2022 29th International Conference on Systems, Signals and Image Processing (IWSSIP) | 978-1-6654-9578-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/IWSSIP55020.2022.9854472

Investigation for the Need of Traditional Data-Preprocessing when Applying Artificial Neural Networks to FMCW-Radar Data

Jakob Valtl
Infineon Technologies AG
Neubiberg, Germany
Technische Universität Braunschweig
Braunschweig, Germany
jakob.valtl@infineon.com

Javier Mendez
Infineon Technologies AG
Neubiberg, Germany
Javier.MendezGomez@infineon.com

Gianfranco Mauro
Infineon Technologies AG
Neubiberg, Germany
Gianfranco.Mauro@infineon.com

Antonio Cabrera
Infineon Technologies AG
Neubiberg, Germany
AntonioJavier.CabreraGutierrez@infineon.com

Vadim Issakov
Technische Universität Braunschweig
Braunschweig, Germany
V.Issakov@tu-braunschweig.de

***Abstract*—Robust functionality of autonomous driving vehicles relies on their ability to detect obstables and various scenarios on the road. This can be only achieved by applying robust, fast and efficient AI-based signal processing to radar data. In this work we present an empirical investigation on the question, whether one can apply artificial neural networks (ANNs) directly to frequency modulated continuous wave (FMCW) radar raw data. We show that preprocessing is not necessary if one has enough raw data. In our experiment we have data of 153 648 frames collected with a 60 GHz FMCW radar. We compare systematically the options of preprocessing the data using variational autoencoder, applying traditional preprocessing or omit data-preprocessing and apply ANN directly to raw data. We show that the last option results in 28% faster signal processing and highest accuracy. This is a promising result, since it enables edge computing and direct signal processing at the sensor level.**

***Index Terms*—Artificial neural networks, Data preprocessing, Radar applications.**

I. Introduction

In recent years, Artificial Neural Networks (ANN) have been used to extract relevant information from data. These networks have been proven to provide high accuracy results for classification tasks using multiple input data types, such as images or parameters [1]. At the same time, numerous ANN structures have been developed to adapt to specific tasks, such as image classification and target detection. As a result of this, ANN structures become more complex and integrate capabilities to further study data for deeper analysis.

This work is a result of a collaboration between the projects "KI-Flex" (project number 16ES1027), funded by the German Federal Ministry of Education and Research (BMBF) within the funding program Microelectronic from Germany innovation driver, and the project "TEACHING" (project number 871385) funded by the Horizon 2020 program.

In most cases, the input data requires to be preprocessed before feeding it into an ANN to filter irrelevant data, as well as transform the data into a more explicit format or to execute a first feature extraction [2]–[4]. The selection of the preprocessing technique depends on the input data type as well as the purpose of the application. In the case of frequency modulated continuous wave (FMCW) radar sensors, the raw data is often preprocessed to extract the Rangle-Doppler Map (RDM) or Rangle-Angle Map (RAM) [3]. These formats are often selected due to their easy feature representation in contrast with the raw radar data. However, recent works have proven it is possible to execute this preprocessing using an ANN [5]. As a result, the efficiency and accuracy of transforming radar data in contrast to the use of raw data when using ANN models is not obvious and has not been reported in the literature so far. Therefore, we will focus in this paper on studying the effect of preprocessing the radar data prior to applying it to an ANN. Consequently, it will be discussed under which conditions these preprocessing techniques may be relevant or when it is better to train an ANN model using the raw data assuming the ANN will be able to execute the data preprocessing at the same time it classifies it.

II. State of the art

Traditionally, radar data is preprocessed in order to extract high level features from the detected targets, such as range and speed. The most popular technique to extract these features is the generation of the RDM using a double Fourier transformation to study the different samples measured by each of the antennas of the radar sensor [3]. This data can be further preprocessed to extract the RAM from each of the detected clusters in the RDM using the beamforming algorithm [6] or MUSIC algorithm [7] among others.

However, the effect of different preprocessing techniques in the overall performance is not always clear. Qi Z. et al. [8] investigated, in the domain of computational fluid mechanics, the effect of different preprocessing techniques on the prediction accuracy of non-isothermal indoor airflow distributions. Multiple preprocessing techniques were applied to the initial data to compare the effect on the output result from the ANN. They found out that most of the possible preprocessing techniques (standardization, normalization, proportion and nondimensionalization among others) led to similar results. However, they concluded it was neccesary to preprocess the data since the results achieved in the manuscript with unprocessed data were not comparable to preprocessed data.

In the specific case of radar sensors, as the ones studied in this manuscript, researchers have worked on trying to execute the radar data preprocessing using ANNs to benefit from the acceleration immanent to ANNs when using emerging devices such as Tensor Processing Units (TPU) or high-power Graphic Processing Units (GPU). Stadelmayer T. et al. [5] followed this research line to develop a parametric Convolutional Neural Network (CNN) that is able to mimic the traditional preprocessing technique applied to raw radar data to extract the RDM. This CNN is based on a $2D\ sinc$ or a $2D\ wavelet\ filter\ kernel$ to extract features from the raw data. This technique was applied to human activity data that was later classified. As a result of this, they achieved a classification accuracy of 98.8% in comparison with a traditional CNN based on preprocessed data, which achieved a 93.2% accuracy. Our investigation determines the limits and characteristics of different preprocessing techniques.

III. Design of Comparison

We compare three different preprocessing approaches on a radar dataset recorded with a moving data collection platform whose task was to follow a preceding target as shown in Fig. 1. The only input for the autonomous vehicle is the information obtained by the radar sensor. Detecting the distance to the vehicle in front, which determines the speed, is much easier than determining the angle in which the vehicle is located, which correlates with the steering position. Therefore, we will focus exclusively on angle determination in this study.

A. Dataset Description

Valtl J. et al. [9] recorded multiple datasets with a FMCW radar on various scenarios with a recording setup as shown in Fig. 1. Using these datasets we investigate the need of preprocessing. The dataset consists of 153 648 frames in total and contains recordings of different days and times and environment conditions. Each frame includes the raw radar data from two horizontally aligned antennas and a label. The radar operation parameters are chosen to be 64 chirps per frame and 128 samples per chirp. Each sample value coming from the ADC is within the boundaries of [0, 1]. The label of the frame is the current steering, which is predicted by the NN. The steering correlates with the position of the prior vehicle, but is not directly the angle of arrival under which it is detected. The steering value for each frame is within the boundaries of [-1, 1] representing left and right steering.

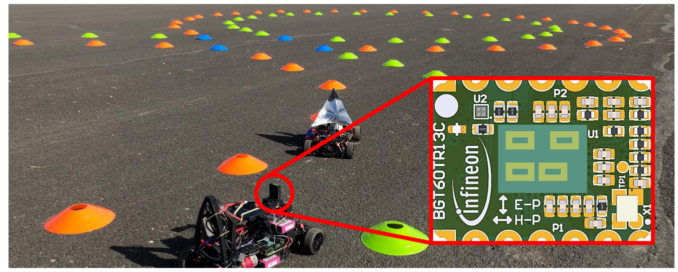

Fig. 1. Data recording setup with used 60 GHz FMCW radar.

For all tests 15 115 consecutive frames of one single take are used.

The remaining 138 533 frames, which will be referred to as the A_{set} or a subset of it, are split in a training and validation dataset. Where the split into training and validation is always performed with the ratio 3:1.

B. Three Different Networks with Different Stages of Preprocessing

The different networks are supposed to handle the same tasks, where one network is given the data following the traditional data processing pipeline, another approach uses a variational autoencoder (VAR) to replace the preprocessing and the third approach processes the data with no preprocessing.

All networks are trained with the same set of hyperparameters, that can be found in the following table, except for the loss function of the VAR during its training with the decoder.

TABLE I
Hyperparameters

hyper parameter	**value**
optimizer	adam
loss function	mean absolute error
learning rate	$1 \cdot 10^{-4}$
decay	$5 \cdot 10^{-7}$
early stopping	true
minimum delta	10^{-10}
patience	5
batch size	512
epochs	300

Especially the hyper parameters that lead to a termination of the training process were chosen in such a way that there is no substantial improvement after about 30 epochs, so that early stopping, after a patience of 5 epochs, with less improvement than the minimum delta, is triggered. The exact values were determined experimentally within the course of several test runs.

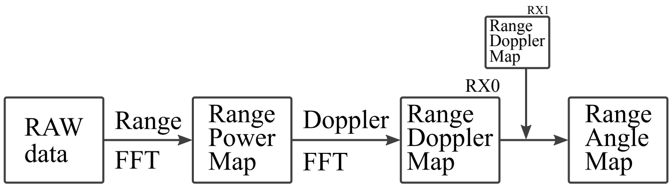

Fig. 2. Traditional preprocessing pipeline.

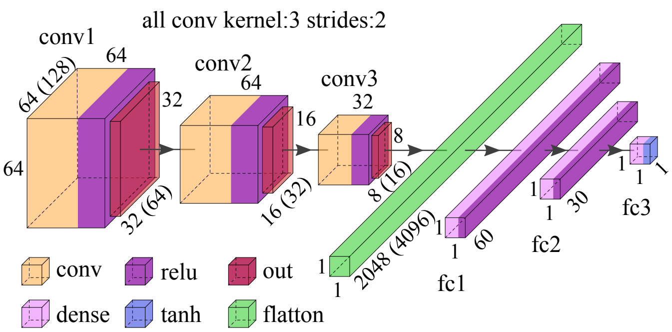

Fig. 3. Architechture of network with RAM or RAW as input.

1) Traditional RDM and RAM as Input: With this traditional approach all raw data gets passed through the preprocessing pipeline shown in Fig. 2. The RDM for each antenna is achieved by two Fast Fourier Transformations (FFTs). From the RDMs of multiple antennas the RAM is computed. Its output is the input of the ANN, whose architecture is shown in Fig. 3.

2) VAR to Overcome RDM and RAM Computation: At first a VAR is trained on the entire A_{set} where the latent dimension is chosen to be 30. The architecture of the encoder is shown in Fig. 4. The VAR learns to reconstruct the raw data that it is given as an input. Training with the entire A_{set} is legitimate as the encoding part of the VAR will act as a replacement of the traditional RDM and RAM computation and its training is independent of the later prediction with the regression block.

The setup and selection of hyperparameters for the VAR are given and explained in detail in [10].

Once the VAR is trained to reconstruct the raw data with the decoder, the weights of the encoder are frozen and the regression block is trained and used to predict the label. During training of the regression block, the entire A_{set} is not always used, but only a subset, as shown in Fig. 5, to investigate the performance of the regression block with limited amount of samples.

3) RAW as Input: The last approach is to train a NN that does not require any preprocessing on the data, but only takes the raw data as an input to predict the label. In fact we use the same architecture as the one for the RAM network and only rearrange the dimensions of the RAW data so that the number of antennas represents the dimension of the filter channel.

IV. Experimental Results

The following results are the average of 3 different runs, initializing different networks, with randomly selected splits of the datasets.

The prediction quality q of the networks describes their performance and is expressed through the following formula:

$$q = \frac{\sum_n^N |p_n - y_n|}{N}, \tag{1}$$

where N is the amount of frames considered for the evaluation. The distance of the output prediction p_n of the network for each frame n and the corresponding correct label y_n are averaged over the entire evaluation dataset, which for all tests consists of the same data, as described in III-A.

Fig. 5 depicts the networks performance dependency on the amount of samples that were used during the training.

The following table shows the network size and the computation time to evaluate the test dataset. The evaluation time of the traditional approach includes the computation of the RDM and the RAM. Both the network size and the runtime of the networks, the VAR-encoder + regression block on the one hand and the RAW-direct-input on the other hand are similar since the network structure is almost the same, except that in the VAR-endocer the preprocessing is learned a priori, which makes it perform especially well already with a small number of available samples.

V. Conclusion

The results in Fig. 5 clearly show in which scenario which topology is most suitable, depending on the amount of available training data. In case of extremely few training data being available a convolutional autoencoder gives the best results though due to its bottleneck at the latent space there seems

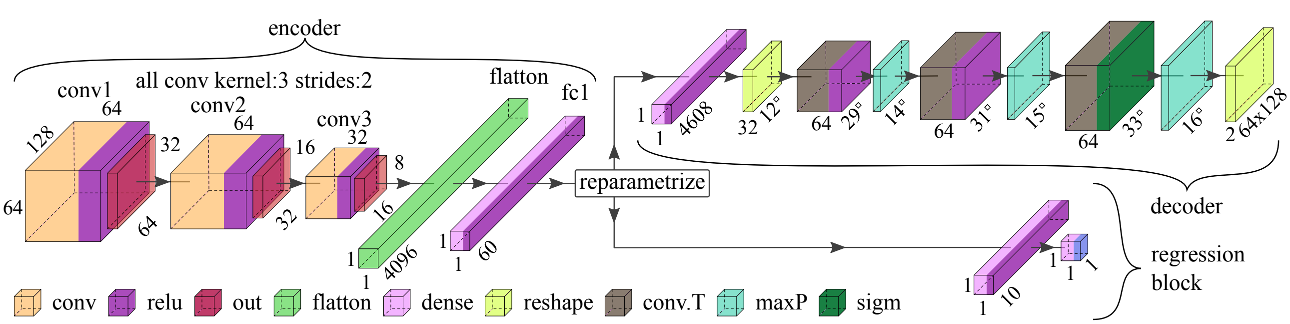

Fig. 4. Architecture of the VAR with the decoder to reconstruct the raw data and the regression block to predict the label.

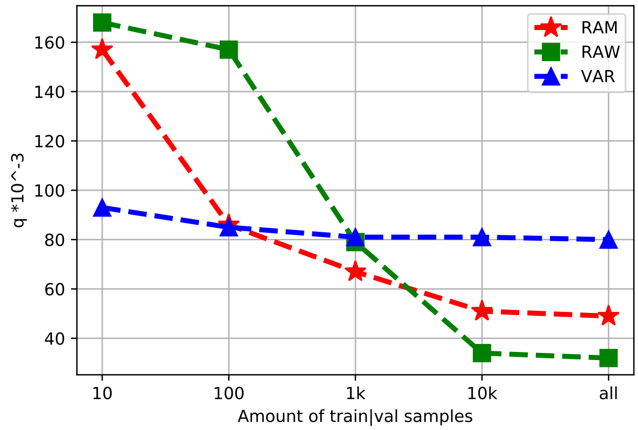

Fig. 5. Dependency of the prediction quality q and the amount of available training and validation samples.

TABLE II
NETWORK SIZE AND EVALUATION TIME OF THE PROPOSED METHODS.

<table>
<tr><th colspan="2">Network architecture / its input</th><th>NN size in MB</th><th>evaluation time in seconds</th></tr>
<tr><td colspan="2">(RDM & RAM) $\rightarrow$ NN</td><td>0.75</td><td>42.0</td></tr>
<tr><td colspan="2">RAW</td><td>1.243</td><td>30.4</td></tr>
<tr><td rowspan="2">VAR + encoder</td><td>decoder</td><td>1.2 + 1.8</td><td>—</td></tr>
<tr><td>regression block</td><td>1.2 + 0.013</td><td>30.3</td></tr>
</table>

to be a glass ceiling that can not be overcome independent of the amount of available training data. Whereas the traditional approach with the computation of the RDM and RAM leads to better results, if more training data is available. In case of the availability of a very large amount of training data, the approach to directly forward it to the NN without any sort of preprocessing leads to the best results in terms of error and execution time.

In terms of speed the methods that are purely ANN based are 28% faster than the traditional data processing pipeline which further emphasises the advantages of pure NN based algorithms, especially as an increase of performance has to be expected for TPUs, thus accelerating ANN approaches even more.

Regarding the size of the networks, models that work on preprocessed data are smaller due to the dimension loss during the preprocessing. The encoder of the VAR on the other hand is rather large as its supposed to extract the features of the preprocessing.

A limitation of this investigation is its restriction of only considering a single dataset, which will be tackled in a follow-up paper.

REFERENCES

[1] W. Wang, Y. Yang, X. Wang, W. Wang, and J. Li, "Development of convolutional neural network and its application in image classification: a survey," *Optical Engineering*, vol. 58, no. 4, p. 040901, 2019.

[2] Z. Zhao, Y. Cheng, X. Shi, and X. Qin, "Classification method of lidar point cloud based on threedimensional convolutional neural network," in *Journal of Physics: Conference Series*, vol. 1168, no. 6. IOP Publishing, 2019, p. 062013.

[3] J. Mendez, S. Schoenfeldt, X. Tang, J. Valtl, M. Cuellar, and D. P. Morales, "Automatic label creation framework for fmcw radar images using camera data," *IEEE Access*, 2021.

[4] A. Safa, F. Corradi, L. Keuninckx, I. Ocket, A. Bourdoux, F. Catthoor, and G. G. Gielen, "Improving the accuracy of spiking neural networks for radar gesture recognition through preprocessing," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[5] T. Stadelmayer, A. Santra, R. Weigel, and F. Lurz, "Data-driven radar processing using a parametric convolutional neural network for human activity classification," *IEEE Sensors Journal*, vol. 21, no. 17, pp. 19 529–19 540, 2021.

[6] J. Qian, Z. He, W. Zhang, Y. Huang, N. Fu, and J. Chambers, "Robust adaptive beamforming for multiple-input multiple-output radar with spatial filtering techniques," *Signal Processing*, vol. 143, pp. 152–160, 2018.

[7] S.-H. Jeong, Y.-S. Won, and D. Shin, "Fast doa estimation method based on music algorithm combined newton method for fmcw radar," in *2019 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*, 2019, pp. 1–5.

[8] Q. Zhou and R. Ooka, "Influence of data preprocessing on neural network performance for reproducing cfd simulations of non-isothermal indoor airflow distribution," *Energy and Buildings*, vol. 230, p. 110525, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378778820317783

[9] J. Valtl and V. Issakov, "Frequency modulated continuous wave radar-based navigation algorithm using artificial neural network for autonomous driving," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 567–571.

[10] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019. [Online]. Available: http://dx.doi.org/10.1561/2200000056