2020 25th International Conference on Pattern Recognition (ICPR)
Milan, Italy, Jan 10-15, 2021

Radar Image Reconstruction from Raw ADC Data using Parametric Variational Autoencoder with Domain Adaptation

Michael Stephan*†‡, Thomas Stadelmayer*†‡, Avik Santra†, Georg Fischer*, Robert Weigel*, Fabian Lurz*
†*Infineon Technologies AG*, Neubiberg, Germany
* Friedrich-Alexander-University Erlangen-Nuremberg, Erlangen, Germany
Email: {thomas.stadelmayer, avik.santra}@infineon.com
, {michael.stephan, georg.fischer, robert.weigel, fabian.lurz}@fau.de
‡equal contribution

***Abstract*—This paper presents a parametric variational autoencoder-based human target detection and localization framework working directly with the raw analog-to-digital converter data from the frequency modulated continuous wave radar. We propose a parametrically constrained variational autoencoder, with residual and skip connections, capable of generating the clustered and localized target detections on the range-angle image. Furthermore, to circumvent the problem of training the proposed neural network on all possible scenarios using real radar data, we propose domain adaptation strategies whereby we first train the neural network using ray tracing based model data and then adapt the network to work on real sensor data. This strategy ensures better generalization and scalability of the proposed neural network even though it is trained with limited radar data. We demonstrate the superior detection and localization performance of our proposed solution compared to the conventional signal processing pipeline and earlier state-of-art deep U-Net architecture with range-doppler images as inputs.**

***Index Terms*—Detection and Localization, Parametric Deep Neural Network, Variational Autoencoder, Domain Adaptation.**

I. Introduction

Human detection and localization is essential for smart home applications that leverage this information to automatically control lighting and heating, ventilation, and air conditioning (HVAC) systems to significantly reduce energy consumption in residential, commercial or mall settings. Several studies have shown that energy consumption can be significantly reduced by 25%-75% [1] by sensing the number of people and their respective locations. Furthermore, human detection and localization is essential for monitoring and maintaining social distancing guidelines, which is of high importance especially under the current corona crisis. There are several sensors that can enable reliable human detection and localization in indoor environments, including vision-based sensors. However, unlike vision-based sensors, radar-based human detection and localization offers a solution that is robust to lighting conditions and preserves privacy [2], [3], [4]. Furthermore, millimeter-wave (mm-wave) radars have small form factors, resulting from integrated silicon and antenna-in-package, that can be easily mounted and aesthetically integrated on the operating device. Radar-based human detection facilitates people counting and density estimation [5], [6], and is also prerequisite for human activity classification [7], [8] that can enable further intelligent control of appliances. Moreover, radar-based detection finds use in automotive in-cabin occupancy sensing [9]. Traditional, the frequency modulated continuous wave (FMCW) radar signal processing pipeline for human detection and localization involves translating the raw analog-to-digital converter (ADC) data along fast-time, i.e., time index within a chirp, and slow-time, i.e., time index across chirps, to range-Doppler domain by 1D FFTs along both axes one after the other. The fast-time axis transforms to range and slow-time to Doppler axis. Reflections from a single human target cause multiple detections on the range and Doppler dimension due to high bandwidth chirps used in mm-wave radars and micro-Doppler components arising from human body parts under motion respectively. Once the range-doppler image (RDI) is generated for all the virtual channels, they are fed into a beamforming algorithm to transform the channel data to explicit angle information. Depending on the algorithm and processing pipeline, a range-angle image (RAI) or range-angle-Doppler 3D data cube is generated. The RAI is then fed into a constant false alarm rate (CFAR) detector to ensure target detection under the Neymann-Pearson criteria. This is followed by clustering to ensure that multiple detections from a single target are grouped together as a single target. Additionally, spurious reflections are rejected as outliers. Human target detection and localization in indoor environments using conventional signal processing is particularly challenging since they lead to ghost targets on the RAI due to multi-path reflections on the human body from walls, chairs, and furniture. The complex relation between range, Doppler, target radar cross section (RCS) and interactions among targets can lead to missed detections on one hand, such that strong reflections from a close-by human can almost occlude the reflection from a farther human target, and spurious targets on the other hand, as reflections from a single human can manifest as several split targets [10]. Further, several hyper-parameters in signal processing algorithms, such as the scaling factor, the guard - and training-window size in CFAR or the

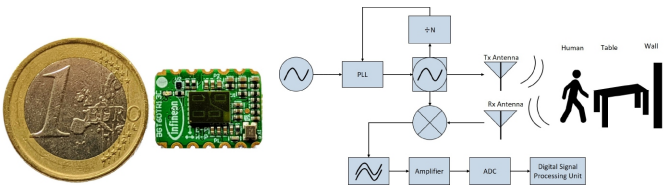
2020 25th International Conference on Pattern Recognition (ICPR) | 978-1-7281-8808-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICPR48806.2021.9412858

minimum number of clustering points and distance separation in density-based spatial clustering of applications with noise (DBSCAN), result in inaccurate target detections in the form of ghost targets or missed targets [11]. Recently, several papers [12], [13], [14], [15], [16] have been proposed that aim to replace the traditional detection and clustering algorithms using deep autoencoder architectures and have demonstrated much better detection performances compared to traditional signal processing approaches. In [10], a deep U-Net with residual connections has been proposed to perform detection and clustering on the RDI, whereas in [11] a complex deep U-Net with residual connections has been proposed to transform RDIs from different channels to a RAI and perform detection and clustering on that domain. In [12], a deep autoencoder has been proposed for interference mitigation and amplitude-phase reconstruction in RAIs for outdoor radars. In [15], a modified deep U-Net with long short-term memory (LSTM) cells has been presented for automotive target detections in polar range-angle domain. In [14], an autoencoder architecture has been suggested for transforming RDIs from multiple channels to RAIs for detecting automotive targets. However, these works use RDIs across channels as input for their proposed neural network, which may not be effective in fully exploiting the potentials of deep neural networks that are, in principle, capable of extracting feature images implicitly. Moreover, 1D FFTs along fast-time followed by 1D FFTs along slow-time to generate RDIs are the computationally most intensive step of the conventional radar signal processing. Furthermore, such hybrid signal processing and deep learning designs are not efficient for embedded solutions as the FFTs need to be performed by the microcontroller or general processor before passing the RDIs into a deep learning accelerator. To address these issues, in this paper we propose a parametrically constrained autoencoder, where the first layer is constrained to parametric functions that enable filtering operation along fast-time - slow-time simultaneously and resembles the (range and Doppler) frequency separability property of FFTs. SincNet-based parametric 1D CNNs have first been proposed for audio signal processing [17]. Similar parametric 2D deep convolutional neural networks (DCNNs) for radar-based human activity classification have been proposed in [18]. For scalability and generalization of the proposed deep neural network, a large training dataset, where data is collected under several different conditions and configurations, is required. Nevertheless, training the network with a dataset comprising such varied scenarios and configuration is practically implausible. To overcome such challenges, we, in this paper, propose a domain adaptation (DA) strategy to first train the proposed architecture using synthetic data generated through ray tracing models under numerous configurations followed by network adaptation to real sensor data, which is generally limited.

II. Radar System Design

A. Radar Chipset

In the paper *Infineon*'s *BGT60TR13C* FMCW radar chipset is used. An image of the chipset and the analog



(a) Photo of the radar chipset (b) Typical FMCW block diagram

Fig. 1. (a) *Infineon*'s *BGT60TR13C* 60-GHz radar sensor. (b) Functional block diagram of FMCW radar RF signal chain depicting 1TX, 1RX channel

radio frequency (RF) signal chain for one transmit and one receiving antenna is depicted in Fig. 1. In the transmit part, the phase-locked loop (PLL), with a reference frequency of $80\,\mathrm{MHz}$, regulates the voltage-controlled oscillator (VCO). The radar is able to generate highly linear chirps covering a frequency range between $57\,\mathrm{GHz}$ and $64\,\mathrm{GHz}$ within a configurable chirp duration by adjusting the divider value and an additional tuning voltage ranging from $1\,\mathrm{V}$ to $4.5\,\mathrm{V}$. The backscattered signal is received by the receiving antenna, then mixed with a replica of the transmitted signal and afterwards low pass filtered in order to obtain the intermediate or beat frequency signal. Further, the signal is sampled with a sampling frequency of $2\,\mathrm{MHz}$ by the $12\,\mathrm{bit}$ ADC. Since the radar has three receiving antennas there are three such receive paths. The antennas are ordered in an L-shape, and therefore two antennas each can be used for azimuth and elevation angle estimation. In this paper, only the two receiving antennas that are necessary to estimate the azimuth angle are used. The relative offset of the used antennas is $d = 2.5\,\mathrm{mm}$, and therefore half the wavelength of the transmitted signal. Moreover, the sensor covers a field of view of $70°$ in elevation and $120°$ in azimuth angle dimension.

B. Signal Model and System Configuration

The mixed and low pass filtered signal of a single chirp assuming a backscatterer at distance d_n is defined as

$$S_{\mathrm{n,ft}}(d_\mathrm{n}, t_\mathrm{ft}) = A\cos(2\pi(\frac{2Bd_\mathrm{n}}{cT}t_\mathrm{ft} + \frac{2f_0 d_\mathrm{n}}{c} + \frac{B}{2T}(\frac{2d_\mathrm{n}}{c})^2)) \approx A\cos(2\pi(\frac{2Bd_\mathrm{n}}{cT}t_\mathrm{ft} + \frac{2f_0 d_\mathrm{n}}{c})) \quad (1)$$

where A is the voltage amplitude, B is the chirp bandwidth, d_n is the radial distance of target n to the radar, f_0 is the center frequency, c is the speed of light, T is the chirp time and t_ft is the fast-time. Due to low velocities in indoor environments, the scene within a chirp is quasi-static. Moreover, the distances are very small for indoor applications, and therefore the quadratic term can be neglected. However, when observing the signal over multiple chirps, i.e. slow-time, the displacement of a moving target can be detected. The received, mixed and low pass filtered signal over slow-time is defined as

$$S_\mathrm{n}(d_\mathrm{n}, t_\mathrm{ft}, t_\mathrm{st}) = A\cos(2\pi(\frac{2B(d_\mathrm{n} + v_\mathrm{n}t_\mathrm{st})}{cT}t_\mathrm{ft} + \frac{2f_0(d_\mathrm{n} + v_\mathrm{n}t_\mathrm{st})}{c})) \quad (2)$$

where the parameters represent the same physical sizes as before and additionally t_{st} is the slow-time axis and v_{n} is the velocity of target n. Relative to the absolute distance d_{n} the movement $\delta d = v_{\mathrm{n}} t_{\mathrm{st}}$ is negligible small. However, since the phase is periodical to the wavelength $\lambda = c/(2f_0)$, which is about $5\,\mathrm{mm}$ for a center frequency of $f_0 = 60\,\mathrm{GHz}$, small displacements still induce a noticeable phase shift. As a result (2) can be approximated as

$$S_{\mathrm{n}}(d_{\mathrm{n}}, t_{\mathrm{ft}}, t_{\mathrm{st}}) \approx A\cos(2\pi(\frac{2Bd_{\mathrm{n}}}{cT}t_{\mathrm{ft}} + \frac{2f_0(d_{\mathrm{n}} + v_{\mathrm{n}}t_{\mathrm{st}})}{c})). \quad (3)$$

Moreover the signal received by an antenna is the superposition of the backscattered signal of multiple targets. The final signal representation of a single antenna used in this paper is therefore defined as

$$S = \sum_{\mathrm{n}\in\mathrm{N}} S_{\mathrm{n}}(d_{\mathrm{n}}, t_{\mathrm{ft}}, t_{\mathrm{st}}) \quad (4)$$

where N is a set of targets.

The signal over multiple receiving antennas differs by a phase shift defined as

$$\phi = \frac{2\pi d \sin(\theta)}{\lambda} \quad (5)$$

where d is the distance between the receiving antennas, θ is the relative azimuth angle of the target and λ is the wavelength. By analyzing the phase offset, the angle of arrival can be estimated. The radar chipset in the paper was configured to send out chirps starting from $f_{\mathrm{min}} = 58\,\mathrm{GHz}$ up to a maximum frequency of $f_{\mathrm{max}} = 62\,\mathrm{GHz}$ within a chirp time of $T_{\mathrm{c}} = 261\,\mu\mathrm{s}$. The received signal is sampled 256 times with a sampling frequency of $f_{\mathrm{s}} = 2\,\mathrm{MHz}$. The total chirp bandwidth $B = 4\,\mathrm{GHz}$ results in a range resolution of $\delta r = 3.75\,\mathrm{cm}$ and therefore a maximum range R_{max} of $4.8\,\mathrm{m}$ can be observed. Moreover a data frame consists of 32 chirps with a chirp repetition time T_{PRT} of $520\,\mu\mathrm{s}$. As a result a Doppler resolution of $\delta v_{\mathrm{max}} = 1.25\,\mathrm{m\,s^{-1}}$ is obtained.

C. Classical Processing Pipeline

For indoor target detection and localization, multiple processing steps have to be performed. First, a 2D FFT is applied on the raw ADC data. Then, a moving target indication (MTI) filter is applied to mitigate static targets. Afterwards, an imaging angle estimation method is applied and finally, a target detection followed by a clustering method has to be performed.

1) 2D FFT: As discussed, the range and Doppler information of one or more targets in the field of view is encoded in the frequency composition of the signal along the fast- and slow-time dimension, respectively. Hence, the signal is traditionally preprocessed using a two staged FFT. First, a 1D FFT is applied along the mean removed chirps. Afterwards, a second 1D FFT is applied on the FFT processed signal across a set of chirps. In both cases, a window function is applied. This processing results in a complex-valued RDI. To mitigate static targets and antenna leakage, MTI filtering is applied over multiple RDIs.

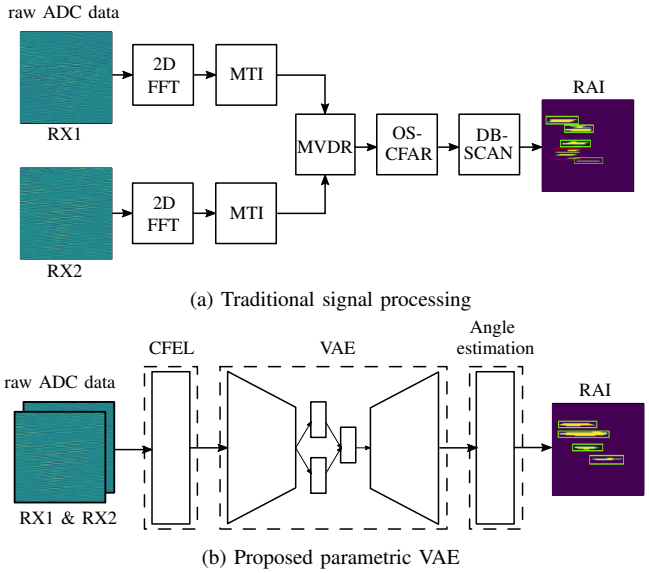


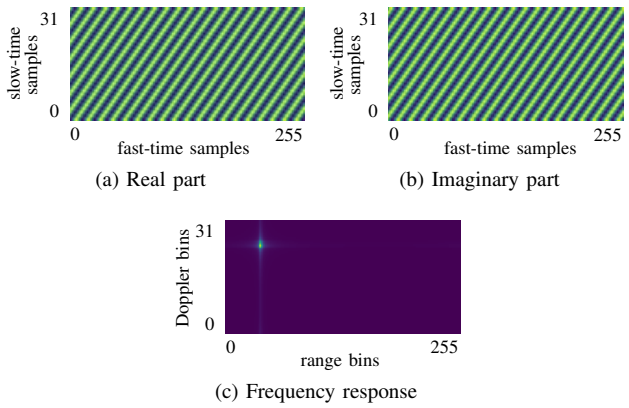
Fig. 2. (a) Traditional signal processing pipeline (b) proposed solution using a parametric VAE

2) Angle Estimation: After the signal was disassembled into different range-Doppler bins, an angle estimation can be performed. As stated in sec. II-B the angle of arrival is defined by the phase shift over multiple receiving antennas. However, due to noise and other disturbances evaluating only the phase difference leads to inaccurate results. There exist advanced angle estimation approaches such as the minimum variance distortionless response (MVDR) beamformer, the estimation of signal parameters via rotational invariance techniques (ESPRIT), and the multiple signal classifier (MUSIC) method. We use the MVDR method to obtain the RAI.

3) Object Detection: From the RAI, one or multiple targets have to be extracted. This can either be done by simple thresholding or by applying more advanced techniques like ordered statistic constant false alarm rate (OS-CFAR). Afterwards, the RAI is given in a binary manner, where zero stands for no target, and one stands for a detected target in a range angle bin. When a person is walking, the signal is not only reflected by the body, but also by the moving limbs which can generate a small isolated signal close to the main target. In order to detect this as a single target, we apply DBSCAN for clustering. Based on the resulting clusters, the location of one or more human targets can be extracted.

III. Proposed Solution

In this paper, we propose a deep learning based human target detection and localization solution that replaces the entire traditional signal processing chain as described in the previous chapter. The raw ADC data is fed into the neural network, enabling a more efficient workflow due to the fact that the data does not have to be preprocessed on a digital signal processor. Instead, the raw data can directly be transferred from the sensor to specialized accelerator hardware. Until now, almost all state of art deep learning based radar systems use



(a) Real part (b) Imaginary part

(c) Frequency response

Fig. 3. (a) Real part, (b) imaginary part and (c) frequency response of an exemplary CFEL filter kernel

preprocessed input data. It turns out that the system often gets stuck in a local minimum when training it on raw ADC data. However, we propose using a parametric layer at the beginning of the neural network for range and Doppler feature extraction. This is then followed by a variational autoencoder (VAE) architecture. Moreover, we propose training the neural network on synthetic radar data and then use DA in order to adapt the neural network to real world data.

A. Complex Frequency Extraction Layer

Processing the data using a 2D FFT directly unveils range and Doppler information but in turn is compute intense and the number of sampling points limits its accuracy. Instead, we propose using a complex frequency extraction layer (CFEL) for range and Doppler feature extraction. This is a parametric layer, which means that the filter kernels are given by a function defined by a finite set of parameters. In the proposed CFEL the filter kernels are defined as

$$f_{\mathrm{M,\,N}}(f_{\mathrm{ft}}, f_{\mathrm{st}}; m, n) = e^{j2\pi f_{\mathrm{ft}} m / f_{\mathrm{s}}^{\mathrm{ft}}} e^{j2\pi f_{\mathrm{st}} n / f_{\mathrm{s}}^{\mathrm{st}}} \tag{6}$$

where m and n are the sample indices, M and N the filter lengths and $f_{\mathrm{s}}^{\mathrm{ft}}$ and $f_{\mathrm{s}}^{\mathrm{st}}$ are the sampling frequencies in fast- and slow-time respectively. Moreover, f_{ft} and f_{st} are the learnable hyperparameters that define the filter kernels. These hyperparameters also define the frequencies that the filter kernel extracts from the signal. Additionally, to create a set of filter kernels the number of filters in fast- N_{ft}, as well as in slow-time N_{st}, has to be given. Although each filter is applied and trained independently, the output channels are reshaped to a two dimensional matrix of size $N_{\mathrm{ft}} \times N_{\mathrm{st}}$ to obtain a similar representation as a RDI. Learning the filter frequencies enables the possibility, unlike in an FFT, to analyze the signal composition in some frequency areas in more detail than in others. In order to enable equal training, the hyperparameters are normalized. When the filters are created using the set of harmonic frequencies, the output of the CFEL equals a 2D FFT. The real and imaginary part, as well as the frequency response of an exemplary filter kernel, is shown in Fig. 3. It can be seen that the frequency response is a single sharp peak. Thus, applying a set of these filter kernels can be seen as sampling the underlying signal in frequency domain at variable positions.

B. Variational Autoencoder

A general idea of encoder-decoder-based structures is to compress some input data into a smaller latent space so that the desired output data can still be obtained from this compressed representation. Ideally, the latent space should capture some high-level features, like the presence/absence of real targets and ghost targets, which are then used in the decoder to highlight targets and remove ghost targets. In practice, the data representation chosen by the encoder network may not be as interpretable, in case of strong overfitting, single input examples may even be mapped to some specific numbers without any high-level meaning. A way to achieve continuity in the latent space and reduce overfitting is to use a VAE instead. In a VAE, the input is not encoded into a single point in latent space, but into a distribution over the latent space. This is achieved, by sampling from normal distributions with mean vectors μ, and standard deviations σ, the values of which are learned by the network. The decoder then learns the mapping from these distributions to the desired outputs. During inference, only the values of μ are used, each input example now corresponds to a deterministic latent vector, where similar inputs should correspond to similar latent vectors. The vectors μ and σ are the outputs of two fully connected layers in the network. As the sampling operation itself is not differentiable, a reparametrization trick is used to move the sampling operation out of the backpropagation-path. Partly in order to avoid the network to set the learned variances to zero, and therefore sample from dirac distributions, the Kullback-Leibler (KL) divergence from the learned distributions defined by μ and σ and Gaussian distributions is added as a regularization term to the overall loss.

C. Domain Adaptation

As is often the case for radar data, the amount of labeled data to train our neural network on is rather limited. While augmentation techniques exist for radar data too, compared to image processing, they are much less powerful, and a lot of possible input scenarios will, therefore, not be covered by the training data. To still steer our network towards general detection capabilities, we use domain adaptation techniques inspired by soft-parameter sharing multi-task learning, as it was shown in [19], for dependency parsing in linguistics.

We first train a network with the same architecture on labeled synthetic point target data, covering the whole output range-angle space. We then initialize the network for the real target data with these pretrained weights and add a weight difference regularization term to our loss function, as shown in (7).

$$L_{DA} = \sum_{i \in T} \frac{||w_i - w_{i0}||^2 + ||b_i - b_{i0}||^2}{||w_{i0}||^2 + ||b_{i0}||^2} \tag{7}$$

Using (7), we slightly punish any divergence of the weights to the weights of the network trained on the synthetic dataset. Here, w_i and b_i are the weights and biases for the ith convolutional layer, T is the set of convolutional layers in our models, while w_{i0} and b_{i0} are the weights and biases for the same convolutional layers trained on the synthetic data. The idea behind this approach is to have the neural network learn a general angle of arrival estimation function on the synthetic data, which should also generally be needed to detect real targets in the range angle space in addition to the ghost target removal and other functions. In effect, this also reduces the experienced overfitting due to the weight regularization.

IV. Architecture & Learning

A. Architecture

Fig.4 shows the full proposed architecture for human target detection from raw ADC data. The neural network input is the two-channel raw ADC data from two receiving antennas of the radar chip. The first layer is the CFEL. The CFEL is initialized so that the center frequencies form a uniform grid over the range-Doppler space. The kernel size in fast-time direction is chosen to be 256 and 32 in slow-time dimension. The initial learnable frequencies in fast-time direction are defined by 128 equally spaced frequencies from 0 to $f_s^{ft}/2$, whereas the initial learnable frequencies in slow-time are given by 32 equally spaced frequencies ranging from 0 to f_s^{st}. In doing so, 128 different positive range bins and 32 Doppler bins, including positive and negative velocities, are created. The 2D kernels are then generated by combining each fast-time frequency with each slow-time frequency resulting in a total set of 4096 kernels. This setup equals a 2D FFT using 128x32 samples. Each kernel is then trained on its own, so its underlying frequencies are independent of other kernels, which allows a maximal flexible and efficient 2D frequency extraction. The same CFEL layer is applied independently to the raw ADC inputs from the two antennas. The output is then of size $128\times32\times2\times2$, where the third dimension describes the two antennas, and the last dimension the channel dimension, where the two channels are the real and the imaginary output part of the CFEL. The encoder part of the network consists of six blocks with the same structure. Each block consists of a $3 \times 3 \times 1$ convolution, followed by a 0.4 dropout and another $3 \times 3 \times 1$ convolution. The input to the block is then added to the convolution output, after undergoing a $1\times1\times1$ convolution to match the channel dimension. Lastly, the feature map size is reduced by a factor of two in both direction through a $3 \times 3 \times 1$ convolution with a stride of $2 \times 2 \times 1$. Between each encoder block, the number of channels is increased by a factor of 1.6, the number is rounded down to the nearest integer. Following the encoder are two fully connected layers, representing the mean-vector and the variance-vector, and a sampling layer to draw from the distribution defined by the fully connected layers. Each fully connected layer has 140 neurons, corresponding to the flattened input, except for the antenna dimension. The decoder part also consists of five blocks of the same structure and of similar structure to the encoder blocks. The input to each such block is the output from the previous block. First it is upsampled by a factor of two, followed by a $2 \times 2 \times 1$ convolution. The convolution output is then concatenated with the output of the add layer of the corresponding encoder block. This is again followed up by two $3 \times 3 \times 1$ convolutions with a 0.4 dropout layer in between. Up until the last add layer in the decoder block, the same network is basically applied separately to the two antenna channels. The information from the two antennas is only brought together in the $1 \times 1 \times 2$ convolution just after the last add layer. The last 1x1 convolution is done to reduce the output channel size to one.

B. Loss Function

The final loss function, as shown in (8), is a combination of the focal loss, the DA regularization term, and the KL divergence loss term.

$$loss = L_{FL} + \beta \cdot L_{DA} + \theta \cdot L_{KL} \tag{8}$$

The focal loss, as described in [20], is used due to the heavy class imbalance of targets and non-targets in our labeled data. Due to the varying number of targets and the varying target sizes, a constant class weighting is not sufficient. Using the focal loss, a higher emphasis is put on miss-classified output pixels during training. The equation for the focal loss is shown in (9).

$$\begin{aligned} L_{FL} &= \alpha_t \cdot (1-p_t)^{\gamma} \cdot \log(p_t) \\ p_t &= \begin{cases} p & \text{if } y=1 \\ 1-p & \text{otherwise} \end{cases} \qquad \alpha_t = \begin{cases} 1 & \text{if } y=1 \\ \alpha & \text{if } y=0 \end{cases} \end{aligned} \tag{9}$$

In (9), p_t describes the probability that a pixel was correctly classified. The parameter α controls the initial class weighting. If the value for γ is chosen to one, the loss is equal to the crossentropy. For higher values of γ, more weight is put on miss-classified examples. We chose $\gamma = 2$, and $\alpha = 0.25$ for our experiments. The other two components of the loss function as shown in (8) are the KL divergence term and the domain adaptation term as described in sections III-B and III-C. Both terms are multiplied with some factors to scale their total loss contribution, $\beta = 0.0001$ for the DA-loss, and $\theta = 0.1$ for the KL-loss. When training the network on the synthetic data, β is set to 0.

V. Results & Discussion

A. Dataset

For training our network with domain adaptation as presented earlier, we need two different datasets. One is the synthetic dataset for learning the general angle of arrival estimation, and the other is our real target dataset. To make it easier for the network to work with both these datasets, we normalize our synthetic and our real input data. Specifically, we map the lowest ADC input value to zero and the highest to one for each input image.

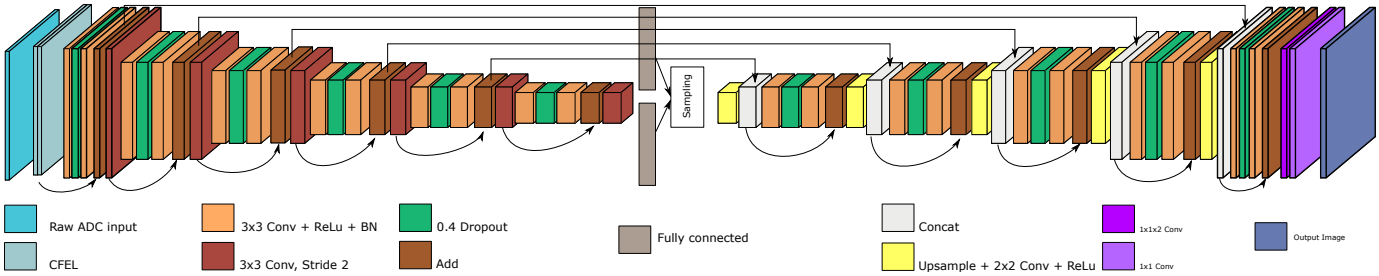


Fig. 4. Proposed parametric VAE architecture for human target detection from raw ADC data

1) Domain Adaptation: Using the phased array toolbox from Matlab, a radar with the same antenna positions and configurations as described in II-B was simulated. The space is discretized in 128 range positions from $3.75\,\mathrm{cm}$ up to $4.8\,\mathrm{m}$ and 32 azimuth angle positions ranging from -50° to 50°. On each point in the discrete space, a point target was placed, and the radar signal was simulated using ray tracing. The received signal is then dechirped and downsampled in order to obtain the IF signal before it is stored on the hard drive.

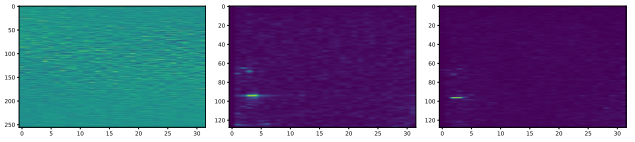
2) Target Domain: To gather our real target dataset, we recorded a person walking around in a typical conference room with a camera and a radar sensor. The labeled data was then created using the traditional signal processing chain and by crosschecking its outputs with the camera data and manually removing any ghost targets or adding missed detections to the labels. The described recording scene is shown in Fig.V-A2. Like this, 1200 labeled examples were created, 1000 for the

Fig. 5. Recording environment

training/validation set, and 200 for the test set. These one target measurements are then taken as a base to create multi-target measurements, up to four targets, by superposition of different one target measurements. To reduce overfitting, it is necessary to employ some augmentation on the one target measurements. We slightly shift the measurements and labels in terms of range via multiplication of complex exponentials $\exp^{j2\pi ft}$, where f describes the applied frequency shift, across the fast-time dimension of our ADC data. In total, we then have 15000 training/validation examples and 1000 examples for testing.

B. Reconstruction Results

It is proposed to feed raw time-domain data into the neural network and extract meaningful features using a parametric



(a) Time domain data (b) RDI (c) Features after CFEL

Fig. 6. Time domain data and corresponding output of the CFEL layer

CFEL. As a result, the CFEL replaces the preprocessing, which typically involves MTI filtering and a 2D FFT. As a comparison, the time domain signal, the absolute values of the preprocessed RDI, and the absolute values of the output of the already trained CFEL of the same scene are shown in Fig. 6. The x-axis goes along slow-time (a) or Doppler (b) (c) and the y-axis along fast-time (a) or range (b) (c) direction. It can be seen that it is hardly possible to obtain information from the time domain signal itself, but after preprocessing, the target gets unveiled as a peak in the range-Doppler domain. After initialization, the features after the CFEL layer look the same as the RDI, but during training the frequencies that are analyzed or extracted are optimized. Thus, if the application requires, it is possible to get a higher frequency resolution in more meaningful frequency areas and less resolution in less meaningful areas. This is an advantage, especially in classification tasks. However, if all frequency regions are of equal interest for the application, the analyzed frequencies are distributed over the whole domain. In Fig. 6 it can be seen that in the CFEL, the target information is successfully extracted from the time domain signal. Both the preprocessed RDI as well as the output of the CFEL show consistent results.

C. Detection Results

We evaluate our proposed approach on a set of 800 test measurements, consisting of one to four moving targets, with 200 examples per target number. The 200 one-target measurements are real measurements that are not within the training/validation set. The 2-4 target measurements were created from these one-target measurements as described in section V-A. We evaluate our model in terms of F1-score by looking at the number of missed detections and false alarms in each output image. We do this by comparing the center of masses of the clusters in the neural network output to the corresponding labels. We do a minimum bipartite matching

in terms of euclidean distance between the clusters and count a target as detected only if the cluster in the network output is within a certain range of the cluster in the labeled image, specifically 37.5 cm.

TABLE I
COMPARISON OF THE DETECTION PERFORMANCE OF THE TRADITIONAL PIPELINE WITH THE AE ARCHITECTURE FROM [11], AND THE PROPOSED METHOD

Approach	Description	F1-Score	Model Size
Traditional	OS-CFAR with DBSCAN	0.61	-
AE	AE with complex RDI as input [11]	0.77	1.23 MB
VAE	VAE with CFEL layer and DA	**0.80**	1.71 MB

Table I shows the results of the traditional signal processing approach, the method from our old paper, and for our new proposal in terms of F1-score on the described test-set. Both deep learning based methods show clear improvement over the traditional chain for the RAI detection task. The new method shows the best results on the test-set, mostly due to changes made in the network architecture. Some examples illustrating the model performance compared to the traditional processing chain are shown in Fig. 8, 9. Here, the (b) are the labels, (c) shows the output of the traditional processing chain, and (d) the output of the proposed approach. All correct detections are marked with a green box, while missed detections or false alarms have a red box instead. In Fig. 8, the proposed approach shows all targets detected at the correct positions, while the traditional processing chain has one missed detection and one false alarm due to a target split. In Fig. 9, there is a ghost target and a missed detection in (c), while two targets were detected as one with the neural network due to just using a simple clustering to avoid any additional processing overhead.

D. Discussion

The proposed method works directly with the ADC data as input, making any additional preprocessing blocks unnecessary, therefore reducing the memory and compute requirements from the preprocessing, which is not factored in the model size in table I. Furthermore, it is more robust to new input data due to the VAE structure and the DA regularization, yet still shows small performance improvements over [11]. Fig. 7 visualizes the effects of adding the DA regularization term. It shows the deviations of the model weights to the model weights of the network trained on the synthetic data over the training epochs. In all cases, the convolutional layers are initialized with the weights from the model trained on the synthetic data. Retraining the network without constraints quickly results in large deviations from the previous model weights. Adding a higher weight regularization factor keeps the model weights closer to the initial weights, therefore ensuring that the model keeps close to the learned general angle estimation without more than a slight drop in performance.

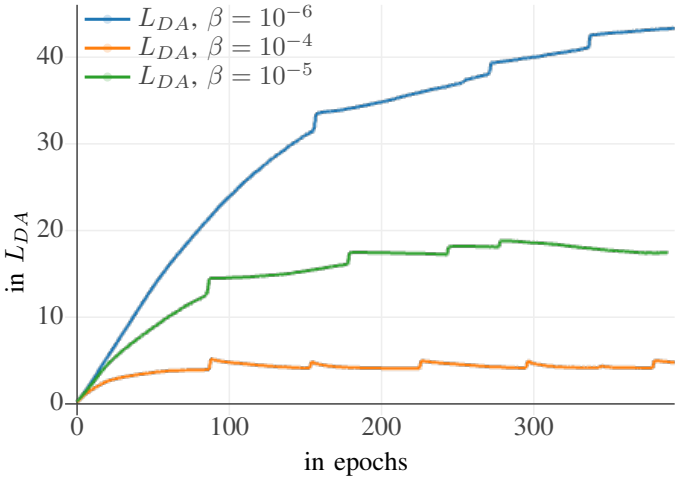
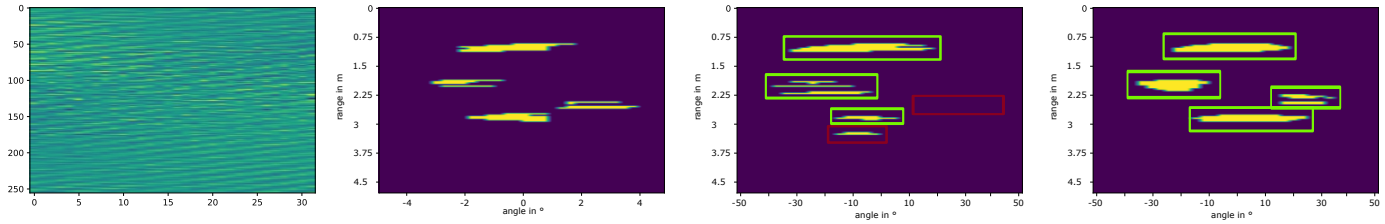


Fig. 7. Weight divergence from the synthetic model

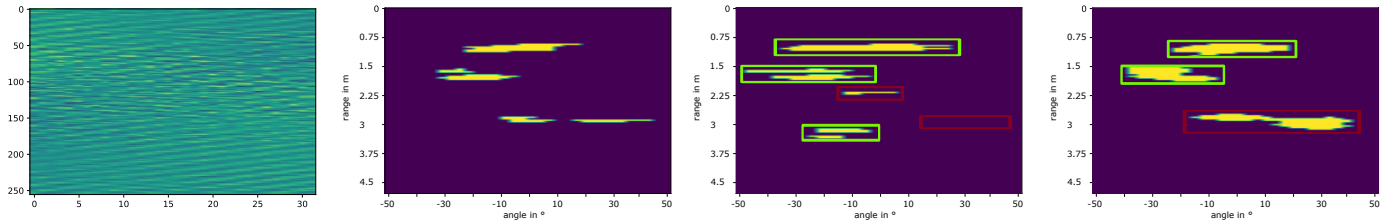
VI. CONCLUSION

In this paper, we propose a novel variational autoencoder, whose first layer is drawn from a family of parametric functions that constrain the convolution layers to perform filtering along fast-time and slow-time and separation of different range-Doppler frequencies into distinct kernels similar to conventional FFTs. To overcome the problem of limited radar data, we propose supplementing training through domain adaptation of the network from ray-tracing mathematical model-based synthetic data generated under diverse configurations. We demonstrate the superior performance of the proposed solution compared to conventional radar signal processing and state-of-art neural network solutions for human detection and localization in indoor environments. Apart from superior detection performance, the proposed solution reduces computation requirements drastically by replacing FFTs that consume most operations and simplifying the data flow in the embedded realization by requiring only deep learning accelerators to process raw ADC data directly.



(a) ADC data after MTI (b) Labels (c) traditional processing chain (d) proposed approach

Fig. 8. (a) ADC data after MTI, (b) Labels indicating the true positions, (c) Processed RAI using the traditional chain , (d) Processed RAI using proposed approach wherein all targets are detected accurately.



(a) ADC data after MTI (b) Labels (c) traditional processing chain (d) proposed approach

Fig. 9. (a) ADC data after MTI, (b) Labels indicating the true positions, (c) Processed RAI using the traditional chain , (d) Processed RAI using proposed approach wherein all targets are detected accurately.

REFERENCES

[1] V. Garg and N. K. Bansal, "Smart occupancy sensors to reduce energy consumption," *Energy and Buildings*, vol. 32, no. 1, pp. 81–87, 2000.

[2] Z. Peng, J. M. Muñoz-Ferreras, Y. Tang, C. Liu, R. Gómez-García, L. Ran, and C. Li, "A portable fmcw interferometry radar with programmable low-if architecture for localization, isar imaging, and vital sign tracking," *IEEE transactions on microwave theory and techniques*, vol. 65, no. 4, pp. 1334–1344, 2016.

[3] G. Wang, C. Gu, T. Inoue, and C. Li, "A hybrid fmcw-interferometry radar for indoor precise positioning and versatile life activity monitoring," *IEEE Transactions on Microwave Theory and Techniques*, vol. 62, no. 11, pp. 2812–2822, 2014.

[4] A. Santra, R. V. Ulaganathan, and T. Finke, "Short-range millimetric-wave radar system for occupancy sensing application," *IEEE sensors letters*, vol. 2, no. 3, pp. 1–4, 2018.

[5] A. Santra and H. Souvik, *Deep Learning Applications of Short-Range Radars*. Artech House Books, 2020.

[6] J. W. Choi, D. H. Yim, and S. H. Cho, "People counting based on an ir-uwb radar sensor," *IEEE Sensors Journal*, vol. 17, no. 17, pp. 5717–5727, 2017.

[7] Y. Lin, J. Le Kernec, S. Yang, F. Fioranelli, O. Romain, and Z. Zhao, "Human activity classification with radar: Optimization and noise robustness with iterative convolutional neural networks followed with random forests," *IEEE Sensors Journal*, vol. 18, no. 23, pp. 9669–9681, 2018.

[8] P. Vaishnav and A. Santra, "Continuous human activity classification with unscented kalman filter tracking using fmcw radar," *IEEE Sensors Letters*, vol. 4, no. 5, pp. 1–4, 2020.

[9] M. Alizadeh, H. Abedi, and G. Shaker, "Low-cost low-power in-vehicle occupant detection with mm-wave fmcw radar," *arXiv preprint arXiv:1908.04417*, 2019.

[10] M. Stephan and A. Santra, "Radar-based human target detection using deep residual u-net for smart home applications," in *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*. IEEE, 2019, pp. 175–182.

[11] M. Stephan, A. Santra, and G. Fischer, *Human Target Detection and Localization with Radars using Deep Learning*. Springer, 2020.

[12] J. Fuchs, A. Dubey, M. Lübke, R. Weigel, and F. Lurz, "Automotive radar interference mitigation using a convolutional autoencoder," in *2020 IEEE International Radar Conference (RADAR)*. IEEE, 2020, pp. 315–320.

[13] L. Wang, J. Tang, and Q. Liao, "A study on radar target detection based on deep neural networks," *IEEE Sensors Letters*, vol. 3, no. 3, pp. 1–4, 2019.

[14] G. Zhang, H. Li, and F. Wenger, "Object detection and 3d estimation via an fmcw radar using a fully convolutional network," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 4487–4491.

[15] B. Major, D. Fontijne, A. Ansari, R. Teja Sukhavasi, R. Gowaikar, M. Hamilton, S. Lee, S. Grzechnik, and S. Subramanian, "Vehicle detection with automotive radar using deep learning on range-azimuth-doppler tensors," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[16] D. Brodeski, I. Bilik, and R. Giryes, "Deep radar detector," in *2019 IEEE Radar Conference (RadarConf)*, 2019, pp. 1–6.

[17] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 1021–1028.

[18] T. Stadelmayer, A. Santra, R. Weigel, and F. Lurz, "Parametric Convolutional Neural Network for Radar-based Human Activity Classification Using Raw ADC Data," 9 2020. [Online]. Available: https://www.techrxiv.org/articles/preprint/Parametric_Convolutional_Neural_Network_for_Radar-based_Human_Activity_Classification_Using_Raw_ADC_Data/12896108

[19] L. Duong, T. Cohn, S. Bird, and P. Cook, "Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Beijing, China: Association for Computational Linguistics, Jul. 2015, pp. 845–850. [Online]. Available: https://www.aclweb.org/anthology/P15-2139

[20] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *CoRR*, vol. abs/1708.02002, 2017. [Online]. Available: http://arxiv.org/abs/1708.02002