviation (Test Data)
Person A	86.50%	0.96	87.32%	1.33
Person B	87.06%	1.13	89.54%	0.23
Person C	86.89%	1.24	88.10%	0.61
Person D	87.11%	0.65	80.88%	1.20
Person E	86.67%	1.04	87.89%	0.86
Person F	86.60%	0.81	84.66%	0.51
Person G	88.13%	0.66	79.97%	0.72
Person H	87.09%	0.85	81.35%	1.08
Person I	87.71%	0.54	81.15%	1.02
Person J	86.85%	0.84	89.23%	0.62
Person K	87.50%	0.48	79.53%	1.33
Person L	87.68%	0.71	76.16%	0.97
Person M	86.61%	0.41	83.25%	1.65
Person N	87.11%	1.11	89.88%	0.52
Mean Value	**87.11%**	**0.82**	**84.21%**	**0.9**

Table 3.10: System performance across all participants. The mean value is the averaged accuracy and standard deviation over all people (from person A to person N).

3.10. SUMMARY

This chapter presents the evaluation results of the proposed HAR system.

Section 3.1 shows the relationship between the system performance and the number of CNN layers. The result indicates that the CNN-RNN architecture has a sweet spot in choosing the number of CNN layers. A subsequent experiment has shown that the number of RNN layers can also influence the depth of the CNN block.

One effective way to reduce the number of model parameters and tackle the limited dataset problem is to apply the weight sharing strategy to the neural network. Section 3.2 provides the related result that shows the exact advantages of the weight sharing strategy. Comparing to the recognition system without weight sharing, the proposed system has fewer model parameters but higher validation accuracy.

The comparison between different types of data fusion methods is explored in Section 3.3. The result indicates that the halfway fusion method is more promising in terms of model performance and complexity than its counterparts. Moreover, the result also shows the advantages of using the data fusion technique as all tested fusion methods outperform the model without data fusion.

In Section 3.5, the influence of the depth of the RNN block on the model performance is investigated. Similar to the conclusion in Section 3.1, the result shows that overshooting the number of RNN layers may not increase the model performance further. Moreover, it is observed that the CNN and RNN block have mutual influence, and the CNN-RNN architecture can actually lead to a more light-weighted neural network.

One disadvantage of the conventional basic RNN is its inability to extract long-term dependencies and exploit future information. Section 3.5 explored the model performance under different types of RNN architecture. The result shows that both the GRU and LSTM outperform the basic RNN. Moreover, the bidirectional structure leads to a massive performance leap in the validation accuracy. Considering the model complexity and validation accuracy, the Bi-GRU is selected for the final recognition system.

Deep learning tools are known to have an overfitting problem. Section 3.6 indicates that the dropout layer is one possible solution to address this problem. As shown in the result, the neural network with the dropout layer added achieves better classification accuracy. Moreover, it is observed that the gap between the training and validation accuracy is reduced due to the dropout layer.

Section 3.7 compares the proposed recognition system with two counterparts from the literature. Although one of the counterparts has a different experimental setting, it is evident that the proposed system is able to handle more realistic HAR scenarios. As for another counterpart, the proposed system provides an end-to-end activity classification and data fusion solution. Moreover, the proposed system also shows a higher recognition accuracy.

Section 3.8 provides the evaluation results in which several famous evaluation metrics for machine learning models are used to inspect the proposed recognition system from a different angle. The result shows that the proposed system is able to score well on other more class-balanced metrics. Moreover, an error analysis is conducted based on the error patterns observed in the confusion matrix.

Finally, Section 3.9 provides a more rigorous test result of the proposed system. The L1PO method and K-fold cross-validation method are applied to measure the system performance across all 14 participants. The result shows that the proposed system is

able to generalize to different human activity patterns well as it achieves an average classification accuracy of 84.21% tested on the unseen dataset.

4

CONCLUSION AND FUTURE WORK

This chapter presents the conclusion and future work for the proposed human activity recognition system.

4.1. CONCLUSION

The following keywords conclude the main results of this thesis work:

1. **An End-to-End Solution**:
   The proposed classifier provides an end-to-end solution for human activity classification problems. Thanks to deep learning approaches, the classifier can conduct automatic feature extraction and data fusion without the need for human engineering. To the best of our knowledge, this is the first work that uses neural networks for continuous and unconstrained human activity classification. The result shows that the proposed system achieves 89.88% accuracy on unseen data for nine-class classification.

2. **Spatio-Temporal Feature Extraction**:
   A hybrid CNN-RNN architecture is proposed to capture the spatio-temporal characteristics in the input spectrograms. The CNN architecture is used to extract the local correlations, while the RNN architecture exploits the long-term and short-term temporal dependencies from both forward and backward directions. The result proves the importance of the two types of architectures in the hybrid model.

3. **Realistic Human Activity**:
   A more challenging and realistic dataset is used to evaluate the proposed classifier. During data collection, the participants are allowed to perform continuous human

activities with unconstrained moving trajectories. Moreover, each recorded data contains a mix of in-place and translational activities with variable durations. The result shows that the proposed classifier is able to score 0.88 on the Macro F1-score metric for nine-class classification.

4. **Hyperparameter Searching**:
   To find the optimal configurations for the hybrid CNN-RNN architecture, a thorough hyperparameter tuning procedure is conducted. The result indicates that the hybrid architecture can lead to a more compact and light-weighted classifier. This observation is important for radar-based classification tasks since acquiring a large dataset for training a complex model is expensive.

5. **Neural Network-based Data Fusion**:
   To handle the challenge of arbitrary moving directions, three fusion strategies are explored in this thesis work, including (1) early fusion (or signal fusion), (2) late fusion (or decision fusion), and (3) halfway fusion (or feature fusion). These fusion strategies are defined depending on the location where the multi-radar information is fused. Three fusion positions are selected to combine the features at different representation levels. The result indicates that the halfway fusion method achieves the best classification accuracy among the three. However, all fusion methods outperform the case when only a single radar is used.

6. **Weight Sharing**:
   Despite the advantages, halfway fusion is often hard to train due to the duplicated multiple channels. To address this issue, the weight sharing method is applied on these duplicates. The result shows the weight sharing method can significantly reduce the model parameters, from 229K to 71K. Moreover, the model with the weight sharing method achieves higher classification accuracy.

7. **Model Evaluation**:
   A more rigorous model evaluation method is proposed in this work. This method combines the popular K-fold cross-validation method with the leave-one-person-out method. It can efficiently use the limited dataset to verify the generalization capability and robustness of the proposed model. The evaluation result shows that the proposed classifier achieved 84.21% accuracy averaged over 14 participants.

4.2. FUTURE WORK

Although the main challenges for human activity recognition problems have been investigated in this thesis, several promising aspects can be examined in future research for further performance improvement, including:

1. **Loss Function**:
   The previous result shows that the transition error is pervasive in the predictions. However, it is hard for the participant to define the starting and ending points of a

set of continuous activities accurately, i.e., a precise activity-label alignment is difficult. To address this issue, other loss functions, e.g., the connectionist temporal classification (CTC) [139] loss, can be considered. Preliminary result shows adding the CTC loss to the proposed recognition system can significantly improve the F1 score of the activity "falling while walking" from 0.71 to 0.92 (tested on the Person L).

2. **Imbalanced Dataset**:
   As shown in Table 2.1, the collected radar data is imbalanced. Using an imbalanced dataset, the neural network will tend to learn how to make correct predictions for the majority class only. Simple remedies for this problem can be:

   (a) Down-sampling the number of data in the majority classes.

   (b) Over-sampling the data in the minority classes.

   (c) Applying class weighting to the loss function.

   More advanced approaches can be:

   (a) Data augmentation [140].

   (b) Adding synthetic data [141].

3. **Boundary Error**:
   The boundary error happens at the two boundaries of the input spectrograms. For a real-time recognition system, using the sliding-window spectrogram method to generate the input data, this type of error is unavoidable due to the lack of future and past information. One possible future direction is to consider applying a weighted moving average filter on the prediction. However, this method will add delay to the real-time system.

4. **Data Representation**:
   Last but not least, different data representations can be further investigated. This thesis work uses the spectrogram to capture the time-frequency features of the moving target. However, previous works [142–144] have shown that combining multi-domain information for classification is advantageous, e.g., combining the range-Doppler, Doppler-time, Cadence Velocity Diagram, and range-time information.

IV

Closing Remarks

1

CONCLUSION AND FUTURE WORK

A detailed discussion over the conclusion and future work for the proposed tracking and activity recognition system is presented in Part II, Chapter 4.1 and 4.2, and Part III, Chapter 4.1 and 4.2, respectively. To avoid repetition, this chapter provides a high-level summary of this thesis work. Having said that, Section 1.1 lists the main conclusions of this work, and Section 1.2 provides several interesting future directions for investigating the radar-based joint tracking and activity classification system.

1.1. CONCLUSION

Joint tracking and classification is the final goal for many radar-based applications. It is especially true for indoor human monitoring since not only knowing where the targets are is important, but also understanding what kinds of activity they are performing. This is helpful to prevent casualties caused by life-threatening activities like "falling on the ground" from happening to vulnerable people.

However, implementing joint tracking and classification is not just connecting two systems together since there are mutual dependencies and requirements between them. For example, targets are allowed to move freely in tracking tasks. In return, targets' moving characteristics such as the Doppler signature are needed in classification tasks. Not to mention the general questions regarding how to tracking multiple targets and how to recognize different activities.

Therefore, this thesis aims to not only build two connectable systems, one for multiple target tracking (MTT) and another for human activity classification (HAR), but also address some of the problems that existed in the joint system. To achieve this goal, this thesis proposed an MTT system and a HAR system based on a distributed IR-UWB radar

sensor network (RSN). The main contributions of this work can be summarized as follows:

1. **System for Multiple Target Tracking**
   The proposed MTT system is capable of tracking multiple extended targets and extract their Doppler signatures for the proposed HAR system. It uses a decentralized tracking architecture to improve the tracking accuracy and its ability against clutter. Also, except for solving the general tracking problems, such as tracking an unknown number of targets, the proposed system addressed two additional issues that were rarely explored in the IR-UWB RSN-based tracking literature. The first problem relates to the target merging effect, and the second problem is about the false alarm introduced by the detection fusion center. Furthermore, a simulator, which models the uncertainties in MTT, is constructed for generating multitarget tracking data. The simulated data provides the means of analyzing the system performance quantitatively. Moreover, the proposed system is tested on experimental radar data. The result shows that the proposed system can successfully extract the Doppler signatures of the target from each radar channel.

2. **System for Human Activity Recognition**
   The proposed HAR system provides an end-to-end solution for data fusion and activity classification. It is built based on deep learning tools, which allow it to conduct automatic feature extraction. In addition, the proposed system uses a hybrid network architecture, which consists of the convolutional and recurrent neural networks, to directly exploit the spatial and temporal characteristics in the input data. Furthermore, the proposed system is evaluated under a more challenging experimental setting. More specifically, it allows the participant to move arbitrarily inside the measurement area while conducting a set of pre-selected activities. To the best of my knowledge, this is the first work that uses neural networks and radar data to classify continuous human activities with unconstrained moving directions and inter-activity transition. For a nine-class classification task, it achieves an accuracy of 89.88% tested on the unseen target.

Due to the time constraint, a thorough investigation of the combination of the proposed MTT and HAR system is left for future exploration. Still, the work of this thesis provided a foundation for their combination and showed improvements in both tracking and activity recognition tasks compared to the state-of-the-art.

1.2. FUTURE WORK

The suggested future works with regard to the joint tracking and recognition system are summarized as follows:

1. **System Interactions Between MTT and HAR**:
   Integrating the tracking system with the activity recognition system provides a

promising future investigation of the possible system interactions. This is because the cooperation among different signal processing components in the joint system may help improve the overall performance. For example, the classification result might be used to help the multitarget tracker select different motion models for the targets under track. In return, the estimated kinematic information provided by the tracker may help the recognition system discriminate in-place and translational activities [145].

2. **Radar Deployment Geometries**:
   A recent work [22] based on synthetic radar data indicates that the radar deployment geometry may significantly influence classification accuracy. Moreover, a similar result was reported in [20], which demonstrates the impact of node positions on coverage percentage, required transmitted power, and localization accuracy. Therefore, a possible future direction can be to improve the performance of the joint tracking and recognition system by considering a different deployment geometry.

3. **System Integration and Evaluation**
   Although the proposed recognition system was tested using experimental radar data, the proposed tracking system was mainly evaluated using simulated multitarget data. Moreover, the two systems were developed based on two different programming languages, i.e., MATLAB and Python. Thus, it might be interesting to migrate these two systems into the same programming environment and measure the performance of the integrated system using the real radar data.

4. **Heterogeneous Sensor Network**
   The radar sensor network used in this work consists of five identical IR-UWB radar sensors. Thus, the data fusion steps in the tracking and classification system are straightforward. However, it might be interesting to consider using a heterogeneous sensor network for joint tracking and classification.

2

ACKNOWLEDGEMENT

I know life is short and time flies, but I never thought that two years of my life could have passed like an electromagnetic wave. Maybe this is why people always say that the pleasant time is always short. Indeed, the time I spent at TU Delft was so memorable and precious. I still remember all the interesting discussions I had with my classmates; I remember all the sleepless nights I had before the exams; I remember all the excellent lectures I had with my professors; I remember all the good things. However, all good things have to come to an end. Before I leave this lovely place, I would like to offer my most sincere thanks to the people who have appeared in my field of view. To those empower me with knowledge, and guide me to propagate through all the obstacles.

First of all, I would like to express my gratitude to my supervisor, Prof. Alexander Yarovoy, for his practical suggestions and constructive advice, which played a decisive role in keeping me always on the right track. I also want to express the most profound appreciation to my daily supervisor, Dr. Francesco Fioranelli, who provided me with so much encouragement and guidance throughout my thesis project. I would like to extend my sincere thanks to Dr. Hans Driessen and Dr. Oleg Krasnov, for those wonderful lectures on radar fundamentals and signal processing.

In addition, thanks should also go to all the PhD and MSc students in the Microwave Sensing, Signals and Systems (MS3) group at TU Delft. Thanks for those informative presentations they prepared for every Friday and the good questions they asked, all of these had made me thinking and improving. Special thanks to Ph.D. Ronny Guendel, who made the radar data available to me. Without his help, I would not have had the chance to start this project. I am also grateful to Ph.D. Ignacio Roldan and Peter Svenningsson, who gave me so many valuable feedbacks and suggestions after my mid-term presentations. I wish to thank the MSc students who had worked with me. Although we were working on different projects, all these ingenious ideas and unparalleled supports

they gave me are invaluable.

Last but not the least, I would like to extend my gratitude to all my friends, relatives, and my parents. Thanks for cheering me up when I was low; Thanks for giving me unconditional support when I was hesitating; Thanks for caring about my health when I felt sick. Without all my loved ones, I would never be able to reach this point.

For all the people who have walked into my life, I wish you all the best for every step in your journey and achieve everything you want!

Simin Zhu
Den Haag, September 2021

BIBLIOGRAPHY

[1] Stefania Bartoletti et al. “Sensor radar networks for indoor tracking”. In: *IEEE Wireless Communications Letters* 3.2 (2014), pp. 157–160.

[2] Francesco Fioranelli, Julien Le Kernec, and Syed Aziz Shah. “Radar for health care: Recognizing human activities and monitoring vital signs”. In: *IEEE Potentials* 38.4 (2019), pp. 16–23.

[3] Bahri Çağlıyan and Sevgi Zübeyde Gürbüz. “Micro-Doppler-based human activity classification using the mote-scale BumbleBee radar”. In: *IEEE Geoscience and Remote Sensing Letters* 12.10 (2015), pp. 2135–2139.

[4] Maria-Gabriella Di Benedetto and Guerino Giancola. “Understanding Ultra Wide Band Radio Fundamentals”. In: (2004).

[5] Graeme E Smith, Fauzia Ahmad, and Moeness G Amin. “Micro-Doppler processing for ultra-wideband radar data”. In: *International Society for Optics and Photonics* 8361 (2012), p. 83610L.

[6] Victor C Chen, David Tahmoush, and William J Miceli. “Radar Micro-Doppler Signatures”. In: (2014).

[7] Yuan He et al. “Decentralised tracking for human target in multistatic ultra-wideband radar”. In: *IET Radar, Sonar & Navigation* 8.9 (2014), pp. 1215–1223.

[8] SangHyun Chang et al. “UWB radar-based human target tracking”. In: *2009 IEEE Radar Conference* (2009), pp. 1–6.

[9] JD Bryan et al. “Application of ultra-wide band radar for classification of human activities”. In: *IET Radar, Sonar & Navigation* 6.3 (2012), pp. 172–179.

[10] Rui Qi et al. “Multi-Classification Algorithm for Human Motion Recognition Based on IR-UWB Radar”. In: *IEEE Sensors Journal* 20.21 (2020), pp. 12848–12858.

[11] Pavlina Konstantinova, Alexander Udvarev, and Tzvetan Semerdjiev. “A study of a target tracking algorithm using global nearest neighbor approach”. In: *Proceedings of the International Conference on Computer Systems and Technologies (CompSysTech’03)* (2003), pp. 290–295.

[12] Thomas Fortmann, Yaakov Bar-Shalom, and Molly Scheffe. “Sonar tracking of multiple targets using joint probabilistic data association”. In: *IEEE journal of Oceanic Engineering* 8.3 (1983), pp. 173–184.

[13] Samuel S Blackman. “Multiple hypothesis tracking for multiple target tracking”. In: *IEEE Aerospace and Electronic Systems Magazine* 19.1 (2004), pp. 5–18.

[14] Karl Granstrom, Marcus Baum, and Stephan Reuter. “Extended object tracking: Introduction, overview and applications”. In: *arXiv preprint arXiv:1604.00970* (2016).

[15] Thomas Wagner, Reinhard Feger, and Andreas Stelzer. “Radar signal processing for jointly estimating tracks and micro-Doppler signatures”. In: *IEEE Access* 5 (2017), pp. 1220–1238.

[16] Youngwook Kim and Hao Ling. “Human activity classification based on micro-Doppler signatures using a support vector machine”. In: *IEEE transactions on geoscience and remote sensing* 47.5 (2009), pp. 1328–1337.

[17] Youngwook Kim and Taesup Moon. “Human detection and activity classification based on micro-Doppler signatures using deep convolutional neural networks”. In: *IEEE geoscience and remote sensing letters* 13.1 (2015), pp. 8–12.

[18] Francesco Fioranelli, Matthew Ritchie, and Hugh Griffiths. “Aspect angle dependence and multistatic data fusion for micro-Doppler classification of armed/unarmed personnel”. In: *IET Radar, Sonar & Navigation* 9.9 (2015), pp. 1231–1239.

[19] Aman Shrestha et al. “Continuous human activity classification from FMCW radar with Bi-LSTM networks”. In: *IEEE Sensors Journal* 20.22 (2020), pp. 13607–13619.

[20] Enrico Paolini et al. “Localization capability of cooperative anti-intruder radar systems”. In: *EURASIP Journal on Advances in Signal Processing* 2008 (2008), pp. 1–14.

[21] Dusan Kocur, Jana Rovňáková, and Daniel Urdzık. “Mutual shadowing effect of people tracked by the short-range UWB radar”. In: *2011 34th International Conference on Telecommunications and Signal Processing (TSP)* (2011), pp. 302–306.

[22] Boyu Zhou et al. “Simulation framework for activity recognition and benchmarking in different radar geometries”. In: *IET Radar, Sonar and Navigation* (2020).

[23] Abu Sajana Rahmathullah, Ángel F Garcıa-Fernández, and Lennart Svensson. “Generalized optimal sub-pattern assignment metric”. In: *2017 20th International Conference on Information Fusion (Fusion)* (2017), pp. 1–8.

[24] SangHyun Chang, Naoki Mitsumoto, and Joel W Burdick. “An algorithm for UWB radar-based human detection”. In: *2009 IEEE Radar Conference* (2009), pp. 1–6.

[25] AG Yarovoy et al. “UWB radar for human being detection”. In: *IEEE Aerospace and Electronic Systems Magazine* 21.3 (2006), pp. 10–14.

[26] Qiuchi Jian et al. “Detection of breathing and heartbeat by using a simple UWB radar system”. In: *The 8th European Conference on Antennas and Propagation (EuCAP 2014)* (2014), pp. 3078–3081.

[27] Antonio Lazaro, David Girbau, and Ramon Villarino. “Analysis of vital signs monitoring using an IR-UWB radar”. In: *Progress In Electromagnetics Research* 100 (2010), pp. 265–284.

[28] Jing Li et al. “Through-wall detection of human being’s movement by UWB radar”. In: *IEEE Geoscience and Remote Sensing Letters* 9.6 (2012), pp. 1079–1083.

[29] Sukhvinder Singh et al. “Sense through wall human detection using UWB radar”. In: *EURASIP Journal on Wireless Communications and Networking* 2011.1 (2011), pp. 1–11.

[30] Matti Hämäläinen et al. “Ultra-Wideband Radar-Based Indoor Activity Monitoring for Elderly Care”. In: *Sensors* 21.9 (2021), p. 3158.

[31] Xuanjun Quan, Jeong Woo Choi, and Sung Ho Cho. “A new thresholding method for ir-uwb radar-based detection applications”. In: *Sensors* 20.8 (2020), p. 2314.

[32] Sungwon Yoo et al. “Adaptive clutter suppression algorithm for detection and positioning using IR-UWB radar”. In: (2018), pp. 40–43.

[33] Bita Sobhani et al. “Target tracking for UWB multistatic radar sensor networks”. In: *IEEE Journal of Selected Topics in Signal Processing* 8.1 (2013), pp. 125–136.

[34] Marco Chiani, Andrea Giorgetti, and Enrico Paolini. “Sensor radar for object tracking”. In: *Proceedings of the IEEE* 106.6 (2018), pp. 1022–1041.

[35] SangHyun Chang et al. “People tracking with UWB radar using a multiple-hypothesis tracking of clusters (MHTC) method”. In: *International Journal of Social Robotics* 2.1 (2010), pp. 3–18.

[36] Van-Han Nguyen and Jae-Young Pyun. “Location detection and tracking of moving targets by a 2D IR-UWB radar system”. In: *sensors* 15.3 (2015), pp. 6740–6762.

[37] Jana Rovňaková and Dušan Kocur. “Weak signal enhancement in radar signal processing”. In: *20th International Conference Radioelektronika 2010* (2010), pp. 1–4.

[38] Matthew Ash, Matthew Ritchie, and Kevin Chetty. “On the application of digital moving target indication techniques to short-range FMCW radar data”. In: *IEEE Sensors Journal* 18.10 (2018), pp. 4167–4175.

[39] Hermann Rohling. “Radar CFAR thresholding in clutter and multiple target situations”. In: *IEEE transactions on aerospace and electronic systems* 4 (1983), pp. 608–621.

[40] Hermann Rohling. “Ordered statistic CFAR technique-an overview”. In: *2011 12th International Radar Symposium (IRS)* (2011), pp. 631–638.

[41] Xuanjun Quan, JeongWoo Choi, and Sung Ho Cho. “A miss-detection probability based thresholding algorithm for an IR-UWB radar sensor”. In: *2018 19th International Radar Symposium (IRS)* (2018), pp. 1–8.

[42] Marco Chiani et al. “Target detection metrics and tracking for UWB radar sensor networks”. In: *2009 IEEE International Conference on Ultra-Wideband* (2009), pp. 469–474.

[43] Filippo Valmori et al. “Indoor detection and tracking of human targets with UWB radar sensor networks”. In: *2016 IEEE International Conference on Ubiquitous Wireless Broadband (ICUWB)* (2016), pp. 1–4.

[44] Yuan He, Timofey Savelyev, and Alexander Yarovoy. “Two-stage algorithm for extended target tracking by multistatic UWB radar”. In: *Proceedings of 2011 IEEE CIE International Conference on Radar* 1 (2011), pp. 795–799.

[45] Mária Švecová et al. “Target localization by a multistatic UWB radar”. In: *20th International Conference Radioelektronika 2010* (2010), pp. 1–4.

[46] Snezhana Jovanoska and Reiner Thomä. “Multiple target tracking by a distributed UWB sensor network based on the PHD filter”. In: *2012 15th International Conference on Information Fusion* (2012), pp. 1095–1102.

[47] Bo Yan, Andrea Giorgetti, and Enrico Paolini. “A Track-Before-Detect Algorithm for UWB Radar Sensor Networks”. In: *Signal Processing* (2021), p. 108257.

[48] Ronald Mahler. “PHD filters for nonstandard targets, I: Extended targets”. In: *2009 12th International Conference on Information Fusion* (2009), pp. 915–921.

[49] Karl Granström, Christian Lundquist, and Umut Orguner. “A Gaussian mixture PHD filter for extended target tracking”. In: *2010 13th International Conference on Information Fusion* (2010), pp. 1–8.

[50] Karl Granstrom and Umut Orguner. “A PHD filter for tracking multiple extended targets using random matrices”. In: *IEEE Transactions on Signal Processing* 60.11 (2012), pp. 5657–5671.

[51] James MacQueen et al. “Some methods for classification and analysis of multivariate observations”. In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* 1.14 (1967), pp. 281–297.

[52] Dongkuan Xu and Yingjie Tian. “A comprehensive survey of clustering algorithms”. In: *Annals of Data Science* 2.2 (2015), pp. 165–193.

[53] David Arthur and Sergei Vassilvitskii. “k-means++: The advantages of careful seeding”. In: (2006).

[54] Arthur P Dempster, Nan M Laird, and Donald B Rubin. “Maximum likelihood from incomplete data via the EM algorithm”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1 (1977), pp. 1–22.

[55] Martin Ester et al. “A density-based algorithm for discovering clusters in large spatial databases with noise.” In: *Kdd* 96.34 (1996), pp. 226–231.

[56] Dominik Kellner, Jens Klappstein, and Klaus Dietmayer. “Grid-based DBSCAN for clustering extended objects in radar data”. In: *2012 IEEE Intelligent Vehicles Symposium* (2012), pp. 365–370.

[57] Rudolph Emil Kalman. “A new approach to linear filtering and prediction problems”. In: (1960).

[58] Zhe Chen et al. “Bayesian filtering: From Kalman filters to particle filters, and beyond”. In: *Statistics* 182.1 (2003), pp. 1–69.

[59] Maria Isabel Ribeiro. “Kalman and extended kalman filters: Concept, derivation and properties”. In: *Institute for Systems and Robotics* 43 (2004), p. 46.

[60] Eric A Wan and Rudolph Van Der Merwe. “The unscented Kalman filter for nonlinear estimation”. In: *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)* (2000), pp. 153–158.

[61] Arnaud Doucet, Simon Godsill, and Christophe Andrieu. “On sequential Monte Carlo sampling methods for Bayesian filtering”. In: *Statistics and computing* 10.3 (2000), pp. 197–208.

[62] Harold W Kuhn. “The Hungarian method for the assignment problem”. In: *Naval research logistics quarterly* 2.1-2 (1955), pp. 83–97.

[63] Dimitri P Bertsekas. “The auction algorithm: A distributed relaxation method for the assignment problem”. In: *Annals of operations research* 14.1 (1988), pp. 105–123.

[64] Katta G Murty. “Letter to the editor—An algorithm for ranking all the assignments in order of increasing cost”. In: *Operations research* 16.3 (1968), pp. 682–687.

[65] Hongyu Qian et al. “PLA-JPDA for Indoor Multi-Person Tracking Using IR-UWB Radars”. In: *2020 IEEE Radar Conference (RadarConf20)* (2020), pp. 1–6.

[66] SangHyun Chang, Michael Wolf, and Joel W Burdick. “An MHT algorithm for UWB radar-based multiple human target tracking”. In: *2009 IEEE International Conference on Ultra-Wideband* (2009), pp. 459–463.

[67] Edmund Brekke and Mandar Chitre. “Relationship between finite set statistics and the multiple hypothesis tracker”. In: *IEEE Transactions on Aerospace and Electronic Systems* 54.4 (2018), pp. 1902–1917.

[68] Tod Luginbuhl, Yan Sun, and Peter Willett. “A track management system for the PMHT algorithm”. In: *Proceedings of the 4th International Conference on Information Fusion* (2001).

[69] Karl Granstrom et al. “Random set methods: Estimation of multiple extended objects”. In: *IEEE Robotics & Automation Magazine* 21.2 (2014), pp. 73–82.

[70] Ba Tuong Vo. “Random finite sets in multi-object filtering”. In: (2008).

[71] Irwin R Goodman, Ronald P Mahler, and Hung T Nguyen. “Mathematics of data fusion”. In: 37 (2013).

[72] Ronald PS Mahler. “Multitarget Bayes filtering via first-order multitarget moments”. In: *IEEE Transactions on Aerospace and Electronic systems* 39.4 (2003), pp. 1152–1178.

[73] B-N Vo and W-K Ma. “The Gaussian mixture probability hypothesis density filter”. In: *IEEE Transactions on signal processing* 54.11 (2006), pp. 4091–4104.

[74] B-N Vo, Sumeetpal Singh, and Arnaud Doucet. “Sequential Monte Carlo methods for multitarget filtering with random finite sets”. In: *IEEE Transactions on Aerospace and electronic systems* 41.4 (2005), pp. 1224–1245.

[75] Ba-Tuong Vo, Ba-Ngu Vo, and Antonio Cantoni. “Analytic implementations of the cardinalized probability hypothesis density filter”. In: *IEEE transactions on signal processing* 55.7 (2007), pp. 3553–3567.

[76] Daniel E Clark, Kusha Panta, and Ba-Ngu Vo. “The GM-PHD filter multiple target tracker”. In: *2006 9th International Conference on Information Fusion* (2006), pp. 1–8.

[77] Yuxuan Xia. “Conjugate Priors for Bayesian Object Tracking”. In: (2020).

[78] Ángel F Garcıa-Fernández et al. “Gaussian implementation of the multi-Bernoulli mixture filter”. In: *2019 22th International Conference on Information Fusion (FUSION)* (2019), pp. 1–8.

[79] Jason L Williams. “Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA, and association-based MeMBer”. In: *IEEE Transactions on Aerospace and Electronic Systems* 51.3 (2015), pp. 1664–1687.

[80] Yuxuan Xia et al. “Performance evaluation of multi-Bernoulli conjugate priors for multi-target filtering”. In: *2017 20th International Conference on Information Fusion (Fusion)* (2017), pp. 1–8.

[81] Ángel F Garcıa-Fernández et al. “Poisson multi-Bernoulli mixture filter: direct derivation and implementation”. In: *IEEE Transactions on Aerospace and Electronic Systems* 54.4 (2018), pp. 1883–1901.

[82] Sen Wang. “Multi-Bernoulli Mixture Filter: Complete Derivation and Sequential Monte Carlo Implementation”. In: *arXiv preprint arXiv:1911.03699* (2019).

[83] Jana Rovňáková and Dušan Kocur. “Short range tracking of moving persons by UWB sensor network”. In: *2011 8th European Radar Conference* (2011), pp. 321–324.

[84] Karl Granstrom, Christian Lundquist, and Omut Orguner. “Extended target tracking using a Gaussian-mixture PHD filter”. In: *IEEE Transactions on Aerospace and Electronic Systems* 48.4 (2012), pp. 3268–3286.

[85] Karl Granström, Maryam Fatemi, and Lennart Svensson. “Gamma Gaussian inverse-Wishart Poisson multi-Bernoulli filter for extended target tracking”. In: *2016 19th International Conference on Information Fusion (FUSION)* (2016), pp. 893–900.

[86] Karl Granström, Maryam Fatemi, and Lennart Svensson. “Poisson multi-Bernoulli conjugate prior for multiple extended object estimation”. In: *ArXiv e-prints* (2016).

[87] Karl Granström, Maryam Fatemi, and Lennart Svensson. “Poisson multi-Bernoulli mixture conjugate prior for multiple extended target filtering”. In: *IEEE Transactions on Aerospace and Electronic Systems* 56.1 (2019), pp. 208–225.

[88] Roy Jonker and Anton Volgenant. “A shortest augmenting path algorithm for dense and sparse linear assignment problems”. In: *Computing* 38.4 (1987), pp. 325–340.

[89] Stephan Reuter et al. “The labeled multi-Bernoulli filter”. In: *IEEE Transactions on Signal Processing* 62.12 (2014), pp. 3246–3260.

[90] Dominic Schuhmacher, Ba-Tuong Vo, and Ba-Ngu Vo. “A consistent metric for performance evaluation of multi-object filters”. In: *IEEE transactions on signal processing* 56.8 (2008), pp. 3447–3457.

[91] John R Hoffman and Ronald PS Mahler. “Multitarget miss distance via optimal assignment”. In: *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 34.3 (2004), pp. 327–336.

[92] Lee R Moyer, Jeffrey Spak, and Peter Lamanna. “A multi-dimensional Hough transform-based track-before-detect technique for detecting weak targets in strong clutter backgrounds”. In: *IEEE Transactions on Aerospace and Electronic Systems* 47.4 (2011), pp. 3062–3068.

[93] Yuki Okamoto, Isamu Matsunami, and Akihiro Kajiwara. “Pedestrian and two-wheeler detection using ultra-wideband vehicular radar”. In: *2012 IEEE Sensors Applications Symposium Proceedings* (2012), pp. 1–4.

[94] Francesco Fioranelli, Matthew Ritchie, and Hugh Griffiths. "Classification of unarmed/armed personnel using the NetRAD multistatic radar for micro-Doppler and singular value decomposition features". In: *IEEE Geoscience and Remote Sensing Letters* 12.9 (2015), pp. 1933–1937.

[95] Qingchao Chen et al. "DopNet: A deep convolutional neural network to recognize armed and unarmed human targets". In: *IEEE Sensors Journal* 19.11 (2019), pp. 4160–4172.

[96] Zhenyuan Zhang, Zengshan Tian, and Mu Zhou. "Latern: Dynamic continuous hand gesture recognition using FMCW radar sensor". In: *IEEE Sensors Journal* 18.8 (2018), pp. 3278–3289.

[97] Youngwook Kim and Brian Toomajian. "Hand gesture recognition using micro-Doppler signatures with convolutional neural network". In: *IEEE Access* 4 (2016), pp. 7125–7130.

[98] Enea Cippitelli et al. "Radar and RGB-depth sensors for fall detection: A review". In: *IEEE Sensors Journal* 17.12 (2017), pp. 3585–3604.

[99] Moeness G Amin et al. "Radar signal processing for elderly fall detection: The future for in-home monitoring". In: *IEEE Signal Processing Magazine* 33.2 (2016), pp. 71–80.

[100] Ronny Gerhard Guendel et al. "Continuous human activity recognition for arbitrary directions with distributed radars". In: *2021 IEEE Radar Conference (RadarConf21)* (2021), pp. 1–6.

[101] Syed Aziz Shah and Francesco Fioranelli. "RF sensing technologies for assisted daily living in healthcare: A comprehensive review". In: *IEEE Aerospace and Electronic Systems Magazine* 34.11 (2019), pp. 26–44.

[102] Neil J Gordon, Simon Maskell, and Thiagalingam Kirubarajan. "Efficient particle filters for joint tracking and classification". In: *Signal and Data Processing of Small Targets 2002* 4728 (2002), pp. 439–449.

[103] Jian Lan and X Rong Li. "Joint tracking and classification of extended object using random matrix". In: *Proceedings of the 16th International Conference on Information Fusion* (2013), pp. 1550–1557.

[104] WD Van Eeden et al. "Micro-Doppler radar classification of humans and animals in an operational environment". In: *Expert Systems with Applications* 102 (2018), pp. 1–11.

[105] Nello Cristianini, John Shawe-Taylor, et al. "An introduction to support vector machines and other kernel-based learning methods". In: (2000).

[106] Yanmin Qian et al. "Very deep convolutional neural networks for noise robust speech recognition". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24.12 (2016), pp. 2263–2276.

[107] Ying Zhang et al. "Towards end-to-end speech recognition with deep convolutional neural networks". In: *arXiv preprint arXiv:1701.02720* (2017).

[108] Haoyang Jiang et al. “Human activity classification using radar signal and RNN networks”. In: (2021).

[109] Haobo Li et al. “Bi-LSTM network for multimodal continuous human activity recognition and fall detection”. In: *IEEE Sensors Journal* 20.3 (2019), pp. 1191–1201.

[110] Haobo Li et al. “Sequential human gait classification with distributed radar sensor fusion”. In: *IEEE Sensors Journal* 21.6 (2020), pp. 7590–7603.

[111] Vishal Passricha and Rajesh Kumar Aggarwal. “A hybrid of deep CNN and bidirectional LSTM for automatic speech recognition”. In: *Journal of Intelligent Systems* 29.1 (2020), pp. 1261–1274.

[112] William Song and Jim Cai. “End-to-end deep neural network for automatic speech recognition”. In: *Standford CS224D Reports* (2015).

[113] Jianping Zhu, Haiquan Chen, and Wenbin Ye. “A hybrid CNN–LSTM network for the classification of human activities based on micro-Doppler radar”. In: *IEEE Access* 8 (2020), pp. 24713–24720.

[114] Julien Maitre, Kevin Bouchard, and Sebastien Gaboury. “Fall detection with UWB radars and CNN-LSTM architecture”. In: *IEEE journal of biomedical and health informatics* 25.4 (2020), pp. 1273–1283.

[115] Dave Tahmoush and Jerry Silvious. “Radar micro-Doppler for long range front-view gait recognition”. In: *2009 IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems* (2009), pp. 1–6.

[116] Chuanwei Ding et al. “Continuous human motion recognition with a dynamic range-Doppler trajectory method based on FMCW radar”. In: *IEEE Transactions on Geoscience and Remote Sensing* 57.9 (2019), pp. 6821–6831.

[117] Ljubisa Stankovic, Miloš Daković, and Thayannathan Thayaparan. “Time-frequency signal analysis with applications”. In: (2014).

[118] Daniel A Brooks et al. “Complex-valued neural networks for fully-temporal micro-Doppler classification”. In: *2019 20th International Radar Symposium (IRS)* (2019), pp. 1–10.

[119] Thomas Stadelmayer et al. “Data-driven Radar Processing Using a Parametric Convolutional Neural Network for Human Activity Classification”. In: *IEEE Sensors Journal* (2021).

[120] Zhaoxi Chen et al. “Personnel recognition and gait classification based on multistatic micro-Doppler signatures using deep convolutional neural networks”. In: *IEEE Geoscience and Remote Sensing Letters* 15.5 (2018), pp. 669–673.

[121] Sergey Ioffe and Christian Szegedy. “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: *International conference on machine learning* (2015), pp. 448–456.

[122] Shibani Santurkar et al. “How does batch normalization help optimization?” In: *Proceedings of the 32nd international conference on neural information processing systems* (2018), pp. 2488–2498.

[123] Yanzhe Wang et al. “End-to-End Mandarin Recognition based on Convolution Input”. In: *MATEC Web of Conferences* 214 (2018), p. 01004.

[124] Chigozie Nwankpa et al. “Activation functions: Comparison of trends in practice and research for deep learning”. In: *arXiv preprint arXiv:1811.03378* (2018).

[125] Vinod Nair and Geoffrey E Hinton. “Rectified linear units improve restricted boltzmann machines”. In: *Icml* (2010).

[126] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. “Deep learning”. In: (2016).

[127] Christian Szegedy et al. “Going deeper with convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 1–9.

[128] Syed Aziz Shah and Francesco Fioranelli. “Human activity recognition: Preliminary results for dataset portability using FMCW radar”. In: *2019 International Radar Conference (RADAR)* (2019), pp. 1–4.

[129] Md Atiqur Rahman and Robert Laganière. “Mid-level Fusion for End-to-End Temporal Activity Detection in Untrimmed Video.” In: (2020).

[130] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory”. In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[131] Kyunghyun Cho et al. “Learning phrase representations using RNN encoder-decoder for statistical machine translation”. In: *arXiv preprint arXiv:1406.1078* (2014).

[132] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. “Learning long-term dependencies with gradient descent is difficult”. In: *IEEE transactions on neural networks* 5.2 (1994), pp. 157–166.

[133] Mike Schuster and Kuldip K Paliwal. “Bidirectional recurrent neural networks”. In: *IEEE transactions on Signal Processing* 45.11 (1997), pp. 2673–2681.

[134] Nitish Srivastava et al. “Dropout: a simple way to prevent neural networks from overfitting”. In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[135] Weizhe Wang, Xiaodong Yang, and Hongwu Yang. “End-to-end low-resource speech recognition with a deep cnn-lstm encoder”. In: *2020 IEEE 3rd International Conference on Information Communication and Signal Processing (ICICSP)* (2020), pp. 158–162.

[136] Tyler S Jordan. “Using convolutional neural networks for human activity classification on micro-Doppler radar spectrograms”. In: *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security, Defense, and Law Enforcement Applications XV* 9825 (2016), p. 982509.

[137] Hao Du, Yuan He, and Tian Jin. “Transfer learning for human activities classification using micro-Doppler spectrograms”. In: *2018 IEEE International Conference on Computational Electromagnetics (ICCEM)* (2018), pp. 1–3.

[138] Yuichiro Anzai. “Pattern recognition and machine learning”. In: (2012).

[139] Alex Graves et al. “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks”. In: (2006), pp. 369–376.

[140] Donghong She, Xin Lou, and Wenbin Ye. “RadarSpecAugment: A Simple Data Augmentation Method for Radar-Based Human Activity Recognition”. In: *IEEE Sensors Letters* 5.4 (2021), pp. 1–4.

[141] Sevgi Z Gurbuz et al. “Cross-frequency training with adversarial learning for radar micro-Doppler signature classification (Rising Researcher)”. In: *Radar Sensor Technology XXIV* 11408 (2020), 114080A.

[142] Zhenghui Li et al. “Multi-domains based human activity classification in radar”. In: (2021).

[143] Shaoxuan Li et al. “Elderly Care: Using Deep Learning for Multi-Domain Activity Classification”. In: *2020 International Conference on UK-China Emerging Technologies (UCET)* (2020), pp. 1–4.

[144] Mu Jia et al. “Human activity classification with radar signal processing and machine learning”. In: *2020 International Conference on UK-China Emerging Technologies (UCET)* (2020), pp. 1–5.

[145] Ronny G Guendel, Francesco Fioranelli, and Alexander Yarovoy. “Derivative target line (DTL) for continuous human activity detection and recognition”. In: *2020 IEEE Radar Conference (RadarConf20)* (2020), pp. 1–6.