Multi-Radar Fusion for Failure-tolerant Vulnerable Road Users Classification

Maxim Rykunov[#1], Eddy De Greef[#], Habib-Ur-Rehman Khalid[*#], Kheireddine Aziz[*#], André Bourdoux[#2], Hichem Sahli[*#]

[#]Interuniversity Microelectronics Centre (imec), Leuven, Belgium
[*]Vrije Universiteit Brussel (VUB), Dept. of Electronics & Informatics (ETRO), Brussels, Belgium
{[1]maxim.rykunov, [2]andre.bourdoux}@imec.be

***Abstract* — Data fusion is one of the key aspects in robust and failure-tolerant vulnerable road user (VRU) perception systems. This paper presents a multi-radar sensor fusion platform that enables automatic detection, tracking and classification of pedestrians and cyclists while aiming to support fail-safe system operation. The sensor fusion platform encapsulates two main modules working concurrently: the first detection-to-detection fusion module performs a spatio-temporal alignment of radar detections, data association of the aligned detections and finally multi-object tracking; the second module is the sensor failure management block, which is responsible for fault-tolerant system operation and encapsulates multilevel verification subsystem. The proposed multi-radar fusion system is experimentally evaluated through multi-target scenarios. Demonstrated results show effectiveness of the proposed platform and failure-tolerance of the system compared to a single sensor solution.**

***Keywords* — Radar, sensor fusion, VRU, failure-tolerant, redundancy, target classification.**

I. Introduction

Multi-sensor data fusion has already demonstrated and been proved to be an efficient approach for accurate, environment-resistant, and failure-tolerant data sensing and processing in many applications and especially in automotive ones like monitoring VRUs such as pedestrians and cyclists. These applications in particular require high precision measurements and fail-tolerant sensors [1]. There are countless number of works showing fusion approaches of different sensor modalities: radar, vision, LIDAR, ultrasound, etc. [2], [3]. However very few are focused on homogeneous radar fusion architectures, despite its advanced environment perception sensing capabilities. Furthermore, increasing the number of radars not only increases area coverage, probability of detection, localization, tracking and classification performance but also significantly improves failure-tolerance of the whole system.

This paper was inspired by our work on a radar-camera fusion framework [2], however it targets different sensor modalities with one of the key focus aiming to support fail-safe system operation.

II. Architecture of the Proposed Fusion Platform

The structure of the proposed VRUs perception system is shown in Fig. 1. More details can be found in [2], where the detailed descriptions of modules A (Radar Signal Processing) and B (Detection-to-Detection Fusion) is outlined. Therefore, this section is mostly focused on the main concept with a high-level description of modules A and B, and more details of module C (Sensor Failure Management).

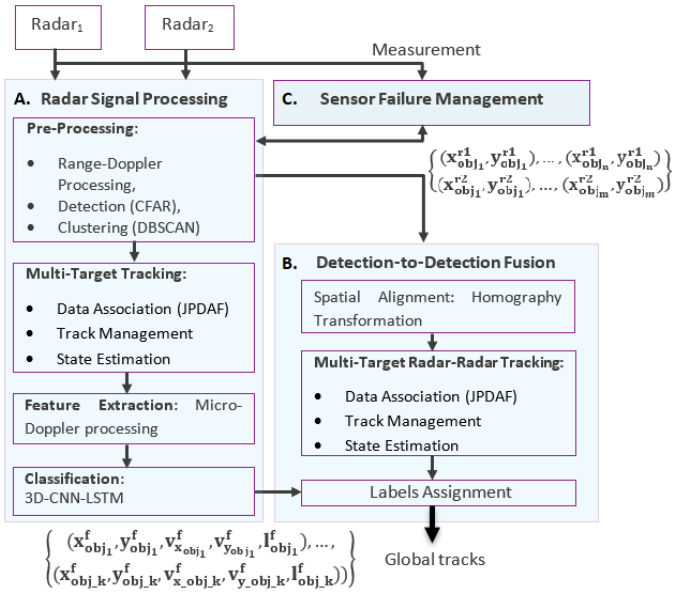


Fig. 1. Overview of the proposed VRUs perception system.

A. Radar Signal Processing

In this work, we used a TI 79 GHz, 3x4MIMO, FMCW (Frequency Modulated Continuous Wave) radar in 3x4 MIMO waveform settings and resulting sensing capabilities summarized in Table 1.

Table 1. 79 GHz radar settings and sensing parameters.

Waveform settings		Sensing parameters	
Chirp BW	0.838GHz	Range Resolution	$0.1789m$
Cube Dimension	(252,3,4,118)	Doppler Resolution	$0.0877m/s$
Sampling Freq	$6.23Msps$	Ambiguous Range	$20m$
Chirp Duration	$29.14\mu s$	Ambiguous Speed	$11.0559m/s$

The Radar Signal Processing module is composed of four sub-modules:

1) Pre-Processing

This stage includes range-Doppler-angle processing with Fourier transforms, detection with a Constant False Alarm Rate algorithm (CFAR) [4] and clustering with the *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN) algorithm.

2) Multi-target tracking

The multi-target tracking block from Fig. 1 includes three main blocks. First the Data Association block where the input measurements are fed to the tracking module, and associated with the current tracks, following their Mahalanobis distance. When two targets are close to each other and share the same observation space, the observations can contribute to update both trajectories. Sharing the observations introduces a coupling into the decision process. JPDAF (Joint Probabilistic Data Association Filter) [5] operates over these possible combinations of matches. Then, at Track Management level, these tracks are treated for track initiation, merging and recovery. Finally, every active track, including the ones in the initialization and tracking states, is predicted and updated in the State Estimation block.

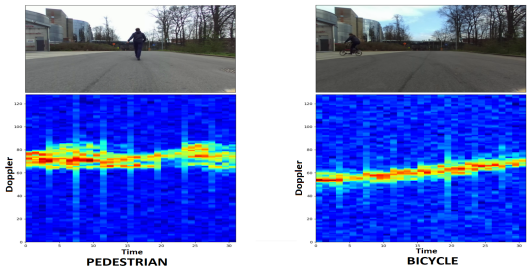


Fig. 2. Example of micro-Doppler images for pedestrian and bicyclist.

3) Feature extraction

This block utilises a 3D tuple (range, Doppler and azimuth coordinates) for each tracked target centroid. Furthermore, to improve classification results, we extract cubelets not only from the current but also from the previous time instances: for each tuple, for the same set of contiguous range and angle coordinates, we select the previous 8 frames in the range-azimuth profile (which amounts to approx. $800msec$ worth of activity) and then apply a Short Time Fourier Transform (STFT) with an overlapping factor of 50%. Fig. 2 shows examples of micro-Doppler images for different targets.

4) Classification

We evaluated several previously developed models [6], [7]. In this work we used a 3D-Convolutional Neural Network (CNN)-Long Short-Term Memory Networks (LSTM) architecture. The architecture is depicted in Fig. 3. The 3D-CNN is used to obtain intermediate short term spatio-temporal features within sub-cubelets, and the LSTM layer is used for the final radar cubelet classification.

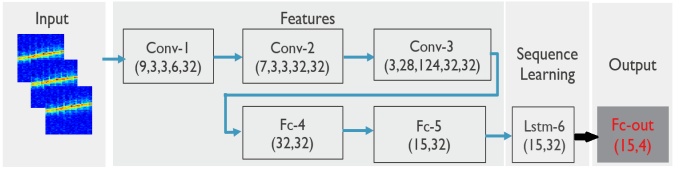


Fig. 3. 3D Convolutional Neural Networks + LSTM scheme.

B. Detection-to-Detection Fusion

The above described radar modules provide detected ROIs of targets, along with their labels. The first step is the projection of the detections on a common coordinate system. This is obtained by estimating the homography between the two radar planes.

1) Spatio-temporal alignment

Based on approaches from the work of [7], radar acquisitions are considered coplanar because they perform a plane projection on the horizontal plane. Consequently, the transformation between the radar planes can be represented by a homography matrix.

Let $(X_1, Y_1, Z_1)^T$ the coordinates of the target in the radar frame and $(X_2, Y_2, Z_2)^T$ its projection on the second radar. For Z = 0 (horizontal plane) they are related as follows:

$$w \times \begin{bmatrix} X_1 \\ Y_1 \\ 1 \end{bmatrix} = H \times \begin{bmatrix} X_2 \\ Y_2 \\ 1 \end{bmatrix} \quad (1)$$

$$where: H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \quad (2)$$

w is a scale factor representing the depth of $(X_1, Y_1, 1)^T$ of one radar relative to another and H is the homography matrix.

To estimate w and H, we used corner reflectors placed at different locations in the environment and observed by both radars. In our experiments, we used 25 correspondences of radar1/radar2 for estimating the homography parameters. Because both radars operate at different acquisition rates, we time-stamped the measurements and updated the system with measurements having the same timestamps. This allows obtaining a uniform synchronization between the sensors.

2) Multi-target radar-radar tracking

After radars alignment, all detections are projected in a common plane. The centralized algorithm takes the current detections as inputs from both sensors and performs tracking in the same way as for the above described multi-objects radar tracking. Each detection consists of three values: 2D coordinates, confidence of classification and label. These detections are then used in the same dynamic model as the one defined for radar-based tracking. Finally, the association of each estimated state i to the corresponding label l_i is made via a voting approach. The association is made with the highest vote for the class.

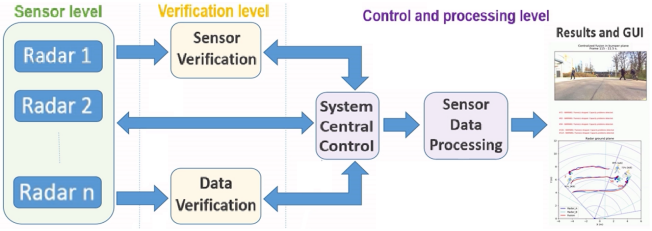


Fig. 4. Multi-level view on the system architecture.

C. Failure sensor management

A homogeneous multi-sensor data fusion system provides a support for perception and processing with increased sensor fail-robustness. The failure sensor management system is the key part responsible of fault-tolerant and fail-operational aspect. This block consists of sensor and data verification parts demonstrated on the verification level see Fig. 4.

1) Sensor verification

As sensors can physically degrade over time, the quality of the received data therefore degrades too. From a pure software architecture point of view, there is very little that can be done about this. Problems like these, however, are detected at the algorithmic level. E.g. noise level and signal-to-noise level measurements are used to detect certain types of signal degradation. Other options, at the fusion level, include cross-correlation of signals between different sensors. E.g. if one sensor misses a signal that is detected by other sensors, this is an indication that this sensor is malfunctioning.

2) Data verification

Several fail-robustness aspects are addressed at this level.

Data integrity: Sensor data can get garbled on the route between the sensor and the processing unit. Therefore, data integrity checks are included, which allows us to verify that the received data frames are not altered and are complete.

Capacity problems: The amount of data to be transferred from the sensors to the processing is very high. Temporary capacity problems result in missing data packets or even complete missing data frames. Data integrity checks are used to identify missing data packets. Frame meta-data (timestamps, sequence numbers) are used to detect missing frames.

Deadlock and excessive latency avoidance: In order to avoid deadlocks (in case of complete sensor failure), and to avoid high latency in the processing (e.g. due to capacity problems) the data processing pipeline should not wait indefinitely for data to arrive from different sensors. If sensor data does not arrive within an acceptable time period, the fusion algorithm should continue with the available data from other sensors.

III. Experimental results

The VRUs perception system was further tested in real life scenarios. For that, we conducted several outdoor measurements to collect real-time samples for further processing and evaluation. The multi-radar setup was placed on two tripods at a height and width targeting a conventional vehicle bumper as illustrated in Fig. 5. There was also a camera placed on a crossbar for obtaining only a ground truth information but not in the main fusion processing pipeline. Furthermore one of two radars was rotated in a horizontal plane towards the other one to increase the overlap of their field of view see Fig. 6b. We collected a dataset for pedestrians and bicycles combining several scenarios of single and multiple heterogeneous and homogeneous targets. The recorded dataset has been used to evaluate the different modules of the proposed above multi-radar fusion framework.

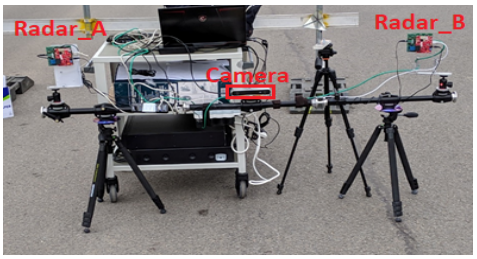


Fig. 5. Overall system and set up for outdoor measurement.

A. Radar Signal Processing

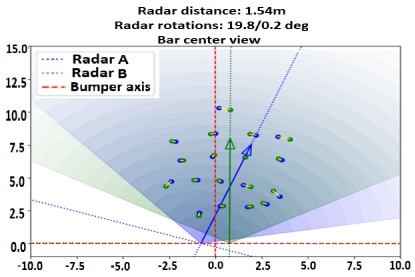
The recorded single-target dataset has been processed by tracking each target and extracting the radar cubes. We then partitioned the extracted data cubes into training, testing and validation data. The 3D-CNN LSTM model discussed above was trained with the batch-size of 8. The input of the model is presented by 3D time-range-Doppler echo of size 64x32x128x6. We train the model with 64 radar data cubes of 32x128 range-Doppler images, where each 2D-image has 6 azimuth channels. The radar data cubes are created around target's centroid with contiguous range and azimuth bins, where the target's centroid is available at each time-step from the JPDAF-tracker. Six azimuth bins along with other dimensions were guided by the cubelet centroid to be focused on the center of the target for the accurate target feature extraction and further into classification. For training, we used an ADAM solver as an optimizer and an initial learning-rate of 1E-4. The model was trained for 50 epochs using an open source machine learning library TensorFlow and NVIDIA TITAN-XP GPU with 12 GB RAM. The effectiveness of the proposed model is tested by a cross-validation technique. Fig. 6a summarizes the obtained classification results of the proposed 3D-CNN+LSTM. As can be seen, the 3D-CNN+LSTM achieves an average recognition accuracy of 94.58%. This confirms that the 3D-CNN provides robust spatio-temporal features able to distinguish between different targets. Combined with the LSTM layer to capture spatio-temporal dependencies, it achieves good results.

B. Detection-to-Detection Fusion

Fig. 6b illustrates the projection average error of the alignment process described above. The blue dots correspond to the different locations of the corner reflector detected in the

	Pedestrian	Cyclist	Recall
Pedestrian	1864	144	92.83
Cyclist	6	754	99.21
Precision	99.68	83.96	94.58

(a)



(b)

Fig. 6. Illustration of the radar signal processing. (a) Confusion matrix for multiple-targets classification, (b) Alignment results

first radar plane. The green dots correspond to the different locations of the corner reflector viewed in the second radar plane and projected on the bumper plane using homography matrix H. We estimated the homography matrix H using the data sets of 18 points. The transformed locations performed in ranges between $[3m, 15m]$ and angles between $[-60°, 60°]$. Fig. 7 illustrates the final results of the fusion algorithm with

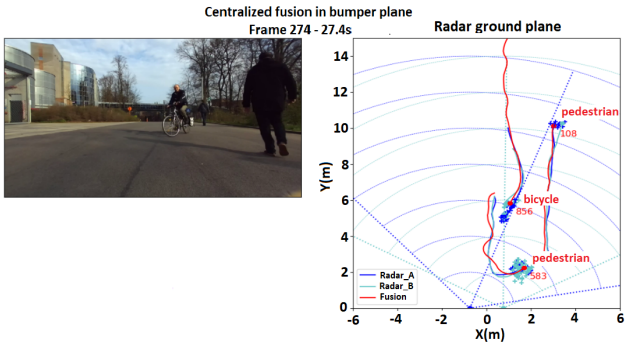


Fig. 7. Results of the centralised fusion framework with multiple targets.

multiple heterogeneous targets. Throughout the experiment, the radar tracks were more stable in the longitudinal direction, which is consistent with the characteristics of the sensor. After fusion processing, the status information of the fusion track and the class were obtained. The information on the fusion track was relatively stable and smooth. In some situations, radar classification fails because of occlusions between the targets. This is explained by the fact that the two targets are closer to each other than the range, Doppler or angle resolution of the radar system.

C. Failure sensor management

In the previous section, we discussed the main structure of the block and types of failures tackled. Here we highlight some real-life results. Although it is not always easy to demonstrate all targeted failures in a real live situation, we highlight one of them - sensor degradation.

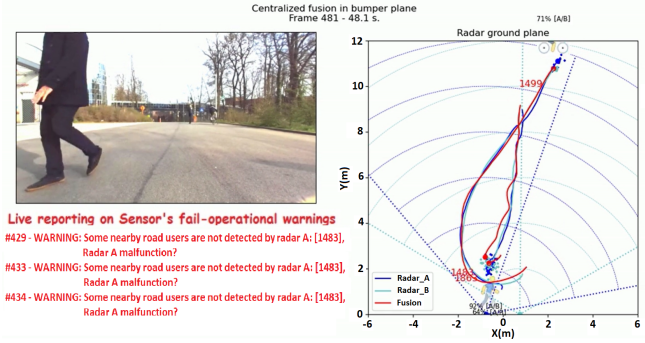


Fig. 8. Pedestrian monitored with one of the radars degraded.

Fig. 8 shows a pedestrian being detected and tracked by our two-radar setup. Later on, we do degradation emulation on one of the radars (Radar A) and it starts completely losing detection of pedestrian. Still, the fusion tracker can track the pedestrian (as is Radar B). Moreover, the failure detection algorithm has generated several warnings over time, as it detects that Radar A is no longer contributing. We detected and indicated other types of failures in a similar way.

IV. Conclusion

In this paper, we proposed a multi-radar data fusion system for failure-tolerant vulnerable road users detection, tracking and classification. The multi-sensor approach was leveraged to fuse the data and improve the quality and robustness of different processing modules. The fusion of the data between two sensors has been improved thanks to the estimated homography transform and the proposed data fusion algorithm. An additional verification level was developed and tested to utilise multi-sensor resources for the sensor fail-robustness aspect of the system.

Acknowledgment

The research leading to these results has received funding from the European Community's ECSEL Joint Undertaking under grant agreement n° 783190 - project PRYSTINE.

References

[1] G. Reina, D. Johnson, and J. Underwood, "Radar sensing for intelligent vehicles in urban environments," *Sensors*, vol. 15, no. 6, 2015.

[2] K.Aziz, E.De Greef, M.Bauduin, M. Rykunov, H. Sahli, and A. Bourdoux, "Radar-camera fusion for road target classification," in *Proceedings of the 2020 IEEE Radar Conferece*, 2020.

[3] Zhengping Ji and D. Prokhorov, "Radar-vision fusion for object classification," in *2008 World Automation Congress*, Sep. 2008, pp. 1–6.

[4] H. Rohling, "Radar cfar thresholding in clutter and multiple target situations," *IEEE TEAS*, pp. 608–621, 1983.

[5] S. Särkkä, T. Tamminen, A. Vehtari, and J. Lampinen, "Probabilistic methods in multiple target tracking - review and bibliography," 2004.

[6] H. Khalid, S. Pollin, M. Rykunov, A. Bourdoux, and H. Sahli, "Convolutional long short-term memory networks for doppler-radar based target classification," in *2019 IEEE RadarConf*, April 2019, pp. 1–6.

[7] A. D. Berenguer, M. C. Oveneke, H. Khalid, M. Alioscha-Perez, A. Bourdoux, and H. Sahli, "Gesturevlad: Combining unsupervised features representation and spatio-temporal aggregation for doppler-radar gesture recognition," *IEEE Access*, vol. 7, pp. 137 122–137 135, 2019.