IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 6 June 2023, accepted 18 July 2023, date of publication 21 July 2023, date of current version 26 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3297724

RESEARCH ARTICLE

Quadratic Residual Multiplicative Filter Neural Networks for Efficient Approximation of Complex Sensor Signals

MUSTAFA UMUT DEMIREZEN, (Senior Member, IEEE)
Department of Data Products, UDemy Inc., San Francisco, CA 94107, USA
e-mail: umut@demirezen.tech

ABSTRACT In this research, we present an innovative Quadratic Residual Multiplicative Filter Neural Network (QRMFNN) to effectively learn extremely complex sensor signals as a low-dimensional regression problem. Based on this novel neural network model, we introduce two enhanced architectures, namely FourierQResNet and GaborQResNet. These networks integrate the benefits of quadratic residual neural networks, multiplicative filter neural networks, and several filters in signal processing to effectively capture complex signal patterns, thereby addressing issues associated with convergence speed, precision, and spectral bias. These architectures indicate effectiveness in reducing spectral bias, thereby improving the accuracy of signal approximation. After conducting comprehensive experiments on ten very complex test signals from diverse application domains, the proposed architectures have demonstrated superior ability in approximating intricate sensor signals and mitigating spectral bias effectively. The numerical results of the experiments reveal that FourierQResNet and GaborQResNet exhibit excellent performance compared to other existing neural network architectures and models in accurately estimating complicated sensor signals, with admiringly small errors. In addition, the findings emphasize the importance of mitigating spectral bias in order to achieve reliable learning from sensor data. The implications of these results are significant in various domains that require precise and reliable sensor data analysis, including healthcare, environmental monitoring, aviation, IoT applications, and industrial automation. This research significantly advances the field of sensor signal approximation and opens new avenues for enhancing data interpretation reliability and accuracy in complex signal environments.

INDEX TERMS Multiplicative filter neural networks, quadratic residual neural networks, spectral bias frequency principle, regression, implicit neural representations, neural networks, machine learning, deep learning.

I. INTRODUCTION

The significance of complex sensor signals is paramount in diverse fields such as healthcare, environmental surveillance, and industrial automation. Neural networks are universal function approximators that represent several types of signal function [1]. Precise estimation and examination of these signals present notable difficulties owing to their complex patterns, vulnerability to noise interference, and spectral partiality [2]. The utilization of neural network (NN) modeling has surfaced as a promising methodology for the regression of sensor signals [3]. This approach has the capability to apprehend intricate relationships and provide precise prognostication. The task of approximating complex sensor signals is a formidable undertaking owing to their intricate patterns and the existence of noise [4]. Conventional techniques frequently encounter difficulties in capturing fundamental frameworks and precisely depicting the signal dynamics. The utilization of neural networks has exhibited exceptional achievements in diverse fields, capitalizing

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

on their ability to acquire intricate patterns from extensive datasets [5]. In the domain of sensor signal approximation, the efficacy of neural networks may be substantially diminished by the presence of noise and spectral biases [6].

The application of neural networks in sensor signal representation problems has been extensively discovered, showcasing their effectiveness across diverse domains [7]. Neural network models have been utilized in the healthcare sector [8], specifically for the analysis of electrocardiogram (ECG) signals in the diagnosis of heart disease, as demonstrated in [9]. The utilization of neural networks in predicting air quality based on sensor readings has been observed in environmental monitoring [10]. In the realm of industrial automation, the utilization of neural network models has been deemed essential for the optimization of manufacturing processes through the analysis of sensor data [11]. The aforementioned instances underscore the adaptability of neural network modeling in the context of sensor-based applications [12]. The efficacy of neural networks in regression tasks has been demonstrated, as they have the ability to provide precise predictions for a range of problems. Neural networks have demonstrated successful applications in the regression of sensor signals. In [13], the authors introduced a deep neural network framework for regression analysis of sensor data in industrial systems. Their approach was found to outperform conventional techniques. In [14], researchers employed neural networks to estimate soil moisture levels using sensor measurements, exhibiting enhanced precision compared to conventional methodologies. The aforementioned studies highlight the capability of neural networks to perform tasks related to regression.

Polynomial neural networks [15] provide a versatile methodology for representing the intricate associations among input signals and output forecasts [16]. Numerous investigations have been conducted to examine the utilization of polynomial neural networks for the purpose of addressing regression issues [17]. In [18], the authors introduced a polynomial neural network model to predict the temperature using sensor data, which resulted in a notable level of precision in forecasting. Researchers employed polynomial neural networks to forecast energy consumption trends in intelligent buildings based on sensor data [19]. The aforementioned instances serve as evidence of the effectiveness of polynomial neural networks when utilized in regression scenarios [15]. As a special case of polynomial neural networks, the utilization of quadratic neural networks presents a robust structure for the acquisition of nonlinear connections and interactions within sensor signals and also proves that they are universal function approximators [20]. Quadratic neural networks have been utilized by researchers in the context of regression problems, yielding encouraging results [21]. In [22], the authors showed that quadratic neurons could be employed as a means of simulating imprecise bearing data and extracting attentional information from it and showed that the presented model possesses inherent interpretability, thereby facilitating efficacious and comprehensible diagnosis of bearing faults. Several studies have underscored the benefits of utilizing quadratic neural networks for sensor signal regression [23], [24]. The utilization of quadratic residual neural networks has surfaced as a propitious methodology for augmenting the representational capacity of neural networks in the context of regression assignments. In [25], researchers suggested Quadratic Residual Networks (QRes), which are a new type of neural network with high efficiency, convergence speed, and accuracy. They showed that these networks can be used in a broader range of ML applications owing to their ability to learn higher frequencies and greater nonlinearity than deep neural networks. In [26], a new form of quadratic deep neural network (QDNN) was proposed as a quadratic neuron design with theoretical effectiveness and efficiency analysis. Extensive experiments on multiple learning tasks demonstrated their superiority compared to first-order NNs. They also showed that QDNNs have significant potential for learning tasks, such as object detection, segmentation, and position recognition, which can improve performance by focusing on important objects while ignoring unimportant backgrounds.

Coordinate-based networks (CBNs) such as Multiplicative Filter Networks (MFNs) [27], Band-limited Coordinate Networks (BACON) [28], and Residual Multiplicative Filter Networks (RMFN) [29] provide a degree of control over the frequency spectrum employed for the representation of continuous signals. Multiplicative neural networks provide a distinct approach to sensor-signal representation and modeling by integrating multiplicative interactions among input features. These networks have demonstrated potential for problems related to regression [27]. Multiplicative filter networks circumvent the traditional approach of compositional depth and instead employ concurrent multiplication of sinusoidal or Gabor wavelet basis functions that are applied to the input. The aforementioned representation possesses a significant benefit in that it enables comprehensive observation of the complete function as a function with a linear approximator [30] across a vast number of Fourier or Gabor basis functions [31]. Notwithstanding their lack of sophistication, multiplicative filter networks demonstrate superior or comparable performance to more advanced methods such as, Fourier feature networks [32] with ReLU activation functions or sinusoidal activation networks [33], in the domains previously mentioned. BACON [28] is a coordinate-based MFN utilized to enhance the interpretability and scalability of coordinate networks. The authors proposed a method in which the intermediate layers of the network can be analyzed and controlled to enable the manipulation of the spectral bandwidth, thereby facilitating the representation of signals at multiple scales. The system also generates intermediate outputs that facilitate the enhancement of inference times through adaptive frequency evaluation. Their findings indicated that BACON exhibits superior performance compared to other single-scale coordinate networks in the domains of

multi-scale image fitting, neural rendering, and 3D scene representation. In [29], researchers presented a study, RMFN, which introduced a novel coordinate network structure and instructional strategy that facilitates optimization from a general to a specific level while allowing for precise regulation of the frequency support of acquired reconstructions. The incorporation of skip connections and a unique initialization scheme are two pivotal advancements that enable the regulation of the model frequency spectrum during each optimization phase. By utilizing skip connections and a uniform initialization scheme, the suggested network was able to guarantee that the lower levels of the network captured broader scales of the signal. In another study, The Wavelet Implicit Neural Representation (WIRE) [34] was introduced and defined as a novel and precise Implicit Neural Representation (INR) that employs a continuous complex Gabor wavelet activation function, thereby enhancing its resilience and dependability. The model exhibits greater representation capacity, attains superior accuracy with increased speed, and possesses robust inductive biases that render it highly suitable for addressing complex inverse problems. Consequently, it has become the preferred solution for representing signals and resolving inverse problems in the field of INR.

The term ''spectral bias'' pertains to the tendency of neural networks to display a predilection towards particular frequency components or patterns in the input signals [6]. The comprehension and reduction of spectral bias are of the utmost importance in achieving precise estimation of intricate sensor signals [35]. Numerous investigations have been conducted on the issue of spectral bias [36] in various types of neural networks, such as multiplicative [37], polynomial [38], and quadratic (also quadratic residual) neural networks [39]. In [40], researchers conducted an analysis of the spectral bias present in Coordinate-based MLPs using a novel model of training dynamics. Their results showed that after a low-frequency representation was created, the high-frequency components of the signal were learned, with the degree of spectral bias dependent on how the network allocated inputs to activation zones by using these CBNs. Another finding is that positional encoding has the potential to improve expressiveness and hasten convergence to high-frequency components. In [41], Their study measured the influences of the spectral behavior of image classification tasks, helping to understand why deep models generalize effectively. Another research [42] proposed two approaches to quantify spectral bias in current image classification neural networks and examined how training decisions affect learned frequencies. Their experimental results indicated that the proposed approach performed well over iterations with a controlled spectral bias and outperformed existing denoising, deblocking, inpainting, super-resolution, and detail enhancement methods. In another work [43], the spectral bias was explained using the ReLU neural network spectral bias, and they proposed that replacing the ReLU activation function with a piecewise linear B-spline, such as the Hat function, the Hat activation function significantly enhanced image classification generalization accuracy. Several studies have proposed using adaptive activation functions to reduce spectral bias [44], [45], [46], [47]. In another study [48], deep Kronecker NNs with rowdy adaptive activation functions were proposed and tested as a function approximation solution inference of partial differential equations using physics-informed neural networks. In [49], a computable depiction of Spectral Bias in neural networks was introduced. In [35] and [50], researchers used Fourier analysis to find an interesting way to explain why feed-forward deep learning works faster. According to their theoretical studies, the impact of depth could be thought of naturally as a state that changes the target function into a function with a lower frequency; however, it could also be saturated. In [30], this study demonstrated that the F-Principle holds true even when using non-gradient-descent-based training for DNNs, such as Powell's method and Particle Swarm Optimization, neither of which relies on gradient information for optimization.

Several studies have attempted o find a remedy for spectral bias (F-Principle) and reduce its effects on NNs. In [51], the study introduced a novel DNN architecture, namely PhaseDNN, which exhibits consistent and broad convergence in approximating high-frequency functions and wave equation solutions. This approach leveraged the idea that moderately sized DNNs were created and trained to address specific high-frequency ranges. By utilizing phase shifts in the frequency domain, it was possible to train each DNN to approximate the higher frequency components of a given function within a specific range while maintaining a convergence rate similar to that of the low-frequency range. Despite the method's high level of accuracy, several experiments have revealed that it is less efficient in fitting broadband signals with smooth spectra [52]. Another study [52] introduced a novel approach, namely the parallel frequency function-deep neural network (PFF-DNN), which leverages the frequency-domain analysis of broadband signals and the spectral bias characteristics of neural networks to significantly enhance their efficiency. Extensive numerical experiments yielded a significant enhancement in efficiency. A recent study [53], introduced a novel approach called coupled frequency predictor-corrector triangular deep neural network (cFPCT-DNN) that demonstrates proficient convergence in the estimation of high-frequency functions. Their numerical findings exhibited the capacity of the cFPCT-DNN to acquire knowledge of the functions and solutions of the Helmholtz equations. This methodology enhanced the frequency of the deep neural network (DNN) and effectively augmented the learning capacity of the Corrector DNN.

After a thorough review of the existing literature, it was observed that most neural network models exhibit insufficient learning capability, are very complex, and have computationally intensive operations. Most of the proposed methods assume that the target signals are uniformly sampled in real-world problems and seek solutions to the spectral bias

problem using the classical Fourier transform. However, the Fourier transform only provides accurate calculations when the signals are uniformly sampled. This approach makes it difficult for the proposed neural network models to provide effective solutions and be used in real problems. Additionally, it was noted that the phenomenon of spectral bias has yet to be thoroughly investigated in novel studies on sensor signal modeling. To the best of our knowledge, there is currently no all-encompassing examination of regression-based modeling for intricate sensor signals using multiplicative filter neural networks. Additionally, there has been no thorough empirical exploration of spectral bias concerning these models.

This study introduces two innovative Quadratic Residual Multiplicative Filter Neural Networks, namely FourierQResNet and GaborQResNet. The present study employed a combination of quadratic residual neural networks, multiplicative filter networks, Fourier filters, and Gabor filters to effectively approximate intricate sensor signals while simultaneously examining their optimal spectral bias performance. The utilization of Fourier filters allows networks to effectively capture frequency components, whereas the implementation of Gabor filters enhances the capacity of the network to capture the local spatiotemporal patterns present in the sensor signals. The objective is to overcome the difficulties related to the estimation of intricate sensor signals and proficiently manage the spectral bias through the integration of these components.

The contributions of this study are twofold. First, we introduce FourierQResNet and GaborQResNet as advanced methods for efficient approximation of complex sensor signals. These networks offer improved accuracy compared with traditional methods and existing neural network architectures, enabling more reliable analysis of sensor data. Second, we investigated the perfect spectral bias performance of these networks, focusing on their ability to eliminate spectral bias without any loss of signal information. Through rigorous evaluation, we assessed their robustness in handling spectral bias and demonstrated their effectiveness in real-world scenarios. The findings of this research have significant implications for various domains, including healthcare monitoring, environmental sensing, and industrial automation, where the accurate and reliable analysis of sensor data is essential. By addressing the challenges of convergence speed and spectral bias, FourierQResNet and GaborQResNet open new possibilities for improved signal approximation and enhance the applicability of sensor data analysis in practical settings.

FourierQResNet and GaborQResNet play a role in enhancing the reliability and accuracy of data interpretation across domains such as healthcare monitoring, environmental sensing, and industrial automation. These networks address challenges encountered in sensor data analysis, such as convergence speed and spectral bias. By incorporating filters like Fourier and Gabor filters, FourierQResNet and GaborQResNet effectively capture patterns in signals and mitigate the impact of spectral bias. For instance, in healthcare monitoring FourierQResNet and GaborQResNet can be utilized to analyze physiological sensor data like electrocardiograms (ECG) or electroencephalograms (EEG). These networks excel in extracting frequency components and identifying patterns in the data, thereby assisting in the diagnosis of abnormalities or neurological disorders. In the realm of sense, FourierQResNet and GaborQResNet empower the analysis of sensor data related to air quality, water quality, or weather conditions. These networks are capable of detecting patterns or textures that indicate pollution levels identifying anomalies in water samples, or accurately predicting weather patterns. Moreover, in the field of automation, FourierQResNet and GaborQResNet offer insights into sensor data for tasks like predictive maintenance, fault detection, or process optimization. These networks possess the capability to detect changes or anomalies in sensor readings, thereby aiding in identifying equipment failures or optimizing manufacturing processes. Utilizing the potentials of FourierQResNet and GaborQResNet allows for a precise and dependable analysis of sensor data. These networks can improve the understanding of data patterns. Facilitate better decision-making in domains that heavily rely on accurate analysis of sensor data.

This paper is organized as follows: Section II provides a comprehensive review of neural network modeling of sensors and the learning of sensor signals for regression-related problems, highlighting various architectures and their applications, and focuses on polynomial neural networks, quadratic neural networks, quadratic residual neural networks, and multiplicative neural networks. We present the proposed architectures, FourierQResNet and GaborQResNet, and the preliminaries for the methods and models used in this study are briefly explained in this section. The studies conducted to verify our research hypotheses are described in Section III. The experimental setup, dataset, measurement criteria, and training methods are described. In addition, it covers precise design decisions and parameter values for GaborQResNet and FourierQResNet. Section IV, the results of the experiments are presented and analyzed. The performance of FourierQResNet and GaborQResNet was compared with that of traditional methods and existing neural network architectures in terms of approximation accuracy for complex sensor signals and spectral bias. The results of the experiments are discussed in detail, highlighting the strengths and limitations of the proposed architectures. Finally, Section V concludes the paper by summarizing the contributions of this study. It revisits the research hypotheses and offers concluding remarks regarding the efficacy of FourierQResNet and GaborQResNet in approximating complex sensor signals and addressing spectral bias. In addition, prospective future directions and areas for further research in the field of sensor-signal approximation are discussed.

This study seeks to advance the field of signal processing by introducing novel neural network architectures and analyzing their efficacy in approximating complex sensor signals. To the best of our knowledge, this is the first research that combines quadratic residual neural networks,

multiplicative filter networks, Fourier filters, and Gabor filters for the efficient approximation of complex sensor signals and the investigation of their perfect spectral bias performance.

II. MATERIALS AND METHODS

Recently, owing to the increasing prevalence of intricate signal approximation using deep neural networks, researchers have directed their focus toward the spectral bias of neural networks. This issue arises when a neural network is employed to fit the intricate sensor signals. Numerous mathematical computing issues entail resolving high-frequency solutions within intricate domains, including but not limited to dealing with high-frequency wave equations in non-uniform media. These arise from phenomena such as the propagation of electromagnetic waves in turbid media, rough surface dispersion, seismic waves, and geophysical problems. The calculation challenges associated with discovering effective solutions, particularly under arbitrary conditions, are significant due to the extremely fluctuating nature of the solutions. Although deep neural network-based techniques may not yield numerical outcomes as precise as those obtained through finite element methods, they offer the benefit of straightforward implementation and circumvent the significant expense of mesh generation while still providing satisfactory accuracy for numerous engineering applications. In order to enhance the proficiency of conventional deep neural networks in acquiring knowledge of highly oscillatory functions in physical spatial variables, we propose two innovative deep neural network models that possess broad learning capabilities to minimize errors in approximating all frequencies of the intended function. This was achieved by utilizing faster convergence of the neural networks at low frequencies during the training process. The resolution of this issue has emerged as a fundamental impediment to the advancement of artificial intelligence technology.

Motivated by factors, the use of residual neural networks (QResNets), in combination with multiplicative filter networks, Fourier filters, and Gabor filters, proves beneficial. Firstly QResNets exhibit learning capacity and expressivity compared to neural network architectures. They achieve this by incorporating weight terms into their structure, enabling them to model complex functions and enhance approximation performance. Additionally, integrating QResNets with filter networks allows for the extraction of features through interactions among input features. This unique approach to signal representation and modeling leads to accuracy and performance in regression tasks. Lastly, the incorporation of Fourier filters and Gabor filters in the network architecture facilitates the capture of components and local spatiotemporal patterns, respectively. These filters enable the networks to extract information from sensor data enhancing signal representation accuracy. By combining QResNets with filter networks, Fourier filters, and Gabor filters, this comprehensive network architecture offers improved approximation performance and feature extraction capabilities. It efficiently addresses the requirements of sensor data analysis with accuracy and efficiency.

In the following sections, we introduce two novel neural networks to solve this problem.

A. PROBLEM DEFINITION

The prevalent use of deep networks is to approximate functions [3] over inputs with high dimensions. However, recent research has sparked a growing interest in utilizing neural networks as function approximations [1] for complex functions with low dimensions. These functions may include expressing pictures as a function of pixel locations [54], solving (partial) differential equations [45], describing signed distance functions, or representing neural radiance fields. The sensor signals [5]. However, these problems often require neural networks with a high degree of representational power. Moreover, spectral bias [6] is another problem that has a negative impact on neural network training and prediction performance [36], and the rate of convergence exhibited by the neural network is associated with the frequency spectrum of the signal that has been fitted, as noted in reference [6]. A quantitative analysis was conducted on the correlation between the convergence speed and frequency of the fitted signal [36]. The training time necessary to fit a signal using a network exhibits exponential growth as the central frequency of the component increases. The convergence speed in network training is affected by spectral bias, resulting in prolonged training times to fit high-frequency components in sensor signals, which can be challenging to endure.

When the related literature is examined in detail, it is found that the spectral bias can be caused by the different architectural components of neural networks [42], initialization processes of neural networks [35], activation functions [43], and frequency values contained in the signals [50]. Even different optimization methods other than gradient-based optimization can not help reduce the frequency bias [30]. In this research, to solve these problems, we proposed two new neural networks based on multiplicative filter neural networks and quadratic residual neural networks to obtain much better learning capacity and less spectral bias than most well-known neural networks.

In this research, we hypothesize the following:

- FourierQResNet and GaborQResNet will outperform the traditional methods and existing neural network architectures in terms of approximation accuracy for complex sensor signals.
- The integration of quadratic residual neural networks and multiplicative filter networks will significantly enhance the feature extraction capabilities, leading to improved approximation performance.
- The inclusion of Fourier filters and Gabor filters in the network architecture will enable the networks to effectively capture frequency components and local spatiotemporal patterns, respectively, resulting in more accurate signal representation.

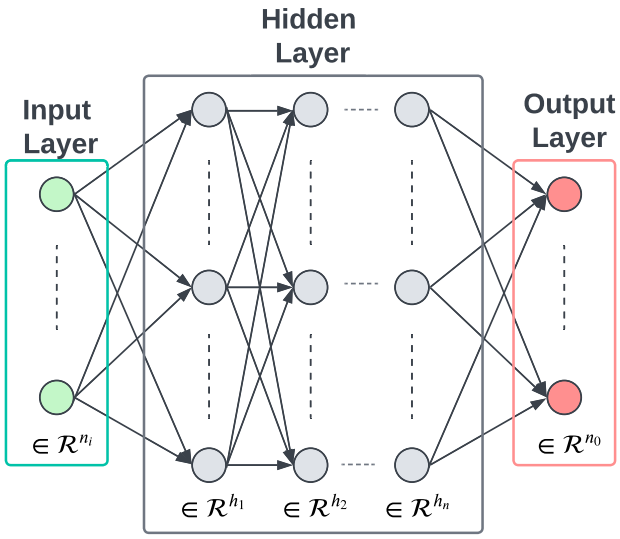

FIGURE 1. General neural network architecture.

By investigating these hypotheses, we aimed to demonstrate the effectiveness of FourierQResNet and GaborQResNet in efficiently approximating complex sensor signals and their ability to handle spectral bias. The outcomes of this research will contribute to the advancement of signal-processing techniques and provide valuable insights for applications such as healthcare monitoring, environmental sensing, and industrial automation. The preliminaries of the methods and models used in this study are briefly explained in the following subsections.

B. MULTILAYER PERCEPTRON FEED FORWARD NEURAL NETWORKS (MLP)

Deep neural networks can be utilized as universal approximators for any nonlinear continuous function by using a sufficient number of neurons [1], [4]. Hence, we employed a DNN model to describe the nonlinear mapping between the state and actions for an integrated guidance and control policy.

Figure 1 shows the general layout of a DNN, which consists of an input layer with an arbitrary number of features, hidden layers stacked together, and an output layer. The input-output mapping is achieved with sequential mathematical operations followed by nonlinear activation functions using a finite number of free parameters called weights and biases. The optimal combinations of these parameters are optimized with gradient-based optimization methods.

Considering x_l as an input vector to a specific layer l, then the mathematical calculations performed can be expressed as follows;

$$f_l = \sigma_l(\omega.x_l + b_l) \tag{1}$$

In Equation 1, $\omega_l \in \mathbb{R}^{N_l \times N_{l+1}}$ are the weight matrices, and the N_l, N_{l+1} account for the number of neurons used in layers l and $l+1$ respectively. The $b_l \in \mathbb{R}^{N_{l+1}}$ denotes bias vector. Lastly, the σ_l corresponds to the non-linear activation function used in the layer. There exist different kinds of functions used in a DNN model in order to provide non-linearity. The most common choices are *Linear*, *Tanh*, *Relu*, *Gelu*, and *Sigmoid*. After defining the formulations for a specific layer, the final prediction of the DNN model can be constructed, as given in Equation 2.

$$\hat{y}(x) = (f_l \circ f_{l-1} \cdots f_2 \circ f_1)(x) \tag{2}$$

In Equation 2, $\circ$ denotes the composition operator and, f_l is the l_{th} NN layer. As mentioned before, the learnable parameters W_l and b_l are trained by using the backpropagation algorithm [55]. However, to perform the backpropagation algorithm, it is necessary to define a performance index. For instance, the first thing that comes to mind, and also the common choice for regression-related tasks, is to utilize the Mean Squared Error (MSE) as a loss function. With the definition of a loss function, the networks are trained to find the best combinations of weights and biases. It is possible to use different variants of optimization algorithms during backpropagation, such as stochastic gradient-descent, Momentum, ADAM, and NADAM [56] according to the nature of the problem.

C. QUADRATIC NEURAL NETWORKS (QNN)

The Quadratic (deep) Neural Network is a burgeoning area of research that draws heavily upon the foundational knowledge of first-order neural networks. Quadratic neurons exhibit inherent improvements in their capacity for representation, which is attributable to their nonlinear computational processes rather than an increase in parameters [20]. The activation function of a regular neuron provides nonlinear mapping, whereas a quadratic neuron incorporates additional non-linear mapping through the utilization of the quadratic combination function. It has been observed that a quadratic neuron cannot be deemed identical to the mixture or aggregation of three traditional neurons. Assuming that the activation function $\sigma(\cdot)$ is a Rectified Linear Unit (ReLU), it can be inferred that the mixture of traditional neurons is limited to a piecewise linear function. However, a quadratic neuron is capable of representing a piecewise polynomial basis function [23]. It is widely acknowledged that a polynomial spline is a superior method for approximating complex functions compared to a linear basis [22].

Contrary to popular belief, while certain studies have demonstrated the superior non-linear and learning capabilities of QNNs in comparison with first-order NNs, their neuron design is plagued by various limitations that hinder their theoretical performance and practical deployment. In [26], the QNN design was classified into four distinct categories according to their neuron architecture. In another approach, we redefined this classification into seven categories. Subsequently, a comprehensive analysis was conducted to identify the limitations of each design type from various perspectives. The mathematical definitions of the classified QNNs are presented in Equations 3-9 based on the manner in which the second-order term of input x is incorporated in each

quadratic neuron.

$$\text{Type 1: } f(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{w}_a \boldsymbol{x} + \boldsymbol{w}_b \boldsymbol{x} \tag{3}$$
$$\text{Type 2: } f(\boldsymbol{x}) = \boldsymbol{w}_a \boldsymbol{x}^2 \tag{4}$$
$$\text{Type 3: } f(\boldsymbol{x}) = (\boldsymbol{w}_a \boldsymbol{x})^2 \tag{5}$$
$$\text{Type 4: } f(\boldsymbol{x}) = (\boldsymbol{w}_a \boldsymbol{x})\,(\boldsymbol{w}_b \boldsymbol{x}) \tag{6}$$
$$\text{Type 5: } f(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{w}_a \boldsymbol{x} + \boldsymbol{w}_b \boldsymbol{x}^2 \tag{7}$$
$$\text{Type 6: } f(\boldsymbol{x}) = (\boldsymbol{w}_a \boldsymbol{x})\,(\boldsymbol{w}_b \boldsymbol{x}) + \boldsymbol{w}_b \boldsymbol{x}^2 \tag{8}$$
$$\text{Type 7: } f(\boldsymbol{x}) = (\boldsymbol{w}_a \boldsymbol{x})\,(\boldsymbol{w}_b \boldsymbol{x}) + \boldsymbol{w}_c \boldsymbol{x} \tag{9}$$

As a result of a detailed examination of the literature on the proposed QNNs and the experimental results of these neural networks, we decided to examine the Type-6 QResNet models within the scope of this study and evaluate their performance by comparing their performance with our proposed models. In all subsequent sections, the type-6 quadratic neural networks given in Equation 8 are referred to as QNN.

1) TYPE - 6 QUADRATIC RESIDUAL NEURAL NETWORKS (QNN)

This type of neuron performs two inner products and one power term operation on the input vector, which are then integrated to produce a nonlinear activation function. The quadratic neuron's output function can be mathematically represented as follows [22], [57]:

$$\phi_l = \sigma\left(\left(\boldsymbol{x}^\top \boldsymbol{w}^r + b^r\right)\left(\boldsymbol{x}^\top \boldsymbol{w}^g + \boldsymbol{b}^g\right) + (\boldsymbol{x} \odot \boldsymbol{x})\boldsymbol{w}^b + c\right) \tag{10}$$

In Equation 10, $\sigma(\cdot)$ is a nonlinear activation function, $\odot$ denotes the Hadamard product, $\boldsymbol{w}^r, \boldsymbol{w}^g, \boldsymbol{w}^b \in \mathbb{R}^n$ are weight vectors, and $b^r, b^g, c \in \mathbb{R}$ are biases. When $\boldsymbol{w}^g = 0, b^g = 1$, and $\boldsymbol{w}^b = 0$, a quadratic neuron degenerates to a conventional neuron: $\sigma(f(\boldsymbol{x})) = \sigma\left(\boldsymbol{x}^\top \boldsymbol{w}^r + b^r\right)$.

2) QUADRATIC RESIDUAL NEURAL NETWORKS

The concept of a quadratic neuron was introduced by [57], and all neural networks are designed utilizing neurons of the same type, which are distinguished by two key characteristics. The first characteristic involves the computation of an inner product between an input vector and a corresponding weighting vector consisting of trainable parameters. The second characteristic pertains to the utilization of a nonlinear excitation function. Whilst it is true that these neurons possess the capability to be interlinked in a manner that can simulate a wide range of functions, it should be noted that the configuration of the network is not necessarily exclusive. On the other hand, with regard to this lack of uniqueness, the identification of appropriate neuron types for general machine learning is a novel area of research, as there is no singular type of neuron that is universally applicable. The utilization of quadratic neurons enables the substitution of the inner product with a quadratic function of the input vector. This upgrade transforms a 1st-degree neuron into a 2nd-degree neuron, which enhances the capabilities of each neuron and simplifies the optimization of neural networks [22]. The Quadratic Residual Network (QResNet) architecture was presented in [25]. The primary intent of this architecture is to enhance the expressive power of the model to capture high-frequency responses with fewer parameters than those of conventional NNs. At each network layer, QResNet delivers quadratic nonlinearity prior to applying activation functions. The formulation of a quadratic neuron is given by Equation 11.

$$\phi_l(\boldsymbol{x}) = \sigma_l(\boldsymbol{w}_1^l \boldsymbol{x}_l \odot \boldsymbol{w}_2^l \boldsymbol{x}_l + \boldsymbol{w}_1^l \boldsymbol{x}_l + \boldsymbol{b}^l) \tag{11}$$

In Equation 11, ϕ_l is the output layer l, $\boldsymbol{x}_l \in \mathbb{R}_l^N$ is the input vector for the layer l, $\boldsymbol{w}_1^l \in \mathbb{R}^{N_l \times N_{l+1}}$ and $\boldsymbol{w}_2^l \in \mathbb{R}^{N_l \times N_{l+1}}$ are the weight matrices at layer numbers l and $l+1$. The N_l, N_{l+1} account for the number of neurons used in layers l and $l+1$ respectively, and $\odot$ denote the Hadamard product. The term $(\boldsymbol{w}_1^l x_l \odot \boldsymbol{w}_2^l \boldsymbol{x}_l)$ is the quadratic residual term and the $\boldsymbol{b}^l \in \mathbb{R}^{N_{l+1}}$ is the bias vector. Finally, the σ_l corresponds to the non-linear activation function used in the layer.

$$\hat{\Phi}(\boldsymbol{x}) = (\phi^{(l)} \circ \phi^{(l-1)} \cdots \phi^{(2)} \circ \phi^{(1)})(\boldsymbol{x}) \tag{12}$$

In Equation 12, $\hat{\Phi}$ is the output of QResNet, $\circ$ denotes the composition operator, $\phi^{(l)}$ is the output of the l_{th} QResNet layer, $\boldsymbol{x}$ is the input vector for the first layer. As mentioned previously, the learnable parameters $\boldsymbol{w}_1^l$, $\boldsymbol{w}_2^l$, and $\boldsymbol{b}^l$ can be trained with the backpropagation algorithm using gradient-based optimization methods.

D. MULTIPLICATIVE FILTER NETWORKS: FourierNet AND GaborNet

A conventional neural network, which consists of n-layers and maps an input vector from $\mathbb{R}^p$ to an output vector in $\mathbb{R}^q$, $f : \mathbb{R}^p \to \mathbb{R}^q$, is commonly formulated using a recursive equation, as given in Equation 13 [27].

$$\begin{aligned} \boldsymbol{z}^{(1)} &= \boldsymbol{x} \\ \boldsymbol{z}^{(l+1)} &= \sigma\left(\boldsymbol{w}^{(l)} \boldsymbol{z}^{(l)} + \boldsymbol{b}^{(l)}\right),\ l = 1, \ldots, n-1 \\ \boldsymbol{f}(\boldsymbol{x}) &= \boldsymbol{w}^{(n)} \boldsymbol{z}^{(n)} + \boldsymbol{b}^{(n)} \end{aligned} \tag{13}$$

The weight and bias of the l_{th} layer are denoted as $\boldsymbol{w}^{(l)} \in \mathbb{R}^{d_{l+1} \times d_l}$ and $\boldsymbol{b}^{(l)} \in \mathbb{R}^{d_{l+1}}$, respectively. In addition, the hidden unit in layer l is represented by $\boldsymbol{z}^{(l)} \in \mathbb{R}^{d_l}$. It should be noted that $\boldsymbol{\sigma}$ denotes the nonlinearity applied element-wise. This terminology is derived from the fact that each nonlinearity is applied in a compositional manner to the outputs of the preceding nonlinearity, with the aim of achieving a higher level of representational complexity. The networks referred to in this context are known as compositional depth networks (CDNs). In [27], the multiplicative filter network employed an alternative recursion method that precludes the composition of nonlinear functions. The definition of a multiplicative filter network is established through a recursive

process, as shown in Equation 14.

$$\begin{aligned} z^{(1)} &= \boldsymbol{h}\left(x;\theta^{(1)}\right) \\ z^{(l+1)} &= \boldsymbol{h}\left(x;\theta^{(l+1)}\right) \circ \left(\boldsymbol{w}^{(l)}z^{(l)} + \boldsymbol{b}^{(l)}\right),\ l = 1,\ldots,n-1 \\ \boldsymbol{f}(\boldsymbol{x}) &= \boldsymbol{w}^{(n)}z^{(n)} + \boldsymbol{b}^{(n)} \end{aligned} \tag{14}$$

The symbol $\circ$ is used to represent element-wise multiplication. The variables $\boldsymbol{w}^{(l)}, \boldsymbol{b}^{(l)}, z^{(l)}$ are defined as stated previously. However, the function $\boldsymbol{h} : \mathbb{R}^p \to \mathbb{R}^{d_l}$ is parameterized by $\theta^{(l)}$ (which can have varying sizes to implicitly establish the output dimensions d_l). Furthermore, h denotes a nonlinear filter directly applied to the input. Of primary significance in this context is the avoidance of applying nonlinearity to the output of the preceding nonlinearity within the network. The nonlinearity of the network is exclusively confined to the h functions. Layers $z^{(l)}$, following their traversal through a linear function, are solely subjected to multiplication by fresh input filters. In [27], [29], and [54], two variations of the MLP were introduced by employing either sinusoids or a Gabor wavelet as the filter function $\boldsymbol{h}$ was introduced. These two models are referred to as FourierNet and GaborNet, respectively. For the implementation of FourierNet, a basic sinusoidal filter is being contemplated, and the definition is given in Equation 15:

$$\boldsymbol{h}_l\left(x;\theta^{(l)}\right) = \sin\left(\boldsymbol{w}^{(l)}\boldsymbol{x} + \phi^{(l)}\right) \tag{15}$$

In Equation 15, involves the use of the parameters $\theta^{(l)} = \left\{\boldsymbol{w}^{(l)} \in \mathbb{R}^{d_l \times n}, \phi^{(l)} \in \mathbb{R}^{d_l}\right\}$. One notable advantage of FourierNet over composition-based networks is its capacity to exhibit a linear function of Fourier bases that can be viewed immediately and compellingly. This is achieved through a low-rank set of coefficients that are defined by the network parameters, despite the large number of Fourier basis functions involved.

Pure Fourier bases are commonly acknowledged to possess an imperfection in that they exhibit global support, which may pose challenges in representing localized features. An often employed substitute for these bases involves the utilization of Gabor filters, which are capable of capturing both frequency and spatial locality components. In [27], [29], and [54], the Gabor filter expressed in Equation 16:

$$\begin{aligned} \boldsymbol{h}_j\left(x;\theta^{(l)}\right) &= \sin\left(\boldsymbol{w}_j^{(l)}\boldsymbol{x} + \boldsymbol{\phi}_j^{(l)}\right) \\ &\quad \cdot \exp\left(-\frac{\boldsymbol{\gamma}_j^{(l)}}{2}\left\|\boldsymbol{x} - \boldsymbol{\mu}_j^{(l)}\right\|_2^2\right) \end{aligned} \tag{16}$$

In Equation 16, The set of parameters $\theta^{(l)}$ consists of real valued variables $\gamma_{1:d_l}^{(l)}$, $\mu_{1:d_l}^{(l)}$ in $\mathbb{R}^n$, $\boldsymbol{w}_{1:d_l}^{(l)}$ in $\mathbb{R}^n$, and $\phi_{1:d_l}^{(l)}$ in $\mathbb{R}$. $\gamma_j^{(l)}$ is used to express the scaling term, and $\gamma_j^{(l)}$ $\mu_j^{(l)}$ is utilized to represent the mean of the j_{th} Gabor filter. For the sake of clarity, the function $h : \mathbb{R}^n \to \mathbb{R}^{d_l}$ is specified in terms of each of its $j = 1\ldots, d_l$ coordinates. Similar to FourierNet, a notable characteristic of GaborNet is that its ultimate function f can be expressed as a linear mixture of Gabor filters.

E. QUADRATIC RESIDUAL MULTIPLICATIVE FILTER NEURAL NETWORKS: FourierQResNet AND GaborQResNet

The study by [27] on Multiplicative Filter Networks highlights a novel approach to avoid the conventional method of compositional depth. This is achieved through the simultaneous multiplication of sinusoidal or Gabor wavelet basis functions, which are then applied to the input. Furthermore, the incorporation of quadratic neurons facilitated the replacement of the inner product with a quadratic function of the input vector. According to [25], the process of upgrading a 1st-order neuron to a 2nd-order neuron results in improved capabilities for each neuron and streamlines the optimization of neural networks. The present study utilizes a multiplicative filter network coordinate-based architecture and quadratic residual neural networks, a specific type of polynomial neural network-based architecture which was recently introduced. Our objective is to utilize the learning capacity of multiplicative neural networks. The filtering operation was employed in lieu of nonlinear activation functions in multiplicative neural networks, and quadratic weight terms were incorporated into the architecture. Furthermore, the utilization of quadratic residual neural networks with reduced spectral bias power in high-order modeling results in increased robustness. Consequently, the suggested structures exhibit rapid convergence and heightened susceptibility to frequency bias.

The output of a FourierNet can be represented by a linear mixture of sinusoidal basis functions [27], [28], [29]. In other words, FourierNet can be described as a mathematical model that expresses its ultimate function as a linear combination of conventional Fourier bases. This property is given by Equation 17.

$$\boldsymbol{f_j}(\boldsymbol{x}) = \sum_{t=1}^{K} \bar{\boldsymbol{\beta}}_t \sin\left(\bar{\boldsymbol{w}}_t\boldsymbol{x} + \bar{\boldsymbol{\phi}}_t\right) + \bar{\boldsymbol{b}} \tag{17}$$

In Equation 17, $\bar{\beta}_{1:K}$ expresses several coefficients, $\bar{\omega}_{1:K}$ represents frequencies, $\bar{\phi}_{1:K}$ represents phase offsets, and $\bar{b}$ is the bias term. The analysis of the intermediate layer frequencies enables comprehension of the inner workings of FourierNet [27], [29]. The utilization of the trigonometric identity given in Equation 18:

$$\begin{aligned} \sin(\boldsymbol{w}\boldsymbol{x} + \phi) \circ \sin(\tau x + \psi) &= \frac{1}{2}\cos((\boldsymbol{w} - \tau)\boldsymbol{x} + \phi - \psi) \\ &\quad - \frac{1}{2}\cos((\boldsymbol{w} + \tau)\boldsymbol{x} + \phi + \psi) \end{aligned} \tag{18}$$

Similar to FourierNet, a noteworthy characteristic of GaborNet is that the ultimate function f can be expressed as a linear mixture of Gabor filters [27], [34]. The Gabor Network produces its output by linearly combining the Gabor basis

functions given in Equation 19.

$$f_j(\boldsymbol{x}) = \sum_{t=1}^{K} \bar{\lambda}_t \exp\left(-\frac{1}{2}\bar{\gamma}_t \left\|\boldsymbol{x} - \bar{\boldsymbol{\mu}}_t\right\|^2\right) \cdot \sin\left(\bar{\boldsymbol{w}}_t \boldsymbol{x} + \bar{\boldsymbol{\phi}}_t\right) + \bar{b} \tag{19}$$

In Equation 19, $\bar{\lambda}_{1:K}$ expresses several coefficients, $\bar{\omega}_{1:K}$ represents frequencies, $\bar{\phi}_{1:K}$ are phase offsets, and $\bar{b}$ is the bias term. Using the property that the output functions specified in Equations 17 and 19 can be represented by sinusoidal and Gabor filter functions, we can define our proposed neural networks. The definition of a multiplicative filter quadratic residual neural network is established through a recursive process, as shown in Equation 20:

$$\begin{aligned} \boldsymbol{z}^{(1)} &= \boldsymbol{h}\left(x; \theta^{(1)}\right), \\ \boldsymbol{Q}^{(l)}(\boldsymbol{z}) &= (\boldsymbol{W}_1^{(l)}\boldsymbol{z}^{(l)} \odot \boldsymbol{W}_2^{(l)}\boldsymbol{z}^{(l)} + \boldsymbol{W}_1^{(l)}\boldsymbol{z}^{(l)} + \boldsymbol{b}_q^{(l)}), \\ \boldsymbol{z}^{(l+1)} &= \boldsymbol{h}\left(x; \theta^{(l+1)}\right) \odot \left(\boldsymbol{Q}^{(l)}\boldsymbol{z}^{(l)} + \boldsymbol{b}_z^{(l)}\right), \; l = 1, \ldots, n-1 \\ \boldsymbol{f}(\boldsymbol{x}) &= \boldsymbol{Q}^{(n)}\boldsymbol{z}^{(n)} + \boldsymbol{b}_f^{(n)} \end{aligned} \tag{20}$$

In Equation 20, The symbol $\odot$ denotes the Hadamard product. The weight and bias of the l_{th} quadratic residual layer $\boldsymbol{Q}^{(l)}(\boldsymbol{z})$ are denoted as $\boldsymbol{W}_1^{(l)}$ and $\boldsymbol{W}_2^{(l)} \in \mathbb{R}^{d_{l+1} \times d_l}$, $\boldsymbol{b}_q^{(l)} \in \mathbb{R}^{d_{l+1}}$, respectively. Additionally, the hidden unit in layer l is represented by $\boldsymbol{z}^{(l)} \in \mathbb{R}^{d_l}$, the bias value of the hidden unit in layer l is $\boldsymbol{b}_z^{(l)} \in \mathbb{R}^{d_l}$, and the bias value of the output unit in layer n is $\boldsymbol{b}_f^{(n)} \in \mathbb{R}^{d_n}$. Note that $\boldsymbol{\sigma}^{(l)}$ denotes the nonlinearity applied element-wise (Tanh). This terminology is derived from the fact that each nonlinearity is applied in a compositional manner to the outputs of the preceding nonlinearity with the aim of achieving a higher level of representational complexity. However, the function $\boldsymbol{h} : \mathbb{R}^p \rightarrow \mathbb{R}^{d_l}$ is parameterized by $\theta^{(l)}$ (which can have varying sizes to implicitly establish the output dimensions d_l). Furthermore, h denotes a nonlinear filter directly applied to the input. Of primary significance in this context is the avoidance of applying nonlinearity to the output of a preceding nonlinearity within the network, and the nonlinearity of the network is exclusively confined to the h functions. Layers $\boldsymbol{z}^{(l)}$, following their traversal through a linear function, are solely subjected to multiplication by fresh input filters. Two variations of the MFN were introduced by employing either sinusoids or a Gabor wavelet as filter function $\boldsymbol{h}$ was introduced. These two models are referred to as FourierQResNet and GaborQResNet, respectively. For FourierQResNet and GaborQResNet, h filter functions are given in Equations 15 and 16, respectively. Training can be done using the standard backpropagation algorithm with first and also with second-order optimization methods such as ADAM, RMSProb, and L-BFGS. etc. The final optimization problem entails minimizing a loss function by optimizing both the newly introduced filter parameters and the NN weights and biases. The trainable parameters for FourierQResNet and GaborQResNet are given by Equations 21 and 22, respectively.

$$\Gamma_{FQRN} = \left\{\boldsymbol{W}_1^{(l)}, \boldsymbol{W}_2^{(l)}, \boldsymbol{b}_q^{(l)}, \boldsymbol{b}_z^{(l)}, \boldsymbol{b}_f^{(l)}, \boldsymbol{w}^{(l)}, \phi^{(l)}\right\}_{l=1}^{n} \tag{21}$$

$$\Lambda_{GQRN} = \left\{\boldsymbol{W}_1^{(l)}, \boldsymbol{W}_2^{(l)}, \boldsymbol{b}_q^{(l)}, \boldsymbol{b}_z^{(l)}, \boldsymbol{b}_f^{(l)}, \boldsymbol{w}^{(l)}, \phi^{(l)}, \gamma_j^{(l)}, \mu_j^{(l)}\right\}_{l=1}^{n} \tag{22}$$

Now, our optimization problem can be defined for the FourierQResNet model parameters, and additional Fourier filter parameters are given in Equation 23. Then, we can update the γ^l parameters using the loss function and its gradients by using the gradient descent step with Equation 24.

$$\gamma^*_{FourierQResNet_l} = \arg\min_{\gamma_l \in \mathbb{R}}(\mathcal{L}(\gamma)) \tag{23}$$

$$\gamma^l_{m+1} = \gamma^l_m - \eta_l \nabla_{\gamma^l} \mathcal{L}_m(\Gamma_{FQRN}) \tag{24}$$

In Equations 23 and, 24, $\gamma^*_{FourierQResNet_l}$ is the optimal value of Γ_{FQRN} at layer l, $\mathcal{L}$ is the loss function, $\mathcal{L}_m$ is the loss value at iteration step m, η_l is the learning rate, and m is the number of iterations. γ^l_m, is the trainable parameter value at the iteration step m in the layer l. ∇_{γ^l} is the gradient operator for the Loss function with respect to γ^l parameter at layer l, and $l = 1, \cdots, L$; $L \in \mathbb{N}$ is the layer number.

$$\lambda^*_{GaborQResNet_l} = \arg\min_{\lambda_l \in \mathbb{R}}(\mathcal{L}(\lambda)) \tag{25}$$

$$\lambda^l_{m+1} = \lambda^l_m - \eta_l \nabla_{\lambda^l} \mathcal{L}_m(\Lambda_{GQRN}) \tag{26}$$

Similarly, we can define another optimization problem for the GaborQResNet model parameters and the additional Gabor filter parameters. This optimization problem is given by Equation 25. Then, we can update the λ^l parameters using the loss function and its gradients by utilizing the gradient descent step with Equation 26. In Equations 25 and, 26, $\lambda^*_{GaborQResNet_l}$ is the optimal value of Λ_{GQRN} at layer l, $\mathcal{L}$ is the loss function, $\mathcal{L}_m$ is the loss value at iteration step m, η_l is the learning rate, and m is the number of iterations. γ^l_m, is the trainable parameter value at the iteration step m in the layer l. ∇_{λ^l} is the gradient operator for the Loss function with respect to λ^l parameter at layer l, and $l = 1, \cdots, L$; $L \in \mathbb{N}$ is the layer number.

Our proposed NNs, GaborQResNet and FourierQResNet, give greater importance to high-frequency elements of the target values, which can mitigate the impact of the F-principle [50]. When the loss function incorporates the gradient of the DNN output with respect to the input, it impairs a greater weight at a higher frequency in the loss function. The existence of an F-principle is contingent upon the interplay between activation regularity and loss function. Our proposed NNs are capable of assigning greater weights to high-frequency components to offset the low-priority attributed by the activation function using Fourier or Gabor filter terms and second-order QNNs gradients during the training. This gradient generation ability penalizes the errors that might occur at the high frequencies and, as a result, marginally reduces the spectral bias.

F. TARGET SIGNALS USED FOR EXPERIMENTS

In order to verify our hypotheses and examine the performance of the proposed innovative models, complex signals with high-frequency components, which are difficult to model, were used from the literature and were designed in this study. These signals and the data set generation process using them are described in the following sections:

1) TARGET SIGNAL 1

In particular, we are looking for signals that are well supported in the Fourier domain, which means that their amplitudes at high frequencies are not zero. The sine integral function attenuated using the Gaussian function was selected as the target function for this purpose [49].

$$f(x) = e^{\frac{-x^2}{2}} \int_0^{\frac{x}{\alpha}} \frac{\sin(x)}{x} dx \tag{27}$$

In Equation 27, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$. This function can be considered as a homogenized step signal, where a true step response is obtained when $\alpha \to 0$. For $\alpha > 0$, the amplitudes in the frequency spectrum decrease at a rate of $\frac{1}{\omega}$ until the frequency $\omega = \frac{1}{\alpha}$, where it falls to zero. Figures 2a and 3a show target signal 1 and the Fourier transform of target signal 1, respectively, for $\alpha = 0.01$. For the experiment, input points $\{x_i\}_{i=1}^{N}$ were sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(0, 1)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling 5120 i.i.d. points from this input distribution, training, and test datasets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

2) TARGET SIGNAL 2

For Target Signal 2, a signal with a fast frequency shift was investigated. This type of signal is often used in system identification and broad-band communication signal modeling [52], and its shape and frequency spectrum are shown in 2b and 3b. The explicit definition of the target Signal 2 is given by Equation 28.

$$f(x) = \cos\left[\pi\left(\omega_0 + \frac{\omega_T - \omega_0}{\beta} x^3\right) x^3\right] \tag{28}$$

In Equation 28, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$. Figures 2b and 3b show target Signal 2 and the Fourier transform of target signal 2, respectively, for $\omega_0 = 0.01$, $\omega_T = 50$, and $\beta = 1$. Figure 2b shows that as x increased, the oscillation frequency of the target signal grew instantly. Figure 3b also shows that the amplitude and oscillation frequency of the spectrum of the target signal decayed gradually. For the experiment, the input points $\{x_i\}_{i=1}^{N}$ were sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(0, 1)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling at 5120 i.i.d. points from this input distribution, training and validation data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

3) TARGET SIGNAL 3

The piecewise signal was utilized in [51] and [52] to investigate several neural networks' modeling performance. We have also included this signal in our experiment. The wave shape and corresponding frequency spectrum are shown in Figures 2c and 3c. The mathematical definition of the signal used in this experiment is given by Equation 29.

$$f(x) = \begin{cases} 10 \times (\sin(x) + \sin(3x)) & x \in [-\pi, 0] \\ 10 \times (\sin(23x) + \sin(137x) & \\ + \sin(203x)) & x \in [0, +\pi] \end{cases} \tag{29}$$

In Equation 29, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$. Figures 2c and 3c show target Signal 3 and the Fourier transform of target signal 3, respectively. The spectrum of this signal includes several apparent spikes and many small oscillations. For the experiment, input points $\{x_i\}_{i=1}^{N}$ were sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(-\pi, +\pi)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling at 5120 i.i.d. points from this input distribution, training and validation data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

4) TARGET SIGNAL 4

This highly nonlinear signal was designed to investigate the prediction performance of all neural networks. The wave shape and its corresponding frequency spectrum are shown in Figures 2d and 3d. The mathematical definition of the signal is given by Equation 30:

$$f(x) = \frac{1}{2}(\sin(k_1 x) + \sin(k_2 x) + \sin(k_3 x) + \sin(k_4 x) + \sin(k_5 x).e^{-\frac{x}{\pi}} + \sin(k_6 x).e^{-\frac{2x}{\pi}} + \sin(k_7 x)) \tag{30}$$

In Equation 30, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$. Figures 2d and 3d show target signal 4, and the Fourier transform of target signal 4, respectively for $k_1 = 2\pi, k_2 = 4\pi, k_3 = 6\pi, k_4 = 8\pi, k_5 = 10\pi, k_6 = 15\pi, k_7 = 20\pi$. The spectrum of this signal includes different frequencies and frequencies with exponentially decreasing components at the same time. For the experiment, input points $\{x_i\}_{i=1}^{N}$ were sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(-\pi, +\pi)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling at 5120 i.i.d. points from this input distribution, training and validation data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

5) TARGET SIGNAL 5

Target Signal 5 was used to test the proposed methods' approximation accuracy on a discontinuous signal, such as square waves [51], [52]. The waveform and its corresponding frequency spectrum are illustrated in Figures 2e and 3e. The mathematical definition of the signal is given by Equation 31:

$$f(x) = \sin(k_0 x) + \text{sign}[\sin(k_1 x)] + \text{sign}[\sin(k_2 x)] + \text{sign}[\sin(k_3 x)] \tag{31}$$

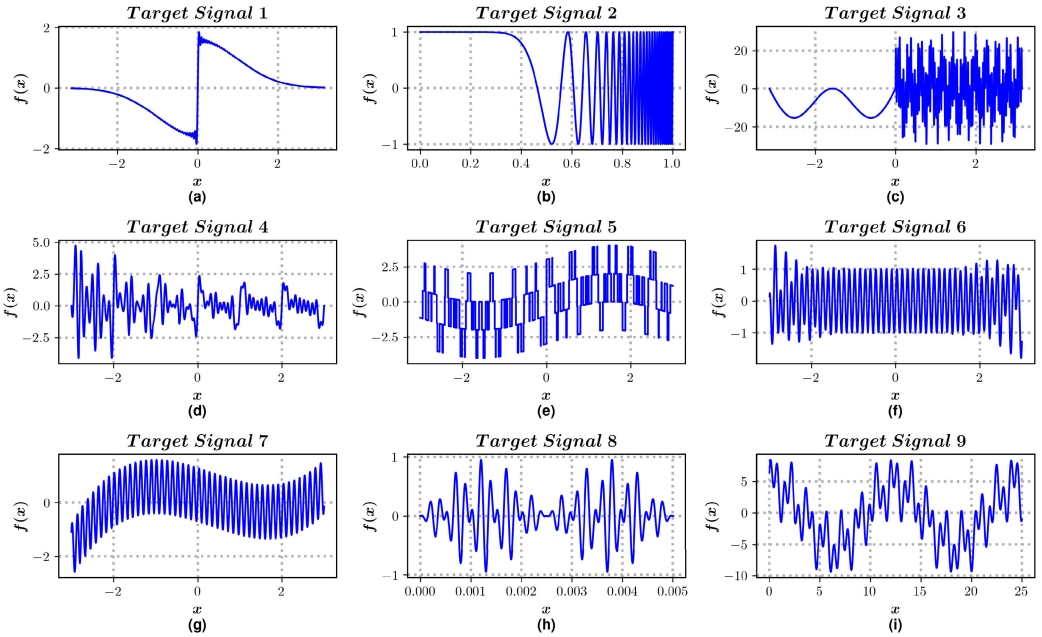

FIGURE 2. Signal waveforms for Experiment 1 (a) Target Signal 1 given in Equation 27. (b) Target Signal 2 given in Equation 28. (c) Target Signal 3 given in Equation 29. (d) Target Signal 4 given in Equation 30. (e) Target Signal 5 given in Equation 31. (f) Target Signal 6 given in Equation 32. (g) Target Signal 7 given in Equation 33. (h) Target Signal 8 given in Equation 34. (i) Target Signal 9 given in Equation 35.

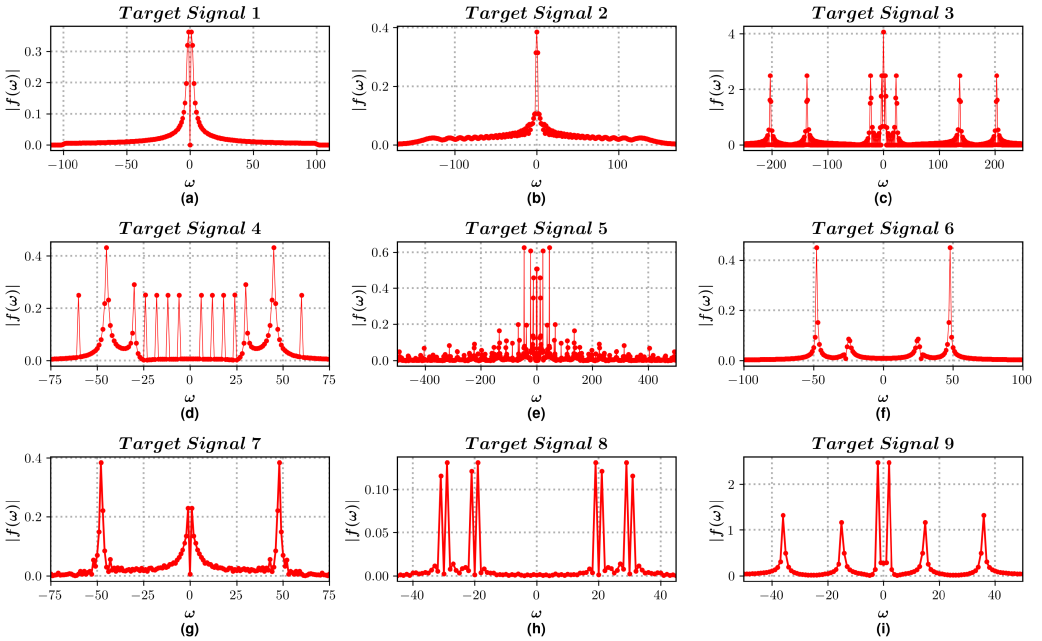

FIGURE 3. Signal spectrum waveforms for Experiment 1 (a) Target Signal 1 given in Equation 27. (b) Target Signal 2 given in Equation 28. (c) Target Signal 3 given in Equation 29. (d) Target Signal 4 given in Equation 30. (e) Target Signal 5 given in Equation 31. (f) Target Signal 6 given in Equation 32. (g) Target Signal 7 given in Equation 33. (h) Target Signal 8 given in Equation 34. (i) Target Signal 9 given in Equation 35.

In Equation 31, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$, and the *sign* is the signum function. Figures 2e and 3e display the target signal 5, and the Fourier transform of the signal, respectively, for the parameters $k_0 = 1, k_1 = 13\pi, k_2 = 23\pi, k_3 = 47\pi$. The spectrum of this signal compromises many highly irregular spikes with different amplitudes and also includes discontinuities in the spatial domain. For the experiment, input points $\{x_i\}_{i=1}^{N}$ were

sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(-\pi, +\pi)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling at 5120 i.i.d. points from this input distribution, training and validation data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

6) TARGET SIGNAL 6

In order to measure the approximation accuracy of the proposed methods on a polynomial signal with different sinusoidal components, another signal from literature was implemented [45], [53]. The waveform and its corresponding frequency spectrum are illustrated in Figures 2f and 3f. The mathematical definition of the signal is given by Equation 32:

$$f(x) = (x^3 - x)\frac{\sin(k_1 x)}{k_1} + \sin(k_2 x) \tag{32}$$

In Equation 32, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$. Figures 2f and 3f show target Signal 6 and the Fourier transform, respectively, for parameters $k_1 = 27, k_2 = 50$. For the experiment, input points $\{x_i\}_{i=1}^{N}$ were sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(-\pi, +\pi)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling at 5120 i.i.d. points from this input distribution, training and validation data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

7) TARGET SIGNAL 7

The signal of sine on the polynomial was used to investigate the neural networks' approximation capability, which is identical to the function implemented in the literature [3], [52]. Target Signal 7 is explicitly defined by Equation 33, whose waveform and corresponding frequency spectrum are shown in Figures 2g and 3g.

$$f(x) = 0.1\,x^3 - 0.1\,x^2 - 0.5\,x + 0.3 + \sin(50x) \tag{33}$$

In Equation 33, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$. Merely the low-frequency component of this signal can be easily approximated by using classical neural networks. However, it has been reported in the literature [52] that even when a large number of sampled points N is used, the high-frequency component cannot be fitted with ordinary neural networks. For the experiment, input points $\{x_i\}_{i=1}^{N}$ were sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(-\pi, +\pi)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling at 5120 i.i.d. points from this input distribution, training and validation data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

8) TARGET SIGNAL 8

The Double Sideband Suppressed Carrier (DSB-SC) transmission scheme is a method of amplitude modulation in which solely the sidebands are transmitted, while the carrier wave is suppressed and not transmitted. The transmission of the carrier signal is devoid of any informational content and is associated with power loss. Therefore, merely sidebands that carry information are transmitted. This leads to a reduction in the energy consumption associated with the transmission. The conserved energy may be incorporated into the two sidebands. Consequently, guarantees a more robust transmission that can cover extended ranges. During the process of suppression, the baseband signal remains unaffected. This signal was used to test the proposed methods' approximation precision. The waveform and its corresponding frequency spectrum are illustrated in Figures 2h and 3h. The mathematical definition of the signal is given by Equation 34:

$$f(x) = \frac{1}{2}\cos(k_1 x)(\cos(k_2 x) - \cos(k_3 x)) \tag{34}$$

In Equation 34, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$ and the parameters $k_1 = 10000\pi$, $k_2 = 1600\pi$ and $k_3 = 2400\pi$. The spectrum of this signal comprises several spikes with different amplitudes at extremely high frequencies. For the experiment, input points $\{x_i\}_{i=1}^{N}$ were sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(0, 5.10^{-3})$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling at only 5120 i.i.d. points from this input distribution, training and validation data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

9) TARGET SIGNAL 9

The signal defined by Equation 35 is frequently used to approximate the ENSO data set [52] in the literature. The waveform and its companion frequency spectrum are shown in Figures 2i and 3i.

$$\begin{aligned} f(x) = {} & 4.7\cos(2\pi\frac{x}{12}) + 1.1\sin(2\pi\frac{x}{12}) \\ & + 0.2\cos(2\pi\frac{x}{1.7}) + 2.7\sin(2\pi\frac{x}{1.7}) \\ & + 2.1\cos(2\pi\frac{x}{0.7}) + 2.1\sin(2\pi\frac{x}{0.7}) - 0.5 \end{aligned} \tag{35}$$

In Equation 35, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$. It has been reported that this signal is more sophisticated than expressed in Equation 33 [52]. For the experiment, input points $\{x_i\}_{i=1}^{N}$ were sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(0, 25)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling at 5120 i.i.d. points from this input distribution, training and validation data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

10) TARGET SIGNAL 10

The extended version of the signal defined in [6] is used to investigate frequency/spectral bias experiments and is given in Equation 36. The waveform and its companion frequency spectrum are shown in Figures 4a and 4b. This signal contains both low and high-frequency components with different phase angles. Therefore, it helps investigate the spectral bias

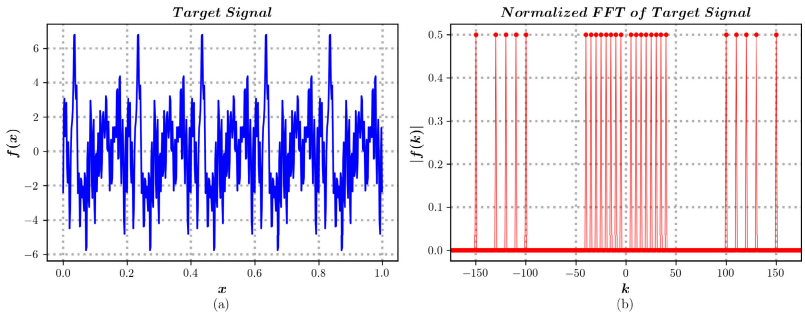

FIGURE 4. Spatial and spectral domain representation of Target Signal 10. (a) Spatial Coordinate vs. Amplitude (b) Normalized FFT of The Signal Representation in Spectral Domain.

phenomena of neural networks.

$$\begin{aligned} f(x) = {} & A_1 \sin(2\pi.5\, x + \phi_1) + A_2 \sin(2\pi.10x + \phi_2) \\ & + A_3 \sin(2\pi.15\, x + \phi_3) + A_4 \sin(2\pi.20\, x + \phi_4) \\ & + A_5 \sin(2\pi.25\, x + \phi_5) + A_6 \sin(2\pi.30\, x + \phi_6) \\ & + A_7 \sin(2\pi.35\, x + \phi_7) + A_8 \sin(2\pi.40\, x + \phi_8) \\ & + A_9 \sin(2\pi.100\, x + \phi_9) + A_{10} \sin(2\pi.110\, x + \phi_{10}) \\ & + A_{11} \sin(2\pi.120\, x + \phi_{11}) + A_{12} \sin(2\pi.130\, x + \phi_{12}) \\ & + A_{13} \sin(2\pi.150\, x + \phi_{13}) \end{aligned} \tag{36}$$

In Equation 36, $f : \mathbb{R} \to \mathbb{R}$ is a one-dimensional function, x_i and $y_i \in \mathbb{R}$. For the experiment, input points $\{x_i\}_{i=1}^{N}$ were sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(0, 1)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling 500 i.i.d. points from this input distribution, $\{A_i\}_{i=1}^{13} = 1.0$, and $\phi_i \overset{iid}{\sim} \{\mathcal{U}(0, 2\pi)\}_{i=1}^{13}$. Using these parameters, training and validation data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{val} = \{(x_i, y_i)\}_{i=1}^{N_{val}}$, were generated.

G. PERFORMANCE METRICS

In this study, we consider only regression-based modeling. The performance metrics implemented for the comparison of the models are given in Equations 37, 38, and 39.

$$\text{MSE}(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2 \tag{37}$$

The Mean Square Error (MSE) [58] is given by Equation 37. Where y is the target label, $\hat{y}$ is the model prediction value, and N is the total number of data points of the sampled target signal. It is a risk function that corresponds to the expected value of the squared (quadratic) error or loss. This can be defined as the average squared difference between the estimated and actual values.

The Mean Absolute Error (MAE) is a metric [58] that quantifies the average magnitude of errors in a set of predictions, disregarding their direction. The mean absolute difference between the predicted values and the actual values is a metric utilized to evaluate the efficacy of a regression model. The definition of this metric is given in Equation 38.

$$\text{MAE}(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} \left|(y_i - \hat{y}_i)\right| \tag{38}$$

The definition of the coefficient of determination (R^2) is given in Equation 39 [58]. In this Equation, y is the target label, $\hat{y}$ is the model prediction value, and N is the number of total data points of the sampled target signal. This represents the proportion of variance of the target label that has been explained by the independent variables in the model. It indicates the goodness of fit and, therefore, measures how well the model will likely predict the unseen samples through the proportion of explained variance.

$$\text{R}^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{N} \left(y_i - \bar{y}_i\right)^2} \;\; and \;\; \bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i \tag{39}$$

For spectral/frequency bias evaluations, we defined and used The Projection Based Relative Error (PBRE) and The Frequency Band Correspondence Metrics (FBCM). Their explicit mathematical definitions are given in Equations 40 and 41: Projection Based Relative Error metric [2] was implemented to measure the spectral bias at different frequencies. In this method, Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ is a dataset with N sample points, the Fourier Transform of this data set is $\mathcal{F}\{\{(x_i, y_i)\}_{i=1}^{N}\}$ can be expressed as $\hat{y}_{\boldsymbol{k}} = \frac{1}{N} \sum_{i=1}^{N} y_i e^{-\text{i}2\pi \boldsymbol{k} \cdot \boldsymbol{x}_i}$. In order to investigate the convergence behavior of various frequency components during training, the relative discrepancy between the neural network prediction and the target function for specified important frequencies k at each training step can be computed. The definition of this metric is given by Equation 40:

$$\Delta_{y,u}(k) = \frac{\left|\hat{y}_k - \hat{u}_k\right|}{\left|\hat{u}_k\right|} \tag{40}$$

where $\hat{y}_k$ and $\hat{u}_k$ are the one-dimensional Fourier transforms of $\{y_i\}_{i=1}^{N}$ and the interrelated neural network output $\{u_i\}_{i=1}^{N}$, respectively, along the first principal component of the input space. During training, each response frequency component, $\{u_i\}_{i=1}^{N}$, of the neural network model output changes and converges to a final value.

In addition to the Projection Based Relative Error metric, we also used the Frequency Band Correspondence Metric [42] to investigate the spectral bias. This metric analyzes the input-output likeness in the frequency domain across several frequency bands. It was proposed to examine the Fourier spectrum of the output signal $\hat{y}(k),\ k = 1, \ldots, N$ to indicate the convergence dynamics of the distinct frequency elements of the target signal. The frequency spectrum of the output signal $\hat{y}$ can be obtained using the Fast Fourier transform, denoted as $\mathcal{F}\{\hat{y}_q(k)\}$ for each training step q (epoch). Similarly, the Fast Fourier transform for the target signal $\hat{u}$, denoted as $\mathcal{F}\{\hat{u}_q(k)\}$ is computed, and then the calculation of an element-wise ratio between both transforms is given in Equation 41.

$$\text{FBCM}_q(\hat{y}, \hat{u}) = \frac{\mathcal{F}\{\hat{y}_q(k)\}}{\mathcal{F}\{\hat{u}_q(k)\}} \tag{41}$$

Intuitively, $FBCM_q$ indicates the degree to which any model prediction in the training step corresponds to the real signal in the frequency domain. As a result, the closer the values are to 1, the higher the correspondence. Because it is interested in the spectral bias of the neural network model, its frequency spectrum can be separated into N subgroups fitting to N non-overlapping frequency bands as the similitude map. Because the similitude map is balanced around the center frequency, it can be uniformly grouped according to the span between its elements and the center. The mean value of the N spectral band was then calculated.

III. EXPERIMENTS

The present investigation utilized a thorough experimental approach to assess the effectiveness of the GaborQResNet and FourierQResNet models in comparison to conventional methods and commonly employed neural network models for the purpose of sensor signal regression. The study comprised five discrete experiments, each aimed at assessing distinct facets of the models' performance and capabilities.

The first experiment involved an empirical investigation of the learning and convergence capacities of the proposed models. This study conducted a comparative analysis of the performance outcomes of various model architectures, namely MLP, QResNet, QNN, FourierNet, and GaborNet, on a dataset comprising nine complex test signals. The models were subjected to a training process with pre-established learning rates, unchanging layers, and neuron dimensions, and without any exploration or adjustment of the hyperparameters. The training procedure utilized Stochastic Gradient Descent optimization and the models were assessed based on their capacity to acquire intricate target signals within constrained circumstances.

The second experiment was conducted to investigate the impact of changes in the depth of the hidden layer on the proposed models. The models, namely GaborNet, GaborQResNet, FourierNet, and FourierQResNet, underwent training and testing procedures utilizing test signals with varying layer depths. The training of the models was conducted using a consistent learning rate and constant neuron size, while the layer depth was varied. The aim of this study is to evaluate the effects of varying layer depths on the learning and convergence abilities of the models.

Experiment 3 aimed to investigate the impact of alterations in the number of neurons within the hidden layers of the proposed models. The GaborNet, GaborQResNet, FourierNet, and FourierQResNet architectures were trained and evaluated using piecewise and square wave signals. The neuron size for each hidden layer varied during the experiment. The objective was to examine the impact of varying neuron sizes on the learning and convergence capabilities of the models while maintaining a consistent layer depth.

The objective of Experiment 4 was to investigate the impact of fluctuations in dataset size on the learning and convergence abilities of the models. The GaborNet, GaborQResNet, FourierNet, and FourierQResNet architectures underwent training and evaluation of the test signals that exhibited diverse quantities of data points. The models were trained with unvarying learning rates, unchanging neuron sizes, and consistent layer depths. The aim of this study was to assess the impact of varying dataset sizes on the capacity of models to acquire intricate target signals.

The fifth experiment was conducted with the aim of examining the spectral bias and frequency principle in the proposed models in comparison with other artificial neural network models. The study employed three distinct methodologies, including the projection-based relative error and the frequency band correspondence metrics, to validate the conjecture that the suggested models demonstrate superior efficacy in the presence of spectral bias and are comparatively less susceptible to its impact. In general, the experiments involved a comprehensive exploration of multiple facets related to the performance of the proposed models. These facets include the models' learning capability, the impact of layer depth and neuron size, variations in dataset size, and spectral bias. The findings derived from these inquiries serve to enhance comprehension of the proposed models' abilities and offer perspectives on their potential benefits compared to current methodologies in the realm of sensor signal regression.

It is worth saying that computational experiments have been carried out on a system with an Intel i7 CPU 3.0 GHz, 128 GB of RAM, and NVIDIA Titan X GPU.

IV. RESULTS

The results of all experiments are shared in this section.

A. EXPERIMENT 1: MODELS' LEARNING CAPABILITY

In this experiment, we conducted an empirical investigation into the learning and convergence capacities of the proposed models. The performance results of our proposed neural network models (GaborQResNet and FourierQResNet) were compared with those of other model architectures (MLP, QResNet, QNN, FourierNet, and GaborNet) using a set of nine intricate test signals. These chosen nine test signals

encompass a variety of frequency and phase information, rendering it challenging to acquire proficiency using conventional neural networks. Signal sampling was conducted in accordance with their respective characteristics in order to produce datasets for both training and testing purposes. Each model underwent training with predetermined learning rates, fixed layers, and neuron sizes without the application of any hyperparameter search or tuning method. For this experiment, the learning rate was set to 0.001, the layer size was set to 6, and the number of neurons allocated for each layer was 256. During the training process, a batch size of 64 was chosen, and optimization was performed using Stochastic Gradient Descent (SGD) optimizer. The models underwent training for 1000 epochs using SGD and were tested on nine distinct signals. Training and test results were recorded. This process was repeated ten times using different random seeds. All the models were initialized according to the patterns presented in Section II. The Adam optimizer was intentionally excluded from this experiment in order to examine the true convergence patterns of the models. The utilization of the Adam optimizer in the optimization processes can prove to be highly efficient and expeditious because of its ability to adapt to the learning rate. This study evaluated the learning capacity of the models in acquiring complex target signals under restricted conditions, including a fixed layer size, predetermined neuron size, and a limited number of training epochs. Upon concluding the training, an examination was conducted on the training loss and test loss values, as well as the performance metrics that were previously established. The definitions of all the test signals and metrics utilized in this experiment are presented in Sections II-F and II-G. Figure 5 shows the training and test loss curves for all models under investigation for each test signal. The results show that the GaborQResNet and FourierQResNet models converge to the lowest loss value the fastest for each test signal without exception. In addition, it was found that the convergence rates of the GaborNet and FourierNet models were much slower and could converge to a rather high loss value. Finally, it was observed that the QResNet, MLP, and QNN models could not effectively learn and converge any of the test signals used. In light of these results, for all target signals used, all the models under investigation can be evaluated by dividing them into three different groups. The first group is models that cannot effectively learn the target signals, including MLP, QNN, and QResNet. The second group is more successful than the first group, which includes GaborNet and FourierNet models but does not show sufficient learning performance. The third group consists of models with fast convergence and high learning capacity, in which the GaborQResNet and FourierQResNet models are suggested in this study. As a result, the following experiment results will be explained according to these model groups.

In Table 1, the training and test loss values of all models with respect to the target signals are listed. When the results were examined, we discovered that the FourierQResNet and GaborQResNet models had the lowest training and test loss values for all the signals. Especially for target signals 3, 5, and 9, when the proposed novel models were compared with the other models, significant differences were observed in the final values of the training and test loss values. In particular, it was determined that GaborQResNet could achieve much lower training and test loss values than FourierQResNet and other models. The MLP, QNN, and QResNet models were found to have the highest training and test loss values. In addition, it was observed that the FourierNet and GaborNet models could not learn the target signals sufficiently, but they provided a very limited performance improvement compared to MLP, QNN, and QResNet.

An additional analysis was conducted to compare the performance metrics of all the models with respect to all the target signals. The results of this analysis are given are presented in Table 2. We discovered that the training and test *MSE* values of FourierQResNet and GaborQResNet were the lowest among all target signals. GaborQResNet achieved the best *MSE* values among all the models for all target signals. The same outcomes were observed for the training and test *MAE* values with FourierQResNet and GaborQResNet for all the target signals. When the test and training R^2 values of the models were examined, it was determined that the MLP, QNN, and QResNet models remained at very low values for all target signals and could not learn the variance in the signal sufficiently; however, all other models reached the maximum value of 1.0.

In Figure 6, the prediction results obtained using the test dataset of all models are visualized for each test signal after the training process was completed. As the figure shows, our proposed novel neural network models exhibit exemplary prediction performance for all very complex target signals. However, the MLP, QNN, and QResNet models did not deliver adequate prediction performances for all target signals.

This experiment confirms one of our hypotheses and shows that our proposed neural network models, FourierQResNet and GaborQResNet, have great convergence and learning capabilities for learning very complex signals. However, we performed this experiment using a specified number of dataset points, fixed layer sizes, and a predefined number of neurons. To further observe the performance of the proposed models and the effect of the variation in hyperparameters, we performed complementary experiments to investigate these models' performances. In the following sections, we discuss the outcomes of the additional experiments.

B. EXPERIMENT 2: THE EFFECT OF VARIATION IN HIDDEN LAYER DEPTH FOR THE PROPOSED MODELS

In this experiment, we empirically investigated the effect of changing the layer depth on the learning and convergence capabilities of the GaborNet, GaborQResNet, FourierNet, and FourierQResNet models by using the test signals given in Equation 30 and Equation 32. All signals were sampled according to their characteristics to generate training and test datasets. All models were trained with fixed learning rates, fixed neuron sizes with changing layer depth sizes

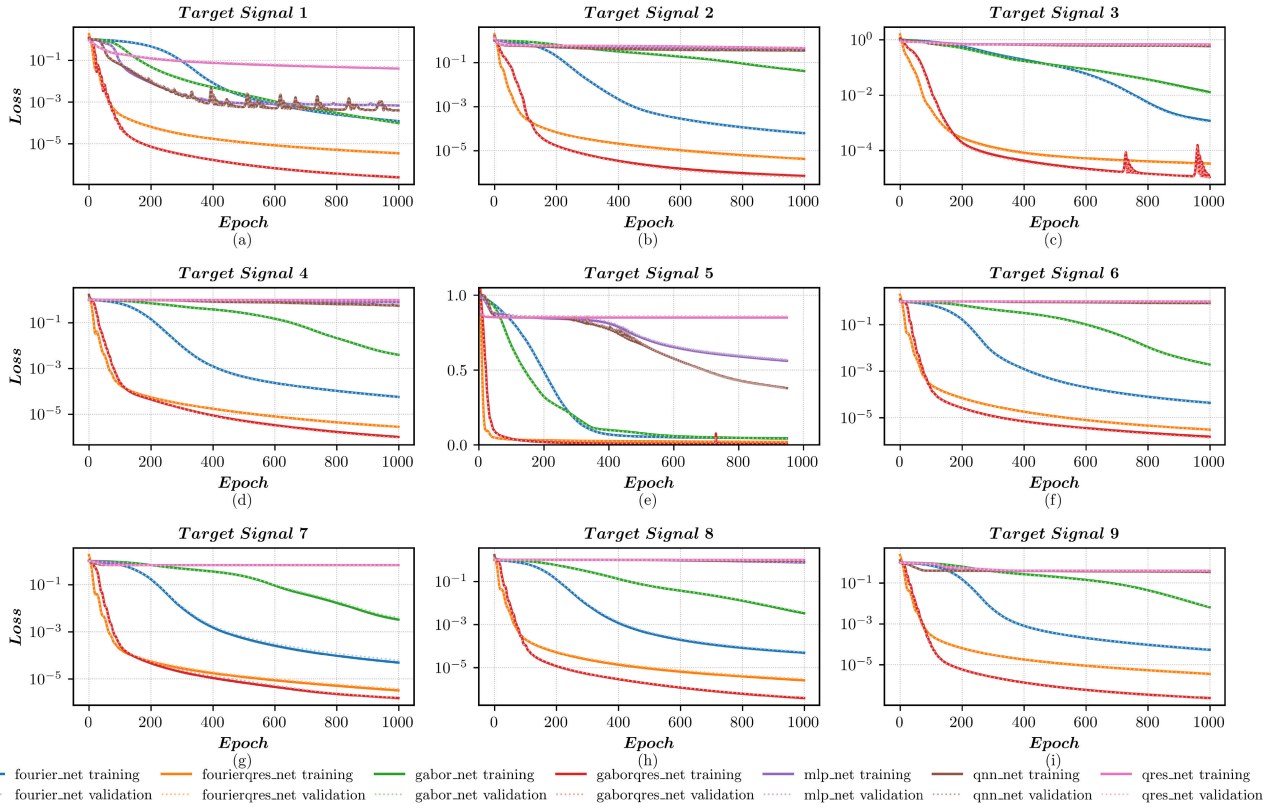

FIGURE 5. Experiment 1: Training and test loss curves for all models (a) Target Signal 1 given in Equation 27. (b) Target Signal 2 given in Equation 28. (c) Target Signal 3 given in Equation 29. (d) Target Signal 4 given in Equation 30. (e) Target Signal 5 given in Equation 31. (f) Target Signal 6 given in Equation 32. (g) Target Signal 7 given in Equation 33. (h) Target Signal 8 given in Equation 34. (i) Target Signal 9 given in Equation 35 (y-axis is log-scaled for all plots).

TABLE 1. Experiment 1: model training and test loss values for all target signals.

	FourierNet		FourierQResNet		GaborNet		GaborQResNet		MLP (FFNN)		QNN		QResNet	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Signal-1	8.32e-5	8.33e-5	2.43e-6	2.42e-6	6.70e-5	6.63e-5	1.67e-7	1.70e-7	4.62e-4	4.55e-4	2.74e-4	2.70e-4	2.75e-2	3.20e-2
Signal-2	3.23e-5	3.19e-5	2.16e-6	2.00e-6	2.14e-2	2.35e-2	3.69e-7	3.14e-7	2.09e-1	2.12e-1	1.80e-1	1.85e-1	2.41e-1	2.44e-1
Signal-3	1.27e-1	1.15e-1	3.60e-3	3.44e-3	1.40e+0	1.23e+0	1.19e-3	1.17e-3	6.90e+1	6.67e+1	6.33e+1	6.07e+1	7.35e+1	7.13e+1
Signal-4	8.97e-5	8.83e-5	4.39e-6	4.41e-6	6.22e-3	6.20e-3	1.59e-6	1.46e-6	1.23e+0	1.22e+0	8.79e-1	8.79e-1	1.53e+0	1.52e+0
Signal-5	1.51e-1	1.50e-1	6.96e-2	6.85e-2	1.42e-1	1.40e-1	2.10e-2	2.41e-2	1.90e+0	1.92e+0	1.25e+0	1.25e+0	2.92e+0	2.96e+0
Signal-6	2.44e-5	2.40e-5	1.69e-6	1.67e-6	1.07e-3	1.06e-3	8.46e-7	7.58e-7	5.42e-1	5.42e-1	4.73e-1	4.76e-1	5.53e-1	5.51e-1
Signal-7	3.69e-5	4.69e-5	2.46e-6	2.87e-6	2.44e-3	2.98e-3	1.14e-6	1.17e-6	5.01e-1	4.98e-1	5.03e-1	4.99e-1	5.04e-1	5.00e-1
Signal-8	6.36e-6	7.05e-6	3.30e-7	3.88e-7	4.23e-4	4.76e-4	4.91e-8	5.31e-8	9.75e-2	9.59e-2	1.15e-1	1.13e-1	1.29e-1	1.27e-1
Signal-9	1.11e-3	1.14e-3	7.40e-5	7.32e-5	1.30e-1	1.36e-1	4.79e-6	4.77e-6	7.73e+0	7.79e+0	7.04e+0	7.08e+0	8.05e+0	8.11e+0

(*layer depth/size* $\in \{2, 3, 4, 5, 6, 7, 8\}$), and no specific hyperparameter search/tuning method was applied. For this experiment, we set the learning rate to 0.001, changed the layer size of each model from 2 to 8, and fixed the number of neurons in each layer to 256. For training, the batch size was set to 64, and the Stochastic Gradient Descent (SGD) optimizer was used for the optimization. All models were trained in 2000 epochs with SGD, and the training and test progress were recorded for each epoch. Again, we intentionally did not use the Adam optimizer in this experiment to investigate the actual convergence behavior of the models. We tested the learning capability of the models for given complex target signals with a changing layer depth in limited training epochs. After completing the training, we investigated training loss and test loss values and performance metrics defined in the previous section (Section II-F and II-G). Results of this experiment is given in Figures 7 and 8, and Tables 3, 4, and 5.

From Table 3, we discovered that FourierNet, GaborNet, and our proposed models' loss values are inversely correlated with the layer size. This result means that the training and test loss values diminish when the layer/dept size of the model increases. This empirical observation is an expected result, as increasing the layer size, increases model complexity and capacity. However, when the FourierQResNet and GaborQResNet loss values were examined closely, even with two layers, their loss values were significantly lower than

TABLE 2. Experiment 1: comparison of all model metrics against all target signals.

		FourierNet		FourierQResNet		GaborNet		GaborQResNet		MLP (FFNN)		QNN		QResNet	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Signal 1	*MSE*	8.32e-5	8.33e-5	2.43e-6	2.42e-6	6.70e-5	6.63e-5	1.67e-7	1.70e-7	4.62e-4	4.55e-4	2.74e-4	2.70e-4	2.75e-2	3.20e-2
	MAE	7.28e-3	7.26e-3	1.26e-3	1.26e-3	5.44e-3	5.42e-3	3.09e-4	3.10e-4	7.55e-3	7.58e-3	7.00e-3	6.99e-3	5.35e-2	5.72e-2
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	1.000	1.000	0.960	0.954
Signal 2	*MSE*	3.23e-5	3.19e-5	2.16e-6	2.00e-6	2.14e-2	2.35e-2	3.69e-7	3.14e-7	2.09e-1	2.12e-1	1.80e-1	1.85e-1	2.41e-1	2.44e-1
	MAE	4.38e-3	4.40e-3	1.07e-3	1.06e-3	5.07e-2	5.24e-2	3.49e-4	3.48e-4	2.73e-1	2.77e-1	2.58e-1	2.63e-1	3.39e-1	3.43e-1
	R^2	1.000	1.000	1.000	1.000	0.959	0.956	1.000	1.000	0.601	0.601	0.655	0.653	0.540	0.541
Signal 3	*MSE*	1.27e-1	1.15e-1	3.60e-3	3.44e-3	1.40e+0	1.23e+0	1.19e-3	1.17e-3	6.90e+1	6.67e+1	6.33e+1	6.07e+1	7.35e+1	7.13e+1
	MAE	2.13e-1	2.12e-1	2.37e-2	2.40e-2	3.97e-1	3.74e-1	1.62e-2	1.61e-2	4.91e+0	4.81e+0	4.76e+0	4.65e+0	5.08e+0	4.98e+0
	R^2	0.999	0.999	1.000	1.000	0.987	0.988	1.000	1.000	0.363	0.372	0.417	0.429	0.322	0.330
Signal 4	*MSE*	8.97e-5	8.83e-5	4.39e-6	4.41e-6	6.22e-3	6.20e-3	1.59e-6	1.46e-6	1.23e+0	1.22e+0	8.79e-1	8.79e-1	1.53e+0	1.52e+0
	MAE	7.54e-3	7.48e-3	1.65e-3	1.66e-3	3.58e-2	3.57e-2	6.64e-4	6.65e-4	7.74e-1	7.74e-1	6.46e-1	6.47e-1	8.94e-1	8.90e-1
	R^2	1.000	1.000	1.000	1.000	0.996	0.996	1.000	1.000	0.221	0.219	0.445	0.438	0.032	0.028
Signal 5	*MSE*	1.51e-1	1.50e-1	6.96e-2	6.85e-2	1.42e-1	1.40e-1	2.10e-2	2.41e-2	1.90e+0	1.92e+0	1.25e+0	1.25e+0	2.92e+0	2.96e+0
	MAE	2.60e-1	2.58e-1	1.51e-1	1.50e-1	2.46e-1	2.43e-1	6.20e-2	6.33e-2	1.11e+0	1.12e+0	8.96e-1	8.99e-1	1.48e+0	1.49e+0
	R^2	0.956	0.956	0.980	0.980	0.959	0.959	0.994	0.993	0.447	0.442	0.637	0.636	0.151	0.140
Signal 6	*MSE*	2.44e-5	2.40e-5	1.69e-6	1.67e-6	1.07e-3	1.06e-3	8.46e-7	7.58e-7	5.42e-1	5.42e-1	4.73e-1	4.76e-1	5.53e-1	5.51e-1
	MAE	3.68e-3	3.69e-3	1.03e-3	1.02e-3	1.66e-2	1.65e-2	4.87e-4	4.78e-4	6.50e-1	6.52e-1	5.92e-1	5.97e-1	6.55e-1	6.56e-1
	R^2	1.000	1.000	1.000	1.000	0.998	0.998	1.000	1.000	0.024	0.021	0.149	0.140	0.004	0.003
Signal 7	*MSE*	3.69e-5	4.69e-5	2.46e-6	2.87e-6	2.44e-3	2.98e-3	1.14e-6	1.17e-6	5.01e-1	4.98e-1	5.03e-1	4.99e-1	5.04e-1	5.00e-1
	MAE	4.27e-3	4.65e-3	1.19e-3	1.28e-3	2.25e-2	2.46e-2	5.21e-4	5.55e-4	6.37e-1	6.34e-1	6.38e-1	6.35e-1	6.39e-1	6.36e-1
	R^2	1.000	1.000	1.000	1.000	0.997	0.996	1.000	1.000	0.332	0.330	0.331	0.329	0.329	0.328
Signal 8	*MSE*	6.36e-6	7.05e-6	3.30e-7	3.88e-7	4.23e-4	4.76e-4	4.91e-8	5.31e-8	9.75e-2	9.59e-2	1.15e-1	1.13e-1	1.29e-1	1.27e-1
	MAE	2.00e-3	2.10e-3	4.66e-4	5.04e-4	1.02e-2	1.08e-2	1.71e-4	1.80e-4	2.29e-1	2.27e-1	2.43e-1	2.40e-1	2.65e-1	2.61e-1
	R^2	1.000	1.000	1.000	1.000	0.997	0.996	1.000	1.000	0.246	0.242	0.114	0.103	0.001	-0.000
Signal 9	*MSE*	1.11e-3	1.14e-3	7.40e-5	7.32e-5	1.30e-1	1.36e-1	4.79e-6	4.77e-6	7.73e+0	7.79e+0	7.04e+0	7.08e+0	8.05e+0	8.11e+0
	MAE	2.47e-2	2.48e-2	6.74e-3	6.68e-3	1.61e-1	1.64e-1	1.73e-3	1.73e-3	2.25e+0	2.26e+0	2.14e+0	2.15e+0	2.30e+0	2.31e+0
	R^2	1.000	1.000	1.000	1.000	0.994	0.993	1.000	1.000	0.624	0.620	0.657	0.654	0.608	0.604

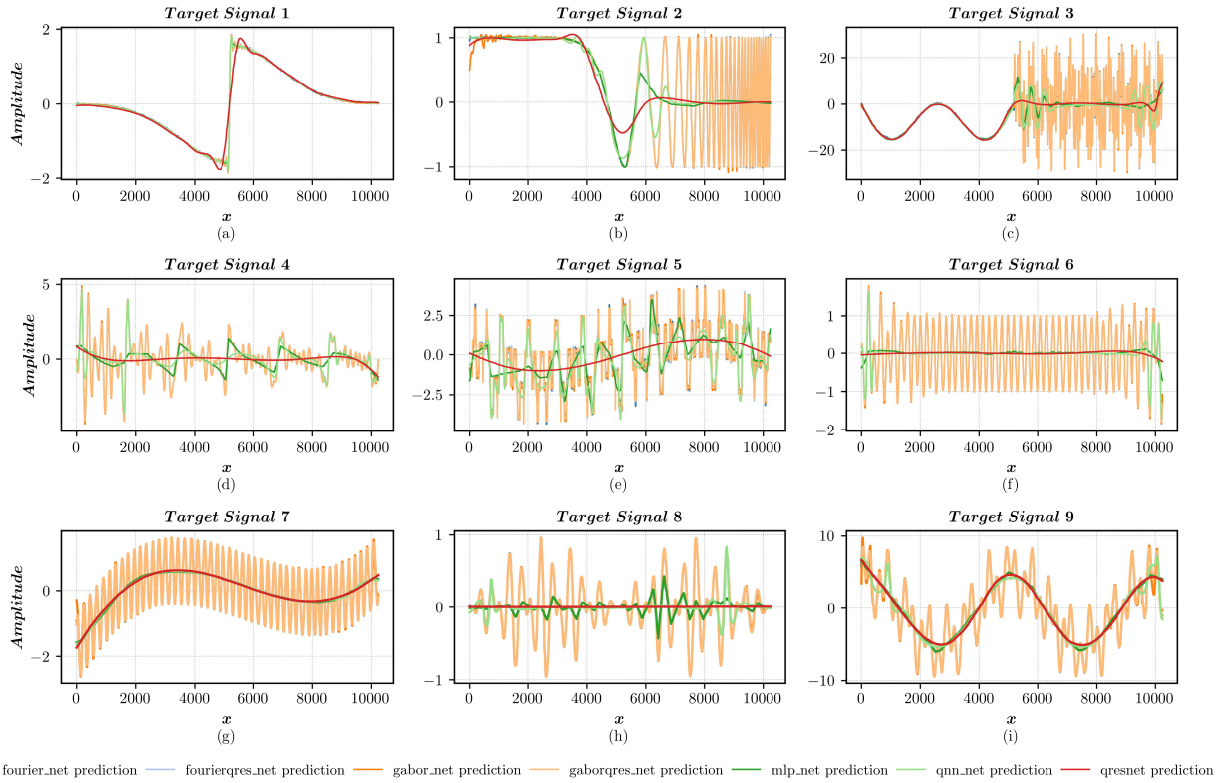

FIGURE 6. Model predictions for all signals (on the test data set) (a) Target Signal 1 given in Equation 27. (b) Target Signal 2 given in Equation 28. (c) Target Signal 3 given in Equation 29. (d) Target Signal 4 given in Equation 30. (e) Target Signal 5 given in Equation 31. (f) Target Signal 6 given in Equation 32. (g) Target Signal 7 given in Equation 33. (h) Target Signal 8 given in Equation 34. (i) Target Signal 9 given in Equation 35.

those of FourierNet and GaborNet. As a result, we can infer that our proposed models have much higher learning capacity and complexity. For both Target Signals 4 and 6, when we increased the layer/depth size, we discovered that the training and test loss values were improved, but this improvement was not significant. Owing to the learning capacity of the

TABLE 3. Experiment 2: comparison of model loss values for Target Signals 4 and 6.

		Layer Size 2		Layer Size 3		Layer Size 4		Layer Size 5		Layer Size 6		Layer Size 7		Layer Size 8	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Signal 4	*FourierNet*	1.15e+0	1.17e+0	6.46e-5	6.09e-5	6.78e-5	6.74e-5	8.97e-5	8.76e-5	1.10e-4	1.06e-4	1.33e-4	1.35e-4	2.25e-4	2.24e-4
	FourierQResNet	7.79e-6	7.57e-6	7.32e-6	6.88e-6	8.33e-6	8.32e-6	6.84e-6	6.81e-6	4.38e-6	4.30e-6	2.86e-6	2.83e-6	2.87e-6	2.85e-6
	GaborNet	1.46e+0	1.49e+0	4.23e-1	4.33e-1	4.68e-2	4.48e-2	4.62e-3	4.29e-3	2.11e-3	2.04e-3	4.75e-3	4.22e-3	6.93e-3	6.75e-3
	GaborQResNet	2.72e-6	2.49e-6	2.34e-6	2.73e-6	1.62e-6	1.69e-6	1.44e-6	1.16e-6	3.51e-6	4.01e-6	2.68e-6	2.69e-6	2.66e-6	2.65e-6
Signal 6	*FourierNet*	4.46e-1	4.42e-1	5.68e-6	4.30e-6	1.76e-5	1.62e-5	2.10e-5	2.04e-5	2.29e-5	2.21e-5	2.27e-5	2.23e-5	2.16e-5	2.11e-5
	FourierQResNet	2.69e-6	2.56e-6	2.66e-6	2.60e-6	2.46e-6	2.44e-6	2.13e-6	2.11e-6	1.41e-6	1.38e-6	1.10e-6	1.12e-6	8.21e-7	7.90e-7
	GaborNet	5.37e-1	5.36e-1	1.47e-1	1.46e-1	1.63e-2	1.58e-2	1.25e-3	1.11e-3	3.88e-4	3.43e-4	3.62e-4	3.26e-4	2.73e-4	2.45e-4
	GaborQResNet	8.36e-7	8.33e-7	9.61e-7	9.48e-7	1.04e-6	1.02e-6	9.24e-7	6.92e-7	8.84e-7	7.77e-7	1.11e-6	9.42e-7	5.65e-7	4.66e-7

proposed models, we can reach very low loss values even with a smaller number of layers, and adding more layers does not provide any significant improvement in either the training or test loss values.

In Figure 7 (a) and (c), the loss values vs. layer/depth size results are shown for Target Signal 4. Owing to the complex structure and several high-frequency components of the target signal, the final test loss value of FourierNet is approximately 2.24×10^{-4}. When we set the layer size to 3, the loss values did not improve even after adding more layers. For GaborNet, its final test loss value was 6.75×10^{-3}, and after using five layers, adding more layers into its architecture did not help further improve the training and test loss values, even worsening the loss values.

From Figure 7 (b) and (d), it can be observed that the FourierQResNet model's training and test loss values were enhanced by including additional layers in its architecture. However, it remained constant when the layer/depth size was set to 7 and 8. Nevertheless, when we set layer/dept size from 4 to 7, we can observe a minimal improvement for the loss values, but this improvement was very small and not significant, and the test and training loss values were approximately 2.85×10^{-6} and 2.87×10^{-6}. The best results for FourierQResNet were obtained by implementing seven layers with training and test loss values around 2.86×10^{-6} and 2.83×10^{-6}. Considering GaborQResNet, our experimental results showed that the model's training and test loss values were not significantly decreased by including additional layers in its architecture. In addition, the final training and test loss values were roughly 2.66×10^{-6} and 2.65×10^{-6} for eight layers. The best model results were obtained by implementing five layers with loss values 1.44×10^{-6} and 1.16×10^{-6}. As we can realize from these results, the proposed architectures have much higher learning capacity and model complexity than the previous models (GaborNet, FourierNet), and GaborQResNet is better than FourierQResNet for learning Target Signal 4 by using a lower number of depths/layer sizes. Their performance and accuracy are higher than those of the original models using minimal layer sizes.

In Figure 8 (a) and (c), the training and test loss values vs. layer/depth size results are shown for Target Signal 6. Target Signal 6 is less complex and includes fewer frequency components than target signal 4. For this signal, the final training and test loss values of the FourierNet area are approximately 2.16×10^{-5} and 2.11×10^{-5}. Again, when we set the layer size to 3, the loss values did not improve even after adding more layers. However, for this signal, lower training and testing loss values were achieved using FourierNet. GaborNet's final test loss value was 2.45×10^{-4} by implementing eight layers. When six layers were used in GaborNet, the training and test loss values were considered minimal, and adding more layers to the model did not significantly improve the training and test loss values. From Figure 8 (b) and (d), it can be observed that the FourierQResNet model's training and test loss values were enhanced by incorporating additional layers into its architecture. The best results for FourierQResNet were obtained by implementing eight layers with training and test loss values around 8.21×10^{-7} and 7.90×10^{-7}. Regarding the GaborQResNet, our experimental results indicated that the model's training and test loss values were not significantly decreased by including extra layers in its architecture. In addition, this model's final loss training and test values were roughly 5.65×10^{-7} and 4.66×10^{-7} for eight layers, and this was the best model result. However, GaborQResNet reached at test loss value of approximately 6.92×10^{-7} with five layers, and although more layers were added to the model, the loss value changed at the level of 10^{-7}. We inferred that loss value levels of 10^{-7} were reached with five layers, but the same loss value level was reached by implementing up to eight layers for FourierQResNet. As we can realize from these results, the proposed architectures have much more learning capacity and model complexity than the previous models (GaborNet, FourierNet), and GaborQResNet is better than FourierQResNet for learning target Signal 6 by using a lower number of depth/layer sizes. Again, their performance and accuracy are beyond those of the original models using minimal layer sizes.

Table 4 lists *MSE*, *MAE*, and R^2 metric values vs. layer size results for target signal 4. According to these results, we observed the same behavior for all four models as for the loss values when we investigated these metrics. FourierNet reached a maximum R^2 value of 1.0 with three layers. GaborNet achieved a maximum R^2 value of 0.999 with six layers and decreased to 0.996 with eight layers. For our proposed models, FourierQResNet and GaborQResNet obtained R^2 value of 1.0 using only two layers. For the *MSE* and *MAE*

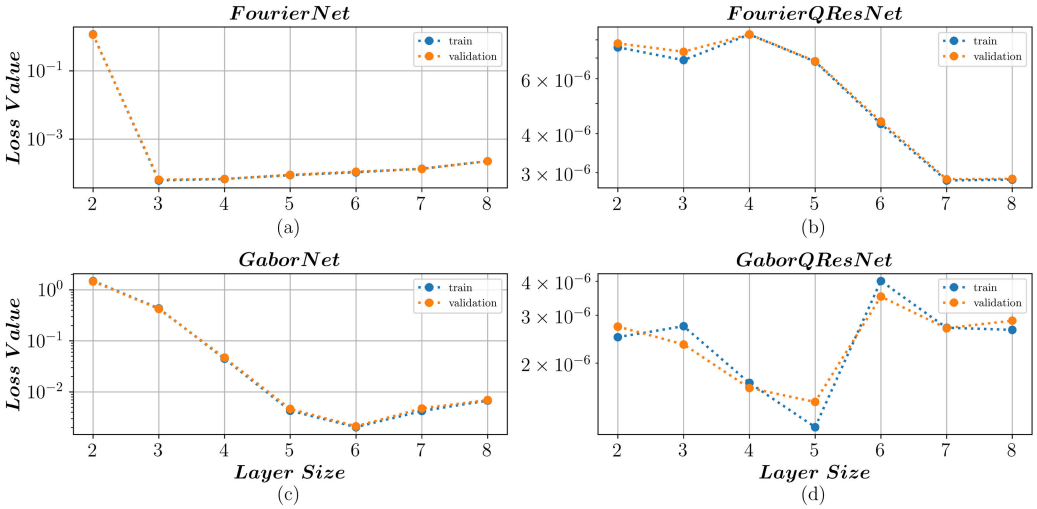

FIGURE 7. Experiment 2: loss value vs layer size for Target Signal - 4 (a) FourierNet (b) FourierQResNet (c) GaborNet (d) GaborQResNet Neural Network Models (y-axis in log scale).

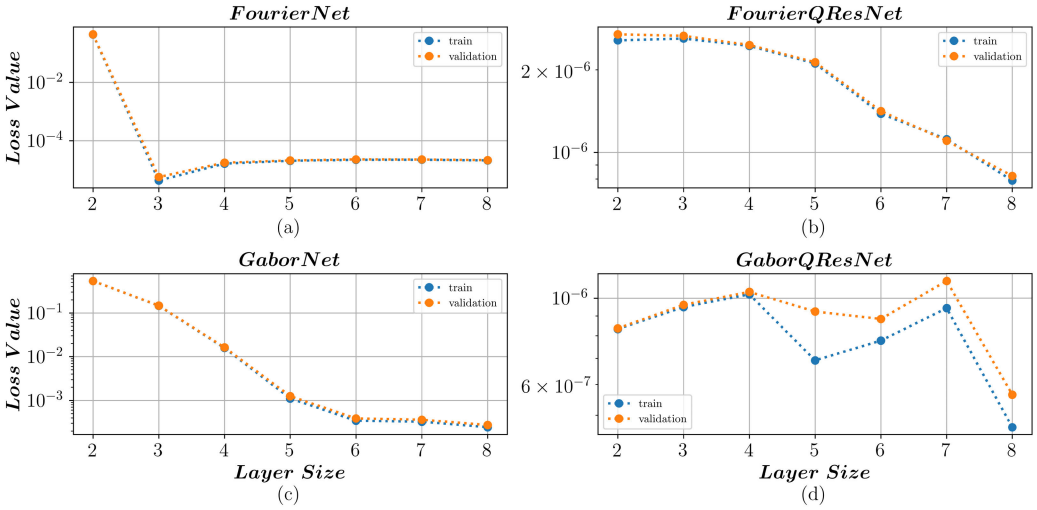

FIGURE 8. Experiment 2: loss value vs layer/depth size for Target Signal - 6 (a) FourierNet (b) FourierQResNet (c) GaborNet (d) GaborQResNet Neural Network Models (y-axis in log scale).

TABLE 4. Experiment 2: comparison of model metrics for Target Signal 4.

		Layer Size 2		Layer Size 3		Layer Size 4		Layer Size 5		Layer Size 6		Layer Size 7		Layer Size 8	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
FourierNet	*MSE*	1.15e+0	1.17e+0	6.46e-5	6.09e-5	6.78e-5	6.74e-5	8.97e-5	8.76e-5	1.10e-4	1.06e-4	1.33e-4	1.35e-4	2.25e-4	2.24e-4
	MAE	8.10e-1	8.15e-1	5.85e-3	5.84e-3	6.54e-3	6.57e-3	7.28e-3	7.28e-3	8.05e-3	7.98e-3	8.23e-3	8.14e-3	1.12e-2	1.12e-2
	R^2	0.276	0.277	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
FourierQResNet	*MSE*	7.79e-6	7.57e-6	7.32e-6	6.88e-6	8.33e-6	8.32e-6	6.84e-6	6.81e-6	4.38e-6	4.30e-6	2.86e-6	2.83e-6	2.87e-6	2.85e-6
	MAE	2.22e-3	2.20e-3	2.03e-3	1.99e-3	2.33e-3	2.32e-3	2.03e-3	2.01e-3	1.67e-3	1.65e-3	1.34e-3	1.34e-3	1.35e-3	1.34e-3
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GaborNet	*MSE*	1.46e+0	1.49e+0	4.23e-1	4.33e-1	4.68e-2	4.48e-2	4.62e-3	4.29e-3	2.11e-3	2.04e-3	4.75e-3	4.22e-3	6.93e-3	6.75e-3
	MAE	8.73e-1	8.80e-1	2.94e-1	2.97e-1	9.62e-2	9.44e-2	3.21e-2	3.15e-2	2.23e-2	2.20e-2	2.82e-2	2.71e-2	3.77e-2	3.70e-2
	R^2	0.077	0.076	0.733	0.732	0.970	0.972	0.997	0.997	0.999	0.999	0.997	0.997	0.996	0.996
GaborQResNet	*MSE*	2.72e-6	2.49e-6	2.34e-6	2.73e-6	1.62e-6	1.69e-6	1.44e-6	1.16e-6	3.51e-6	4.01e-6	2.68e-6	2.69e-6	2.86e-6	2.65e-6
	MAE	8.47e-4	8.37e-4	6.75e-4	6.78e-4	6.41e-4	6.36e-4	6.01e-4	5.94e-4	7.88e-4	7.81e-4	7.54e-4	7.37e-4	7.58e-4	7.35e-4
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

metrics, FourierNet attained its best test values of 6.09×10^{-5} and 5.84×10^{-3} respectively, with three layers, and GaborNet achieved its best test *MSE* and *MAE* values of 2.04×10^{-3} and 2.20×10^{-2} respectively using six layers. We also observed that increasing the layer size from 3 to higher values degenerated *MSE* and *MAE* metrics for FourierNet. In addition, we noticed that increasing the layer size from six even deteriorated the *MSE* and *MAE* metrics for GaborNet. For our

proposed models, FourierQResNet obtained its best test *MSE* and *MAE* values of 2.83×10^{-6} and 1.34×10^{-3} respectively with seven layers, and GaborQResNet achieved its best test *MSE* and *MAE* values of 1.16×10^{-6} and 5.94×10^{-4} respectively using five layers. We also observed that increasing the layer size from seven did not change *MSE* and *MAE* metric values for FourierQResNet. In addition, we observed that increasing the layer size from five even declined *MSE* and *MAE* metrics for GaborQResNet. According to the experimental results obtained for test signal-4, our proposed novel neural network models had much less training and test losses, *MAE*, and *MSE* values than the other models.

Table 5 shows *MSE*, *MAE*, and R^2 metric values vs. layer size results for Target Signal 6. Target Signal 6 is slightly complicated and contains smaller frequency parts than Target Signal 4. For The FourierNet, it reached the maximum R^2 value of 1.0 with 3 layers. GaborNet achieved its maximum R^2 value of 0.999 with 6 layers and diminished to 0.996 with 8 layers included. For our proposed models, FourierQResNet and GaborQResNet obtained R^2 value of 1.0 using only 2 layers. For the *MSE* and *MAE* metrics, the FourierNet attained its best test *MSE* and *MAE* values of 4.30×10^{-6} and 1.15×10^{-3} respectively with 3 layers, and the GaborNet reached its best test *MSE* and *MAE* values of 2.73×10^{-4} and 2.45×10^{-3} respectively by using 8 layers. We also observed that increasing the layer size from 3 degenerated *MSE* and *MAE* metrics for FourierNet. In addition, we noticed that increasing the layer size from 5 improved *MSE* and *MAE* metrics for GaborNet. For our proposed models, FourierQResNet obtained its best test *MSE* and *MAE* values of 7.90×10^{-7} and 7.09×10^{-4} respectively with 8 layers, and the GaborQResNet achieved its best test *MSE* and *MAE* values of 4.66×10^{-7} and 4.09×10^{-4} respectively by using 8 layers. We also observed that GaborQResNet with 2 layers had much better *MAE* and *MSE* values than FourierQResNet with 6 or more layers. According to the experimental results obtained for Test Signal-6, our proposed novel neural network models had much less training and test losses, *MAE*, and *MSE* values than the other modes evaluated.

When all the experimental results were examined, it was found that the proposed novel artificial neural network models, FourierQResNet and GaborQResNet, accomplished higher performance levels than others that could not be managed. Our experimental results also showed that GaborQResNet achieved better learning capacity and expressivity than FourierQResNet. In almost all cases (target signals), GaborQResNet has significantly lower training and test loss values, *MAE*, *MSE* values, and higher R^2 values. FourierQResNet has these capabilities. Exploiting Hadamard-product-based neural quadratic networks in conjunction with multiplicative operations improves the performance of the proposed neural network models studied in this work. In addition to this layer/depth size comparison experiment, in the next section, we planned another experiment to study the effect of the number of neuron sizes for each layer to acquire much better intuition on the suggested novel neural networks' learning capability and expressivity.

C. EXPERIMENT 3: EFFECTS OF VARIATION IN THE NUMBER OF NEURONS IN HIDDEN LAYERS FOR THE PROPOSED MODELS

In Experiment 3, we aimed to empirically analyze the effect of the change in the neuron sizes for each hidden layer on the learning and convergence capacities of the GaborNet, GaborQResNet, FourierNet, and FourierQResNet models by using the test signals given in Equation 29 and Equation 31 (piecewise and square wave signals). All signals were sampled according to their attributes to generate the training and test datasets. All models were trained with selected learning rates and fixed layer depth with changing neuron sizes in each layer, and no hyperparameter search/tuning method was applied. For this experiment, we set the learning rate to 0.001, changed the neuron size of each layer from 64 to 1024 (*neuron size in each hidden layer* $\in$ {64, 128, 256, 512, 1024}), and kept layer dept constant at 5. We intentionally selected a hidden layer/depth size of five from the results of the previous experiment (Experiment 2). For training, the batch size was set to 64, and the Stochastic Gradient Descent (SGD) optimizer was used for the optimization. All models were trained in 2000 epochs using SGD, and the training and testing progress was recorded. Again, we did not use the Adam optimizer in this experiment to investigate the actual convergence behavior of the models. We tested the learning capability of the models for extremely complex target signals with changing neuron sizes in limited training epochs. After completing the training, we investigated the training and test loss values and other performance metrics defined in the previous section. The results of this experiment are shown in Figures 9 and 10, and Tables 6, 7, and 8.

In Table 6, and Figures 9 and 10, the variation of the model training and test loss values of the models under investigation (GaborNet, GaborQResNet, FourierNet, and FourierQResNet) according to the number of neuron sizes in the hidden layers for target signals 3 and 5 are shown. For target signal-3, the training and test loss values of FourierNet and GaborNet decreased when the neuron size was increased. The final test loss values for FourierNet and GaborNet are 1.32×10^{-2} and 2.39×10^{-3}, respectively, for 1024 neurons. For FourierQResNet, increased neuron size yielded a reduction in loss values; however, with 256 neurons, this architecture started to overfit. The best loss value on the test set was 7.13×10^{-3} for 512 neurons. For GaborQResNet, increased neuron size provoked a decline in training and test loss values, but after 256 neurons, this architecture also started to overfit. The best loss value on the test set was 3.03×10^{-4} for 1024 neurons.

For target signal-5, the training and test loss values for FourierNet and GaborNet diminished when the neuron size was increased. The final test loss values for FourierNet and GaborNet were 3.58×10^{-7} and 1.97×10^{-6}, respectively, for 1024 neurons, respectively. For FourierQResNet, increased

TABLE 5. Experiment 2: comparison of model metrics for Target Signal 6.

		Layer Size 2		Layer Size 3		Layer Size 4		Layer Size 5		Layer Size 6		Layer Size 7		Layer Size 8	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
FourierNet	*MSE*	4.46e-1	4.42e-1	5.68e-6	4.30e-6	1.76e-5	1.62e-5	2.10e-5	2.04e-5	2.29e-5	2.21e-5	2.27e-5	2.23e-5	2.16e-5	2.11e-5
	MAE	5.86e-1	5.84e-1	1.20e-3	1.15e-3	3.20e-3	3.17e-3	3.68e-3	3.63e-3	3.74e-3	3.73e-3	3.62e-3	3.59e-3	3.53e-3	3.51e-3
	R^2	0.196	0.202	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
FourierQResNet	*MSE*	2.69e-6	2.56e-6	2.66e-6	2.60e-6	2.46e-6	2.44e-6	2.13e-6	2.11e-6	1.41e-6	1.38e-6	1.10e-6	1.12e-6	8.21e-7	7.90e-7
	MAE	1.29e-3	1.28e-3	1.25e-3	1.22e-3	1.26e-3	1.25e-3	1.13e-3	1.12e-3	9.29e-4	9.23e-4	8.35e-4	8.45e-4	7.19e-4	7.09e-4
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GaborNet	*MSE*	5.37e-1	5.36e-1	1.47e-1	1.46e-1	1.63e-2	1.58e-2	1.25e-3	1.11e-3	3.88e-4	3.43e-4	3.62e-4	3.26e-4	2.73e-4	2.45e-4
	MAE	6.44e-1	6.44e-1	2.86e-1	2.85e-1	5.88e-2	5.80e-2	1.59e-2	1.54e-2	9.75e-3	9.51e-3	9.11e-3	8.97e-3	9.03e-3	8.85e-3
	R^2	0.033	0.032	0.736	0.737	0.971	0.971	0.998	0.998	0.999	0.999	0.999	0.999	1.000	1.000
GaborQResNet	*MSE*	8.36e-7	8.33e-7	9.61e-7	9.48e-7	1.04e-6	1.02e-6	9.24e-7	6.92e-7	8.84e-7	7.77e-7	1.11e-6	9.42e-7	5.65e-7	4.66e-7
	MAE	5.29e-4	5.23e-4	5.33e-4	5.24e-4	4.97e-4	4.90e-4	4.70e-4	4.62e-4	4.29e-4	4.24e-4	5.34e-4	5.20e-4	4.19e-4	4.09e-4
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

TABLE 6. Experiment 3: comparison of model loss values for Target Signal 3 and 5.

		32 Neurons		64 Neurons		128 Neurons		256 Neurons		512 Neurons		1024 Neurons	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Signal 3	*FourierNet*	8.08e+0	7.59e+0	1.21e-1	1.22e-1	2.58e-2	2.61e-2	1.83e-2	1.81e-2	1.35e-2	1.39e-2	1.25e-2	1.32e-2
	FourierQResNet	1.43e-2	1.43e-2	3.90e-3	4.30e-3	1.52e-3	1.97e-3	7.80e-4	1.20e-3	3.87e-4	7.13e-4	1.32e-3	1.68e-3
	GaborNet	9.84e-2	1.09e-1	1.86e-2	1.91e-2	1.12e-2	1.18e-2	5.13e-3	5.28e-3	2.25e-3	2.58e-3	2.06e-3	2.39e-3
	GaborQResNet	8.32e-2	8.08e-2	1.09e-2	1.24e-2	1.39e-3	1.80e-3	4.69e-4	6.95e-4	2.30e-4	4.00e-4	1.61e-4	3.03e-4
Signal 5	*FourierNet*	3.02e-2	2.94e-2	8.61e-4	8.23e-4	7.15e-5	7.15e-5	1.31e-5	1.09e-5	2.76e-6	2.02e-6	7.33e-7	3.58e-7
	FourierQResNet	1.55e-4	1.49e-4	1.59e-5	1.59e-5	3.84e-6	3.63e-6	1.00e-6	1.00e-6	1.95e-7	2.10e-7	1.30e-5	1.46e-5
	GaborNet	3.83e-4	3.37e-4	1.08e-4	5.92e-5	5.79e-5	2.74e-5	4.24e-5	1.29e-5	1.99e-5	5.57e-6	5.88e-6	1.97e-6
	GaborQResNet	1.64e-4	1.15e-4	2.10e-5	1.54e-5	7.33e-6	4.29e-6	8.85e-7	4.93e-7	6.81e-7	3.65e-7	1.41e-7	8.73e-8

neuron size yielded a reduction in loss values; however, after 512 neurons, this architecture started to overfit. The best loss value on the test set was 2.10×10^{-7} for 512 neurons. For GaborQResNet, increased neuron size caused a drop in loss values, but this time we did not detect an overfit for the architecture. The best loss value on the test set was 8.73×10^{-8} for 1024 neurons.

In Figure 9, It can be seen that the performances of all models performances were generally improved when the neuron size was increased for Test Signal 3. However, the performance of the FourierQResNet model worsened when the neuron size increased from 512 to 1024. For GaborQResNet, this behavior was not observed, and loss values continued to decrease while neuron size increased, and the loss's final training and test values were the smallest.

In Figure 10, It can be observed that the performances of all models generally were enhanced when the neuron size was raised for Test Signal 5. However, the performance of the FourierQResNet model deteriorated when the neuron size increased from 512 to 1024. For GaborQResNet, this behavior was not detected, and all the loss values persisted in decreasing as the neuron size increased, and the final training and test loss values were the smallest among the models under investigation.

Table 7 shows *MSE*, *MAE*, and R^2 metric values vs. neuron size results for Target Signal 3. Target Signal 3 is slightly complicated and contains smaller frequency parts than Target Signal 5. For the FourierNet, it reached the maximum R^2 value of 1.0 with 128 neurons. GaborNet achieved its maximum R^2 value of 0.999 with 6 layers and diminished to 0.996 with 8 layers included. For our proposed models, FourierQResNet and GaborQResNet obtained R^2 value of 1.0 using only 64 neurons. For the *MSE* and *MAE* metrics, the FourierNet attained its best test *MSE* and *MAE* values of 1.32×10^{-2} and 2.02×10^{-2} respectively with 1024 neurons, and the GaborNet reached its best test *MSE* and *MAE* values of 2.39×10^{-3} and 1.04×10^{-2} respectively by using 1024 neurons. For our proposed models, FourierQResNet obtained its best test *MSE* and *MAE* values of 7.13×10^{-4} and 5.19×10^{-3} respectively with 512 neurons, and the GaborQResNet achieved its best test *MSE* and *MAE* values of 3.03×10^{-4} and 5.09×10^{-3} respectively by using 1024 neurons. We also observed that GaborQResNet with 1024 neurons had much better *MAE* and *MSE* values than FourierQResNet with 1024 neurons. According to the experimental results obtained for Test Signal - 3, our proposed novel neural network models had much less training and test *MAE* and *MSE* values than the other models evaluated.

Table 8 lists *MSE*, *MAE*, and R^2 metric values vs. layer size results for Target Signal 5. According to these results, we observed the same behavior for all four models as for the loss values when we investigated these metrics. FourierNet reached at maximum R^2 value of 1.0 with 128 neurons. GaborNet achieved a maximum R^2 value of 1.0 with 32 neurons. For our proposed models, FourierQResNet and GaborQResNet obtained R^2 value of 1.0 using only 32 neurons. For the *MSE* and *MAE* metrics, the FourierNet attained its best test *MSE* and *MAE* values of 3.58×10^{-7} and 4.29×10^{-4} respectively with 1024 neurons, and the GaborNet achieved its best test *MSE* and *MAE* values

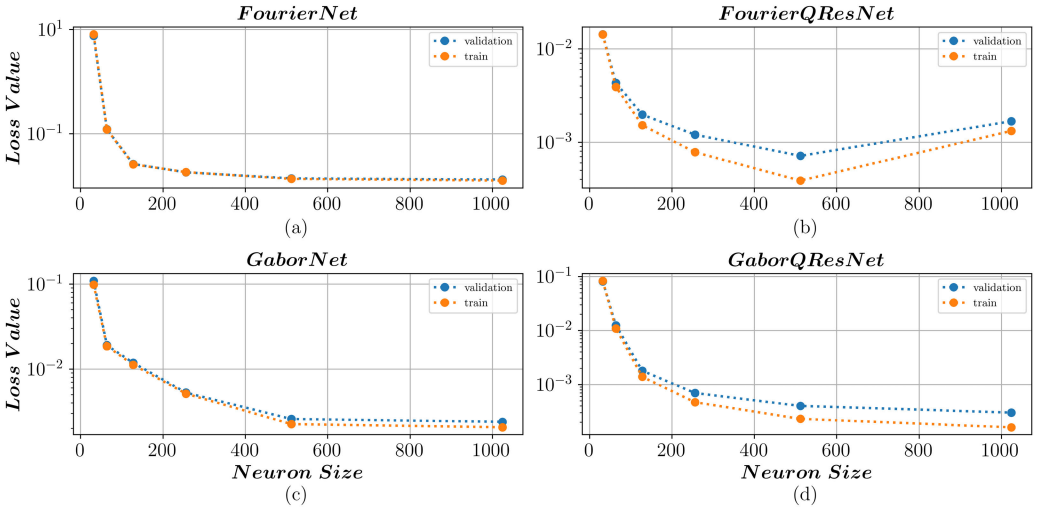

FIGURE 9. Experiment 3: loss value vs neuron size for target signal - 3 (a) FourierNet (b) FourierQResNet (c) GaborNet (d) GaborQResNet Neural Network Models (y-axis in log scale).

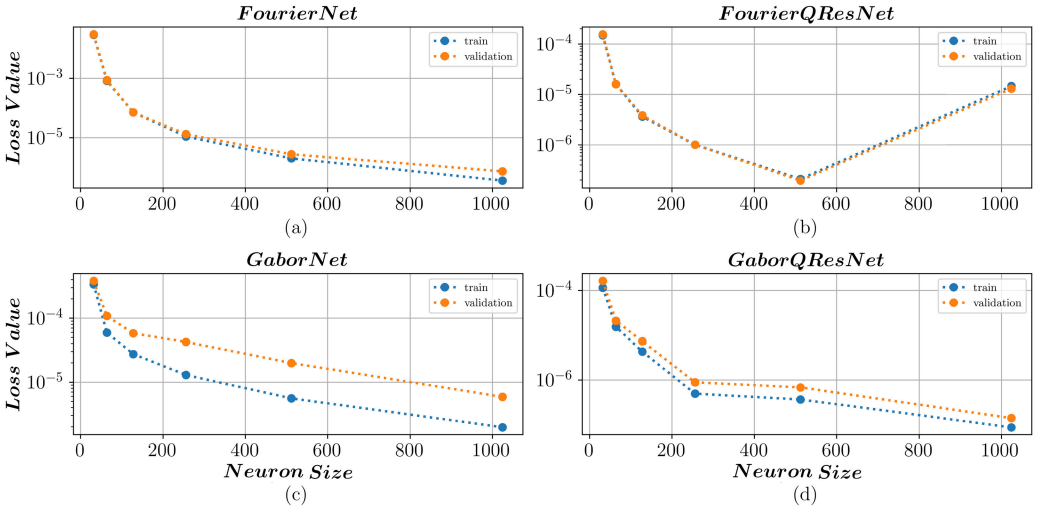

FIGURE 10. Experiment 3: loss value vs neuron size for target signal - 5 (a) FourierNet (b) FourierQResNet (c) GaborNet (d) GaborQResNet Neural Network Models (y-axis in log scale).

TABLE 7. Experiment 3: comparison of model metrics for Target Signal 3.

		32 Neurons		64 Neurons		128 Neurons		256 Neurons		512 Neurons		1024 Neurons	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
FourierNet	*MSE*	8.08e+0	7.59e+0	1.21e-1	1.22e-1	2.58e-2	2.61e-2	1.83e-2	1.81e-2	1.35e-2	1.39e-2	1.25e-2	1.32e-2
	MAE	2.14e+0	2.10e+0	2.56e-1	2.58e-1	7.39e-2	7.50e-2	4.60e-2	4.48e-2	2.31e-2	2.30e-2	1.96e-2	2.02e-2
	R^2	0.925	0.927	0.999	0.999	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
FourierQResNet	*MSE*	1.43e-2	1.43e-2	3.90e-3	4.30e-3	1.52e-3	1.97e-3	7.80e-4	1.20e-3	3.87e-4	7.13e-4	1.32e-3	1.68e-3
	MAE	7.99e-2	7.89e-2	3.98e-2	3.91e-2	1.91e-2	1.90e-2	1.02e-2	1.04e-2	4.93e-3	5.19e-3	1.63e-2	1.69e-2
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GaborNet	*MSE*	9.84e-2	1.09e-1	1.86e-2	1.91e-2	1.12e-2	1.18e-2	5.13e-3	5.28e-3	2.25e-3	2.58e-3	2.06e-3	2.39e-3
	MAE	2.13e-1	2.26e-1	7.79e-2	7.78e-2	4.66e-2	4.68e-2	2.52e-2	2.47e-2	1.36e-2	1.34e-2	1.09e-2	1.04e-2
	R^2	0.999	0.999	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GaborQResNet	*MSE*	8.32e-2	8.08e-2	1.09e-2	1.24e-2	1.39e-3	1.80e-3	4.69e-4	6.95e-4	2.30e-4	4.00e-4	1.61e-4	3.03e-4
	MAE	1.95e-1	1.94e-1	6.61e-2	6.47e-2	2.17e-2	2.25e-2	1.12e-2	1.13e-2	7.42e-3	7.53e-3	5.82e-3	5.90e-3
	R^2	0.999	0.999	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

of 1.97×10^{-6} and 4.51×10^{-4} respectively by using 1024 neurons. For our proposed models, FourierQResNet obtained its best test *MSE* and *MAE* values of 2.10×10^{-7} and 3.61×10^{-4} respectively, with 512 neurons, whereas GaborQResNet achieved its best test *MSE*, *MAE* values of 8.78×10^{-8} and 2.02×10^{-4} respectively, using 1024 neurons.

TABLE 8. Experiment 3: comparison of model metrics for Target Signal 5.

		32 Neurons		64 Neurons		128 Neurons		256 Neurons		512 Neurons		1024 Neurons	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
FourierNet	*MSE*	3.02e-2	2.94e-2	8.61e-4	8.23e-4	7.15e-5	7.15e-5	1.31e-5	1.09e-5	2.76e-6	2.02e-6	7.33e-7	3.58e-7
	MAE	1.35e-1	1.34e-1	2.31e-2	2.30e-2	6.76e-3	6.76e-3	2.57e-3	2.55e-3	1.14e-3	1.11e-3	4.59e-4	4.29e-4
	R^2	0.981	0.982	0.999	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
FourierQResNet	*MSE*	1.55e-4	1.49e-4	1.59e-5	1.59e-5	3.84e-6	3.63e-6	1.00e-6	1.00e-6	1.95e-7	2.10e-7	1.30e-5	1.46e-5
	MAE	9.78e-3	9.52e-3	3.20e-3	3.23e-3	1.55e-3	1.51e-3	7.68e-4	7.76e-4	3.48e-4	3.61e-4	1.55e-3	1.64e-3
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GaborNet	*MSE*	3.83e-4	3.37e-4	1.08e-4	5.92e-5	5.79e-5	2.74e-5	4.24e-5	1.29e-5	1.99e-5	5.57e-6	5.88e-6	1.97e-6
	MAE	1.19e-2	1.22e-2	4.70e-3	4.38e-3	2.64e-3	2.44e-3	1.40e-3	1.15e-3	8.91e-4	7.48e-4	5.31e-4	4.51e-4
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GaborQResNet	*MSE*	1.64e-4	1.15e-4	2.10e-5	1.54e-5	7.33e-6	4.29e-6	8.85e-7	4.93e-7	6.81e-7	3.65e-7	1.41e-7	8.73e-8
	MAE	8.70e-3	8.11e-3	3.01e-3	2.86e-3	1.51e-3	1.45e-3	5.23e-4	4.83e-4	4.00e-4	3.83e-4	2.10e-4	2.02e-4
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

We also observed that increasing the neuron size from 512 to 1024 did not improve the *MSE* and *MAE* metric values for the FourierQResNet. According to the experimental results obtained for Test Signal 5, our proposed novel neural network models had much less training and testing *MAE* and *MSE* values than the other models under investigation. When all the experimental results were examined, it was discovered that the proposed novel quadratic multiplicative filter-based neural network models, FourierQResNet and GaborQResNet, achieved higher performance levels than the others that could not be managed. Our experimental results also show that GaborQResNet performed better than FourierQResNet, and has a much higher learning capacity and expressivity. In almost all cases (target signals), GaborQResNet has considerably lower training and test loss values, *MAE* and *MSE* values, and higher R^2 values. FourierQResNet possesses these capabilities. Exploiting Hadamard-product-based neural quadratic networks in conjunction with multiplicative operations improves the performance of the proposed neural network models studied in this work. In addition to this neuron size comparison experiment, in the next section, we design another experiment to investigate the effect of the number of dataset sizes for each model to gain much more intuition on the learning capacity and expressivity of the proposed novel neural networks.

D. EXPERIMENT 4: THE EFFECT OF THE DATA SET SIZE VARIATION

We empirically examined the effect of varying the dataset size on the learning and convergence capabilities of the GaborNet, GaborQResNet, FourierNet, and FourierQResNet models by using the test signals given in Equation 28, Equation 30, and Equation 34 in Experiment 4. All signals were sampled with different numbers of points (*data set sample points* $\in$ {512, 1024, 2048, 4096, 10240}) according to their characteristics to generate the training and test datasets. All models were trained with specified learning rates, a fixed neuron size with constant layer depth, and no hyperparameter search/tuning method was applied. For this experiment, we set the learning rate to 0.001, the layer size of each model to five, and the number of neurons for each layer to 256. For training, the batch size was set to 64, and the Stochastic Gradient Descent (SGD) optimizer was used for the optimization. All models were trained in 2000 epochs with SGD for three different test signals, and the training and testing progress was recorded. We did not use the Adam optimizer for this experiment to investigate the actual convergence behavior of the models. We tested the learning ability of the models for the given complex target signals with different data set sizes in limited training epochs. After completing the training, we investigated the training and test loss values and other performance metrics defined in the previous section. The results of this experiment are shown given in Figures 11, 12 and 13, and Tables 9, 10, 11, and 12.

Table 9 shows a comparison of the model training and test loss values for Target Signals 2, 4, and 8. Figures 11, 12 and 13 depict the training and test loss values with respect to the dataset size for target signals 2, 4, and 8, respectively. From the results in Table 9, it can be seen that training and test loss values of the FourierNet and GaborNet models were not improved by increasing the dataset size and remained nearly constant around the values of 3.0×10^{-5} and 2.0×10^{-2} respectively, for Test Signal 2. However, FourierQResNet and GaborQResNet models' test loss values were around 1.84×10^{-6} and 2.67×10^{-7} for 1024 data points, respectively. In light of these numerical results, our proposed models provided much better performance using only 1024-point data for Test Signal 2, and adding more data did not further improve the loss values. Similar results were obtained for Test Signal 4. For Test Signal 8, we observed similar behavior with the number of data points. Because this signal includes very high-frequency components, the number of required data points for the train models was higher than the other models. For all models, only 4096 data points were required, and increasing this value beyond 4096 did not help to improve the model performance. We obtained the best results with FourierQResNet and GaborQResNet with test loss values of 3.37×10^{-7} and 7.53×10^{-8}.

In Figure 11, the training and test loss values of the models with respect to the dataset size for Target Signal - 2 are shown. The training loss values of the FourierNet and GaborNet models were not enhanced by expanding the dataset size and remained almost constant for Test Signal 2. Nonetheless, training and test loss values of the FourierQResNet and GaborQResNet models were less than those of the FourierNet and GaborNet models for every data set size, and adding more

TABLE 9. Experiment 4: comparison of model loss values for Target Signals 2, 4 and 8.

		512 Points		1024 Points		2048 Points		4096 Points		10240 Points	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Signal 2	*FourierNet*	3.28e-5	2.46e-5	3.96e-5	3.41e-5	3.92e-5	3.36e-5	3.00e-5	2.81e-5	2.95e-5	2.97e-5
	FourierQResNet	5.16e-14	1.03e-4	1.50e-6	1.84e-6	2.12e-6	1.83e-6	1.24e-6	1.16e-6	1.96e-6	1.87e-6
	GaborNet	1.94e-2	2.00e-2	2.14e-2	2.22e-2	2.47e-2	2.48e-2	1.87e-2	1.65e-2	2.55e-2	2.91e-2
	GaborQResNet	1.20e-8	2.82e-4	2.67e-7	7.46e-7	2.66e-7	2.49e-7	3.57e-7	2.77e-7	5.01e-7	4.62e-7
Signal 4	*FourierNet*	8.64e-5	9.64e-5	7.52e-5	6.96e-5	8.06e-5	7.47e-5	8.85e-5	9.52e-5	1.00e-4	9.93e-5
	FourierQResNet	2.02e-13	2.50e-4	4.61e-6	5.14e-6	3.76e-6	3.76e-6	4.80e-6	4.80e-6	4.32e-6	4.33e-6
	GaborNet	6.73e-3	7.13e-3	2.91e-3	3.68e-3	3.82e-3	4.45e-3	5.14e-3	4.87e-3	7.17e-3	7.11e-3
	GaborQResNet	1.11e-7	7.51e-6	1.96e-6	2.21e-6	2.82e-6	1.53e-6	1.52e-6	1.45e-6	2.33e-6	2.34e-6
Signal 8	*FourierNet*	3.57e-5	1.15e-2	1.69e-5	2.11e-4	9.27e-6	2.85e-5	6.75e-6	8.86e-6	7.90e-6	8.59e-6
	FourierQResNet	4.77e-7	1.86e-2	4.42e-7	8.68e-5	3.06e-7	3.86e-6	3.38e-7	3.37e-7	3.18e-7	3.59e-7
	GaborNet	1.33e-3	1.68e-2	6.21e-4	1.36e-3	5.67e-4	7.55e-4	1.28e-3	1.49e-3	8.71e-4	9.01e-4
	GaborQResNet	2.89e-7	1.34e-3	9.55e-8	2.79e-6	8.08e-8	2.75e-7	5.60e-8	7.53e-8	6.55e-8	7.25e-8

data points to the dataset did not further enhance the loss values.

In Figure 12, the training and test loss values of the models with respect to the dataset size for Target Signal - 4 are shown, and we observed similar results for Test Signal 2. Training and test loss values of the FourierNet and GaborNet models were not improved by extending the dataset size and nearly remained steady for Test Signal 4. Nevertheless, FourierQResNet and GaborQResNet models' training and test loss values were smaller than the FourierNet and GaborNet models for every dataset size, and including more additional data points in the dataset did not further enhance the loss values.

In Figure 13, the training and test loss values of the models with respect to the dataset size for Target Signal - 8 are shown. We can see that only 4096 data points were required to train all the models, and increasing this value beyond 4096 did not help to improve the model performances. We obtained the best results with FourierQResNet and GaborQResNet. In addition, the training and test loss values of the FourierQResNet and GaborQResNet models were less than those of FourierNet and GaborNet models for every data set size, and including extra data points in the data set did not positively affect the loss values further.

In Table 10, the effect of dataset size on *MSE*, *MAE*, and R^2 metric values for Target Signal 2 is presented. FourierNet reached a maximum R^2 value of 1.0 with 512 data points. GaborNet achieved a maximum R^2 value of 0.96 with 4096 data points, and it did not reach a value of 1.0. For our proposed models, FourierQResNet and GaborQResNet obtained R^2 value of 1.0 using only 512 data points, and further increasing the data point size did not affect this metric. For the *MSE* and *MAE* metrics, FourierNet attained its best test *MSE* and *MAE* values of 2.81×10^{-5} and 4.25×10^{-3} respectively with 4096 data points, and GaborNet achieved its best test *MSE* and, *MAE* values of 1.65×10^{-2} and 4.27×10^{-2} respectively, using 4096 data points. For our proposed models, FourierQResNet obtained its best test *MSE* and *MAE* values of 1.16×10^{-6} and 8.23×10^{-4} respectively with 4096 data points, and the GaborQResNet achieved its best test *MSE* and, *MAE* values of 2.49×10^{-7} and 3.25×10^{-4} respectively, using 2048 data points. We also observed that increasing the dataset size from 1024 to 10240 did not improve *MSE* and *MAE* metric values for FourierQResNet and GaborQResNet. According to the experimental results obtained for Test Signal 2, our proposed novel neural network models had much less training and testing *MAE* and *MSE* values than the other models under investigation for Test Signal 2.

In Table 11, the impact of dataset size on *MSE*, *MAE*, and R^2 metric values for Target Signal 4 is presented. FourierNet reached a maximum R^2 value of 1.0 with 512 data points. GaborNet achieved a maximum R^2 value of 0.998 with 2048 data points, and it did not reach a value of 1.0. For our proposed models, FourierQResNet and GaborQResNet archived an R^2 value of 1.0, using only 512 data points. For the *MSE* and *MAE* metrics, FourierNet achieved its best test *MSE* and *MAE* values of 6.96×10^{-5} and 6.71×10^{-3} respectively, with 1024 data points, and GaborNet achieved its best test *MSE* and *MAE*1 values of 3.68×10^{-3} and 2.94×10^{-2} respectively, using 1024 data points. The additional data points did not improve the results of these models. For our proposed models, FourierQResNet obtained its best test *MSE*, *MAE* values of 3.76×10^{-6} and 1.56×10^{-3} respectively with 2048 data points, and the GaborQResNet achieved its best test *MSE*, *MAE* values of 1.46×10^{-6} and 6.93×10^{-4} respectively by using 4096 data points. We also observed that increasing the dataset size from 2048 to 10240 did not improve *MSE* and *MAE* metric values for FourierQResNet or GaborQResNet. Based on the experimental findings for Test Signal 4, compared to the other models under consideration for Test Signal 4, our innovative neural network models exhibited much lower training and test *MAE* and *MSE* values.

The effect of the dataset size on the *MSE*, *MAE*, and R^2 metric values for Target Signal 8 is shown in Table 12. Using 512 data points, FourierNet achieved the highest *R*2 value of 1.0. Using 512 data points, GaborNet reached a maximum R^2 value of 0.994 with 2048 data points but fell short of the value of 1.0. With just 512 data points, FourierQResNet and GaborQResNet produced an *R*2 value of 1.0 for our suggested models. The best test *MSE* and *MAE* values for FourierNet and GaborNet, respectively, were 8.59×10^{-6} and 2.28×10^{-3} for the *MSE* and *MAE* metrics for 10240 data points, and 7.55×10^{-4} and $1.39 \times 10{-}2$, respectively, for the *MSE* and *MAE* metrics for 2048. The findings of these models do not benefit from additional data

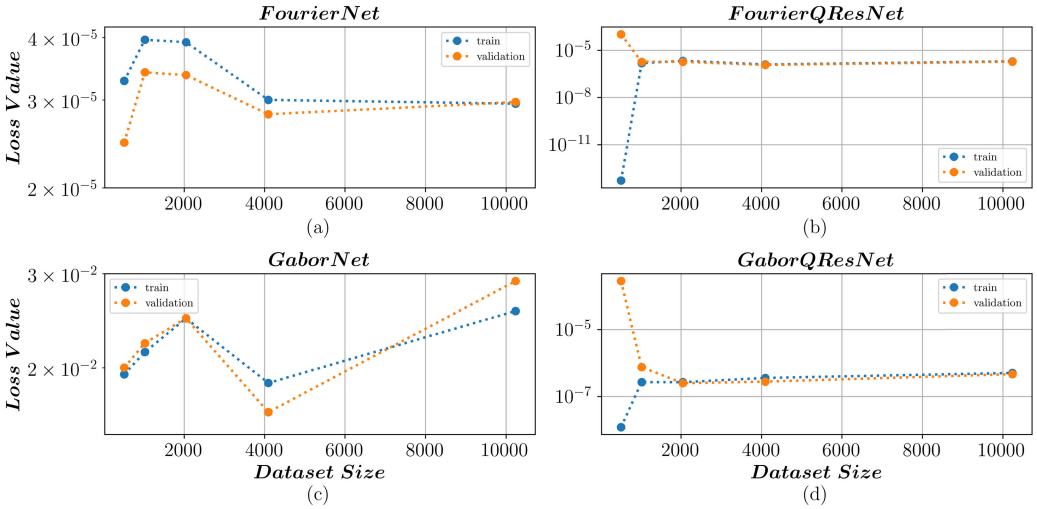

FIGURE 11. Loss value vs. dataset size for Target Signal - 2 (a) FourierNet (b) FourierQResNet (c) GaborNet (d) GaborQResNet Neural Network Models (y-axis in log scale).

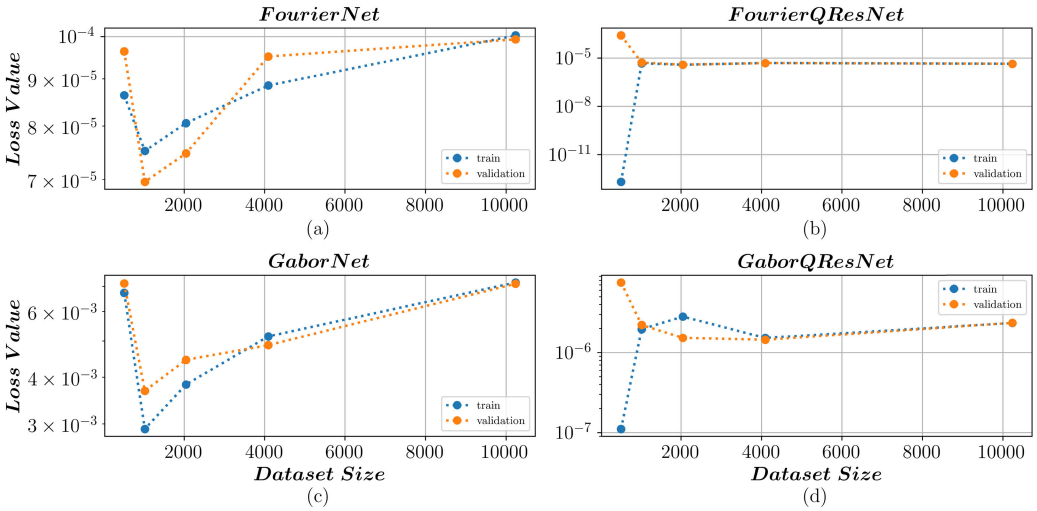

FIGURE 12. Loss value vs. dataset size for Target Signal - 4 (a) FourierNet (b) FourierQResNet (c) GaborNet (d) GaborQResNet Neural Network Models (y-axis in log scale).

TABLE 10. Experiment 4: comparison of model metrics for Target Signal 2.

		512 Points		1024 Points		2048 Points		4096 Points		10240 Points	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
FourierNet	*MSE*	3.28e-5	2.46e-5	3.96e-5	3.41e-5	3.92e-5	3.36e-5	3.00e-5	2.81e-5	2.95e-5	2.97e-5
	MAE	4.28e-3	3.92e-3	4.67e-3	4.39e-3	4.44e-3	4.39e-3	4.30e-3	4.25e-3	4.24e-3	4.27e-3
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
FourierQResNet	*MSE*	5.16e-14	1.03e-4	1.50e-6	1.84e-6	2.12e-6	1.83e-6	1.24e-6	1.16e-6	1.96e-6	1.87e-6
	MAE	1.35e-7	7.04e-3	9.52e-4	9.42e-4	1.06e-3	1.01e-3	8.28e-4	8.23e-4	9.43e-4	9.51e-4
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GaborNet	*MSE*	1.94e-2	2.00e-2	2.14e-2	2.22e-2	2.47e-2	2.48e-2	1.87e-2	1.65e-2	2.55e-2	2.91e-2
	MAE	4.57e-2	4.94e-2	4.67e-2	4.66e-2	5.33e-2	5.14e-2	4.61e-2	4.27e-2	5.40e-2	5.70e-2
	R^2	0.963	0.964	0.959	0.957	0.953	0.951	0.964	0.968	0.951	0.946
GaborQResNet	*MSE*	1.20e-8	2.82e-4	2.67e-7	7.46e-7	2.66e-7	2.49e-7	3.57e-7	2.77e-7	5.01e-7	4.62e-7
	MAE	3.53e-5	5.33e-3	3.56e-4	3.91e-4	3.27e-4	3.25e-4	3.21e-4	3.11e-4	4.26e-4	4.32e-4
	R^2	1.000	0.999	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

points. FourierQResNet and GaborQResNet each achieved their best test *MSE* and *MAE* values with 10240 data points, and the best test *MSE* and *MAE* values were 7.25×10^{-8} and 2.28×10^{-3}, respectively. FourierQResNet's best test *MSE* and *MAE* values were 3.59×10^{-7} and 2.10×10^{-4}, respectively. Based on the test results for Test Signal 4, our

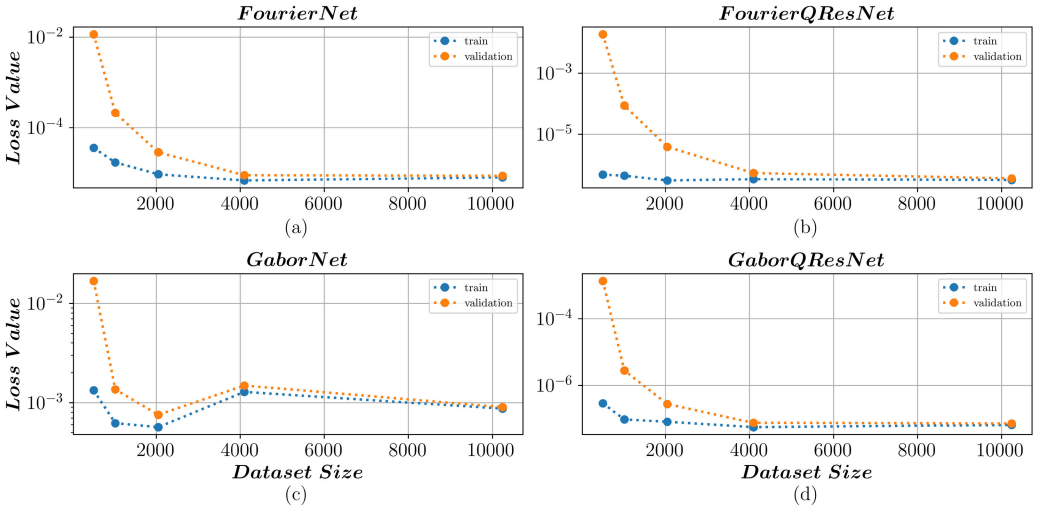

FIGURE 13. Loss value vs. dataset size for Target Signal - 8 (a) FourierNet (b) FourierQResNet (c) GaborNet (d) GaborQResNet Neural Network Models (y-axis in log scale).

TABLE 11. Experiment 4: comparison of model metrics for Target Signal 4.

		512 Points		1024 Points		2048 Points		4096 Points		10240 Points	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
FourierNet	*MSE*	8.64e-5	9.64e-5	7.52e-5	6.96e-5	8.06e-5	7.47e-5	8.85e-5	9.52e-5	1.00e-4	9.93e-5
	MAE	7.34e-3	7.83e-3	6.95e-3	6.71e-3	7.09e-3	6.91e-3	6.81e-3	6.91e-3	7.62e-3	7.60e-3
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
FourierQResNet	*MSE*	2.02e-13	2.50e-4	4.61e-6	5.14e-6	3.76e-6	3.76e-6	4.80e-6	4.80e-6	4.32e-6	4.33e-6
	MAE	2.24e-7	1.12e-2	1.72e-3	1.77e-3	1.56e-3	1.56e-3	1.73e-3	1.73e-3	1.65e-3	1.66e-3
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GaborNet	*MSE*	6.73e-3	7.13e-3	2.91e-3	3.68e-3	3.82e-3	4.45e-3	5.14e-3	4.87e-3	7.17e-3	7.11e-3
	MAE	3.59e-2	4.09e-2	2.73e-2	2.94e-2	3.02e-2	3.26e-2	3.44e-2	3.40e-2	4.16e-2	4.12e-2
	R^2	0.996	0.996	0.998	0.998	0.998	0.997	0.997	0.997	0.995	0.995
GaborQResNet	*MSE*	1.11e-7	7.51e-6	1.96e-6	2.21e-6	2.82e-6	1.53e-6	1.52e-6	1.45e-6	2.33e-6	2.34e-6
	MAE	1.69e-4	1.67e-3	7.22e-4	7.62e-4	7.59e-4	7.36e-4	7.11e-4	6.93e-4	6.16e-4	6.25e-4
	R^2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

novel neural network models showed much lower training and test *MAE* and *MSE* values than the other models taken into account for Test Signal 4.

The findings of this experiment led us to conclude that FourierQResNet and GaborQResNet, our proposed innovative quadratic multiplicative filter-based neural network models, outperformed other models with smaller data sets. Additionally, the findings of our experiment demonstrate that GaborQResNet is more capable of learning than FourierQResNet. GaborQResNet has significantly reduced training and test loss, *MAE* and *MSE* values, and higher R^2 values in virtually all circumstances (target signals). Likewise, these abilities were possessed by FourierQResNet. The performance of our suggested models was enhanced by using Hadamard-product-based neural quadratic networks together with multiplicative operations.

The success of the proposed new models and the changes in the hyper-parameters that affect this success, specific to different signals from previous experiments, were investigated. From these experiments, it was discovered that the performance of the proposed innovative models was better than that of other models when evaluated from different perspectives. The fact that sensor signals contain different frequency components shows sudden changes, and contain noise makes it very difficult to model such signals using deep neural networks. An experiment was designed to examine and evaluate the proposed innovative neural network models and other neural network models in terms of spectral bias. The findings are presented in the following section.

E. EXPERIMENT 5: SPECTRAL/FREQUENCY BIAS

This experiment aims to compare the spectral bias of our proposed GaborQResNet and FourierQResNet models with other artificial neural network models. All models were examined using three different approaches to confirm our other hypothesis, which claims that the proposed models can achieve more successful results under the influence of spectral bias and are less affected by this phenomenon. For spectral/frequency bias inspections, we used the projection-based relative error and the frequency band correspondence metrics, and their explicit definitions are given in Equations 40 and 41 in Section II-G. In addition, an extended version of the signal defined in [6] was used to investigate the frequency/spectral bias, and its definition is given by Equation 36. The waveform and its accompanying frequency spectrum are shown in Figures 4a and 4b, respectively. This signal contains both low and high-frequency components with different phase angles. It also contains very low and

TABLE 12. Experiment 4: comparison of model metrics for Target Signal 8.

		512 Points		1024 Points		2048 Points		4096 Points		10240 Points	
		Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
FourierNet	*MSE*	3.57e-5	1.15e-2	1.69e-5	2.11e-4	9.27e-6	2.85e-5	6.75e-6	8.86e-6	7.90e-6	8.59e-6
	MAE	3.78e-3	4.57e-2	2.93e-3	7.14e-3	2.14e-3	3.09e-3	1.98e-3	2.23e-3	2.18e-3	2.28e-3
	R^2	1.000	0.906	1.000	0.998	1.000	1.000	1.000	1.000	1.000	1.000
FourierQResNet	*MSE*	4.77e-7	1.86e-2	4.42e-7	8.68e-5	3.06e-7	3.86e-6	3.38e-7	5.37e-7	3.18e-7	3.59e-7
	MAE	3.74e-4	5.22e-2	4.48e-4	3.70e-3	4.11e-4	9.12e-4	4.40e-4	5.26e-4	4.54e-4	4.78e-4
	R^2	1.000	0.848	1.000	0.999	1.000	1.000	1.000	1.000	1.000	1.000
GaborNet	*MSE*	1.33e-3	1.68e-2	6.21e-4	1.36e-3	5.67e-4	7.55e-4	1.28e-3	1.49e-3	8.71e-4	9.01e-4
	MAE	1.77e-2	6.84e-2	1.19e-2	1.91e-2	1.17e-2	1.39e-2	1.66e-2	1.85e-2	1.41e-2	1.46e-2
	R^2	0.990	0.862	0.995	0.990	0.996	0.994	0.990	0.988	0.993	0.993
GaborQResNet	*MSE*	2.89e-7	1.34e-3	9.55e-8	2.79e-6	8.08e-8	2.75e-7	5.60e-8	7.53e-8	6.55e-8	7.25e-8
	MAE	3.52e-4	1.15e-2	2.15e-4	7.10e-4	2.08e-4	3.26e-4	1.80e-4	2.08e-4	2.00e-4	2.10e-4
	R^2	1.000	0.989	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

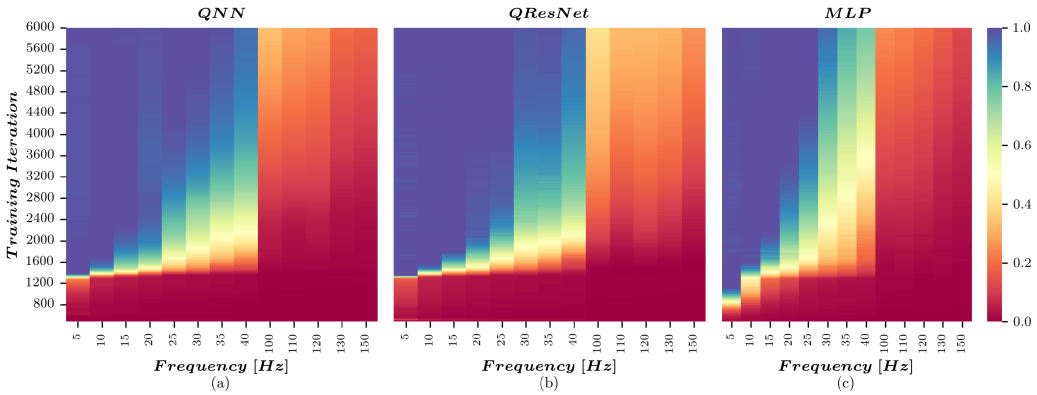

FIGURE 14. Evolution of the frequency spectrum (a) QNN Model (b) QResNet Model (c) MLP model.

high-frequency components; some of the frequency components are very close (densely placed), and some are distant from each other. Therefore, it helps investigate the spectral bias concept of neural networks at both low and high frequencies with different spectral distances.

In this experiment, all models examined were divided into three different groups according to the results obtained in Experiment 1. The first of these groups was determined to be models that could not effectively learn target signals, such as MLP, QNN, and QResNet. The second group, which includes the GaborNet and FourierNet models, was chosen as the model that did not show sufficient learning performance but was more successful than the first group. The third group consists of the GaborQResNet and FourierQResNet models proposed in this study, which exhibit fast convergence and high learning capacity. The groups determined in Experiment 1 were used to present the results of this experiment to facilitate our analyses and to make more evident inferences when comparing the models.

For the first analysis, we assessed the evolution of the frequency spectrum for all models using target signal 10 given in Equation 36. The input points $\{x_i\}_{i=1}^{N}$ are sampled from a uniform distribution $x_i \overset{iid}{\sim} \mathcal{U}(0, 1)$. The output values $\{y_i\}_{i=1}^{N}$ are the function values $y_i = f(x_i)$, and by uniformly sampling 500 i.i.d. points from this input distribution, $\{A_i\}_{i=1}^{13} = 1.0$, and $\phi_i \overset{iid}{\sim} \{\mathcal{U}(0, 2\pi)\}_{i=1}^{13}$. By using these parameters, training, and testing data sets, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N_{train}}$ and $\mathcal{D}_{test} = \{(x_i, y_i)\}_{i=1}^{N_{test}}$, were generated. All models were trained with specific learning rates, fixed neuron size, and fixed layer depth, and no hyperparameter search/tuning method was applied. For this experiment, we set the learning rate to 0.001, the layer size of each model to six, and the number of neurons for each layer to 256 for all models. For training, the batch size was set to 64, and the Stochastic Gradient Descent (SGD) optimizer was used for the optimization. All the models were trained in 6000 epochs with SGD with the data generated from the target signal 10, and training and testing progress, *PBRE*, and *FBCM* metrics were recorded for all models. This process was repeated ten times with different random number seeds. Our target signal includes the frequencies of $f \in \{5, 10, 15, 20, 25, 30, 35, 40, 100, 110, 120, 130, 150\}$ Hz with different phase information (sampled randomly). We assessed the models' prediction power on these specific frequencies for the target signal 10 and investigated the evaluation of the effect of the spectral bias phenomenon on all models during the training by using the metrics described previously. (In Figures 14,15, and 16, the colors in the visual representation indicate the magnitude of the network spectrum at a given frequency, which has been normalized by the target magnitude at that same frequency, represented as $|\tilde{f}_{k_i}|/A_i$. The color bar has been adjusted to display values between 0 and 1.)

As depicted in Figure 14, it can be observed that the QNN, QResNet, and MLP models (group one models) are

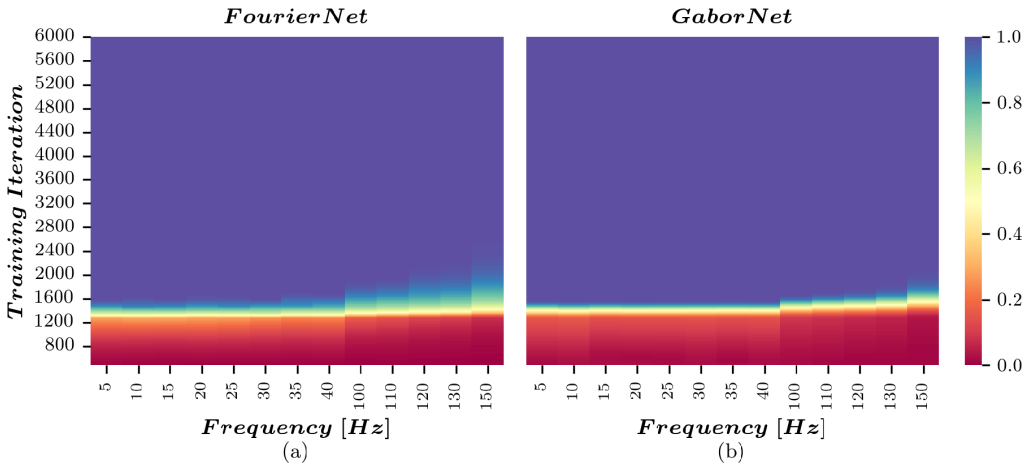

FIGURE 15. Evolution of the frequency spectrum (a) FourierNet Model (b) GaborNet model.

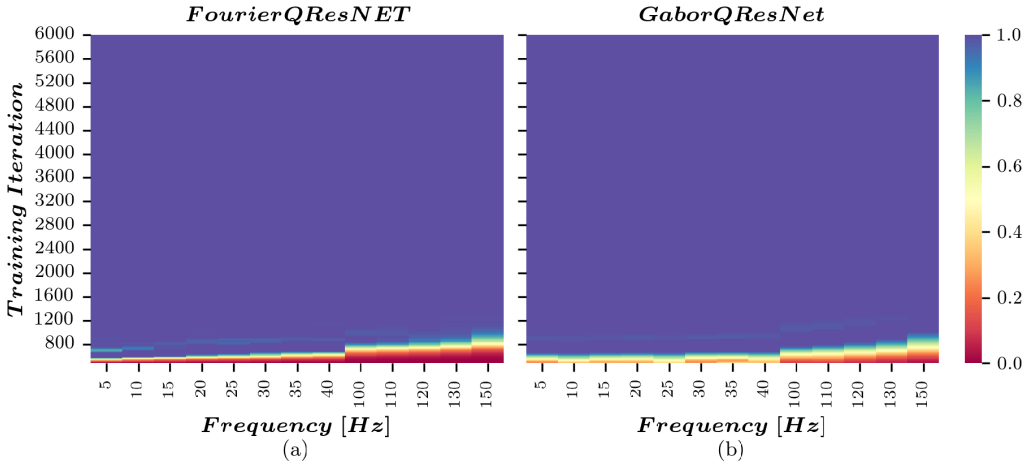

FIGURE 16. Evolution of the frequency spectrum (a) FourierQResNet Model (b) GaborQResNet model.

significantly affected by the phenomenon of spectral bias. It was discovered that these models did not acquire information for frequencies greater than 100 Hz even after 6000 epochs. All the models were capable of acquiring information at frequencies below 50 Hz after reaching 1200 revolutions. After 1200 epochs, information in the frequency range of 0-50 Hz was gradually learned with increasing iteration/epoch count in all models. However, when the iteration number was increased to 6000, the QNN, QResNet, and MLP models were unable to reach accurate values at these frequencies. Among these models, we discovered that QResNet was the least affected by the phenomenon of spectral bias, QNN was the second least affected model, and MLP was the worst affected model. When we investigate the group two model (FourierNet and GaborNet) results, depicted in Figure 15, we notice that these models were slightly affected by the phenomenon of spectral bias. It was discovered that these models accurately acquired information for frequencies between 0-50 Hz in 1200 epochs. After 1200 epochs, the signal information in the 50-150 Hz frequency range was almost equally learned with increasing iteration/epoch count by all models. However, when the iteration count reached 6000 epochs, all models could reach accurate final values of 1.0 in the entire frequency range. Among these models, we found that GaborNet was the least affected spectral bias and was negligibly better than FourierNet. Figure 16 shows the outcomes of the evolution of the frequency spectrum for FourierQResNet and GaborQResNet training, which were evaluated as part of the final visual inspection of the group three models. It was determined that the impact of spectral bias on these models was negligible. The findings indicate that the models effectively obtained signal information within the 0-50 Hz frequency range after a mere 20-40 epochs. Following 40-50 iterations, all models demonstrated comparable acquisition of signal information within the 50-150 Hz frequency range as the number of iterations/epochs increased. Upon reaching approximately 60 epochs, all models were able to attain precise final values of 1.0 across the entire frequency range. Among the set of models examined, GaborQResNet exhibited the least susceptibility to spectral

bias and demonstrated a marginal improvement over FourierQResNet. In conclusion, the suggested models exhibited rapid convergence in the frequency intervals of 40-150 Hz and 0-40 Hz.

According to the findings of the preceding analysis, the proposed models exhibit superior performance in terms of spectral bias, rapid convergence, and learning capacity. Despite being affected by spectral bias, the influence of this phenomenon was negligible. Furthermore, these models were able to acquire knowledge of intricate signal features, including high-frequency components, within a short training period. In a subsequent analysis, we demonstrated this proficiency with regard to two distinct metrics. As a brief reminder, in order to measure the spectral bias at distinct frequencies, we used *FBCM* and *PBRE*. *FBCM* reveals the degree to which any model prediction at the training step corresponds to the actual signal in the frequency domain, and *PBRE* is the relative discrepancy between the neural network prediction and the target function for defined target frequencies k at each training step that can be calculated. These metrics have considerable advantages and provide numerical values for each training step.

Our analysis reveals that the group one models, namely QNN, QResNet, and MLP, were notably affected by the occurrence of spectral bias. The *PBRE* metric variations of the QNN, QResNet, and MLP models during the training are presented in Figure 17. The *PBRE* values exhibited an earlier convergence within the frequency range of 5-40 Hz in the QNN model compared to frequencies exceeding 100 Hz. However, it is noteworthy that this metric remained constant even at the 6000th epoch for higher frequencies. The QResNet model exhibited an earlier convergence of the *PBRE* values within the frequency range of 5-25 Hz. Nevertheless, the aforementioned metric exhibited a comparatively lower rate of decline for frequencies of 30, 35, and 40 Hz. Ultimately, the *PBRE* metric value stabilized for frequencies exceeding 100 Hz subsequent to the conclusion of training. Similar results were observed for the MLP model, as in the case of QNN. However, the convergence rates for the frequencies of 30, 35, and 40 Hz were comparatively slower. However, the convergence rate for the frequencies of 5, 10, 15, 20, and 25 Hz was even slower than that of QNN and QResNet. In conclusion, it was observed that the *PBRE* metric exhibited significant oscillations for the MLP and QResNet models in the lower frequency range. The increase in frequency results in a reduction in the amplitude of the ripple.

According to the observations, the spectral bias phenomenon had little impact on the GaborNet and FourierNet models (which are part of Group Two). The *PBRE* metric variations of the GaborNet and FourierNet models during training are shown in Figure 18. In contrast to prior models in group one, it was observed that during training, the decrease rates of *PBRE* values for frequencies ranging from 5 to 140 Hz reached comparable final values (around $\sim 10^{-6}$) earlier than for the 150 Hz frequency in both models. Furthermore, GaborNet exhibited a marginally superior convergence rate compared with FourierNet, particularly during the initial 1000 epochs. In conclusion, there was no variability in the *PBRE* measure across all frequencies in the aforementioned models.

Empirical observations revealed that the spectral bias phenomenon barely affected the GaborQResNet and FourierQResNet models (group three models). The *PBRE* metric variations of the GaborQResNet and FourierQResNet models during training are shown in Figure 19. In contrast to earlier group one and two models, the reduction rates of *PBRE* values across all frequencies ranging from 5 to 150 Hz converged to comparable final values (approximately $\sim 10^{-9}$) in a rapid manner during the training process. The ultimate *PBRE* metric outcome exhibited a significantly lower magnitude than the models of groups one and two. In conclusion, we noted a marginal fluctuation in the *PBRE* measure across all frequencies subsequent to the attainment of the ultimate values, which persisted until the termination of the training process for these models.

Figure 20 illustrates the variation in the QNN, QResNet, and MLP models during training, specifically in the context of the *FBCM* variation. The QNN model demonstrated a faster convergence of *FBCM* values in the frequency range of 5-40 Hz as opposed to frequencies above 100 Hz. It is important to note that the aforementioned metric exhibited a relatively stable trend after the 1000th epoch with respect to lower frequencies. The QResNet architecture demonstrated accelerated convergence of *FBCM* metrics in the frequency range of 5-25 Hz. However, the metric demonstrated a relatively lower rate of escalation for frequencies of 30, 35, and 40 Hz. The *FBCM* metric value exhibited no improvement beyond 100 Hz at the end of the training. Comparable results were noted in the MLP model, similar to those in the QNN and QResNet scenarios. The convergence rates of QNN and QResNet were higher than that of 5, 10, 15, 20, and 25 Hz frequencies. However, the convergence rates for the frequencies of 30, 35, and 40 Hz were comparatively slower. Ultimately, the *FBCM* metric displayed notable fluctuations across all frequency spectra for each model. The heightened occurrence of this phenomenon led to a decrease in the oscillation magnitude. In all the models examined, it was observed that the *FBCM* metric did not attain the best value of 1.0. Instead, the metric remained either below or above the ideal value at each frequency for all models in group one.

The interpretation of the training process of the FourierNet and GaborNet models, which belong to group two models, using *FBCM* is depicted in Figure 21. The frequency components of the signal reached convergence before 1500 epochs for both models in group two. Both models demonstrated a marginal overshoot before converging to an optimal metric value of 1.0. In contrast to the group one models, the group two models exhibited a constant *FBCM* value throughout the training process without any observed oscillatory behavior. The disparity in the convergence rate between the lower and

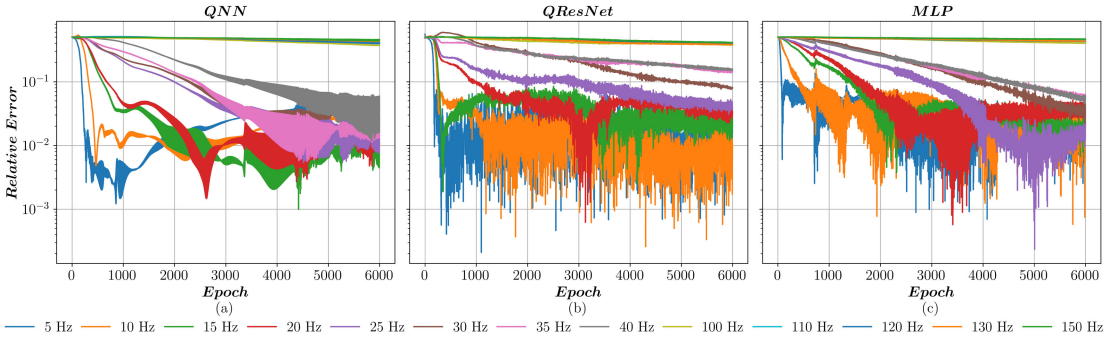

FIGURE 17. Relative errors with projection method (a) QNN Model (b) QResNet Model (c) MLP model.

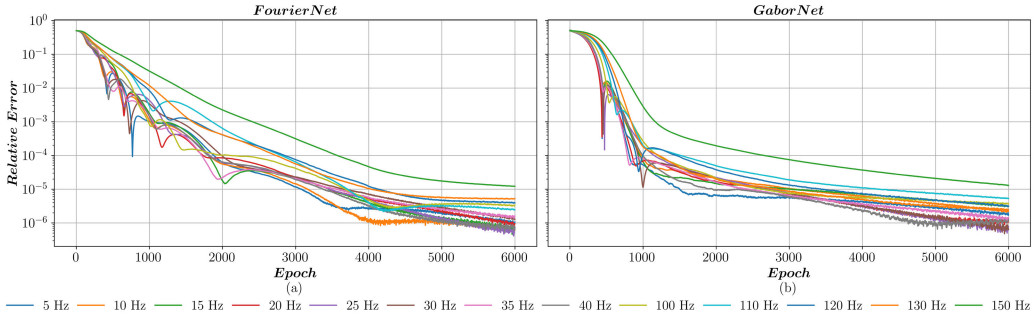

FIGURE 18. Relative errors with projection method (a) FourierNet Model (b) GaborNet model.

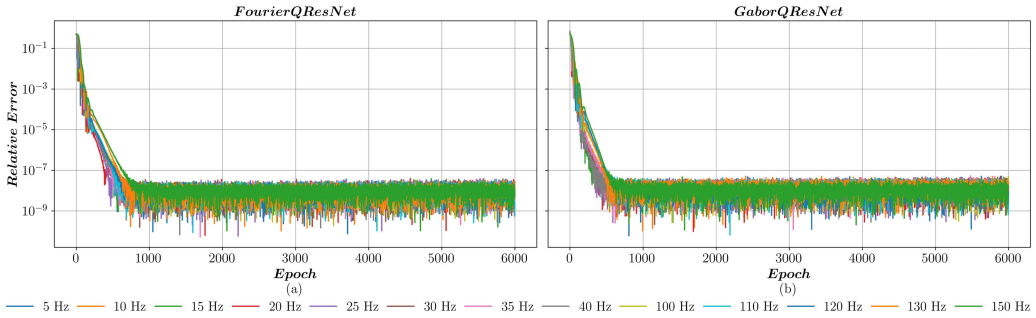

FIGURE 19. Relative errors with projection method (a) FourierQResNet Model (b) GaborQResNet model.

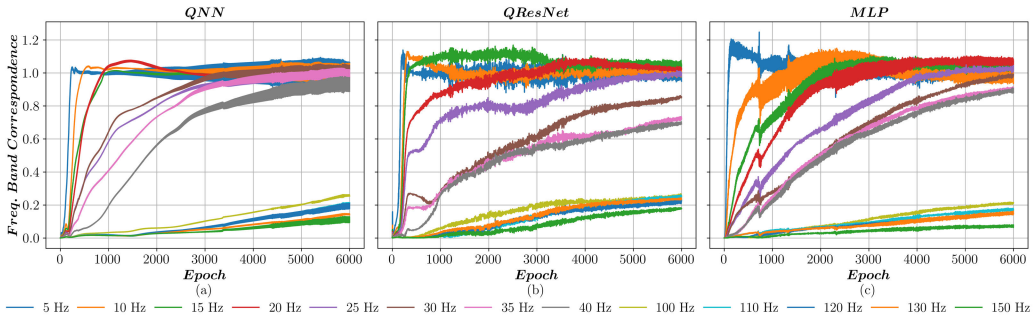

FIGURE 20. Frequency band correspondence metric (a) QNN Model (b) QResNet Model (c) MLP model.

higher frequencies was negligible. In summary, the study found that spectral bias had a minor effect on the performance of the group two models, yet it remained present in these models.

Figure 22 depicts the change in the value of *FBCM* value for the FourierQResNet and GaborQResNet models belonging to group three over the course of their training. The frequency components of the signal achieved convergence prior to 80 epochs for both models in group three. The FourierQResNet model demonstrated a notable overestimation phenomenon which was particularly evident in the lower frequency range of up to 40 Hz. The GaborQResNet model exhibited a minor overshoot for the aforementioned frequencies, after which both models reached a state of stability and

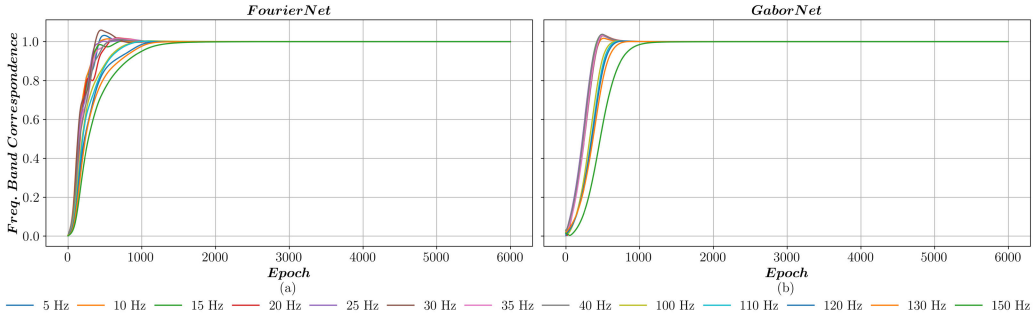

FIGURE 21. Frequency band correspondence metric (a) FourierNet Model (b) GaborNet model.

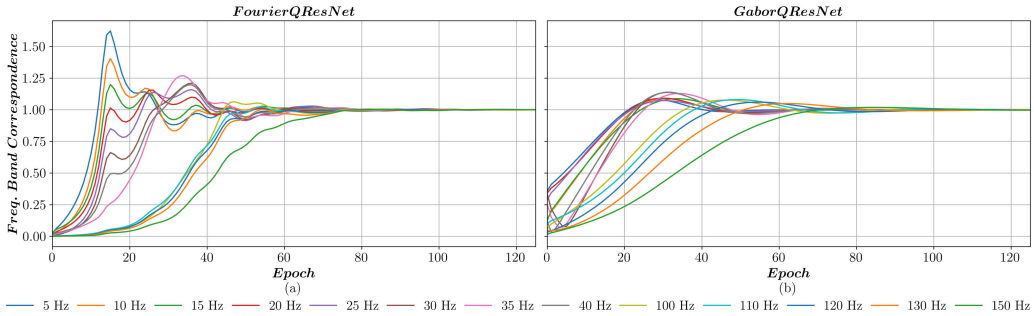

FIGURE 22. Frequency band correspondence metric (Zoomed View Epochs From 0 to 125) (a) FourierQResNet Model (b) GaborQResNet model.

ultimately converged toward an optimal metric value of 1.0. In contrast to the models in group one, the group three models exhibited a consistent *FBCM* value throughout the training process without any observed oscillatory behavior. The rate of convergence exhibited a disparity between the lower and higher frequencies, which was comparatively lower than that observed in the models belonging to groups one and two. Furthermore, the rapidity of the convergence of the frequency components was observed to be the highest among all the models employed in this experiment. In summary, it was discovered that the models still exhibited spectral bias, although their influence was found to be insignificant in group three models.

This study validates our prior hypotheses and demonstrates that the neural network models we have proposed, namely FourierQResNet and GaborQResNet, exhibit remarkable convergence and learning aptitudes for the purpose of learning. Significantly, the proposed neural network models exhibited negligible spectral bias in comparison to the other models examined in this research.

V. CONCLUSION

In this study, we introduced two novel Quadratic Residual Multiplicative Filter Neural Networks (QRMFNNs), FourierQResNet and GaborQResNet, for the efficient approximation of complex sensor signals and investigated their perfect spectral bias performance. The study also examined their optimal spectral bias performance. The empirical findings revealed the enhanced efficacy of FourierQResNet and GaborQResNet compared to conventional techniques and pre-existing neural network architectures. Both networks demonstrated exceptional precision in approximating intricate sensor signals, proficiently mitigating convergence speed. In addition, the networks exhibited remarkable capability in addressing spectral bias, a crucial challenge frequently encountered in analyzing sensor signals and other learning-related tasks.

Through careful examination and assessment, we measured the optimal spectral bias efficacy of FourierQResNet and GaborQResNet in diverse scenarios. The exhibited networks demonstrated resilience and dependability in mitigating the unfavorable impacts of spectral bias, rendering them exceedingly appropriate for practical implementations. The ability of these networks to capture intricate patterns and minimize the impact of spectral bias makes them powerful tools for sensor signal analysis. As a result, the investigation of the perfect spectral bias performance of FourierQResNet and GaborQResNet revealed their robustness and reliability in mitigating spectral bias. By effectively handling variations in signal acquisition conditions and sensor limitations, these networks performed remarkably in eliminating spectral bias, thus ensuring more accurate and reliable signal approximation. These findings have important implications for real-world applications in which spectral bias can significantly affect the quality of sensor data analysis.

FourierQResNet and GaborQResNet present a range of opportunities and potentials to enhance the reliability and appropriateness of sensor data analysis. These advanced neural network architectures effectively address existing challenges and introduce improvements to the field. Firstly they incorporate Fourier and Gabor filters to capture patterns and textures in sensor data. This approach overcomes

convergence speed and spectral bias issues, which often impede data interpretation. FourierQResNet makes use of the Fourier bases to capture patterns in the domain, while GaborQResNet leverages Gabor filtering to extract local spatiotemporal patterns. Both networks excel in approximating sensor signals and exhibit resilience in handling bias resulting in more precise and trustworthy analysis without using activation functions. Moreover, by integrating properties of quadratic residual neural networks and multiplicative filter neural networks, FourierQResNet and GaborQResNet demonstrate improved learning capacity, expressivity, and error reduction when approximating sensor signals compared to traditional methods. The combination of recursive filtering operations instead of activation functions and quadratic residual neural network properties positions FourierQResNet and GaborQResNet as tools for enhancing the reliability and suitability of sensor data analysis across diverse domains.

Although the results are promising, it is essential to acknowledge the limitations of this study. First, the experiments were conducted on a diverse dataset of sensor signals. However, further validation on more extensive and various datasets would strengthen the generalizability of the findings. Additionally, this research focused on perfect spectral bias performance, and future investigations could explore network performance in scenarios with imperfect spectral bias. Future research could explore several areas for further improvement and application. Firstly, the optimization of network architectures and training strategies can enhance the efficiency and accuracy of FourierQResNet and GaborQResNet. In addition, the integration of other advanced filtering techniques and data augmentation methods may offer further improvements in signal approximation and noise reduction. As another future study, experimenting with these models in high-dimensional learning problems would prove their effective use in the domains that require processing high-dimensional data, such as image and sensor signal classification.

In conclusion, the FourierQResNet and GaborQResNet models constitute noteworthy progress within the domain of sensor signal approximation. The findings of this study contribute to the advancement of sensor signal analysis and offer new possibilities for improving the reliability and accuracy of data interpretation in various domains. By addressing these limitations and exploring future research directions, we can continue to refine and enhance the capabilities of these networks, fostering progress in sensor data analysis and its practical applications.

REFERENCES

[1] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0893608089900208

[2] G.-Y. Chen, Y.-H. Yu, M. Gan, C. L. P. Chen, and W. Guo, "Properties and potential applications of random functional-linked types of neural networks," 2023, *arXiv:2304.00957*.

[3] S. Yang, T. Ting, K. L. Man, and S.-U. Guan, "Investigation of neural networks for function approximation," *Proc. Comput. Sci.*, vol. 17, pp. 586–594, Jan. 2013.

[4] T. Chen and H. Chen, "Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems," *IEEE Trans. Neural Netw.*, vol. 6, no. 4, pp. 911–917, Jul. 1995.

[5] S. Yao, A. Piao, W. Jiang, Y. Zhao, H. Shao, S. Liu, D. Liu, J. Li, T. Wang, S. Hu, L. Su, J. Han, and T. Abdelzaher, "STFNets: Learning sensing signals from the time-frequency perspective with short-time Fourier neural networks," in *Proc. World Wide Web Conf.*, May 2019, pp. 2192–2202.

[6] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, and A. Courville, "On the spectral bias of neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 5301–5310.

[7] F.-L. Fan, Y. Li, H. Peng, T. Zeng, and F. Wang, "Towards NeuroAI: Introducing neuronal diversity into artificial neural networks," 2023, *arXiv:2301.09245*.

[8] S. Yu, J. Yang, T.-H. Huang, J. Zhu, C. J. Visco, F. Hameed, J. Stein, X. Zhou, and H. Su, "Artificial neural network-based activities classification, gait phase estimation, and prediction," *Ann. Biomed. Eng.*, vol. 51, no. 7, pp. 1471–1484, Jul. 2023.

[9] S. Romagnoli, F. Ripanti, M. Morettini, L. Burattini, and A. Sbrollini, "Wearable and portable devices for acquisition of cardiac signals while practicing sport: A scoping review," *Sensors*, vol. 23, no. 6, p. 3350, Mar. 2023.

[10] U. Schilt, B. Barahona, R. Buck, P. Meyer, P. Kappani, Y. Möckli, M. Meyer, and P. Schuetz, "Low-cost sensor node for air quality monitoring: Field tests and validation of particulate matter measurements," *Sensors*, vol. 23, no. 2, p. 794, Jan. 2023.

[11] G. Chryssolouris, K. Alexopoulos, and Z. Arkouli, "Artificial intelligence in manufacturing processes," in *A Perspective on Artificial Intelligence in Manufacturing*. Cham, Switzerland: Springer, 2023, pp. 15–39.

[12] X. Liu, Y. Liu, X. Kong, L. Ma, A. H. Besheer, and K. Y. Lee, "Deep neural network for forecasting of photovoltaic power based on wavelet packet decomposition with similar day analysis," *Energy*, vol. 271, May 2023, Art. no. 126963. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544223003572

[13] E. Dejband, Y. C. Manie, Y.-J. Deng, M. A. Bitew, T.-H. Tan, and P.-C. Peng, "High accuracy and cost-effective fiber optic liquid level sensing system based on deep neural network," *Sensors*, vol. 23, no. 4, p. 2360, Feb. 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/4/2360

[14] J. Trejo-Alonso, S. Fuentes, N. Morales-Durán, and C. Chávez, "Evaluation and development of pedotransfer functions and artificial neural networks to saturation moisture content estimation," *Water*, vol. 15, no. 2, p. 220, Jan. 2023.

[15] G. G. Chrysos, S. Moschoglou, G. Bouritsas, Y. Panagakis, J. Deng, and S. Zafeiriou, "P-Nets: Deep polynomial neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7325–7335.

[16] S.-K. Oh, W. Pedrycz, and B.-J. Park, "Polynomial neural networks architecture: Analysis and design," *Comput. Electr. Eng.*, vol. 29, no. 6, pp. 703–725, Aug. 2003.

[17] J. Kileel, M. Trager, and J. Bruna, "On the expressive power of deep polynomial neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–10.

[18] J. J. Moghaddam, D. Momeni, and G. Zarei, "Thermal, ANFIS, and polynomial neural network models for predicting environmental variables in an arch greenhouse," *J. Agricult. Sci. Technol.*, vol. 24, no. 3, pp. 617–633, 2022.

[19] W. Waheeb, R. Ghazali, and T. Herawan, "Ridge polynomial neural network with error feedback for time series forecasting," *PLoS ONE*, vol. 11, no. 12, Dec. 2016, Art. no. e0167248.

[20] F. Fan, J. Xiong, and G. Wang, "Universal approximation with quadratic deep networks," *Neural Netw.*, vol. 124, pp. 383–392, Apr. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608020300095

[21] F.-L. Fan, H.-C. Dong, Z. Wu, L. Ruan, T. Zeng, Y. Cui, and J.-X. Liao, "One neuron saved is one neuron earned: On parametric efficiency of quadratic networks," 2023, *arXiv:2303.06316*.

[22] J.-X. Liao, H.-C. Dong, Z.-Q. Sun, J. Sun, S. Zhang, and F.-L. Fan, "Attention-embedded quadratic network (qttention) for effective and interpretable bearing fault diagnosis," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–13, 2023.

[23] F.-L. Fan, M. Li, F. Wang, R. Lai, and G. Wang, "On expressivity and trainability of quadratic networks," 2021, *arXiv:2110.06081*.

[24] K. Xu, H. Bastani, and O. Bastani, "Robust generalization of quadratic neural networks via function identification," 2021, *arXiv:2109.10935*.
[25] J. Bu and A. Karpatne, "Quadratic residual networks: A new class of neural networks for solving forward and inverse problems in physics involving PDEs," in *Proc. SIAM Int. Conf. Data Mining (SDM)*. Philadelphia, PA, USA: SIAM, 2021, pp. 675–683.
[26] Z. Xu, F. Yu, J. Xiong, and X. Chen, "QuadraLib: A performant quadratic neural network library for architecture optimization and design exploration," 2022, *arXiv:2204.01701*.
[27] R. Fathony, A. K. Sahu, D. Willmott, and J. Z. Kolter, "Multiplicative filter networks," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–11.
[28] D. B. Lindell, D. Van Veen, J. J. Park, and G. Wetzstein, "Bacon: Band-limited coordinate networks for multiscale scene representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 16231–16241.
[29] S. Shekarforoush, D. Lindell, D. J. Fleet, and M. A. Brubaker, "Residual multiplicative filter networks for multiscale reconstruction," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 8550–8563.
[30] Y. Ma, Z.-Q. J. Xu, and J. Zhang, "Frequency principle in deep learning beyond gradient-descent-based training," 2021, *arXiv:2101.00747*.
[31] Z. J. Xu, "Understanding training and generalization in deep learning by Fourier analysis," 2018, *arXiv:1808.04295*.
[32] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, and R. Ng, "Fourier features let networks learn high frequency functions in low dimensional domains," in *Advances in Neural Information Processing Systems*, vol. 33, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds. Red Hook, NY, USA: Curran Associates, 2020, pp. 7537–7547.
[33] V. Sitzmann, J. N. P. Martel, A. W. Bergman, D. B. Lindell, and G. Wetzstein, "Implicit neural representations with periodic activation functions," 2020, *arXiv:2006.09661*.
[34] V. Saragadam, D. LeJeune, J. Tan, G. Balakrishnan, A. Veeraraghavan, and R. G. Baraniuk, "WIRE: Wavelet implicit neural representations," 2023, *arXiv:2301.05187*.
[35] Z.-Q. J. Xu and H. Zhou, "Deep frequency principle towards understanding why deeper learning is faster," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 10541–10550.
[36] Z.-Q. J. Xu, Y. Zhang, and T. Luo, "Overview frequency principle/spectral bias in deep learning," 2022, *arXiv:2201.07395*.
[37] S. Ramasinghe, L. E. MacDonald, and S. Lucey, "On the frequency-bias of coordinate-MLPs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 796–809.
[38] M. Choraria, L. Dadi, G. G. Chrysos, J. Mairal, and V. Cevher, "The spectral bias of polynomial neural networks," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–30.
[39] Y. Wu, Z. Zhu, F. Liu, G. G. Chrysos, and V. Cevher, "Extrapolation and spectral bias of neural nets with Hadamard product: A polynomial net study," 2022, *arXiv:2209.07736*.
[40] J. Lazzari and X. Liu, "Understanding the spectral bias of coordinate based MLPs via training dynamics," 2023, *arXiv:2301.05816*.
[41] S. Fridovich-Keil, R. G. Lopes, and R. Roelofs, "Spectral bias in practice: The role of function frequency in generalization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 7368–7382.
[42] Z. Shi, P. Mettes, S. Maji, and C. G. M. Snoek, "On measuring and controlling the spectral bias of the deep image prior," *Int. J. Comput. Vis.*, vol. 130, no. 4, pp. 885–908, Apr. 2022.
[43] Q. Hong, J. W. Siegel, Q. Tan, and J. Xu, "On the activation function dependence of the spectral bias of neural networks," 2022, *arXiv:2208.04924*.
[44] M. M. Lau and K. Hann Lim, "Review of adaptive activation function in deep neural network," in *Proc. IEEE-EMBS Conf. Biomed. Eng. Sci. (IECBES)*, Dec. 2018, pp. 686–690.
[45] A. D. Jagtap, K. Kawaguchi, and G. E. Karniadakis, "Adaptive activation functions accelerate convergence in deep and physics-informed neural networks," *J. Comput. Phys.*, vol. 404, Mar. 2020, Art. no. 109136.
[46] A. D. Jagtap, K. Kawaguchi, and G. Em Karniadakis, "Locally adaptive activation functions with slope recovery for deep and physics-informed neural networks," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 476, no. 2239, Jul. 2020, Art. no. 20200334.
[47] A. Filgöz, G. Demirezen, and M. U. Demirezen, "Applying novel adaptive activation function theory for launch acceptability region estimation with neural networks in constrained hardware environments: Performance comparison," in *Proc. IEEE/AIAA 40th Digit. Avionics Syst. Conf. (DASC)*, Oct. 2021, pp. 1–10.
[48] A. D. Jagtap, Y. Shin, K. Kawaguchi, and G. E. Karniadakis, "Deep Kronecker neural networks: A general framework for neural networks with adaptive activation functions," *Neurocomputing*, vol. 468, pp. 165–180, Jan. 2022.
[49] J. Kiessling and F. Thor, "A computable definition of the spectral bias," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2022, vol. 36, no. 7, pp. 7168–7175. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/20677
[50] Z.-Q.-J. Xu, "Frequency principle: Fourier analysis sheds light on deep neural networks," *Commun. Comput. Phys.*, vol. 28, no. 5, pp. 1746–1767, Jun. 2020.
[51] W. Cai, X. Li, and L. Liu, "A phase shift deep neural network for high frequency approximation and wave problems," *SIAM J. Sci. Comput.*, vol. 42, no. 5, pp. A3285–A3312, Jan. 2020, doi: 10.1137/19M1310050.
[52] Z. Zeng, P. Shi, F. Ma, and P. Qi, "Parallel frequency function-deep neural network for efficient approximation of complex broadband signals," *Sensors*, vol. 22, no. 19, p. 7347, Sep. 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/19/7347
[53] R. Zhang, "Learning high frequency data via the coupled frequency predictor-corrector triangular DNN," *Japan J. Ind. Appl. Math.*, vol. 40, pp. 1259–1285, Mar. 2023.
[54] Z. Huang, S. Bai, and J. Z. Kolter, "(Implicit)^2: Implicit layers for implicit representations," in *Advances in Neural Information Processing Systems*, vol. 34, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds. Red Hook, NY, USA: Curran Associates, 2021, pp. 9639–9650.
[55] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.
[56] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, *arXiv:1609.04747*.
[57] F. Fan, W. Cong, and G. Wang, "A new type of neurons for machine learning," *Int. J. Numer. Methods Biomed. Eng.*, vol. 34, no. 2, p. e2920, Feb. 2018.
[58] N. Matloff, *Statistical Regression and Classification: From Linear Models to Machine Learning*, 1st ed. Boca Raton, FL, USA: CRC Press, 2017.

MUSTAFA UMUT DEMIREZEN (Senior Member, IEEE) received the B.S., M.Sc., and Ph.D. degrees in electrical and electronics engineering from Gazi University in 2000, 2002, and 2015, respectively. From 2000 to 2013, he was an Engineer Officer with the Turkish Naval Forces. After resigning from the Turkish Naval Forces, he was with several defense companies, such as STM Defence Inc. and ROKETSAN Missiles Inc., as a Researcher and then the Group Manager for research and development projects in artificial intelligence, from 2013 to 2021. He was also the Research and Innovation Manager with Huawei Research and Development Center. Since 2023, he has been with the Department of Data Products, UDemy Inc., as the Senior Manager of Machine Learning Engineering. He is the author of five artificial intelligence subjected books and more than 20 scientific articles. His research interests include artificial intelligence, deep learning, neuromorphic computing, spiking neural networks, scientific machine learning, and physics-informed neural networks. He was a recipient of the AIAA/IEEE Digital Avionics Systems Conference (DASC) Best Session Paper Award, in 2021, and IEEE UBMK-2022, and the International Conference on Computer Science and Engineering, Best Paper (3rd place) Award, in 2022. He is an Associate Editor of the journal of *Measurement and Control*.