IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received December 27, 2020, accepted January 4, 2021, date of publication January 8, 2021, date of current version January 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3049950

Convolutional Neural Filtering for Intelligent Communications Signal Processing in Harsh Environments

ZHUO SUN, (Member, IEEE), JINGJING LI, AND JINPO FAN

Key Laboratory of Universal Wireless Communications, Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Zhuo Sun (zhuosun@bupt.edu.cn)

This work was supported in part by the China 2020 Industrial Internet Innovation and Development Project under Grant TC200H01L.

ABSTRACT Aiming at utilizing artificial neural networks to enhance intelligent filtering for interfered wireless communication signal in harsh environments, a new method named *convolutional neural filtering* is designed and presented in this paper. This method is based on model-driven deep learning princeple, by analyzing the theoretical connection between the filter model and the convolutional neural layer, it attempts to use one-dimensional convolution kernels to learn a matched or bandpass filter. Moreover, the model introduces a kernel-wise attention mechanism between different convolution kernels to selectively emphasize informative filters. The results show that in terms of interference and noise suppression for received wireless signal, the filtering method has highlighted dynamic adaptability to variation of signals and interference, and it also reveals that the performance is affected by the initialization parameters and the number of convolution kernels. Based on this method an embeddable filtering unit fully based on neural network is provided, which can be easily integrated into a deep learning network targeting such as wireless signal detection and recognition applications, avoiding complex preprocessing for end-to-end wireless signal learning.

INDEX TERMS Linear filter, convolution neural network, neural filtering, model-driven deep learning.

I. INTRODUCTION

For communications in harsh environments, an intelligent physical (PHY) layer is fundamental and inevitable to achieving the envisioned communication requirements. Machine learning at PHY layer holds the potential to perform intelligent signal processing that can offer significant performance enhancements over traditional approaches [1]. Basically, these approaches can be classified into two generic groups: data-driven [2] and model-driven [3]. Among these approaches, many of them are designed by taking radio signals as the input of deep learning network, such as channel decoding, channel estimation and modulation classification [2], [3]. However, due to the dynamics of wireless transmission channels, there are random and burst variations between training samples and test data. These factors may be caused by interference generated by channel dynamics, thermal noise fluctuations, adjacent band interference, etc, which have seriously affected the generalization ability of the deep learning network under actual channel conditions, and even caused a sharp decline in performance.

To cope with the degraded generalization ability of deep learning under actual wireless transmission conditions, conventional solutions are to preprocess sampled signal, such as channel equalization and interference suppression by through adaptive filtering methods. However, the idea of filtering preprocessing has disadvantages in two aspects. First of all, these preprocessing before deep learning models are often complicated and rely on manual labor, making it impossible to support time-critical applications, and also destroying the global optimization characteristics of end-to-end learning. Secondly, to reduce the distribution difference between training samples and real samples, designing of filter in the above preprocessing requires a priori or feedback of certain channel influencing factors, otherwise the performance of the deep learning model cannot be guaranteed.

Therefore, the challenge is that, can we resort to the ability of deep learning to learn by itself an adaptive filtering method

The associate editor coordinating the review of this manuscript and approving it for publication was Qilian Liang.

from the radio signals? The learned filter also is expected to automatically track the target signal and improve the signal to interference noise ratio, in other words, we expect that the raw signal is fed into a neural layer to extract the band of interest without preprocessing.

The idea of "*neural filtering*" is most likely the solution which presented as early as the 1990s [4], [5], while it has been recently applied and verified in deep learning applications outside of communication signals [6]–[8]. The work of this paper will continue this way of neural filtering, but put our focus on exploring the connection between communication signal filtering and convolutional neural layer. Our purpose is to establish a neural filtering unit through model-driven way, which is adaptive to carrier frequency and amplitude of signal, even to interference change.

First, in order to verify whether the convolutional layer can independently learn a filter that eliminates out-of-band interference, we construct an end-to-end learning network, which is composed by a convolutional layer, a fully connected layer and an activation function. The result is exactly the same as we expected, and an effective band filter is obtained. Second, in order to further optimize the effectiveness of the filter in the convolutional layer, we introduced an attention mechanism on the convolution kernels based on its contribution to the filtering performance, and finally proposed a convolutional neural filtering unit that combines the channel attention mechanism. We also proved that the unit has excellent adaptability to changes in the signal-to-noise ratio, interference frequency, and channel bandwidth (at least not lower than the traditional filter method). It is worth noting that what we propose is not a dedicated deep learning application network, but a neural processing unit that completes the adaptive filtering function. This unit can be arbitrarily embedded in the end-to-end wireless communication deep learning network, to improve the generalization ability of deep learning in communication applications under actual wireless propagation conditions.

II. RELATED WORKS

Generally, there are two routes of machine learning approaches at PHY layer: data-driven and model-driven approaches. As a representative of full data-driven approach, end-to-end learning communications [2] has recently received widespread attention, because it gets rid of the process of hand-crafted and explicit feature selection. Compared with the traditional method based on feature engineering, it has achieved performance great improvement. Besides, model-driven deep learning on physical layer communications design have emerged in recent years [9], [10], in which the network is constructed based on known physical mechanisms and domain knowledge for achieving intelligent communications. The model-driven deep learning are proved requiring less training data and training time because of not heavily depending on the huge volume of labeled data.

Neural filtering was proposed to use a fully connected layer of neural network to approximate IIR/FIR filters with a nonlinear method [4], [5]. In [11], The author uses variable time delay neural network to demodulate pulse amplitude modulation(PAM) signal, and proved be able to learn feature detection equivalent to a matched filter or equalizing filter, depending on the modulation pulse shape. Different from the neural structures used in these works, herein we want to discover the capability of conventional neural layer other than feed-forward neural network.

Moreover, as a popular structure of artificial neural networks, the convolutional neural layer is naturally consistent with the filter in signal processing in terms of mathematical model representation. This has led to some research in deep learning applications outside of communication signals, such as audio or electroencephalo-graph(EEG) signals, to learn filtering for signal enhancement. In [12], where the raw EEG signal is fed into a convolutional layer to extract the band of interest without preprocessing, followed by a feedforward convolutional neural network(CNN) model or recurrent neural network(RNN) model for epileptic spike and non-spike classification. Bell *et al.* [13] train a convolutional neural network directly on raw acoustic-phonetic continuous speech corpus waveforms, and show that the network tends to learn matching filters when trained to do phone classification. In [14], the 2-D spectro-temporal modulation filters learned from the convolutional variational autoencoder(CVAE) model in an unsupervised fashion are used to process the speech spectrogram for deriving robust spectrogram representations.

The above research results confirm the feasibility of the convolutional neural network structure in signal equalization. In the following work, we will innovatively use the linear modeling theoretical model to supervise our utilization of the convolutional neural layer on a different target, that is, the communication signal, to realize the process of intelligent filtering.

III. FROM LINEAR FILTERING TO CONVOLUTIONAL NEURAL LAYER

Convolutional neural network is a multi-layer perceptron specially designed for recognizing two-dimensional shapes. This network structure is highly invariant to translation, scaling, tilt, or other forms of deformation, but it is also suitable for speech and text recognition applications. Some well-known improved CNN models were proposed. In the convolutional neural network, the convolutional layer is the core unit. Assuming that the input layer is the (l-1)th layer, its input characteristics are $X^{(l-1)} \in R^{m\times m}$,the corresponding convolution kernel $K^{(l)} \in R^{n\times n}$, the output of the convolution layer $Z^{(l)} \in R^{(m-n+1)\times(m-n+1)}$, and each output plus a bias unit $B^{(l-1)} \in R^{1\times n}$, then the mathematical process completed by the convolution layer can be expressed as:

$$z_{u,v}^{(l)} = \sum\nolimits_{i=1}^{n} \sum\nolimits_{j=1}^{n} x_{i+u,j+v}^{(l-1)} k_{i,j}^{(l)} + b^{(l)} \tag{1}$$

When the input is a one-dimensional sequence in the time domain, such as in end-to-end learning of speech or communication signals, the mathematical expression of the convolutional neural layer is:

$$z_u^{(l)} = \sum\nolimits_{i=1}^{n} x_{i+u}^{(l-1)} k_i^{(l)} + b^{(l)} \tag{2}$$

where the weights $k_i^{(l)}$ specify the convolution filter. It can be seen that when the bias is removed, the process of the convolutional neural layer in the above-mentioned one-dimensional input case is a form of finite convolution sum. If the weight of the convolution kernel is set as the filter time-domain impulse response, the coefficient is equivalent to the bandpass filter of length N.

If the coefficient of the convolutional neural layer is X, it happens to form a matched filtering process:

$$z_u^{(l)} = \sum\nolimits_{i=1}^{n} x_{i+u}^{(l-1)} x_i^{(l-1)} \tag{3}$$

From theoretical point of view, each kernel of the convolutional neural layer may learn to become a bandpass filter or a matched filter. In addition, the weight parameters in the convolutional neural layer are obtained through training, which is based on the backward propagation(BP) algorithm, the key is to solve the loss function minimization problem to perform parameter estimation and update the weight according to the obtained gradient. This is completely consistent with the idea of obtaining filter weighting coefficients by minimizing the average error method in the least mean square(LMS) or normalized LMS adaptive filter. The possible difference is that when the error is propagated by BP, the error that the convolutional neural layer may get is the result of nonlinear processing, such as the error output of the pooling layer or Rectified Linear Units(ReLU).

Therefore, it is theoretically feasible to use convolutional neural layers to learn adaptive filters. In addition, since the convolutional neural layer in general deep learning generally has multiple convolution kernels, training is equivalent to obtaining multiple filters which have different filtering characteristics. In some research on speech processing, it is also confirmed that different convolution kernels are equivalent to learning a filter bank of bandpass filter. This also reminds us that the diverse filter can be optimized for specific task. For example, it can improve the adaptive ability to noise and interference changes.

IV. LEARNING FILTERING WITH SINGLE CONVOLUTIONAL LAYER

In order to verify that the convolutional layer of the neural network can also learn an adaptive filter with noise reduction and interference cancelation, we built a simple convolutional neural network around the convolutional layer(Fig.1). This is a simple signal detection application in cognitive radio. By inputting labeled noise only and noisy binary phase shift keying(BPSK) signals, we expect the model can correctly judge whether the sample contain the signal of interest. Compared with the traditional CNN network, we did not use a pooling layer (because we did not use too many kernels), only with a convolutional layer, activation function, and softmax layer.

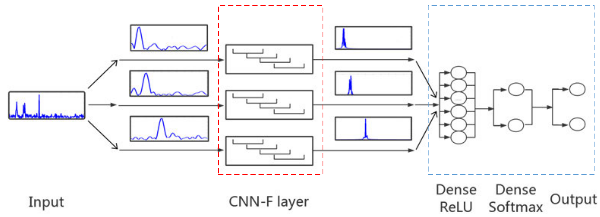

FIGURE 1. Learning filter through a simple convolutional neural network with single convolutional.

Specifically, convolutional layer consists of several filters whose parameters are optimized by the back-propagation algorithm. The output of each filter is passed through a ReLU unit, which is used as activation function. Softmax layer acts as a classifier in the whole network. A cross entropy is used as a loss function to measure the difference between the two signal types, and then adjust the parameters during training to reduce this difference.

When training the network to achieve convergence and the detection accuracy reaches more than 95%, we analyze the current convolutional layer with updated parameters. In order to analyze whether we learned a needed filter from the data, we performed FFT transformation on each convolution kernel to observe the frequency domain characters of the filter. At the same time, for observing the effect of convolution filtering, we directly take out the output of the convolution layer, and analyze it separately from its constellation diagram in time domain.

First of all, we considered an extreme configuration case: the convolutional layer has only one kernel, which is different from the usual CNN applications. Either in image or natural language processing, multiple kennels are often used to extract multiple channels feature. However, here we want to explore if we want to learn a unique feature of matched filter. The result proves that under the constraint of such loss function, the frequency domain characteristics learned by the only convolution kernel (as shown in the Fig. 2), its passband and stopband characteristics are consistent with the features of model-based matched filter we designed exactly for this signal of interest.

We compare the constellation diagrams of BPSK signal(as shown in the Fig. 3) after convolution filtering at -8dB signal to noise ratio(SNR), and it can be clearly seen that the aggregation of the constellation diagrams has been greatly improved after filtering, this result is very meaningful. Although we only design the loss function with signal detection as the application goal, the convolution filter learned by training not only obtains spectral domain features that support signal detection, but also the phase and amplitude characteristics are also retained and even improved, and it is expected to support afterward signal processing.

Based on the above results, we expand the convolution kernel to multiple, expecting to explore the filter characteristics learned by different convolution kernels under

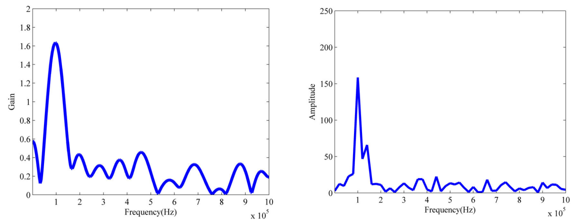

FIGURE 2. **Frequency domain characteristics of Chebyshev bandpass filter(left) and Learned filter(right).**

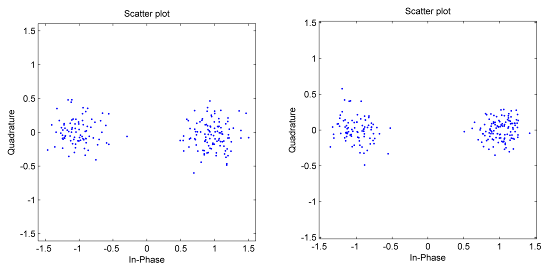

FIGURE 3. **Constellation diagrams of BPSK signal, before(left) and after(right) convolutional neural filtering.**

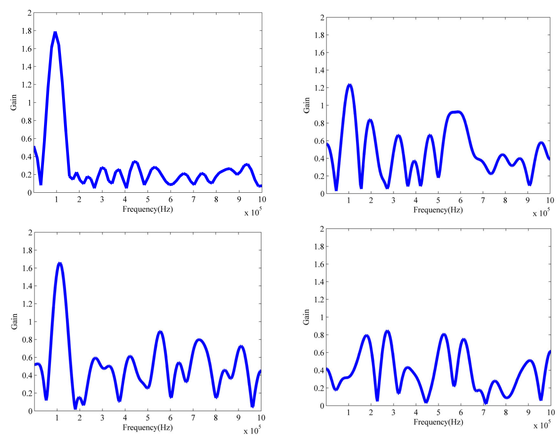

FIGURE 4. **Frequency domain characteristics of 4 learned convolution kernels.**

this condition. As shown in Fig.4, we obtain the frequency domain characteristics of the convolutional layer under the scenario where other parameters are unchanged and only increased to 4 convolution kernels. It can be seen that the first and third convolution kernels learn the frequency domain filtering and denoising characteristics of the signal of interest well, but the second and fourth learn some other characteristics that we cannot explain. The results of these four sets of convolution kernels are passed to the next layer of network indiscriminately, but it is obvious that the contributions of these four sets of learning results to the final filter denoising goal are various. This reminds us that if the learning of the neural network is not guided, the computing resources are not effectively used, and it is hoped that the resource consumption of the performance improvement brought by multiple convolution kernels will be wasted. Therefore, in the next section, we will try to add the attention mechanism on the convolution

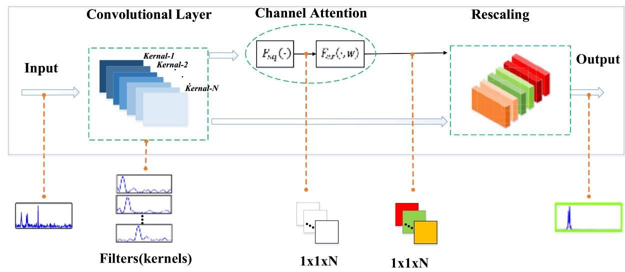

FIGURE 5. **Convolutional Neural Filtering with Channel Attention.**

kernels to help us selectively emphasize informative filters and suppress less useful ones.

V. CONVOLUTIONAL NEURAL FILTER ENHANCED BY ATTENTION MECHANISM

There are many types of attention mechanism under the deep learning framework. For example, Inception Architectures [12] which improve accuracy through embedding multi-scale processes, and attention mechanism of spatial dependence [13]. But here, we need the convolution kernel-wise or channel-wise attention, so we chose squeeze-and-excitation block(abbreviated SEnet) [14] to explicitly model interdependencies between channels to recalibrate filter responses. Based on multiple convolution kernels and convolution kernels, we construct the learning filer unit based on convolutional neural layer with SEnet as the Fig.5:

In the above model, the processing process of the convolutional layer with multiple kernels can be expressed as follows. Let $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_C]$ denote the learned group of filter kernels, where $\mathbf{v}_i$ refers to the parameters of the i-th filter. The outputs of convolution layer can be expressed as $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_C]$, where

$$\mathbf{u}_i = \mathbf{v}_i * \mathbf{X} = \sum\nolimits_{s=1}^{P} v_i^s \times x^s \tag{4}$$

Here $i = 1, 2, \cdots C$ and $(*)$ and $(\times)$ denote convolution and product operator, P is the number of kernel parameters, $\mathbf{v}_i = [v_i^1, v_i^2, \cdots, v_i^P]$ and $\mathbf{X} = [x^1, x^2, \cdots, x^P]$. v_i^s is the *s-th* coefficient of one-dimensional kernel $\mathbf{v}_i$ which acts on the corresponding channel of $\mathbf{X}$.

To obtain the global information of each filter, we perform a squeeze operation on the output of each filter above to obtain a filter descriptor. The squeeze operation has uses global average pooling in the original research. However max-pooling is used in this article, which is expected approximate to the abstraction of the impulse response of the signal filter. A statistic $\mathbf{z}$ is output by shrinking $\mathbf{U}$, the i-th element of $\mathbf{z}$ is achieved by:

$$z_i = F_{sq}(\mathbf{u}_i) = \max(\mathbf{u}_i) \tag{5}$$

To apply the above aggregated information for adaptive recalibration, we also use simple gating mechanism with a sigmoird activation on $\mathbf{z}$.Finally, use the generated scalar to rescaling the filter output:

$$\tilde{x}_{\mathbf{i}} = \mathbf{u}_i \cdot z_i \tag{6}$$

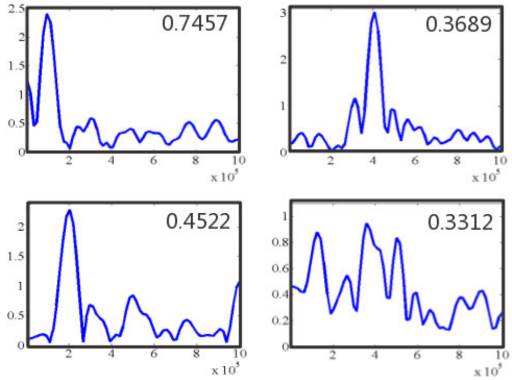

FIGURE 6. Frequency domain characteristics of 4 learned convolution kernels and their scalar value.

Were $\tilde{\mathbf{X}} = [\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_C]$, which is channel-wise multiplication between scalar and feature map.

When training the above-mentioned convolution filter unit with the attention mechanism, in the same way we added two fully connected layers after the entire unit. The goal of network training is to distinguish pure noise from noisiy signal. The signal is a BPSK signal with a carrier frequency of 100Khz. For analyzing what the convolutional layer and attention mechanism have learned, we extracted the convolution kernel coefficients and its scalar generated by the attention mechanism from the trained model, and combined the scalar and the frequency characteristic of the corresponding convolution kernel are put together. Fig.6 and Fig.7 show the results when the number of convolution kernels is 4 and 16 respectively. It can be seen that in both cases, the amplitude-frequency characteristics of the convolution kernel with the maximum scalar value are closest to the signal of interest. We infer that the convolutional layer based on the attention mechanism has learned a matched filter of the signal in the signal detection scenario. In the following experiments in section VI, when the training sample includes out-of-band interference, the learned filter tends to be a bandpass filter. In addition, it is obvious that when the convolution kernel is 16 than the convolution kernel is 4, the learned convolution filter has significantly better suppression of noise. However, does the more convolution kernels bring better performance? We will analyze this issue in next section.

In addition to analyzing the amplitude-frequency characteristics of the convolution kernel coefficients, we compared the constellation diagrams of the signals before and after filtering (take the received BPSK signal at SNR 0dB as an example) to explore whether the filtered signal retains the modulation symbol information to the best extent. From the constellation diagram shown in Fig.8, it can be seen that the noise impact after filtering is significantly reduced, which is greatly improved compared to the effect of only using the convolution unit in the previous section. In addition, we compared the time-domain waveforms of the same signal segment before and after filtering.

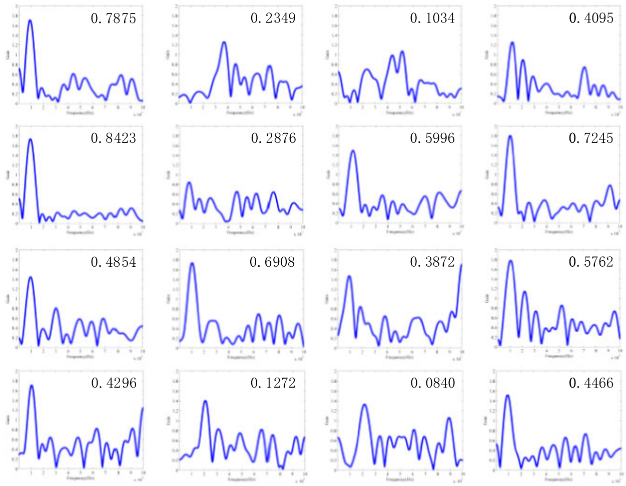

FIGURE 7. Frequency domain characteristics of 16 learned convolution kernels and their scalar value.

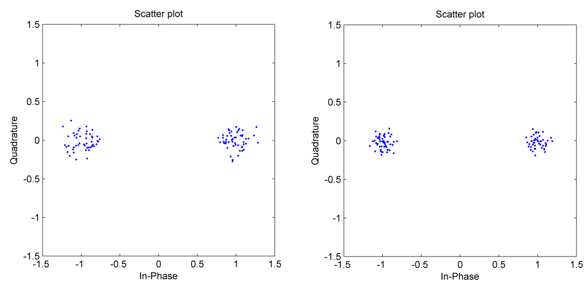

FIGURE 8. Constellation diagrams of BPSK signal(SNR=0dB), before(a) and after(b) convolutional neural filtering.

VI. PERFORMANCE ANALYSIS, OPTIMIZATION AND APPLICATION

A. ADAPTIVITY TO SIGNAL VARIATION

Firstly, considering that in practical applications, the bandwidth, frequency offset and other parameters of the signal of interest may change. We hope that the above learning model has sufficient adaptability to these changes, that is, the convolutional neural filter can still learn from the data the corresponding filter without relying on a priori for parameter adjustment. Such performance is very useful for spectrum sensing applications, which has sufficient adaptive and identification capabilities for specific signals.

We first consider the scenario where the effective bandwidth of the signal changes. We still use signal detection as the learning task. We use two types of training sets. In both data sets, the signals to be identified are BPSK signals with a carrier frequency of 100Khz. However the bandwidth of the former type of BPSK signal is 80Hz, the latter type of signal bandwidth is 20Hz. Fig.8 shows the frequency domain characteristics of the convolution filter obtained through the same network and training conditions. By measuring its 3dB bandwidth, we can find that the learned filter is well matched to the bandwidth change of the signal.

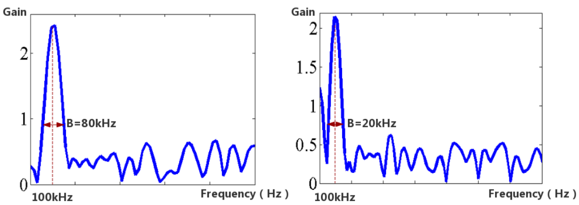

FIGURE 9. Frequency domain characteristics of learned filter for signal with different bandwith.

Second, we considered the scenario when the carrier frequency of the signal of interest changes. Similarly, using signal detection as the learning task. The only difference between the two types of signals is that the carrier frequencies are 100Khz and 200Khz respectively. It shows the frequency domain characteristics of the convolution filter learned under the unchanged condition of network and training. It can be clearly seen that the filter has accurately track signal carrier frequency.

It can be seen from the above results that the convolutional neural filter with attention mechanism has good learning and adaptability for target signals with different parameters. Compared with traditional adaptive filter signal processing, the advantage of our proposed convolutional neural filter is that when the filtered target signal has bandwidth and carrier frequence variation, it does not need to be manually updated prior to the signal change.

B. ROBUSTNESS TO DYNAMIC INTERFERENCES

In complex spectrum sensing scenarios, there are often unknown and variable interferences in the monitoring bandwidth where the target signal is located, such as unlicensed signals, single-frequency interference, etc. These interferences often need to be suppressed by effective bandpass filters. In order to verify the robustness of our proposed convolutional neural filter for out-of-band dynamic interference suppression, we conducted corresponding experimental verification.

It is worth noting that in order to explore the robustness of the convolutional neural filter, especially to changes in the signal-to-noise ratio and interference, we only input signals under the influence of Gaussian white noise during training, while the test signals are subject to interference of out-of-band signals and single-frequency signals of different carrier frequencies. At the same time, we also set the signal-to-noise ratio of the test signal to be different from that during training. Specifically, in the learning task of signal detection, the training signal is a BPSK signal, with carrier frequency of 100Khz and signal-to-noise ratio of -2dB, and noise with the same power spectral density. During the test, we respectively showed the frequency domain characteristics of the filter learned under the conditions of three different interference scenarios.

The spectral characteristics of the filtered signal, as shown in Fig.10. In the first test scenario, we first keep the signal-to-noise ratio between the test signal and the training signal unchanged, and only add an out-of-band BPSK interference signal to the test signal with a carrier frequency interval of 200Khz. Obviously, it can be seen from the filtered signal and the learned filter amplitude-frequency characteristics that it is ideal for suppression of interference BPSK signal and noise. In the second scenario, we increased the signal-to-noise ratio of the test signal to 6dB, but in addition to the out-of-band BPSK interference signal, we also added a strong single-frequency interference whose instantaneous amplitude was even greater than target signal. Even in this complex interference situation, the whole sidelobe decay is very ideal, except that the suppression of single-frequency interference is worse than the BPSK signal that is farther from the carrier frequency. The third scenario is more challenging. In addition to the signal-to-noise ratio will be -6dB, two BPSK signals are placed outside the test signal band. The carrier frequency of the interference BPSK signal is only 100Khz from the target signal, and the average amplitude is higher than the target signal. It can be seen that although compared with the previous two scenarios, mainlobe power are somewhat leaky, the overall interference and noise are still much reduced compared to before filtering. We also show the frequency domain characteristics of the filter learned under this condition, and obviously the bandpass characteristics are still ideal.

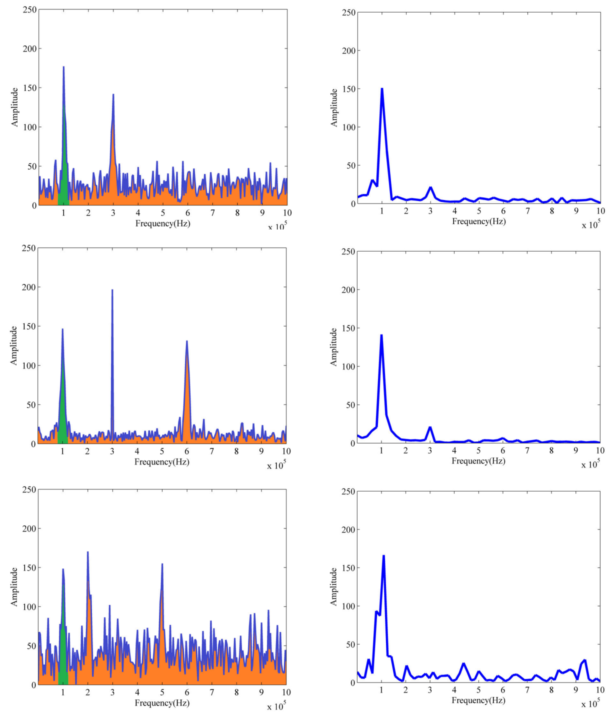

FIGURE 10. Three test scenarios from top to down, signal before (left) and after filtering(right).

C. OPTIMIZATION OF PARAMETERS

1) FILTER QUANTITY

In the convolution filter layer composed of multiple convolution kernels, in fact, each convolution kernel learns a

TABLE 1. Average accuracy in different models.

Model	Accuracy [A] (%)
Baseline	92.92
Conv-based, 1kernel	93.40
Conv-based ,4kernel	94.49
Conv-based, 16kernel	94.26
Conv-SE, 4kernel	95.50
Conv-SE, 16kernel	87.33

[A]Average over SNR from -14dB to 2dB

certain characteristic filter. But, is it better to have more convolution kernels, especially for simple applications like signal perception? To verify this assumption, we tested the noise signal from -14dB to 2dB, and compared the 5 cases of convolution filtering with 1 core, 4 cores, and 16 cores, and 4 cores and 16 cores with attention. It can be seen from the results in Table 1 that the performance of the model trained in this application scenario has increased by about 1% from 1 core to 4 cores, but the performance has dropped somewhat from 4 cores to 16 cores. This is an interesting result, which shows that in the use of deep learning for filter learning, due to the diverse nature of neural network learning, it is not that the more convolution kernels, the better the application performance, and there is likely a ceiling threshold.

2) FILTER QUANTITY

In the previous filter learning process, we randomly initialize the convolution kernel coefficients. However, in the deep learning of speech signals, it has been verified that the initialization filter kernel has a certain influence on the learning results. Therefore, here we also want to verify whether different initializations have an effect on the results. We selected the designed Chebyshev filter coefficients to initialize the convolution kernel coefficients, and used the above-mentioned signal perception accuracy as the evaluation standard. In practice, we test both two models of the single convolution filter layer and it combined with the attention mechanism. As a result in Table 2, we do not see the variation of detection performance under different initialization conditions. This also shows that under the two initialization conditions, the finally learned filter characteristics should be consistent. However, a significant change is that time consumption of training process with coefficients initialized Chebyshev filter coefficients are much shorter than that under the condition of random initialization.

D. DEMONSTRATION OF APPLICATION

We give a network construction method of signal detection applications with the corresponding embedded convolutional neural filter unit. It demonstrats how the proposed convolutional neural filter can be embedded in the deep learning application network, and verifies the performance improvement for signal deep learning tasks. In the same time, we compare it with the original performance without neural filter unit is added. In particular, we also show the two training methods of the proposed convolutional neural filter unit, that is, the application network is jointly trained, and the convolution filter unit and application network are trained separately.

TABLE 2. Average accuracy in different models.

Model	Accuracy [A] (%)	Time Consumption [B] (s)
CNN-DNN 1kernel Rand Init	93.44	12
CNN-DNN 4kernel Rand Init	94.49	19
CNN-DNN 1kernel Cheb Init	93.45	11
CNN-DNN 4kernel Cheb Init	94.46	16

[A]Average over SNR from -14dB to 2dB
[B] On a platform with an Intel i7 processor

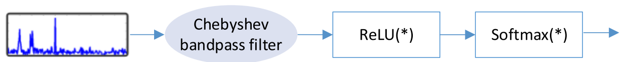

FIGURE 11. A baseline model: Chebyshev bandpass filter as feature extractor and DNN classifier.

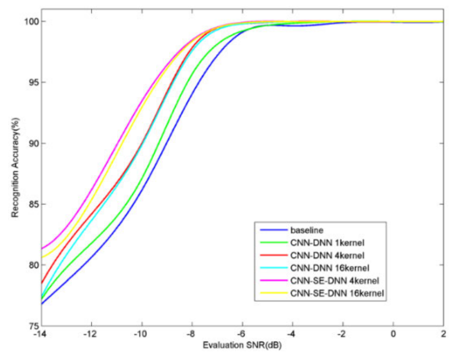

FIGURE 12. Performance of detection accuracy.

We built corresponding signal detection networks on the convolutional neural filter unit with the attention mechanism and without it respectively as shown in Fig.11. A similar fully connected layer is added after the convolutional neural filter with attention, its input is the filtered signal, and the output is the result of whether the signal is detected.

For evaluation the performance of traditional preprocessing methods, we also constructed a baseline model. Traditional recognition task is to extract signal features manually then do classification. We choose the Chebyshev bandpass filter as feature extractor whose efficients are designed using Matlab tool, The extracted results are sent into a DNN classifier. The left boundary of the passband is 80kHz, the right boundary of the passband is 120kHz, the attenuation cutoff is 60kHz to the left boundary, the attenuation cutoff is 140kHz, the attenuation of the sideband is 0.1, the attenuation of the cutoff is 20, and the sampling frequency is 1MHz.

The data to be tested includes two types, one is white noise plus a single-frequency interference and a BPSK interference signal, and the other is the BPSK signal of interest superimposed on the above noise and interference. The BPSK signal of interest does not overlap with single-frequency interference and interference signals in the frequency domain. We have calculated the detection accuracy of convolutional neural filtering with or without attention mechanism under the baseline model and different core configurations under different signal-to-noise ratios (from −14dB to 2dB). It can be seen from the results in Fig.12 that the signal-to-noise ratio is below −5dB, and the convolutional neural filter and its attention enhancement version are better than the baseline model, especially the best performance is the model that combines the attention mechanism and 4 convolution kernels., It is 5% more accurate than the baseline model. However, as the signal-to-noise ratio increases, the accuracy of each model basically tends to be the same level.

VII. CONCLUSION

Based on the analysis of the theoretical connection between the filter model and the convolutional neural layer, this article leverages the convolutional neural structure to learn signal filtering, which follows a model-driven route for combining known filtering mechanisms. We analyzed the performance of convolutional neural filtering from the number of convolution kernels, parameter initialization, attention mechanism, *etc*. The results show that the learned filter has better dynamic adaptability to signals and interference. However, it should be pointed out that the method used in this article is still composed by offline training combined with online applications, which is difficult for some signal processing applications that do not have offline training data. Therefore, a research idea that can be considered is that the convolutional neural filter proposed in this paper has the online learning ability, and the training of the neural filter is completed in the application at the same time, which will effectively improve the practicability of the method.

REFERENCES

[1] A. Jagannath, J. Jagannath, and T. Melodia, "Redefining wireless communication for 6G: Signal processing meets deep learning," 2020, *arXiv:2004.10715*. [Online]. Available: http://arxiv.org/abs/2004.10715

[2] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cognit. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017, doi: 10.1109/TCCN.2017.2758370.

[3] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Commun.*, vol. 14, no. 11, pp. 92–111, Nov. 2017, doi: 10.1109/CC.2017.8233654.

[4] M. Sugisaka, "Neural filtering method for stationary signal processes," *Artif. Life Robot.*, vol. 1, no. 4, pp. 165–168, Dec. 1997.

[5] L. Yin, J. Astola, and Y. Neuvo, "A new class of nonlinear filters-neural filters," *IEEE Trans. Signal Process.*, vol. 41, no. 3, pp. 1201–1222, Mar. 1993, doi: 10.1109/78.205724.

[6] K. Fukumori, H. T. T. Nguyen, N. Yoshida, and T. Tanaka, "Fully data-driven convolutional filters with deep learning models for epileptic spike detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Brighton, U.K., May 2019, pp. 2772–2776.

[7] D. Palaz, M. Magimai-Doss, and R. Collobert, "End-to-end acoustic modeling using convolutional neural networks for HMM-based automatic speech recognition," *Speech Commun.*, vol. 108, pp. 15–32, Apr. 2019, doi: 10.1016/j.specom.2019.01.004.

[8] P. Agrawal and S. Ganapathy, "Modulation filter learning using deep variational networks for robust speech recognition," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 2, pp. 244–253, May 2019.

[9] Z. Xu and J. Sun, "Model-driven deep-learning," *Nat. Sci. Rev.*, vol. 5, no. 1, pp. 22–24, Jan. 2018.

[10] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y. Li, and Z. Xu, "Model-driven deep learning for physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 77–83, Oct. 2019, doi: 10.1109/MWC.2019.1800447.

[11] P. Gorday, N. Erdol, and H. Zhuang, "LMS to deep learning: How DSP analysis adds depth to learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Brighton, U.K., May 2019, pp. 7765–7769.

[12] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn. (PMLR)*, vol. 37, 2015, pp. 448–456.

[13] S. Bell, C. L. Zitnick, K. Bala, and R. Girshick, "Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2874–2883.

[14] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020, doi: 10.1109/TPAMI.2019.2913372.

ZHUO SUN (Member, IEEE) was born in Shanxi, China, 1976. He received the B.S. degree from Information Engineering University, Zhengzhou, in 1998, and the Ph.D. degree from the Beijing University of Posts and Telecommunications, Beijing, in 2007.

From 2008 to 2012, he was a Leturer with the Beijing University of Posts and Telecommunications. Since 2013, he has been an Associate Professor with the Key Laboratory of Universal Wireless Communications, Ministry of Education, Beijing University of Posts and Telecommunications. He has authored three books, more than 60 articles, and more than ten inventions. His research interests include wireless communication, wireless signal processing, and machine learning in communication.

JINGJING LI was born in Henan, China, in 1995. She received the B.S. degree in communication engineering from North China Electric Power University, Hebei, China, in 2019. She is currently pursuing the M.S. degree in information and communication engineering with the Beijing University of Posts and Telecommunications.

Her current research interests include machine learning and wireless physical layer security.

JINPO FAN was born in Yucheng, Henan, China, in 1997. He received the B.S. and M.S. degrees from Beijing Electronic Science and Technology Institute, Beijing, China, in 2015 and 2020, respectively, both in communication engineering. He is currently pursuing the Ph.D. degree in information and communication engineering with the Beijing University of Posts and Telecommunications, Beijing.

His research interests include machine learning and signal recognition, multidimensional data processing, and wireless sensor network security.