FMCW radar2radar Interference Detection with a Recurrent Neural Network

Julian Hille[1,2], Daniel Auge[1], Cyprian Grassmann[2], Alois Knoll[1],
[1] Department of Informatics, Technical University of Munich, Munich, Germany
[2] Infineon Technologies AG, Neubiberg, Germany
julian.hille@infineon.com, knoll@in.tum.de

***Abstract*—FMCW radar sensors receive target reflections from the environmental surrounding at frequencies around 77 GHz. The increasing number of sensors on the road operating in this frequency range leads to a higher likelihood of unfavorable radar-to-radar interference. Consequences can be the appearance of artificial targets or the degradation of the noise spectrum, where targets with a small RCS might disappear. We use a Neural Network-based outlier detection method to identify corrupted samples in the time domain signal after the ADC. The architecture consists of a recurrent Neural Network with Long-Short-Term-Memory cells to extract the temporal information. The small network and the stream processing make it suitable for embedded devices. The semi-supervised trained network can detect various interference patterns with reduced training effort. We evaluate the system in a complete pipeline with zeroing mitigation on simulated randomized FMCW data. The method increases the Signal-to-Noise-Ratio ratio by up to 30 dB in the presence of interference and increases the overall system performance and reliability.**

***Index Terms*—Interference, FMCW, Neural Network, LSTM, Semi-Supervised**

I. INTRODUCTION AND BACKGROUND

Radar sensors are relevant elements for environmental sensing in different driving scenarios. In contrast to other sensors such as cameras, they provide direct information about the distance, velocity, and azimuth angle of a target and are capable to work in various weather conditions [1]. Therefore, they are suitable sensors for use in driver assistance systems and autonomous vehicles.

Instead of passive cameras, radar receives target reflections from the transmitted signal and can, therefore, be influenced by other signals. In the literature, this radar-to-radar signal superposition is called interference. More sensors per vehicle increase the probability of interference in the same frequency band under different conditions as shown by [2]. In this work, we focus on radar-to-radar interference of Frequency Modulated Continuous Wave (FMCW) radar sensors.

Outlier detection is an open challenge in different applications such as images, time series, or video sequences. Neural Networks (NNs) outperform traditional methods like Autoregressive–Moving-Average (ARIMA) on time series prediction, as shown with Long-Short-Term-Memory (LSTM) networks by [3]. Therefore, we demonstrate an LSTM network for outlier detection of radar-to-radar interference in the time domain Intermediate Frequency (IF) signal. The network is capable to detect deviations from the expected normality and enables the identification of interfered samples without pre-processing the signal.

This work is part of the KI-ASIC project that is funded by the German Federal Ministry of Education and Research (Grant Number 16ES0992K).

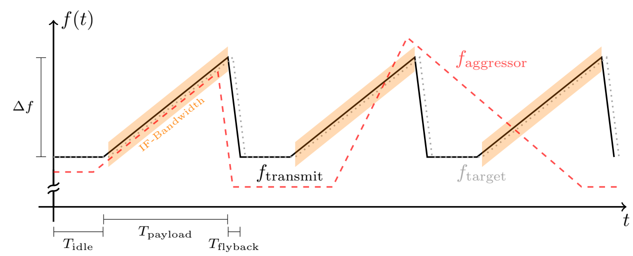

Fig. 1. Possible interference patterns with FMCW radar chirp sequence (black line), example target reflection (dotted gray), and an aggressor chirp sequence (dashed red). The IF-Bandwidth (orange) is defined by the radar settings of the victim sensor. The first chirp with parallel slopes generates a ghost target and the other two interference intersections increase the noise level.

The work is structured as follows: We start with the description of FMCW radar and interference background. Section II provides an overview of interference detection and mitigation methods and Section III describes our proposed methodology. We then explain our setup and the corresponding results in Section IV and follow with a discussion of our observations. The last section outlines conclusions and highlights the importance of real-world datasets for the comparison of interference detection and mitigation methods.

A. FMCW Radar

In the automotive sector, mostly 77 GHz FMCW radar sensors are used. They send out a transmit signal and receive the superposition of the target reflections. The receiver generates the IF signal by mixing the input with the transmit signal. The transmit signal consists of frequency chirps with a linear frequency change, as indicated in Figure 1. The chirp sequence is defined by the individual chirp frequency bandwidth Δf and the chirp timing T_{idle}, $T_{payload}$, and $T_{flyback}$. The received signal from a target is delayed by the round trip time from the sensor to the target and back. Hence, the time delay is proportional to the frequency difference between the transmit and receive signal of a target reflection. The velocity in relation to the radar sensor can be obtained by observing the phase

2022 IEEE Radar Conference (RadarConf22) | 978-1-7281-5368-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/RADARCONF2248738.2022.9764236

978-1-7281-5368-1/22/$31.00 ©2022 IEEE

change across consecutive chirps, called chirp sequence or frame. This phase shift comes from a minor change in the round trip time of the reflected signal. Radar signal processing uses the Fast Fourier Transform (FFT) to extract the range and velocity. The first (range) FFT is applied on every chirp and the second (Doppler shift, velocity) FFT across the chirps. We refer to [4], [5] for more radar signal processing details.

The receiver signal s_{RX} is a superposition of all target reflections and unwanted components like interference. It is defined as

$$s_{\mathrm{RX}}(t) = \underbrace{\sum_{j=0}^{N_{target}} A_j \cdot cos(\varphi_j(t))}_{\text{targets}} + \underbrace{\sum_{i=0}^{N_{interf}} A_i \cdot cos(\varphi_i(t))}_{\text{interference}}, \tag{1}$$

where each target out of N is defined by the amplitude A_j and the phase $\varphi_j(t)$, which depends on the radar transmit power, the antenna gain, the receiver gain, and the target material and distance-dependent Radar Cross Section (RCS). The interference component is depicted by the interference amplitude A_i depending on the transmit power of the aggressor and the time-dependent frequency component $\varphi_i(t)$. The receiver signal is mixed with the transmitter signal to obtain the IF signal. For further details, we refer to an analytical investigation of interference [6].

B. FMCW Interference

Interference occurs in different forms and leads to various performance degradations such as an increase of the noise floor, the loss of targets at specific ranges, or even ghost targets [2]. The reasons are various signal combinations, as indicated by the crossings of the dashed and the constant line in Figure 1. The ego transmitter – defined as the victim – has a constant chirp sequence and a bandwidth-limited IF due to receiver design and settings, indicated with the orange area. A target reflection within the IF band has a stable frequency, amplitude, and phase per chirp. Interference mainly occurs if the instantaneous frequency of the aggressor's transmit signal is in the bandwidth of the victim. In the case of the parallel slope, the IF frequency of the aggressor is constant and will produce a ghost target. Such a scenario, indicated in the first ramp of Figure 1, is improbable [7], [2]. The more common problem is the non-coherent interference that occurs during the crossing of the chirps, as indicated in the two following ramps. The IF frequency of the interference changes linearly but is only defined within a time-limited interval. This interference has a wide frequency spectrum and increases the noise level [7], which can exceed target reflections [8]. The duration and amplitude of the interference affect the performance degradation. The amplitude of an interference signal is often higher than for targets at the same distance because a target has twice the signal path [2]. Hence, an aggressor can affect the victim even outside the detectable distance.

II. RELATED WORK

Interference is a well-known effect in the literature and it is possible to reduce or detect interference in various application domains, such as the optimization of the antenna design, randomization of the chirp sequence, sensor communication, or the processing of the signal in time, or frequency domain. These different methods can be sorted into three categories: avoidance, detection, and mitigation, which we describe in the following.

Avoidance methods reduce the possibility of interference by narrow antenna beams [9], or the randomization of the chirp sequence, by individual ramps with different timing, bandwidth, or initial frequency [10]. Also, the utilization of car-to-car communication can provide deterministic information to adjust the transmitter sub-band or the chirp sequence. Avoidance techniques can reduce the possibility of interference, especially coherent interference [2], but can also affect the processing parameters or introduce delays due to communication overhead.

The interference detection is a crucial component to apply interference mitigation in the time or frequency domain. In the time domain, often dynamic thresholding methods are applied, because the power of the interfered signal is typically higher than the reflections from targets at the same distance [2]. After the first FFT, the change of the Signal-to-Noise-Ratio (SNR) across different chirps indicates interference in one chirp. The exact interference interval cannot be extracted due to its wideband power spectrum. Another method discussed by [9] is to apply a Finite Impulse Response filter in the frequency domain, which combines adjacent frequency bins.

Interference mitigation reduces the affected samples which have been flagged by the detection algorithm. In the time domain, the detected samples can be replaced by zeros, extrapolation, known signal characteristics, or filtered samples [11], [8]. Zeroing is a simple and effective method with low computational costs to mitigate interference.

Other methods utilize the Range-FFT or the Range-Doppler-Map (RDM) for noise reduction and interference detection with mitigation. [12] applies a Convolutional Neural Network (CNN) on the range Short-Time Fourier Transform (STFT), where the 2-dimensional input data size is defined by the FFT points and the window size of the STFT. They propose two CNN architectures with different numbers of convolutional layers to be able to reduce the input dimension from different sizes to a one-dimensional FFT. Therefore, they use up to 21 convolutional layers with Rectified Linear Unit (ReLu) activation and optional pooling layers. The network is trained on an artificial dataset consisting of 40,000 samples in the time domain with and without interference. Another approach proposed by [13], uses a CNN on complex-valued RDMs. After hyperparameter optimization, they focused on a network with 4 layers and 562 kernels of size 3x3. Training is performed on data with and without interference based on a simulated and a combination of simulated interference and real data. [14] shows a similar method for Range-FFT and RDM denoising.

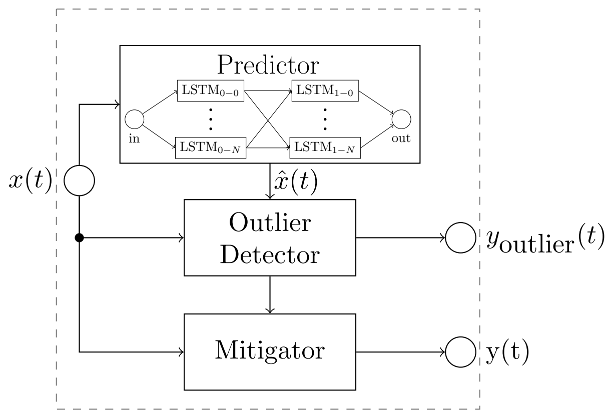

Fig. 2. Overview of the system architecture for FMCW radar-to-radar interference detection in the time domain by using an LSTM network to predict normality. It shows an ideal implementation without temporal latency between input and output.

An often neglected problem of NNs is the generalization of such networks, because during training they are biased to specific noise and interference patterns, and they are not able to perform well on deviating samples. Besides, the computational cost correlates with the number of parameters and the input dimension. The processing of Range-FFTs or RDMs uses a multi-dimensional input and reduces the temporal processing capability. Therefore, we propose a temporal solution for radar-to-radar interference detection to benefit from the small input dimension and enable stream processing.

III. METHOD

The overall architecture is shown in Figure 2 with the continuous time series input $x(t)$ and the corresponding output $y(t)$ with the indication if an outlier was detected $y_{\text{outlier}}(t)$. In the following, we will use the word outlier instead of interference because the approach is not only limited to detect interference. Instead, it is capable to detect deviations from normality, which can be for example radar-to-radar interference or other deviations in the time domain signal.

A. Outlier Detection

We propose the following outlier detection system based on a Recurrent Neural Network (RNN). We use a one-step-ahead prediction consisting of two layers, each with 25 LSTM neurons, to track the temporal dependencies on long and short time spans. The LSTM [15] is a gated neuron with a latent cell state and an input, an output, and a forget gate. These neurons are more complex than traditional neurons with recurrent connections, but reduce the problem of vanishing or exploding gradient during training. The neuron output state depends on the historical and current input values. The hidden state of the last LSTM layer is a limited representation of extracted features from the time series. The network output combines the hidden state in a single linear neuron. For more details about RNNs and LSTMs, we refer to [16]. We have selected our network based on the complexity and the point of diminishing returns of the prediction performance. The network comprises 8026 parameters due to the recurrent connections and the gated LSTM cells. The network is trained in a semi-supervised manner, which reduces the training effort greatly. The LSTM network is thus trained by the expected normality without any outliers. Therefore, the network is highly adjustable to different radar settings and the system can differentiate between normality and various unseen outlier patterns.

Based on the historical and current input, the network predicts the next value, which is used to define the outlier score:

$$score(t) = x(t) - \hat{x}(t), \tag{2}$$

where $x(t)$ is the input value at time t and $\hat{x}(t)$ is the predicted value from the network. We use a static threshold

$$outlier(t) = \begin{cases} 1, & \text{if } |score(t)| > threshold \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

to differentiate between outlier and normality. The performance is similar for dynamic threshold techniques, but it is more hardware-friendly. A lower threshold increases the sensitivity and the probability of false-positive detections. Hence, it is crucial to select the threshold based on the expected variations.

In the task of radar-to-radar interference detection, multiple successive samples are affected, but the algorithm detects mainly the edges of the interference pattern, because the frequency deviation equals zero for the center of the interference. Therefore, we apply an outlier smoothing function to generate sequences of outliers, which we can further use for interference mitigation. If an outlier was detected at time t the probability of an outlier at $t+1$ increases and the same for subsequent outliers. We implemented a counter c with zero as a lower limit to keep all previously detected outliers and an upper limit to adjust the smoothing length to the real interference windows. The upper limit depends on the sampling frequency of the signal and can be selected during simulation. The counter c is defined as:

$$c|_{[0,max]} = \begin{cases} c+1, & \text{if } outlier(t) == 1, \\ c-1, & \text{otherwise.} \end{cases} \tag{4}$$

The network is trained to predict the normality by utilization of Back-Propagation-Through-Time (BPTT). The method unfolds the network over time for back-propagation learning. The computational graph size is defined by the length of the time series with the same weights for all time-steps [16]. We reduce the problem of vanishing gradient in deep computational graphs by splitting the training series into smaller chunks. We use the Adam optimizer for gradient optimization [16].

B. Outlier Mitigation

We combine our NN interference detection with the zeroing mitigation method, which has already shown good performance [11]. It replaces the detected outliers with zeros and

TABLE I
SETTINGS OF THE VICTIM FMCW SENSOR AS DESCRIBED IN FIGURE 1.

Parameter		Value
f_0	initial frequency	77 GHz
Δf	sweep bandwidth	275 MHz
T_{idle}	idle time	$10\mu s$
$T_{payload}$	ramp time	$42\mu s$
$T_{flyback}$	return time	$2\mu s$

TABLE II
PARAMETER RANGE OF THE RANDOMLY GENERATED INTERFERENCE SCENARIOS. SOME SCENARIOS HAVE MINOR EFFECTS ON THE SYSTEM BUT ARE NECESSARY TO SIMULATE REAL-WORLD BEHAVIOR.

Parameter		Range	
f_0	initial frequency	76 GHz	78 GHz
Δf	sweep bandwidth	0.2 GHz	1.5 GHz
T_{idle}	idle time	$2\mu s$	$12\mu s$
$T_{payload}$	ramp time	$40\mu s$	$90\mu s$
$T_{flyback}$	return time	$1\mu s$	$12\mu s$
INR	interference-noise-ratio	10 dB	40 dB

removes interference and signal components. The indication of interfered samples from the outlier detection method is highly affecting the performance of the mitigation. Hence, wrongly flagged outliers result in a signal loss and a bad performance. A limitation of the zeroing mitigation occurs for long interference windows because a large portion of the measurement is zeroed and thus, relevant signal information are lost. This limitation could be reduced by more advanced and complex methods like extrapolation or interpolation, but we state zeroing is an interesting technique due to its simplicity.

IV. SETUP

We generate a simulated dataset with a radar simulator, which includes noise from various sources within the radar receiver chain to emulate real-world behavior. The RX signal is the superposition of target reflections and aggressor signals as in equation 1. Before the Analog-to-Digital Converter (ADC) we apply a band-pass filter to limit the bandwidth of the IF signal. In the digital module, we perform a decimation to reduce the sampling rate. The victim sensor uses eight chirps with the settings in Table I and the aggressor settings vary randomly in the range defined in Table II. The randomly generated aggressor chirps have a high probability to interfere with the victim device. Also, we generated random targets in the environment of the sensor with the setting boundaries shown in Table III. Additionally, we emulate transmitter to receiver leakage, which is a typical effect in real systems.

TABLE III
PARAMETER RANGE FOR EACH RANDOM GENERATED TARGET IN THE RADAR SIMULATION.

Parameter		Range	
d	distance	1 m	200 m
α	azimuth angle	-50°	+50°
v	velocity	0 m/s	40 m/s
RCS	radar cross-section	-20 dBsm	+ 80 dBsm

The system overview with interference detection and mitigation method is shown in Figure 2. The system input is the time domain signal for a complete chirp sequence after the digital decimation. We generate the dataset based on various combinations of target scenarios and interference patterns. The training dataset consists of radar time series without interference but with random target scenarios, where each target is in the parameter range defined in Table III. Due to randomization, we also get data with close targets and high RCS, which leads to a saturation of the limited signal range. During training, we remove these saturated samples and in the evaluation, we use these saturation samples to show that the proposed method can detect further abnormal situations.

The training data consists of 410 radar time series with randomly generated target scenarios without interference. We transform the series into sequences of 100 samples from t_0 until t_{99} with the label t_{100} by using a one-step moving window. The reduced length is important during training to minimize the vanish gradient problem and the training time, which is not necessary during inference. The training loss is the mean squared error of predicted and expected values. During training, it is crucial to continuously observe the performance of unseen data. Thus, we use another dataset of 50 time series without interference to calculate the network validation loss, which is again the mean squared error of expected and predicted samples. Hence, we use complete step-by-step time series predictions for validation, where the input is always the real time series and the output is the prediction of the next value.

For the verification of our outlier detection method, we use a test set with random targets and interference scenarios. It consists of 2266 sequences of which 51.4% include interference scenarios. As shown in Figure 2, we forward the series into the predictor, outlier detector, and mitigator. The outlier labels are generated automatically during the simulation at the known intersection points of the chirps. The interference level is set based on the Interference-to-Noise-Ratio (INR), which we defined between 10 dB and 40 dB. Below that interval, the interference power is low and has a minor effect on the system performance, but exceeding the range leads to saturation of the normalized time signal and can be detected by simple saturation detectors or by the proposed method. Important to mention is that the detected outliers in the first 20 time steps of the proposed method are not valid, due to the settling time of the LSTM network. As already indicated earlier, we use zeroing mitigation to evaluate the improvement of the SNR based on the detected interference after the smoothing function with an upper limit of ten.

V. RESULTS

In the following section, we evaluate the proposed method's performance on the randomly generated dataset. Therefore, the network is trained in PyTorch with 50 epochs, a learning rate of 0.001, and a batch size of 512. The training and validation loss is defined as equal or greater than zero, which means that the deviation of the true from the predicted value is smaller

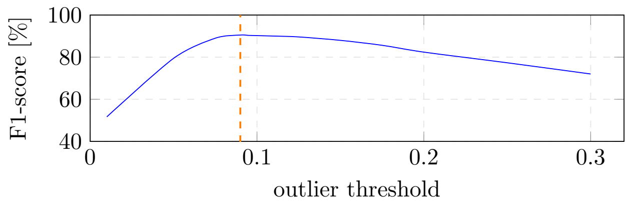

Fig. 3. Dependency between the F1-score and the selected outlier threshold of the test set. The best-performing threshold is highlighted with a dashed orange line at 0.09.

in a system with a loss close to zero. In this work, we reach a loss of 0.0004 and 0.0016 for the training and validation dataset on the given task.

The evaluation of the outlier detection is challenging because the automated labeling process does not perfectly cover the interference pattern due to oscillation effects that occur after interference. Since evaluating the detected outlier for each time step is not possible, we compare whether an outlier was detected in the chirp valid region, which are the important samples for further signal processing. Therefore, we extract the region of interest per chirp for the series and labels. Also, we simulate one additional chirp because the LSTM network needs time to stabilize in the simulation environment, but this chirp is removed before evaluation.

We follow standard statistical metrics to evaluate the performance of the system as described in [17]. The representation of accuracy can be misleading in unbalanced datasets [17]. Hence, we use precision, recall, and the F1-score, which combines precision and recall. Figure 3 shows the relation between the outlier detector threshold value and the F1-score in the test dataset. A low threshold leads to a better recall but reduces heavily the precision. A higher threshold decreases the number of false positive and increases the false negative detections. Based on these results we use the best performing outlier threshold score of 0.09 for further evaluation. The reached precision indicates that 88% of all detected outliers are expected outliers. Deeper observations of the false positive detections show that some cases are valid outliers, which occur due to oscillating disturbances followed by a previous interference outside the valid samples or a deviation in the initial settling. The network reaches a recall of 93%, which indicates the percentage of correctly detected outliers among all labeled outliers. A further investigation of the undetected outliers shows that 89% have a minor influence on the overall performance because the interference affects less than 1% of the samples of each chirp. Overall, the network reaches an F1-score of 90.5%.

The network shows a high detection rate and is also capable to detect unlabeled outliers, which negatively affects the F1-score. Therefore, we also evaluate the performance with the optional zeroing mitigation method. Figure 2 indicates that the proposed system can continuously forward the detected outliers to mitigate interference. For simulation purposes, we

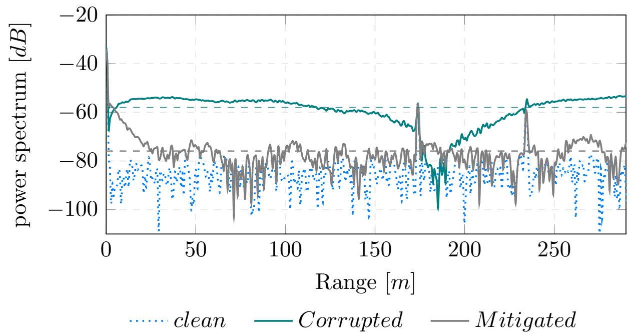

Fig. 4. Example Range-FFT of clean, interfered, and mitigated signal. Mitigation is done by zeroing out the corrupted samples, detected by our proposed outlier detection method. Target at 173 meters is 1 dB below the noise level and not detectable. The interference mitigation improves the noise floor by 18 dB and increases the SNR of the 170-meter target from -1 dB to 19 dB.

extract the valid samples from the radar time series before further processing. We analyze the SNR with and without interference by applying the first FFT. The test dataset contains additional time series without interference because we want to detect all targets with Constant False Alarm Rate (CFAR). We use this CFAR-mask to calculate the mean noise level and signal power for different interference scenarios. Applying CFAR detection to interference samples leads to an undetected loss of targets and the misinterpretation of the SNR. Figure 4 shows an example Range-FFT. The clean signal has a noise level of around -86 dB and there are two targets, one at 173 meters with -56 dB and another at 234 meters with -62 dB. Due to interference, the noise floor increases from -86 dB to -58 dB, and target one has a power level of -59 dB and is not detectable because it is slightly below the noise floor. Applying our proposed interference detection and the zeroing mitigation method improves the noise floor by 18 dB and the targets are well separable from the noise level. The SNR for target one improves by 19 dB and for target two by 12 dB compared to the interfered signal. This architecture achieves an improvement of the noise floor up to 32.17 dB and an average of 1.45 dB for the complete dataset by accounting all detected outliers. It also increases the SNR by up to 30 dB. Additionally, we observe that the performance of the noise floor and SNR can decrease for correctly detected outliers, which indicates limitations of zeroing mitigation, as already reviewed by [11].

VI. Discussion

In this work, we presented a deep learning inspired outlier detection method which we apply to the task “FMCW radar-to-radar interference detection”. The proposed outlier detector consists of a multilayer LSTM network, with high flexibility and pattern independent detection, due to the semi-supervised training.

Other NN-inspired methods use the Range-FFT or RDM to apply convolutional filters to perform image denoising.

The proposed method uses the time domain signal to enable early detection and to reduce the memory footprint because it does not store the intermediate samples such as it would be necessary for denoising of RDMs. Also, the proposed network uses a one-dimensional temporal input compared to the multidimensional input of CNN based methods, where the dimension correlates with the radar settings. Therefore, a change in the number of samples per ramp increases or decreases the input dimension and can affect the necessary number of convolutional layers and kernels. Our proposed system can be retrained to the specific scenario without input dimension modifications, which makes it better suitable for hardware implementations.

Another crucial factor is the training of NN methods. The fact that outliers appear sparsely and in various shapes, often unsupervised methods such as clustering [18] are utilized. The unsupervised methods reduce the controllability of the learned features, which can be an important aspect in safety-related applications. Thus, we use semi-supervised training for our proposed method, where we train only the normality and can distinguish between deviations from the expected values. This reduces the training effort and the need for extensive datasets containing sparse outliers.

Additionally, the proposed method comprises a self-monitoring potential based on the number of detected outliers per chirp. The reason for many outliers within a chirp is twofold: (1) An incorrect prediction of the LSTM or (2) the signal saturates due to strong target reflections or interference. The differentiation between these two cases is challenging, and further analysis provides the possibility for avoidance or mitigation by adjusting the radar settings or retraining the predictor. An opportunity for differentiation would be to observe whether interference occurs outside the valid samples. In these situations, the zeroing mitigation cannot improve the signal because crucial signal information are removed.

Besides, we analyze the extension of the detector with zeroing mitigation because of its low hardware complexity compared to other methods like extrapolation. As shown by [11], we expect better performance with other methods with higher computational complexity.

VII. Conclusion and Future Scope

Detecting outliers has a wide field of application, from fraud detection to faulty sensors. We use a semi-supervised LSTM approach to score the deviation from normality for radar-to-radar interference detection. The network achieves an F1-score of 90% and improves the noise floor by up to 31 dB with zeroing mitigation. We highlight the trade-off between low complexity mitigation and reduced performance. We also show a monitoring capability of the network and the sensor by observing the number of outliers per chirp. This method expands the detectability by observing the complete IF signal time series, which provides more information for dynamic interference avoidance. The proposed method uses the temporal stream to reduce hardware complexity, utilizing the memory capability of an LSTM network. We realize that a benchmark dataset from real radar measurements would increase the comparability of the algorithms. Therefore, we suggest the creation of a dataset with and without interference in real-world radar applications based on the temporal signal for high flexibility in the input signal selection.

References

[1] Abdul Sajeed Mohammed, Ali Amamou, Follivi Kloutse Ayevide, Sousso Kelouwani, Kodjo Agbossou, and Nadjet Zioui. The Perception System of Intelligent Ground Vehicles in All Weather Conditions: A Systematic Literature Review. *Sensors*, 20:34, 2020.

[2] Graham M. Brooker. Mutual Interference of Millimeter-Wave Radar Systems. *IEEE Transactions on Electromagnetic Compatibility*, 49(1):170–181, 2007.

[3] Sima Siami-Namini, Neda Tavakoli, and Akbar Siami Namin. A Comparison of ARIMA and LSTM in Forecasting Time Series. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1394–1401, Orlando, FL, 2018. IEEE.

[4] Hermann Winner, Stephan Hakuli, Felix Lotz, and Christina Singer. *Handbook of driver assistance systems*. Springer International Publishing Amsterdam, The Netherlands, 2014.

[5] M.A. Richards. *Fundamentals of Radar Signal Processing, Second Edition*. McGraw-Hill Education, 2014.

[6] Mate Toth, Paul Meissner, Alexander Melzer, and Klaus Witrisal. Analytical Investigation of Non-Coherent Mutual FMCW Radar Interference. In *2018 15th European Radar Conference (EuRAD)*, pages 71–74, Madrid, September 2018. IEEE.

[7] Markus Goppelt, Hans-Ludwig Blöcher, and Wolfgang Menzel. Analytical investigation of mutual interference between automotive FMCW radar sensors. In *2011 German Microwave Conference*, pages 1–4. IEEE, 2011.

[8] Francesco Laghezza, Feike Jansen, and Jeroen Overdevest. Enhanced interference detection method in automotive fmcw radar systems. In *2019 20th International Radar Symposium (IRS)*, pages 1–7. IEEE, 2019.

[9] B.-E. Tullsson. Topics in FMCW radar disturbance suppression. In *Radar 97 (Conf. Publ. No. 449)*, pages 1–5, 1997.

[10] Youn-Sik Son, Hyuk-Kee Sung, and Seo Heo. Automotive Frequency Modulated Continuous Wave Radar Interference Reduction Using Per-Vehicle Chirp Sequences. *Sensors*, 18(9):2831, 2018.

[11] Sharef Neemat, Oleg Krasnov, and Alexander Yarovoy. An Interference Mitigation Technique for FMCW Radar Using Beat-Frequencies Interpolation in the STFT Domain. *IEEE Transactions on Microwave Theory and Techniques*, 67(3):1207–1220, 2019.

[12] Nicolae-Cătălin Ristea, Andrei Anghel, and Radu Tudor Ionescu. Fully convolutional neural networks for automotive radar interference mitigation. In *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*, pages 1–5. IEEE, 2020.

[13] Johanna Rock, Mate Toth, Paul Meissner, and Franz Pernkopf. Deep interference mitigation and denoising of real-world fmcw radar signals. In *2020 IEEE International Radar Conference (RADAR)*, pages 624–629. IEEE, 2020.

[14] Johanna Rock, Mate Toth, Elmar Messner, Paul Meissner, and Franz Pernkopf. Complex signal denoising and interference mitigation for automotive radar using convolutional neural networks. In *2019 22th International Conference on Information Fusion (FUSION)*, pages 1–8. IEEE, 2019.

[15] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[17] Jiawei Han, Micheline Kamber, and Jian Pei. 8 - classification: Basic concepts. In Jiawei Han, Micheline Kamber, and Jian Pei, editors, *Data Mining*, The Morgan Kaufmann Series in Data Management Systems, pages 327–391. Morgan Kaufmann, third edition, 2012.

[18] Guilherme O. Campos, Arthur Zimek, Jörg Sander, Ricardo J. G. B. Campello, Barbora Micenková, Erich Schubert, Ira Assent, and Michael E. Houle. On the evaluation of unsupervised outlier detection: Measures, datasets, and an empirical study. *Data Mining and Knowledge Discovery*, 30(4):891–927, 2016.