IEICE Communications Express, Vol.13, No.12, 470–474

ComEX

Special Cluster in Conjunction with IEICE General Conference 2024

LETTER

Comparison of learning models for wideband interference mitigation in automotive chirp sequence radar systems

Yudai Suzuki[1], **Xiaoyan Wang**[1, a)], **and Masahiro Umehira**[2]

Abstract CS (Chirp Sequence) radar plays a crucial role in the safety of autonomous driving. However, its widespread adoption increases the probability of wideband inter-radar interference, leading to miss-detection of targets. To address this problem, we utilize RNN (Recurrent Neural Network) and self-attention models to mitigate wideband interference in automotive radar systems and compare 12 different learning models in terms of SNR (Signal-to-Noise Ratio) and processing time.

Keywords: CS radar, RNN models, self-attention, SNR, processing time
Classification: Wireless communication technologies

1. Introduction

To address traffic accidents and congestions, the demand for autonomous driving is increasing. To accurately perceive the surrounding environment, autonomous vehicles conventionally are equipped with three types of sensors: cameras, LiDAR (Light Detection And Ranging), and millimeter-wave radar. Among these, millimeter-wave CS (Chirp Sequence) radar [1] is particularly advantageous due to its robustness against nighttime and adverse weather conditions, as well as its cost-effectiveness.

With the widespread use of radar, inter-radar interference has become a significant issue. Inter radar interference [2] can be categorized into two types: wideband interference, which leads to missed detections due to increased noise levels; and narrowband interference, which causes false detections due to ghost targets. In this study, we focus on wideband interference, which has a higher probability of occurrence.

Various threshold-based interference mitigation algorithms [3] have been proposed, however, they struggle to adapt to the diverse scenarios of wideband interference, especially when the number of interfering radars is large, and the interference duration is long. To this end, threshold-free learning-based interference mitigation methods, which can adapt to diverse environments and exhibit high performance, have gained significant attention in recent years [4, 5]. In this paper, we utilize RNN (Recurrent Neural Network) [6] and self-attention models to mitigate wideband inter-radar interference and evaluate the performance and processing time of 12 different learning models through simulations.

[1] Graduate School of Science and Engineering, Ibaraki University, 4-12-1 Nakanarusawa, Hitati, Ibaraki, 316-8511 Japan
[2] Dept. of Electronics and Communication Technology, Nanzan University, 18 Yamasato, Showa-ku Nagoya, 466-8637, Japan
[a)] xiaoyan.wang.shawn@vc.ibaraki.ac.jp (Corresponding author)

DOI: 10.23919/comex.2024COL0013
Received June 13, 2024
Accepted July 23, 2024
Publicized August 9, 2024
Copyedited December 1, 2024

2. Interference in CS radar

2.1 Principle of CS radar

The block diagram of the CS radar is shown in Fig. 1. The CS radar transmits a signal that is frequency-modulated in a sawtooth pattern and receives the reflected signal from the targets. The transmitted signal is given by Eq. (1).

$$f(t) = f_c + \frac{\Delta f}{\Delta T} t, \tag{1}$$

where f_0 is carrier frequency, Δf is sweep bandwidth, and ΔT is sweep period. The transmitted signal and the reflected signal from the target are mixed at the mixer and passed through a Low Pass Filter (LPF) to obtain the beat signal. The beat signal could be represented by Eq. (2).

$$f_B = \frac{2R\Delta f}{c\Delta T} + \frac{2v f_0}{c}, \tag{2}$$

where R is target's relative range, v is target's relative velocity, and c is the speed of light. Subsequently, the beat signal undergoes a range-FFT (Fast Fourier Transform) to derive the target's range information, and then a Doppler-FFT to extract the target's velocity information.

2.2 Wideband interference

When radar signals from other vehicles are received, inter-radar interference can occur. Figure 2 illustrates wideband interference, which arises when the chirp rate of the interference signal differs from that of the victim radar's signal. When the beat signal of the interference radar falls below the LPF bandwidth, a pulse-like signal appears in the time domain. Whereas in the frequency domain, it leads to an elevation in the noise level across the entire spectrum. This heightened noise level can result in the failure to detect targets.

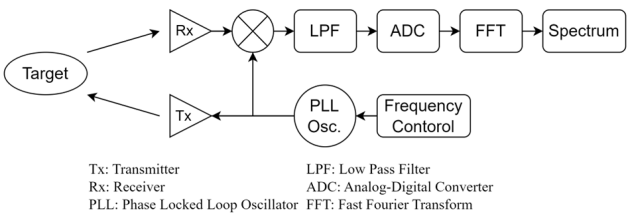

Fig. 1 Block diagram of CS radar

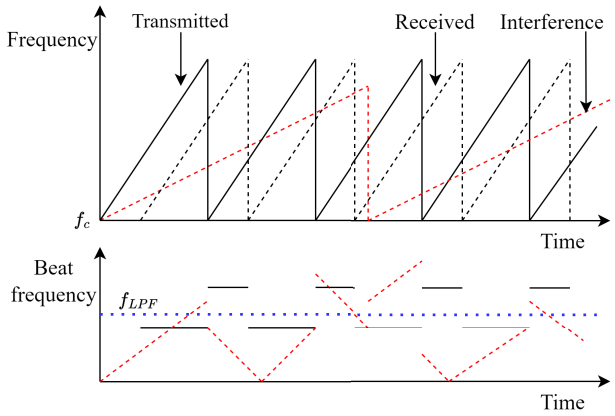

Fig. 2 Wideband interference of CS radar

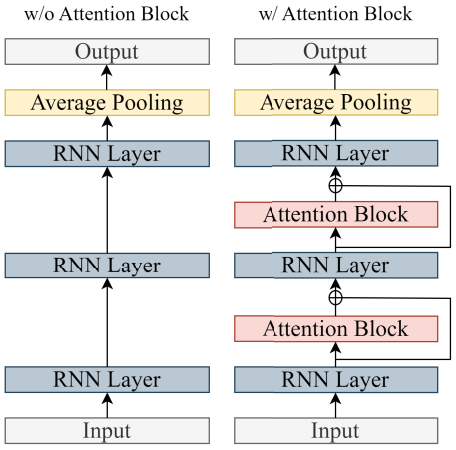

Fig. 3 The architecture of RNN and self-attention based wideband interference mitigation methods

3. Interference mitigation method

In this work, we utilize RNN and self-attention models to mitigate inter-radar interference. The input x_t of the model is the beat signal with interference, and the label $\widehat{y_t}$ is the beat signal without interference. The loss function L is defined as the (MSE) Mean Squared Error between label $\widehat{y_t}$ and output y_t as Eq. (3).

$$L = \frac{1}{N}\sum_{1}^{N}(\widehat{y_t} - y_t)^2 \tag{3}$$

where N is the number of samples.

Figure 3 illustrates the architecture of the RNN-based interference mitigation methods. Specifically, RNN processes the data sequentially thus considering the data's order. On the other hand, self-attention processes the entire input sequence simultaneously, accounting for the significance of each sample. For the model without attention block, 3 layers of RNNs are stacked. When the attention block with skip connections is employed, it is inserted between the RNN layers, resulting in a total of 5 layers. The output of the final RNN layer undergoes average pooling to generate the model's output which has the same size of the input. In this work, we employ 3 widely used RNN models, i.e., VanillaRNN, GRU (Gated Recurrent Units), and LSTM (Long-Short Term Memory), and their respective bidirectional versions, i.e., BiRNN, BiGRU, and BiLSTM.

3.1 VanillaRNN

VanillaRNN is the most basic RNN model. The forward propagation of VanillaRNN is given by following equation.

$$y_t = h_t = \tanh\left(Wx_t + Uh_{t-1} + b\right), \tag{4}$$

where x_t represents input at time step t, y_t represents output at time step t, h_{t-1} and h_t represent hidden state at time step $t-1$ and t respectively, W, U and b represent weight matrix and bias vector. By incorporating the hidden state h_{t-1} in the calculation of the output y_t, the model conducts inference that accounts for the time series nature of the data.

3.2 GRU

To regulate the hidden state h_{t-1} and the input x_t, GRU uses stochastic gates throughout its processing. The forward propagation of GRU is given by following equations.

$$r_t = \sigma\left(W_r x_t + U_r h_{t-1} + b_r\right) \tag{5}$$

$$z_t = \sigma\left(W_z x + U_z h_{t-1} + b_z\right) \tag{6}$$

$$\widehat{h_t} = \tanh\left(W_h x + U_h(r_t \odot h_{t-1}) + b_h\right) \tag{7}$$

$$y_t = h_t = (1 - z_t) \odot \widehat{h_t} + z_t \odot x_t \tag{8}$$

where r_t and z_t are reset gate and update gate respectively, $\widehat{h_t}$ is the candidate activation, and the symbol $\odot$ is Hadamard product.

3.3 LSTM

LSTM employs three stochastic gates to capture long-range dependencies in sequential data. The forward propagation of LSTM is given by following equations.

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{9}$$

$$i_t = \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right) \tag{10}$$

$$o_t = \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right) \tag{11}$$

$$\widetilde{c_t} = \tanh\left(W_c x_t + U_c h_{t-1} + b_c\right) \tag{12}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c_t} \tag{13}$$

$$h_t = y_t = o_t \odot \tanh(c_t) \tag{14}$$

where f_t, i_t, o_t are respectively known as forget gate, input gate, and output gate, $\widetilde{c_t}$ and c_t respectively represent the candidate cell state and update cell state.

3.4 Self-attention

To further capture the relationships between different samples of a sequence, we insert self-attention block into RNN models. Figure 4 illustrates the architecture of self-attention block. In the self-attention, the input $\boldsymbol{x}$, which is the output of the previous RNN layer, is first linearly transformed to query $\boldsymbol{Q}$, key $\boldsymbol{K}$, and value $\boldsymbol{V}$ based on the following equations.

$$\boldsymbol{Q} = \boldsymbol{x}W_Q + \boldsymbol{b}_Q \tag{15}$$

$$\boldsymbol{K} = \boldsymbol{x}W_K + \boldsymbol{b}_K \tag{16}$$

$$\boldsymbol{V} = \boldsymbol{x}W_V + \boldsymbol{b}_V \tag{17}$$

Subsequently, they are used in the calculation of Scaled Dot-Product Attention. Specifically, the dot product of $\boldsymbol{Q}$ and $\boldsymbol{K}$

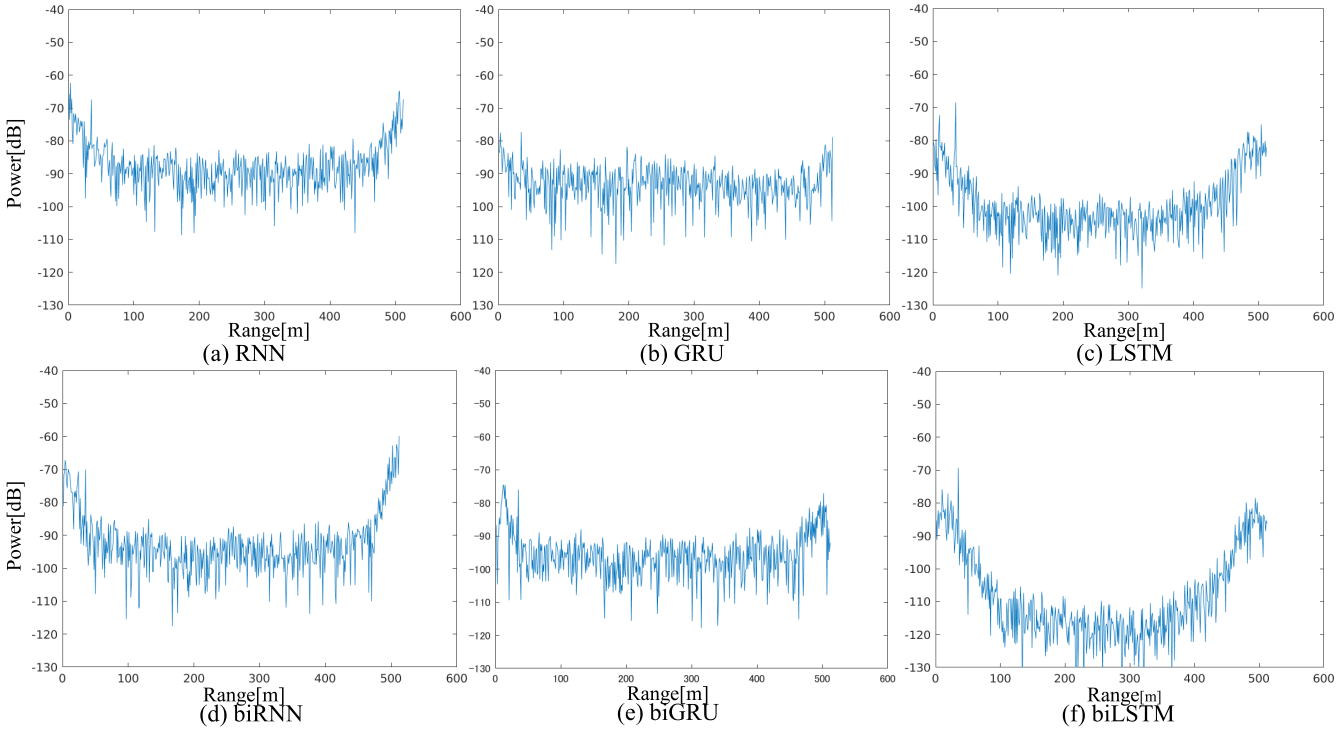

Fig. 6 Frequency spectrum of RNN models

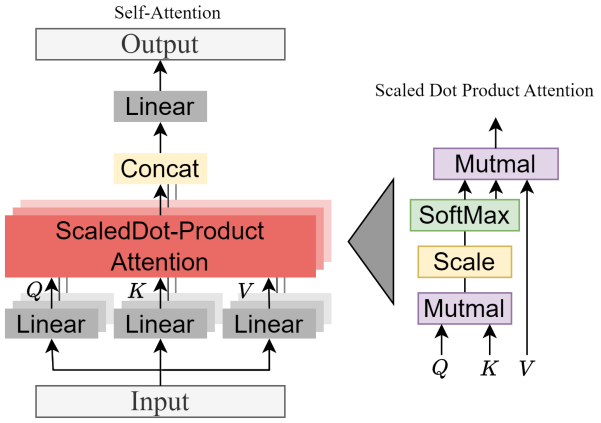

Fig. 4 The architecture of self-attention block

is computed to determine the similarity between the query and each key. This result is then scaled, by which an attention score is obtained. By applying the Softmax function to the attention score, an attention weight is computed, transforming the scores into a probability distribution. The attention weight indicates the importance of information from each time step for a given input at a particular time step. The calculation of Scaled Dot-Product Attention is given by Eq. (18).

$$\text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}}\right)\boldsymbol{V} \qquad (18)$$

where d_k is the dimension of $\boldsymbol{Q}$ and $\boldsymbol{K}$. The output of self-attention is obtained by concatenating the outputs of all heads and passing them through a linear layer to transform them into a matrix with the same shape as the input.

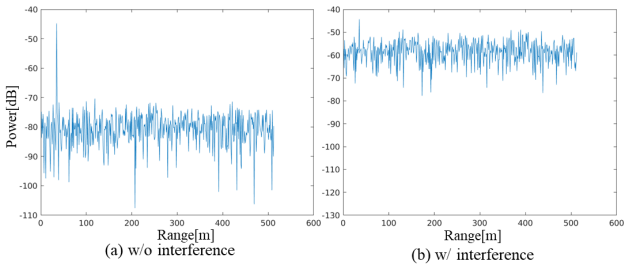

Fig. 5 A frequency spectrum example of test data

4. Simulations and evaluation results

We train the models using randomly generated simulation data and evaluate their performance. Table I shows the radar parameters in the simulations. The training dataset consists of 50 scenarios, each comprising 75 chirps. Among them, 3375 instances are used for learning and 375 instances are for validation. The Adam optimizer was adopted with a learning rate of 0.001 and a batch size of 128 inputs. The training process is ended after 100 epochs. All models are trained using the Nvidia RTX A5500 Laptop GPU.

Firstly, Figs. 5, 6, and 7 depict frequency spectrum examples of test data (without and with interference), the mitigation results of RNN models, and the mitigation results of Attention RNN models, respectively. From Fig. 6, we observe that bidirectional RNNs generally exhibit higher SNR than un-directional RNNs, with LSTM and biLSTM outperforming other models. From Fig. 8, it is evident that the performance could be greatly enhanced by Attention LSTM, Attention biGRU, and Attention biLSTM, but degraded in the Attention RNN and Attention biRNN, by which the peaks cannot be detected at all.

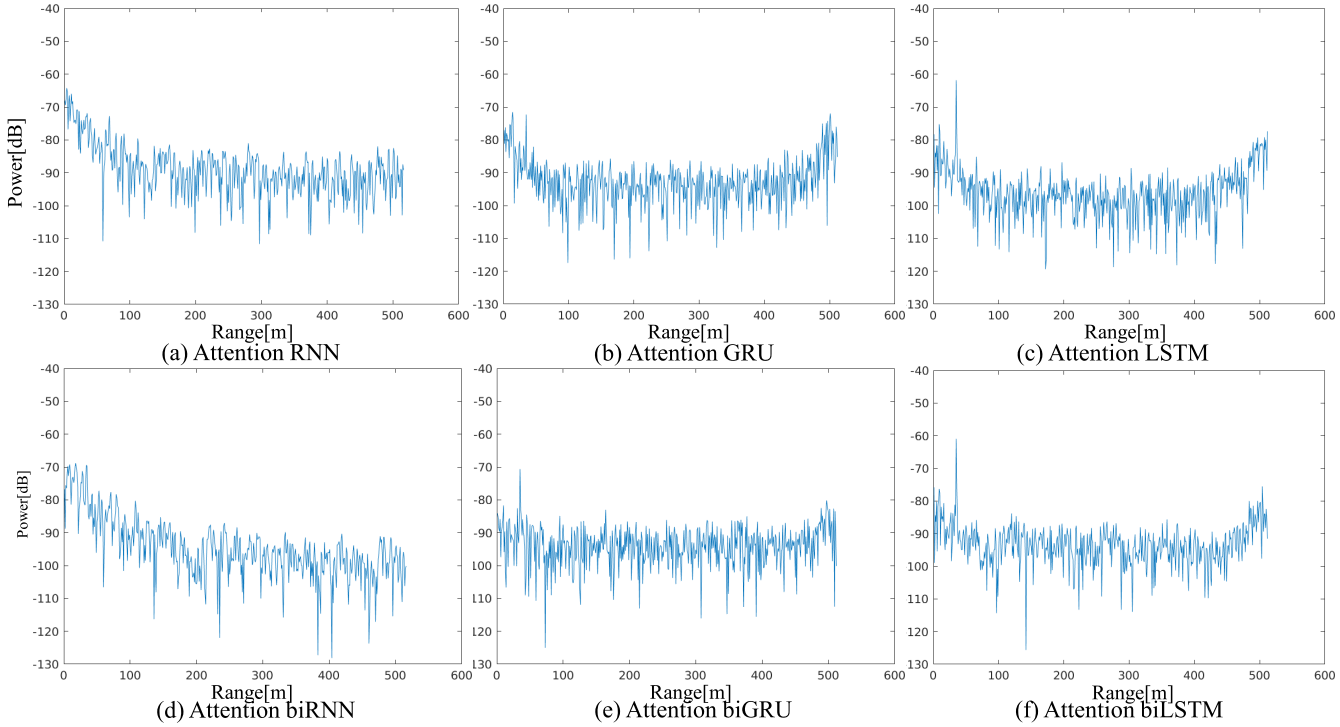

Fig. 7 Frequency spectrum of attention RNN of models

Table I Parameters of radar simulations

Paramaters	Range
Center frequency	76~78GHz
Distance	1~130m
Velocity	1~50km/h
Chirp period	20~40μs
Sweep bandwidth	100~200MHz
Number of targets	1
Number of interferences	1~4

Table II Result of RNN models

	w/ int	RNN	GRU	LSTM
SNR[dB]	16.25	20.5	21.1	21.7
time w/ CPU[ms]		20.9	222	205
time w/ GPU[ms]		8.69	17.9	19.2
	w/o int	biRNN	biGRU	biLSTM
SINR[dB]	29.5	22.3	23.07	23.8
time w/ CPU[ms]		89.7	526	432
time w/ GPU[ms]		17.4	32.3	31.1

Table III Result of attention RNN models

w/ Attention Block				
	GRU	biGRU	LSTM	biLSTM
SNR[dB]	21.0	29.1	22.7	27.0
time w/ CPU[ms]	339	647	284	552
time w/ GPU[ms]	23.9	50.1	22.0	42.9

Subsequently, we evaluated the mitigation performance of these models in terms of average SNR and processing time. The processing time was measured on both intel core-i9 12900H CPU and Nvidia RTX A5500 Laptop GPU. Tables II and III present the SNR and processing time results averaged across 10 testing scenarios. Among the RNN models, bidirectional architectures typically offer an SNR increase of approximately 2 dB, with biLSTM exhibits the best performance. Incorporating the attention block, yields a significant enhancement in SNR for both biGRU and biLSTM models. Particularly, the attention biGRU achieves the highest SNR performance at 29.1 dB, closely approaching the SNR of clean data, which is 29.5 dB.

Finally, we examine the processing time on both CPU and GPU. On the CPU, the processing time increases in the order of VanillaRNN, LSTM, and GRU. Additionally, bidirectional models generally have much longer processing time than un-directional ones. For all models, the processing time on GPU is significantly faster than on the CPU. Notably, the attention biGRU demonstrates the best SNR performance, but also has the longest processing time among all architectures.

5. Conclusion

In this work, we employ RNN and self-attention models for wideband interference mitigation and evaluate the effectiveness and processing time of 12 different models by extensive simulations.

Acknowledgments

This research and development work was supported by the MIC/SCOPE ## JP225003006.

References

[1] C. Waldschmidt, J. Hasch, and W. Menzel, "Automotive radar — From first efforts to future systems," *IEEE J. Microw.*, vol. 1, no. 1, pp. 135–148, Jan. 2021. DOI: 10.1109/jmw.2020.3033616

[2] M. Umehira, Y. Watanabe, X. Wang, S. Takeda, and H. Kuroda, "Inter radar interference in automotive FMCW radars and its mitigation challenges," 2020 IEEE International Symposium on Radio-Frequency Integration Technology (RFIT), pp. 220–222, 2020. DOI: 10.1109/rfit49453.2020.9226222

[3] M. Umehira, T. Okuda, X. Wang, S. Takeda, and H. Kuroda, "An adaptive interference detection and suppression scheme using iterative processing for automotive FMCW radars," Proc. IEEE Radar Conference, pp.1–5, 21-25 Sept. 2020. DOI: 10.1109/radarconf2043947.2020.9266712

[4] J. Mun, S. Ha, and J. Lee, "Automotive radar signal interference mitigation using RNN with self attention," Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), pp. 3802–3806, May 2020. DOI: 10.1109/icassp40776.2020.9053013

[5] R. Koizumi, X. Wang, M. Umehira, R. Sun, and S. Takeda, "Experimental evaluations on learning-based inter-radar wideband interference mitigation method," *IEICE Trans. Fundamentals*, vol.E107-A, no.8, pp.1255–1264, 2024. DOI: 10.1587/transfun.2023eap1122

[6] Goodfellow, I., Bengio, Y., Courville, A., Deep learning (Vol. 1), Cambridge: MIT Press. 2016. 367-415.