Sensors & Transducers, Vol. 262, Issue 3, September 2023, pp. 30-39

Sensors & Transducers

Published by IFSA Publishing, S. L., 2023

http://www.sensorsportal.com

Robust DOA Estimation in Low SNR Radar Systems Using CNN

[*] Webert Montlouis and Yiyang Li

Johns Hopkins University, Baltimore MD, 21218, USA

E-mail: wmontlouis@jhu.edu, yli302@jhu.edu

Received: 22 July 2023 Accepted: 29 August 2023 Published: 30 September 2023

Abstract: Radar systems generate large amounts of data, which can create bottlenecks in the system data and signal processors if not handled properly. In recent years, Radio Frequency (RF) sampling has received popularity due to its potential benefits to the systems. However, RF sampling can generate a large amount of data, presenting processing, communication, and storage challenges. In the past, addressing these challenges required a combination of efficient signal processing algorithms, appropriate computational resources, robust data transmission, and fast access optimized storage. Despite advances in high-speed communication systems, data processing techniques like parallel processing and GPU, and storage technology, the large amount of data streaming for applications like radar systems and wireless communications is still a problem. Deep learning (DL) has emerged as a promising approach for processing radar data and extracting relevant features. In this paper, we investigate the use of Convolutional Neural Networks (CNNs) as a fast signal processor for classifying the direction of targets from radar returns. We evaluate the performance of CNNs in Direction of Arrival (DOA) recognition, particularly in extremely low Signal-to-Noise Ratio (SNR) conditions. Our proposed CNN-based DOA recognition system leverages a special configuration that minimizes delay in the signal processor, where In-phase and Quadrature (IQ) components of the data are fed directly to the trained network. Experimental results show that the proposed CNN-based DOA recognition system achieves high accuracy in low SNR conditions, demonstrating the effectiveness of CNNs for radar signal processing. This study provides insights into the design and optimization of CNNs for DOA estimation in radar systems.

Keywords: Artificial intelligence, Deep learning, Convolutional neural networks, Radar signal processing, Direction of arrival, Low signal-to-noise ratio.

1. Introduction

The current environment requires that we make decisions for complex systems at machine speed. The proliferation of deep learning architectures provides the framework to tackle many problems that we thought impossible to solve a decade ago [1, 2]. The introduction of faster CPU, GPU, and parallel processing makes the application of deep leaning feasible [3-5]. With advances in GPU, more sophisticated deep learning architectures are developed [6, 7]. Therefore, replacing traditional signal processing techniques with deep learning is logical.

Deep learning is a specific type of machine learning (ML) type that utilizes neural networks to make predictions about certain types of data [8]. It is called “deep” learning because these networks often have great depth, propagating data through many layers to perform classification. Deep learning applications include facial recognition, natural language processing, and self-driving cars, to name a few [9-13]. In our case, deep learning will be used to complete the task of radar signal processing. More specifically, it performs angle of arrival classification on a section radar signal. Papers have been published for radar applications but are concentrated more on signal classification [14, 15].

Sequence-to-sequence classification takes a temporal data sequence and classifies each time step using a label [16-19]. This paper applies the trained CNN network on each radar receive window. In this case, the classification of time steps of a reflected radar signal is performed. The complex data from the received window is transformed into real sequences to accomplish this classification process, which is more efficient than further adjusting the deep learning model to accommodate complex datasets [20].

First proposed in [21], CNN has shown its ability to train a multi-layer network using gradient descent for complex, high-dimensionality, and a large amount of data. More specifically, the CNN model uses multiple layers of trainable filters to convolve with test samples, extracting local features in the lower and global features in the higher layers. More recent applications of CNN involve facial emotion recognition [22], PET image reconstruction [23], and many other classifications of image processing applications. CNN's capability in determining the angle of arrival is explored by Kokkins et al. [24]. Yet, the authors use a sample correlation matrix transformation on the input data, demanding additional computational power to transform the raw input signals.

Data transformation is a common step in deep learning signal processing, but it can be problematic in applications where hardware is limited, and system requirements demand low latency. In radar, for instance, the computational burden and time delay associated with these transformations can significantly impact some radar modes and render the system inefficient. To address this issue, we develop a CNN model that analyzes the direction of arrival of a target using the raw signal at the output of a Uniform Linear Array (ULA) and evaluates its performance. This modern approach using the trained network replaces traditional radar signal processors, providing a faster response time.

2. Radar Model

The radar signal data model that will be used in the simulation is based on the constant low-velocity target model [25]. The radar uses a linear frequency modulated (LFM) signal. The transmit signal for a single pulse is modeled as shown in equation (1) by

$$s(t) = u(t)e^{j2\pi f_0 t} = rect\left(\frac{t}{T}\right)e^{j2\pi\left(f_0 t+\frac{1}{2}Kt^2\right)} \quad (1)$$

where $u(t)$ is the envelope of the LFM waveform, T is the pulsewidth, and K is the linear modulation rate. This transmit signal is then down-converted to a baseband signal given by

$$s_b(t) = rect\left(\frac{t}{T}\right)e^{j\pi K t^2} \quad (2)$$

When the signal is incident upon a target, since the constant, low-velocity target model is being used, this target is taken as static while the wave is reflected. Because of this, the reflected baseband signal is modeled by the equation

$$s_{rb}(t) = rect\left(\frac{t-t_r}{T}\right)e^{j\pi K t^2} + N(\sigma), \quad (3)$$

where $t_r = 2R/c$ is the time delay for the signal to propagate to the target and back to the radar receiver, and $N(\sigma)$ is the random Gaussian noise of the environment. Since the detection of the target is of interest here, it is necessary to set up the return signal intensity for the single pulse system.

The intensity of the return will be based on the radar range equation (RRE). For the single pulse case, the RRE is given by

$$SNR = \frac{P_t G^2 \lambda^2 \sigma}{(4\pi)^3 R^4 k T_0 BFL} \quad (4)$$

with the parameters

P_t: $Peak\ transmit\ power$
G: $Array\ gain$
λ: $Transmit\ signal\ wavelength$
σ: $Target\ radar\ cross\ section$
k: $Boltzman\ constant$
B: $Radar\ bandwidth$
F: $Noise\ figure$
L: $Total\ system\ losses$

It is assumed that a 13dB signal-to-noise ratio is needed on the return for a general detection.

3. Radar Signal Processing Models

The signal processor is an integral component of the radar system, playing a crucial role in its overall functionality [26]. It is responsible for various essential functions, such as return signal processing, extracting pertinent information, and facilitating accurate target detection and tracking [27-29]. As the complexity of the radar system's application increases, so does the level of difficulty associated with the tasks performed by the signal processor. The signal processor serves as the backbone of the radar system, ensuring that the received signals are efficiently processed and transformed into actionable data. Advanced algorithms and signal processing techniques effectively separate the desired information from noise and clutter, allowing for precise target detection, classification, and identification.

Furthermore, the signal processor is vital in extracting valuable insights from the radar returns. It employs sophisticated algorithms to analyze the signals, discerning crucial details such as target range, speed, direction, and even shape. This extracted information aids in making accurate decisions, whether in military applications, weather monitoring, air traffic control, or any other radar-dependent field [30-33]. In complex applications, the signal processor faces heightened challenges. The signal processor's tasks become more demanding as the radar system encounters scenarios with increased interference, varied environmental conditions, or highly maneuvering multiple targets. It must adapt to changing needs, optimize signal-to-noise ratios, and employ advanced algorithms to mitigate distortions and enhance target detection and tracking capabilities.

Over the past decade, radar signal processing has undergone significant advancements driven by remarkable transformations in chip design processes. These advancements have played a crucial role in revolutionizing radar technology. One of the primary factors responsible for these breakthroughs is the continuous improvement in chip design processes, leading to remarkable achievements. The continuous evolution of chip design processes has made it possible to manufacture smaller transistors, thereby increasing the transistor count on a single die. This miniaturization trend has paved the way for higher levels of integration and enabled the development of more sophisticated radar systems. These systems benefit from the higher transistor density's increased processing power and computational capabilities.

Simultaneously, new technologies have facilitated the realization of low nano-meter processes, enabling components to operate at higher frequencies. This is particularly noteworthy for key components like mixers and analog-to-digital converters, which play critical roles in receiver designs. The ability to operate at higher frequencies enhances the overall performance of the radar system, allowing for improved signal quality, better resolution, and increased sensitivity. The culmination of these advancements has given rise to the RF-System-On-Chip (RFSOC) concept [34], which represents a significant leap forward in radar design. Integrating multiple RF components onto a single chip reduces the overall component count in the receiver, leading to a less complex and more robust design. This integration not only enhances the system's efficiency but also simplifies manufacturing processes, reduces power consumption, and optimizes the use of physical space.

The benefits of the RFSOC approach extend beyond the radar system's hardware. The reduced component count and increased integration enable streamlined signal processing algorithms and architectures, resulting in improved system performance, faster data processing, and enhanced real-time capabilities. These advantages directly impact the radar system's reliability, responsiveness, and overall effectiveness. The last decade has witnessed significant transformations in radar signal processing, largely driven by advancements in chip design processes. The ability to manufacture smaller transistors and utilize low nano-meter processes has paved the way for increased integration, higher frequencies, and the realization of RF-System-On-Chip (RFSOC) technology. This has led to radar systems with lower component counts, simpler designs, improved performance, and greater efficiency. The continuous evolution of chip design processes is expected to further propel radar technology, opening doors to even more advanced and capable systems in the future.

A. *Conventional Radar Signal Processor*

A conventional radar signal processor refers to the traditional approach or standard methods for processing radar signals in a radar system. These methods have been in use for decades. It typically consists of a series of interconnected hardware and software components designed to perform specific functions related to signal processing. Together, these blocks perform signal conditioning, pulse compression, range-doppler processing, clutter mitigation, target detection, data fusion, and tracking. A block diagram of the conventional signal processor is given in Fig. 1.

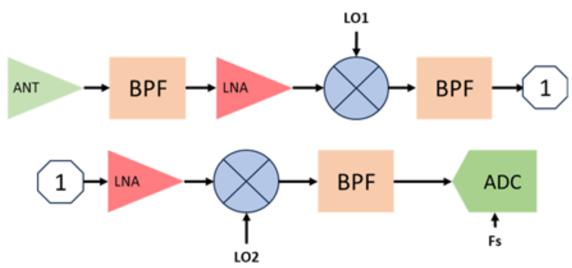

Fig. 1. Conventional receiver chain architecture.

B. *Deep Learning Radar Signal Processor*

In recent years, deep learning has shown promise in enhancing the performance and capabilities of radar systems. Many key aspects of signal processing have been investigated using deep learning. For instance, deep learning models excel at automatically learning relevant features from raw data. In radar signal processing, this means that deep learning models can extract meaningful features directly from the radar returns without relying on feature engineering or domain-specific knowledge. By learning features from the data, deep learning models can capture intricate patterns and discriminate between different types of radar echoes.

A deep learning radar signal processor can employ neural network architectures suitable for specific tasks. Convolutional Neural Networks (CNNs) are commonly used for tasks like radar target classification and image-based radar signal processing, where spatial relationships are important

[35]. Recurrent Neural Networks (RNNs), such as Long Short-Term Memory (LSTM) networks, are useful for processing sequential radar data, such as time series or pulse-Doppler signals. Transformers, which have gained popularity in natural language processing tasks, have also been adapted for radar signal processing to handle sequence-to-sequence tasks.

One aspect that is always the subject of conversation is the availability of datasets for training because deep learning models often require large annotated datasets to generalize well. In the case of classification in radar signal processing, labeled radar datasets are used to train the deep learning models. These datasets may contain radar returns from different waveforms, targets, clutter, noise, and various environmental conditions. Training on such datasets enables the deep learning models to learn the underlying patterns and characteristics of different radar signatures.

Another aspect that is also of interest to many is the ability to apply deep learning for real-time signal processing applications. In this case, deep learning models can be optimized for real-time radar signal processing by leveraging efficient architectures and model optimizations. Specialized hardware, such as Graphics Processing Units (GPUs) or dedicated accelerators like Tensor Processing Units (TPUs), can be employed to speed up the inference process and achieve the required processing rates.

The question is, how do we effectively combine the advances in RF sampling and the real-time capability of deep learning? This paper aims to explore the effective integration of advances in RF sampling and the real-time capability of deep learning. By combining these two advancements, we aim to optimize radar signal processing. In this paper, we begin by demonstrating the impact of RF sampling on the radar receiver. Subsequently, we leverage deep learning techniques, specifically sequence-to-sequence classification, to reduce computational load and streamline the radar signal processor. To illustrate this concept, a block diagram of a radar receiver employing deep learning is presented in Fig. 2.

By leveraging the benefits of RF sampling, which allows for direct digitization of radio frequency signals, we can enhance the quality and resolution of radar measurements. This enables more accurate detection and tracking of targets. However, the increased data volume generated by RF sampling poses computational challenges.

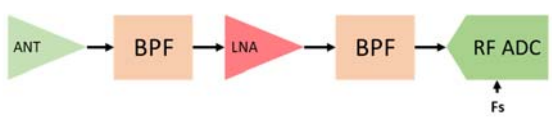

Fig. 2. RF Sampling and Deep Learning receiver chain architecture.

To address this, we introduce deep learning, a powerful tool capable of learning intricate patterns and features from radar signals. We can effectively extract valuable information and reduce the computational burden on the radar signal processor by employing deep neural networks for sequence-to-sequence classification. This combination of RF sampling and deep learning offers a promising solution for achieving real-time radar signal processing with improved accuracy and efficiency.

The block diagram illustrates the integration of deep learning within the radar receiver. It showcases how the received RF signals are processed through the deep learning model, which performs sequence-to-sequence classification. The model has been trained on a dataset of radar signals to learn the patterns and features associated with different target signatures. This approach enables rapid and accurate classification of radar returns, providing valuable information for target identification and tracking.

This research aims to demonstrate the potential of combining RF sampling and deep learning techniques to optimize radar signal processing. By reducing computational load and enhancing the real-time capability of the radar system, we can achieve more efficient and accurate detection, classification, and tracking of targets.

4. Data Formulation and Model

In this section, we will delve into the data generation, labeling, and training process within the framework of our deep learning model. We first explore the data properties in section 4-A, gaining insights into the characteristics and intricacies of the dataset. Subsequently, in section 4-B, we present a detailed description of the convolutional neural network (CNN) architecture employed in our model.

Adopting this approach ensures a comprehensive understanding of the entire process. The data properties lay the foundation for comprehending the dataset's nature. At the same time, the subsequent section offers valuable insights into the architecture, enabling a more profound comprehension of the model's structure and functioning.

A. Data Properties

The training datasets are generated by employing a sampling technique based on the radar model, specifically the receiver output. For diversity and applicability, the target distances of these samples are randomly selected within the range of 200 km to 300 km while maintaining a constant target velocity of 10 meters per second.

Simulating the receiving antenna involves utilizing 40 channels, encompassing a 5km range window (equivalent to a 1.67e-5 second time window). As a result, each sample forms a 40x400 complex-valued matrix. A decomposition process is applied to facilitate the incorporation of this data into the deep learning networks without introducing complex-valued weights. Each signal sample is split into corresponding real and imaginary components, resulting in 3D matrices. Consequently, the final dimensions of the input data are represented as 40x400x2, providing an optimal format for seamless integration into the network architecture.

B. CNN Model

We are using a standard CNN model with 16 layers, denoted as follows:

$$f(X) = f_{15}\left(f_{14}\left(\dots f_1(\mathrm{X})\right)\right) \quad (3)$$

where layer 1,6,11 denoted by $\{f_i(\cdot)\}_{i=1,6,11}$ are the three convolutional layers. Due to the nature of the dataset, it is important to keep the filters in the first convolutional layer large so that signal across multiple channels could be analyzed, so the first convolutional layer has kernel size of 5×5. The other two deeper convolutional layers has a smaller kernel with the size of 3×3. For the first layer, stride size of 2 is chosen, while the rest of the layers had a stride size of 1. Padding is added to the deeper two layers (6 and 11) so that the output's dimensions are the same as inputs. The number of filters also follows a typical CNN network, where the first two layers have 16 filters, and the last layer has 32 filters.

Layers 2, 7 and 12 denoted by $\{f_i(\cdot)\}_{i=2,7,12}$ are the batch normalization layers, which helps to avoid overfitting while speeding up the training process. During training, mini batches are used to update the gradients after each iteration, resulting in small noises that accumulate into a more generalized model. This approach ensures that the model is robust towards variations. To strike a balance between training time and robustness, a batch size of 128 samples is used in the training process.

Layers 3, 8 and 13 denoted by $\{f_i(\cdot)\}_{i=3,8,13}$ are the dropout layers with a dropout factor of 0.5. This sets 50% of the weights to 0 to counter overfitting. Layers 4, 9 and 14 are Leaky Rectified Linear Unit layers $\{f_i(\cdot)\}_{i=4,9,14}$ used as activation functions that helps avoid vanishing gradient compared to normal ReLU layers. The whole model is finished with a fully connected layer f_{15}, and a SoftMax layer f_{16} for one-hot encoding for the angle classification.

5. Training and Evaluation

In this section, we comprehensively explore the crucial aspects of our study, including the input datasets, training methodology, and evaluation metrics. The discussion unfolds in a structured manner, with each key element allocated a designated subsection. Firstly, in section 5-A, we meticulously examine the input datasets utilized in our research. This segment sheds light on the composition and characteristics of the datasets, providing essential context for understanding our model's training process.

Secondly, section 5-B delves into the intricacies of our chosen training methodology. We elucidate the techniques, algorithms, and procedures employed to optimize our deep learning model, ensuring clarity on the approach we adopted to maximize its effectiveness.

Lastly, section 5-C is dedicated to a detailed description of the evaluation metrics utilized to assess the performance of our model. By thoroughly explaining these metrics, we offer transparency in our evaluation process, providing readers with a comprehensive grasp of our model's capabilities and limitations. Through this structured approach, we aim to present a cohesive and informative narrative, facilitating a deeper understanding of the key components and outcomes of our research.

A. Input Dataset

The DOA estimation problem is formulated as a single-label classification problem. The sample space is $-59.67°$ to $60°$, quantized into 360 classes with 1/3 degrees intervals to achieve higher accuracy. A generated signal with $\phi = x°$ is fed into the network with categorical label x, and later evaluations of the network are done by reverting the categorical labels back to angles. For each degree category, 100 samples are generated with random noise and random target distances discussed in section III.A, for a total of 36,000 samples per dataset.

To test the performance of the model under various SNR conditions, multiple training datasets are generated in -25 dB to 20 dB SNR in 5 dB intervals. Each model is trained with one dataset, and the resulting models are compared against each other by evaluating the RMSE of new test signals at different SNR levels.

B. Training Methodology

Our training methodology used cross-entropy loss and the Adam optimizer for loss evaluations and parameter updates. The Adam optimizer is especially worth mentioning since it combines the advantages of adaptive learning rate methods and momentum-based methods for data from a low SNR environment. It adjusts the learning rate for each parameter based on the estimate of the first and second moments of the gradients, allowing it to adapt to different gradients and learning rates for different parameters. In addition, the optimizer also incorporates a momentum term, which helps accelerate the convergence process by dampening the effect of local minimums, which occurs frequently in a high variance, noisy data input.

One crucial hyperparameter to consider when using the Adam optimizer is the learning rate, which controls the adjustment made to the model weights after each iteration. We start with a high learning rate of 0.001 to ignore noise patterns in the training dataset. However, to ensure that our model extracts more informative features from the target signals in later stages of training, we decrease the learning rate by 10% every ten epochs. This approach balances convergence speed with accuracy while mitigating the risk of overlooking important features.

Another hyperparameter that we tune is the epoch number, which directly affects the training time and the risk of overfitting. We begin with a baseline of 100 epochs and implement early stopping when convergence is achieved around 30 epochs. We have conducted several runs to ensure that 40 epochs capture the convergence without inducing overfitting.

C. Evaluation Metrics

For the aforementioned datasets created over a range of SNR values from -25 dB to 20 dB with 5dB intervals, we trained multiple models, with each model trained with one SNR value dataset. To assess the model's performance under varying levels of noise, we calculated the root mean squared error (RMSE) for each model given newly generated data and plotted the results over the range of -30 dB to 20 dB. This approach allowed us to visualize the performance of each model and assess the impact of the varying SNR values on the model's accuracy. By comparing the RMSE plots for each model, we were able to determine the optimal SNR range for each model and identify any limitations or strengths in its performance. It is customary in DOA estimation to use the RMSE curve to evaluate the performance of an estimation algorithm [36-40]. In radar, the Probability of Detection (Pd) curve provides a mechanism to evaluate the performance of the radar system [41-43].

6. Results

This section discusses results from the trained models tested using newly generated datasets under different SNR conditions. The RMSE plots are shown in Fig. 3 and discussed in section 6-A, and the accuracies of different models are discussed in section 6-B.

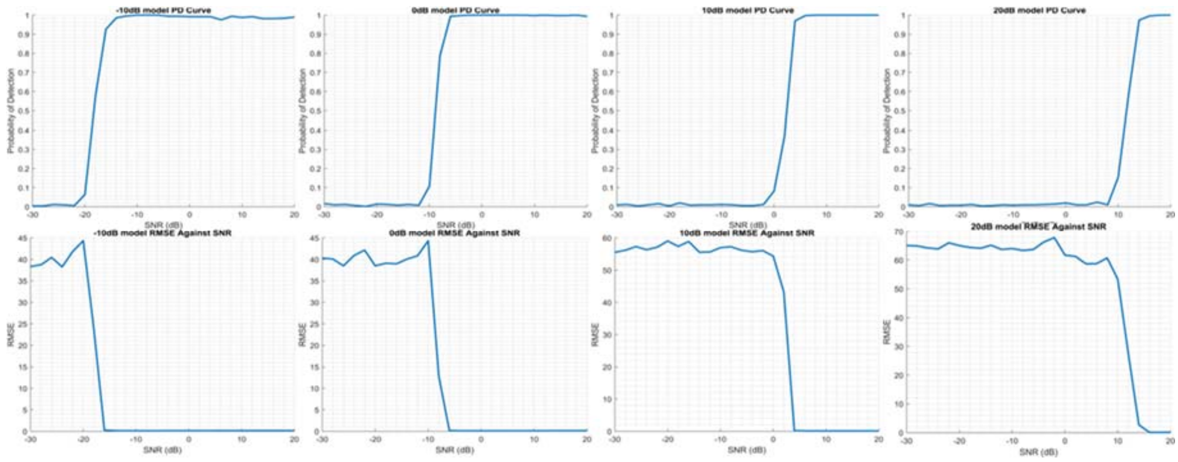

Fig. 3. Performance results for SNR values -10, 0, 10, and 20 dB. Top row are the probability of detection curves. Bottom row are the Root Mean Square Error curves.

A. RMSE Curves

Below are the RMSE curves over the -30 dB to 20 dB ranges. We could observe that:

1. In radar systems and other signal processing applications, the signal-to-noise ratio (SNR) represents the ratio of the power of the desired signal to the power of the background noise. A higher SNR implies a stronger and more distinguishable signal relative to the noise, which typically leads to improved detection performance.

However, an intriguing phenomenon arises in the context of training deep learning models for radar applications. It has been observed that the lower bounds of the detection range, i.e., the ability of the model to detect weak signals, are heavily influenced by the SNR of the training environment. Specifically, as the SNR in the training environment increases, the trained model tends to exhibit worse performance when operating in lower SNR conditions. This counterintuitive behavior can be attributed to the training data distribution. When the training environment provides high SNR examples, the model becomes proficient at learning and recognizing strong signals, prioritizing these patterns during training. Consequently, the model may become less sensitive to weaker signals with lower SNR levels, as they were not as prevalent in the training data.

In contrast, when the training environment includes more instances of lower SNR scenarios, the model is compelled to pay closer attention to weaker signals during training. Consequently, the model becomes more adept at detecting and handling such low SNR conditions, resulting in improved performance when encountering weaker signals in real-world scenarios. It is essential to consider the implications of this behavior when designing and evaluating deep learning models for radar and signal processing applications. Striking a balance in the training data distribution and considering a diverse range of SNR levels can help mitigate the adverse effects of over-relying on high SNR training samples and ultimately lead to better overall performance in various real-world conditions.

2. The upper bounds of SNR accuracy refer to the highest levels of SNR at which the model exhibits optimal performance. Surprisingly, our analysis found that the upper bounds of SNR accuracy remain relatively consistent across a range of training conditions. This stability is a promising indicator, suggesting that the model's effectiveness is not heavily influenced by the specific conditions in which it was trained. This consistency implies that the model can handle a diverse array of SNR levels with comparable accuracy, offering a robust and reliable performance profile. However, one notable exception in our findings is the -25 dB SNR level. The RMSE plot demonstrates higher error values on the right side for this specific SNR value, indicating a decrease in model accuracy. The right side of the plot corresponds to scenarios with low SNR levels. This observed degradation in performance can be attributed to a peculiar aspect of the training process.

In deep learning models, there is often a trade-off between noise suppression and signal preservation. During training, the model learns to distinguish signal patterns from background noise. In cases where the training data predominantly comprises low SNR samples, the model might prioritize noise suppression to enhance the recognition of weak signals. This focus on noise reduction can be beneficial in noisy environments but may lead to overaggressive filtering in scenarios with very little noise. Consequently, at -25 dB SNR, where noise levels are considerably lower, the model's excessive noise suppression might inadvertently interfere with the actual signal, resulting in a drop in accuracy on the right side of the RMSE plot. To mitigate this issue, future model improvements may involve carefully calibrating noise suppression mechanisms or incorporating adaptive filtering techniques, which can adaptively adjust their filtering based on the prevailing SNR conditions. By doing so, the model can strike a better balance between noise reduction and signal preservation, leading to improved performance across a broader range of SNR levels and ensuring consistent accuracy even in challenging real-world conditions.

3. A deep learning model's stability is vital, especially in real-world applications where robustness and reliability are essential. In our investigation, we observed that our trained model possesses exceptional stability, particularly when dealing with signals with higher SNR values than what the model encountered during training. A low root mean square error (RMSE) curve signifies that the predicted angles closely align with the ground truth angles for such high SNR scenarios. This consistently low RMSE indicates the model's proficiency in effectively capturing and generalizing from the patterns in the training data to new, more favorable conditions where the signals are stronger.

The stability of the model in predicting angles accurately, even in less favorable SNR conditions, provides valuable insights into its generalization capabilities. This characteristic is a testament to the robustness achieved during the training process, where the model learns relevant features and patterns in a manner that is transferrable to new, unseen data. While acknowledging this achievement, it's also essential to continue evaluating the model's performance across various SNR ranges and real-world conditions. Maintaining a diverse dataset that encompasses a wide spectrum of SNR levels can help further optimize the model's adaptability and ensure reliable performance in a broader range of practical scenarios.

B. Accuracy

The form below shows the accuracy of different models tested against 1000 new signals generated for each SNR level, allowing an error of $\pm 0.5°$. The matrix's diagonal elements indicate the network's stability when tested at the same SNR as that of training. The table shows that the models show consistent accuracy when tested at the same SNR as that of training. The only exception to this would be the 63 % accuracy achieved by the model trained under -25 dB and tested under -25 dB, which could result from the model's limitations, as it could not further denoise and collect meaningful information under very low SNR.

The off-diagonal elements of the matrix indicate variations in performance when tested under different SNR conditions than that of training. It can be observed that the accuracy of the models degrades as the SNR decreases. This could be observed from all models except the model trained at -25 dB, where its performance decreased as SNR increased. One possible explanation could be that the model has learned too much about offsetting noise, influencing the classification when less noise is presented. Similar behavior could also be observed from other models; for example, the model trained at -20 dB has accuracy peaked at -15 dB. However, this effect is much less significant in other models compared to the -25 dB model. This indicates that while the low-SNR models are generally robust enough to handle both low SNR and high SNR signals, the low-SNR training environment has an influence on their classification for high SNR signals.

The table also reveals that the models trained and tested at low SNR conditions may not be sufficient for some applications. For instance, the cutoff point for military applications seems to be -15 dB SNR. The models trained and tested at -15 dB SNR show an accuracy of around 98 %, which is reasonably good but may not be sufficient for some military applications that require higher accuracy.

Table 1. Deep Learning Models Performance: Training SNR Vs Performance Testing SNR.

Model Training SNR \ Model Performance Testing SNR	-25	-20	-15	-10	-5	0	10	20
-25	**66.7**	91.4	93.7	92.1	83.1	73.8	65.7	65.9
-20	63.0	**95.1**	97.9	97.3	96.3	95.7	95.6	95.8
-15	15.8	76.8	**98.4**	99.1	98.9	98.2	98.4	97.2
-10	1.1	23.5	90.6	**99.9**	99.5	99.4	99.1	98.9
-5	0.8	2.2	30.3	92.2	**99.7**	99.8	99.5	99.5
0	0.7	1.0	1.1	34.6	97.2	**100**	99.9	99.7
10	0.9	0.6	0.7	0.8	1	23.8	**100**	99.9
20	0.9	0.9	0.4	0.7	1.3	2.0	32.3	**99.8**

7. Conclusions

In conclusion, our study demonstrates the potential of using deep learning techniques, specifically CNNs, to accurately estimate target DOA in low SNR conditions up to -20 dB. The CNN model presented in this paper was trained on raw radar data without preprocessing and achieved high accuracy in real-time DOA estimation. The results suggest that CNNs can effectively address the challenges of high computational complexity associated with the use of RF sampling in radar receiver designs. By leveraging deep learning capabilities, our approach enables efficient and cost-effective DOA estimation, which can improve the overall performance of radar systems. Our findings highlight the importance of continued research in the field of deep learning for radar signal processing and the potential for further innovation and optimization of CNN-based techniques for real-world applications.

References

[1]. S. Yang *et al.*, Learning COVID-19 Pneumonia Lesion Segmentation from Imperfect Annotations via Divergence-Aware Selective Training, *IEEE Journal of Biomedical and Health Informatics,* 26, 8, 2022, pp. 3673-3684.

[2]. A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet classification with deep convolutional neural networks, *Commun. ACM,* Vol. 60, No. 6, 2017, pp. 84–90.

[3]. K. Aida-Zade, E. Mustafayev, and S. Rustamov, Comparison of Deep Learning in Neural Networks on CPU and GPU-based frameworks, in *Proceedings of the 11th International IEEE Conference on Application of Information and Communication Technologies (AICT)*, 20-22 Sept. 2017, pp. 1-4.

[4]. T. Haryanto, H. Suhartanto, and X. Lie, Past, present, and future trend of GPU computing in deep learning on medical images, in *Proceedings of the International Conference on Advanced Computer Science and Information Systems (ICACSIS' 2017)*, 28-29 Oct. 2017, pp. 21-28.

[5]. Ebubekir Buber, Banu Diri, Performance Analysis and CPU vs GPU Comparison for Deep Learning, in *Proceedings of the 6th International Conference on Control Engineering & Information Technology (CEIT' 2018)*, 25-27 Oct. 2018, pp. 1-6.

[6]. A. A. Badawi, B. Veeravalli, J. Lin, N. Xiao, M. Kazuaki, and A. K. M. Mi, Multi-GPU Design and Performance Evaluation of Homomorphic Encryption on GPU Clusters, *IEEE Transactions on Parallel and Distributed Systems,* Vol. 32, No. 2, 2021, pp. 379-391.

[7]. K. Cho *et al.*, Signal Integrity Design and Analysis of Silicon Interposer for GPU-Memory Channels in High-Bandwidth Memory Interface, *IEEE Transactions on Components, Packaging and Manufacturing Technology,* Vol. 8, No. 9, 2018, pp. 1658-1671.

[8]. H. C. Kaskavalci and S. Gören, A Deep Learning Based Distributed Smart Surveillance Architecture using Edge and Cloud Computing, in *Proceedings of the International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML' 2019)*, 26-28 Aug. 2019, pp. 1-6.

[9]. R. A. Borgalli and S. Surve, Deep Learning Framework for Facial Emotion Recognition using CNN Architectures, in *Proceedings of the International Conference on Electronics and Renewable Systems (ICEARS' 2022)*, 16-18 March 2022, pp. 1777-1784.

[10]. S. M. S. Islam, H. Mahmood, A. A. Al-Jumaily, and S. Claxton, Deep Learning of Facial Depth Maps for Obstructive Sleep Apnea Prediction, in *Proceedings of the International Conference on Machine Learning and Data Engineering (iCMLDE' 2018)*, 3-7 Dec. 2018, pp. 154-157.

[11]. M. Gupta, S. K. Verma, and P. Jain, Detailed Study of Deep Learning Models for Natural Language Processing, in *Proceedings of the 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN' 2020)*, 18-19 Dec. 2020, pp. 249-253.

[12]. P. Kłosowski, Deep Learning for Natural Language Processing and Language Modelling, in *Proceedings of the Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA' 2018)*, 19-21 Sept. 2018, pp. 223-228.

[13]. R. Kulkarni, S. Dhavalikar, and S. Bangar, Traffic Light Detection and Recognition for Self Driving Cars

Using Deep Learning, in *Proceedings of the 4th International Conference on Computing Communication Control and Automation (ICCUBEA' 2018)*, 16-18 Aug. 2018, pp. 1-4.

[14]. M. J. Sagayaraj, V. Jithesh, and D. Roshani, Comparative Study Between Deep Learning Techniques and Random Forest Approach for HRRP Based Radar Target Classification, in *Proceedings of the International Conference on Artificial Intelligence and Smart Systems (ICAIS' 2021)*, 25-27 March 2021, pp. 385-388.

[15]. H. Li, W. Jing, and Y. Bai, Radar emitter recognition based on deep learning architecture, in *Proceedings of the CIE International Conference on Radar (RADAR' 2016)*, 10-13 Oct. 2016, pp. 1-5.

[16]. H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, and M. D. Vos, SeqSleepNet: End-to-End Hierarchical Recurrent Neural Network for Sequence-to-Sequence Automatic Sleep Staging, *IEEE Transactions on Neural Systems and Rehabilitation Engineering,* Vol. 27, No. 3, 2019, pp. 400-410.

[17]. Y. Tang, J. Xu, K. Matsumoto, and C. Ono, Sequence-to-Sequence Model with Attention for Time Series Classification, in *Proceedings of the IEEE 16th International Conference on Data Mining Workshops (ICDMW' 2016)*, 12-15 Dec. 2016, pp. 503-510.

[18]. Y. Deng, L. Wang, H. Jia, X. Tong, and F. Li, A Sequence-to-Sequence Deep Learning Architecture Based on Bidirectional GRU for Type Recognition and Time Location of Combined Power Quality Disturbance, *IEEE Transactions on Industrial Informatics,* Vol. 15, No. 8, 2019, pp. 4481-4493.

[19]. S. Arniriparian, M. Freitag, N. Cummins, M. Gerczuk, S. Pugachevskiy, and B. Schuller, A Fusion of Deep Convolutional Generative Adversarial Networks and Sequence to Sequence Autoencoders for Acoustic Scene Classification, in *Proceedings of the 26th European Signal Processing Conference (EUSIPCO' 2018)*, 3-7 Sept. 2018, pp. 977-981.

[20]. C. Lee, H. Hasegawa, and S. Gao, Complex-Valued Neural Networks: A Comprehensive Survey, *IEEE/CAA Journal of Automatica Sinica,* Vol. 9, No. 8, 2022, pp. 1406-1426.

[21]. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, *Proceedings of the IEEE,* Vol. 86, No. 11, 1998, pp. 2278-2324.

[22]. E. Pranav, S. Kamal, C. S. Chandran, and M. H. Supriya, Facial Emotion Recognition Using Deep Convolutional Neural Network, in *Proceedings of the 6th International Conference on Advanced Computing and Communication Systems (ICACCS' 2020)*, 6-7 March 2020, pp. 317-320.

[23]. K. Gong ***et al.***, Iterative PET Image Reconstruction Using Convolutional Neural Network Representation, *IEEE Transactions on Medical Imaging,* Vol. 38, No. 3, 2019, pp. 675-685.

[24]. G. Kokkinis, Z. D. Zaharis, P. I. Lazaridis, and N. V. Kantartzis, Direction of Arrival Estimation Applied to Antenna Arrays using Convolutional Neural Networks, in *Proceedings of the 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC' 2022)*, 30 May-4 June 2022, pp. 1-4.

[25]. M. A. Richards, Fundamentals of radar signal processing, 2nd ed., *McGraw-Hill Education LLC.*, Chicago, Ill., 2014.

[26]. T. Yang, A. D. Maio, J. Zheng, T. Su, V. Carotenuto, and A. Aubry, An Adaptive Radar Signal Processor for UAVs Detection With Super-Resolution Capabilities, *IEEE Sensors Journal,* Vol. 21, No. 18, 2021, pp. 20778-20787.

[27]. X. Chen ***et al.***, Implementation of an Adaptive Wideband Digital Array Radar Processor Using Subbanding for Enhanced Jamming Cancellation, *IEEE Transactions on Aerospace and Electronic Systems,* Vol. 57, No. 2, 2021, pp. 762-775.

[28]. C. J. Cochrane, K. B. Cooper, S. L. Durden, R. R. Monje, and R. J. Dengler, An FPGA-Based Signal Processor for FMCW Doppler Radar and Spectroscopy, *IEEE Transactions on Geoscience and Remote Sensing,* Vol. 58, No. 8, 2020, pp. 5552-5563.

[29]. H. Su, Q. Bao, and Z. Chen, Parameter Estimation Processor for Chirp Signals Based on a Complex-Valued Deep Neural Network, *IEEE Access,* Vol. 7, 2019, pp. 176278-176290.

[30]. G. J. Frazer and C. G. Williams, Decametric Radar for Missile Defence, in *Proceedings of the IEEE Radar Conference (RadarConf' 19)*, 22-26 April 2019, pp. 1-6.

[31]. L. Li-hua, W. Zhuang, and H. Wei-dong, Precession Period Extraction of Ballistic Missile Based on Radar Measurement, in *Proceedings of the 2006 CIE International Conference on Radar*, 16-19 Oct. 2006, pp. 1-4.

[32]. W. Huaiyi, J. Johnson, C. Baker, Y. Lixin, and Z. Chenglin, On spectrum sharing between communications and air traffic control radar systems, in *Proceedings of the IEEE Radar Conference (RadarConf' 15)*, 10-15 May 2015, pp. 1545-1550.

[33]. M. Yeary ***et al.***, An update on the multi-channel phased array Weather Radar at the National Weather Radar Testbed, in *Proceedings of the IEEE RadarCon (RADAR' 2011)*, 23-27 May 2011, pp. 971-973.

[34]. F. Michalak, W. Zabołotny, P. Ł, M. Malanowski, M. Piasecki, and K. Kulpa, Universal RFSoC-based Signal Recorder for Radar Applications, in *Proceedings of the 23rd International Radar Symposium (IRS' 2022)*, 12-14 Sept. 2022, pp. 136-140.

[35]. S. P. Kannojia and G. Jaiswal, Ensemble of Hybrid CNN-ELM Model for Image Classification, in *Proceedings of the 5th International Conference on Signal Processing and Integrated Networks (SPIN' 2018)*, 22-23 Feb. 2018, pp. 538-541.

[36]. S. Kim, D. Oh, and J. Lee, Joint DFT-ESPRIT Estimation for TOA and DOA in Vehicle FMCW Radars, *IEEE Antennas and Wireless Propagation Letters,* Vol. 14, 2015, pp. 1710-1713.

[37]. Y. M. Ning, S. Ma, F. Y. Meng, and Q. Wu, DOA Estimation Based on ESPRIT Algorithm Method for Frequency Scanning LWA, *IEEE Communications Letters,* Vol. 24, No. 7, 2020, pp. 1441-1445.

[38]. M. Li and Y. Lu, Source Bearing and Steering-Vector Estimation using Partially Calibrated Arrays, *IEEE Transactions on Aerospace and Electronic Systems,* Vol. 45, No. 4, 2009, pp. 1361-1372.

[39]. X. T. Meng, F. G. Yan, S. Liu, Y. Zhang, and M. Jin, Real-Valued DOA Estimation for Non-Circular Sources via Reduced-Order Polynomial Rooting, *IEEE Access,* Vol. 7, 2019, pp. 158892-158903.

[40]. N. Ahmed ***et al.***, Performance Analysis of Efficient Computing Techniques for Direction of Arrival Estimation of Underwater Multi Targets, *IEEE Access,* Vol. 9, 2021, pp. 33284-33298.

[41]. A. Farina, F. Gini, M. V. Greco, and P. Lombardo, Coherent radar detection of targets against a combination of K-distributed and Gaussian clutter, in

Proceedings of the International Radar Conference, 8-11 May 1995, pp. 83-88.

[42]. J. Han and M. R. Inggs, Cumulative detection probability of stationary and moving targets by MIMO radar, in *Proceedings of the 2011 IEEE RadarCon Conference (RADAR)*, 23-27 May 2011, pp. 974-978.

[43]. Y. Yang, S. P. Xiao, X. S. Wang, and Y. Z. Li, Non-Coherent Radar Detection Probability for Correlated Gamma Fluctuating Targets in K Distributed Clutter, *IEEE Access*, Vol. 6, 2018, pp. 3824-3827.
