Received 23 February 2026; revised 16 March 2026; accepted 18 March 2026.

Digital Object Identifier 10.1109/OJAP.2026.3676263

Accuracy and Computational Cost of Machine Learning Models for Electromagnetic Object Classification

ERGUN SIMSEK (Senior Member, IEEE)

University of Maryland, Baltimore County, Baltimore, MD 21250, USA

e-mail: simsek@umbc.edu

ABSTRACT Classification of randomly oriented objects from electromagnetic data is important in applications such as remote sensing and security screening, where robustness to simulation/measurement configuration and data variability is critical. While prior work has shown that attention mechanisms can significantly improve classification performance for long, highly structured electromagnetic scattering sequences, their benefit in simpler data regimes remains unclear. In this numerical work, we study classification using plane-wave excitation and electromagnetic field data recorded by a limited number of receiver antennae, resulting in short input sequences. We show that under these conditions, a simple one-dimensional convolutional neural network achieves nearly the same accuracy as attention-based models. We further compare neural networks with classical machine learning methods, including support vector machines (SVMs) and extreme gradient boosting (XGB), and demonstrate that for large datasets these methods perform comparably to neural networks, whereas neural networks offer a clear advantage in the small-data regime. However, neural networks' tuning and training times can be substantially higher than those of SVMs and XGB. These results provide practical guidance on selecting model complexity for electromagnetic classification based on dataset size and structure, computational resources, and time.

INDEX TERMS Machine learning, electromagnetic waves, classification, neural networks.

I. INTRODUCTION

ELECTROMAGNETIC inversion and electromagnetic classification represent two complementary approaches for interpreting wave–matter interactions from the electromagnetic field data. Electromagnetic inversion focuses on reconstructing the spatial distribution of material (such as permittivity, conductivity, or permeability [1], [2], [3], [4], [5], [6], [7]) or structural properties (such as shape [8]) of an object or medium, by solving an ill-posed and nonlinear inverse problem. Due to the dispersive nature of electromagnetic wave propagation, inversion problems are often computationally intensive and sensitive to noise, requiring careful regularization and iterative [1], [2], [3], [4] or machine learning [6], [7] based solvers to obtain physically meaningful solutions.

Electromagnetic classification, in contrast, seeks to identify or categorize objects directly from their electromagnetic response without explicitly reconstructing their internal material distributions [9], [10], [11], [12], [13]. Rather than estimating continuous-valued parameters, classification assigns discrete labels based on characteristic signatures embedded in the scattered fields. This is particularly appealing in applications where object recognition is the primary objective and detailed reconstructions are unnecessary. Given the high dimensionality and structured nature of electromagnetic scattering data, machine learning (ML) methods have emerged as effective tools for both inversion [5], [14], [15], [16], [17] and classification [9], [10], [11], [12], enabling data-driven extraction of discriminative features that are difficult to capture analytically.

While it is well-established that qualitative methods in inverse scattering theory, such as those pioneered in [8], are highly effective at retrieving the support or shape of targets, particularly when the scatterer size is comparable to the wavelength (e.g., $\lambda/2$), the objective of the present study is fundamentally different. Qualitative methods focus on geometric reconstruction and solving the inverse problem to determine the physical boundary of an object. In contrast, the approach proposed here bypasses explicit inversion entirely. By adopting machine learning models for direct object classification, we aim to map scattered electromagnetic signatures to discrete labels without the intermediate step of shape

reconstruction. Much like optical character recognition, the challenge here is not merely to recover a geometric boundary, but to recognize the identity of the class despite significant variations in object orientation, as discussed next. Thus, what we propose here can serve as a complementary tool for scenarios where categorical identification is the primary goal, and where the computational overhead of full or qualitative reconstruction may be unnecessary for decision-making.

A key challenge in electromagnetic classification is robustness to object orientation [9], [13], [18], [19]. Classical ML methods such as support vector machines (SVMs), extreme gradient boosting (XGB), and conventional neural networks (NNs) can achieve high classification accuracy (87%) when objects are observed under fixed or known orientations [9]. However, their performance might degrade significantly for arbitrarily oriented targets; for example, the accuracy reported in [9] drops to approximately 57% under random orientations. This sensitivity to orientation-induced variability limits the practicality of such approaches in realistic sensing scenarios [18], [19]. In [18], a physics-based model is developed that is parameterized by decay constants (Laplace-plane poles) derived from the measured electromagnetic induction response. These poles are theoretically invariant to object position and orientation. The researchers jointly estimate location, orientation, and pole parameters using a maximum-likelihood framework, and demonstrate that the extracted poles can serve as robust classification features, particularly under high signal-to-noise ratios. Similarly, in [19], the researchers employ the singularity expansion method to obtain compact representations of ultra-wideband scattered fields that are independent of object orientation and observation angle. By coupling singularity-expansion-based preprocessing with convolutional neural networks, they show improved classification robustness across varying noise levels and object sizes not included in the training set.

In our prior work [13], we addressed this limitation by introducing an attention-based neural network architecture tailored to highly structured, long-sequence electromagnetic data. In that study, the scattered field data were collected from 12 receiver antennae for each of 12 transmitting antennae, excited sequentially. This simulation configuration produced long sequences (i.e., $12 \times 12 \times 2$ components $= 288$ or 576 real-valued entries) and the attention mechanism played a critical role by selectively emphasizing the most informative transmitter–receiver interactions, leading to a substantial improvement in classification accuracy compared to conventional architectures (87% vs. 81%).

In the present study, however, the data characteristics are fundamentally different. We consider plane-wave excitations and data collected at 16 uniformly distributed receiver antennae, resulting in a significantly shorter (i.e., 16×2 components $= 32$ or 64 real-valued entries), more homogeneous input sequence. Because the temporal and spatial dependencies in the data are far less complex, the representational power of an attention mechanism is no longer essential. Our results show that a simple one-dimensional convolutional neural network (1D CNN) is sufficient to capture relevant scattering features and achieves nearly the same classification accuracy as more complex attention-based models. In this regime, attention constitutes an unnecessary overhead rather than a performance-enabling component.

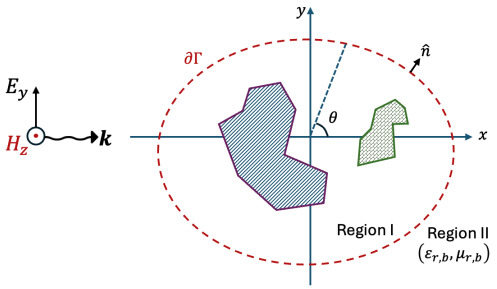

FIGURE 1. Schematic of a 2D electromagnetic scattering problem. The artificial boundary ($\partial\Gamma$), shown with a red dashed curve, divides the computational domain into Regions I and II. The two shapes represent the scatterers. A plane wave is excited along the x-axis.

Moreover, a systematic comparison across dataset sizes reveals an important trade-off between model complexity and data availability. When the dataset is large, classical ML methods such as SVM and XGB achieve accuracies comparable to those of neural networks, albeit with substantially lower training and inference complexity. As evidenced by our results, SVM and XGB performances steadily improve with increasing data volume and approach neural network accuracy for sufficiently large datasets. In contrast, the primary advantage of neural networks becomes evident in the small-data regime, where they consistently outperform SVM and XGB by learning more robust feature representations from limited training samples. These observations underscore that the choice of model architecture should be guided not only by achievable accuracy, but also by the size and structure of the available electromagnetic dataset.

The remainder of this paper is organized as follows. Section II briefly reviews the hybrid spectral integral-finite element (SI–FE) method used for electromagnetic field simulations. Section III describes the construction of the numerical dataset. Section IV details the hyperparameter tuning process, the NN architectures used in this study, and presents numerical results and performance comparisons with classical machine learning techniques. Then, we conclude.

II. SPECTRAL INTEGRAL-FINITE ELEMENT METHOD

Consider a general 2D inhomogeneous object or a collection of objects in an unbounded homogeneous medium, as illustrated in Fig. 1.

The objects are assumed to be infinitely long in the z direction so that the problem can be modeled in two dimensions in the x–y plane. The structure is illuminated by a single transverse magnetic (TM) polarized plane wave whose magnetic field is polarized along the z direction. The incident plane wave propagates along the positive x-axis, i.e., $H_z^{inc}(x, y) = H_0 e^{-jk_b x}$, where H_0 is the amplitude of the

incident magnetic field and k_b is the wavenumber of the homogeneous background medium in Region II.

To compute the electromagnetic fields scattered by the objects using the finite-element method (FEM), a radiation boundary condition (RBC) is required to truncate the computational domain, enabling FEM to be applied within the interior region (Region I). By imposing an appropriate RBC on the boundary $\partial\Gamma$, the electromagnetic field in the homogeneous exterior region (Region II) can be determined once the solution for Region I, including its boundary, is obtained. In Region I, the material properties vary spatially, characterized by a relative permeability $\mu_r(x, y)$ and a relative permittivity $\varepsilon_r(x, y)$. In contrast, Region II consists of a homogeneous medium with constant relative magnetic permeability $\mu_{r,b}$ and relative permittivity $\varepsilon_{r,b}$, encompassing Region I.

Because the objects are invariant along the z direction, the electromagnetic problem reduces to a 2D scalar formulation for TM polarization. The only nonzero magnetic field component is the longitudinal field $H_z(x, y)$. Although the governing equation is scalar, the associated electric field remains vectorial and has components in the x and y directions given by

$$E_x = \frac{1}{j\omega\varepsilon_0\varepsilon_r}\frac{\partial H_z}{\partial y}, \quad E_y = -\frac{1}{j\omega\varepsilon_0\varepsilon_r}\frac{\partial H_z}{\partial x} \tag{1}$$

where ω is the angular frequency and ε_r is the relative permittivity of the medium.

For transverse magnetic (TM) waves, the total magnetic field $H_z(x, y)$ in Region I (bounded by $\partial\Gamma$) is determined by solving the following scalar wave equation:

$$\nabla \cdot \left(\varepsilon_r^{-1}(x, y)\nabla H_z\right) + k_0^2\mu_r(x, y)H_z = S(x, y), \tag{2}$$

where k_0 is the wavenumber in free space, and $S(x, y)$ represents the excitation. To discretize Eq. (2), we multiply the equation by a testing function $W_m(x, y)$, where m is the index of the testing function, and integrate over Region I. Applying vector identities and Gauss's theorem, we derive the weak-form equation:

$$\int\int_\Gamma \left[k_0^2\mu_r W_m H_z - \frac{1}{\varepsilon_r}\nabla W_m \cdot \nabla H_z\right] dxdy + \oint_{\partial\Gamma} W_m \frac{1}{\varepsilon_r}\frac{\partial H_z}{\partial n} ds = \int\int_\Gamma W_m S dxdy, \tag{3}$$

where $\hat{n}$ is the outward unit normal vector on the boundary $\partial\Gamma$. Note that in Eq. (3), ε_r, μ_r, W_m, and S are all functions of x and y.

To solve Eq. (3) using the FEM scheme, Region I is discretized into triangular elements. A linear pyramid basis function $P_n(x, y)$ is used to expand the magnetic field $H_z(x, y)$ in the interior region, while a triangular basis function $T_n(x, y)$ is employed to expand the boundary value $H_{z,n}$ on $\partial\Gamma$. The nodal points of the boundary basis function are collocated with those of the pyramid basis function on the boundary. We use pyramid functions for testing as well and obtain the final set of linear equations [20]. Since there are more unknowns than the number of equations, we need additional conditions to obtain a unique solution of the system. In this work, the spectral integral method (SIM) [13], [20], [21] serves as an RBC to provide these additional conditions by relating the fields inside Region I to those on the outer boundary $\partial\Gamma$. To achieve this, we utilize Green's theorem to derive a surface integral equation that expresses the field outside $\partial\Gamma$ in terms of the field values on the boundary.

We define the boundary integral equation using Green's function $G(k_b, R)$ and its normal derivative $\partial G(k_b, R)/\partial n'$ [21]. We express the fields along $\partial\Gamma$ in terms of a smooth boundary parameterized by θ. By leveraging Fourier series expansions, we approximate the field $H(\theta')$ and its normal derivative $\partial H_z(\theta')/\partial n'$ using a finite number of Fourier coefficients. This transforms the integral equation into a system of algebraic equations in Fourier space. However, since Green's function is singular at $\theta = \theta'$, we apply a singularity subtraction technique to improve numerical convergence [21].

Once we compute the Fourier coefficients of the unknown magnetic field and its normal derivative, we obtain the final equation in matrix form, allowing us to enforce the RBC at discrete boundary points. By using the fast Fourier transform (FFT) and spectral interpolation, we efficiently compute these coefficients, resulting in fast convergence and high spectral accuracy. Note that the SIM RBC effectively suppresses spurious solutions (such as fictitious resonant frequencies) commonly encountered in integral equation solvers. This makes our solver a robust and efficient approach for coupling FEM with an accurate radiation boundary condition.

III. DATASET CREATION

The Modified National Institute of Standards and Technology (MNIST) dataset, a widely recognized resource in machine learning and computer vision, provides a collection of handwritten digits, as detailed in [22]. This dataset is frequently employed for training and evaluating algorithms, particularly those designed for image classification and character recognition. Each image in MNIST is a 28×28-pixel grayscale square representing a handwritten digit, ranging from 0 to 9, with each image accompanied by a corresponding label indicating the digit's identity.

In this study, we utilize 12,000 digital images from the MNIST dataset to construct a scatterer database. Specifically, the grayscale intensity of each pixel, denoted as $d_{u,v}$, where u and v represent the row and column indices ranging from 1 to 28, is converted into relative electrical permittivity values. These pixel values, scaled between 0 and 255, where 0 signifies white and 255 signifies black, are transformed into permittivity values ranging from $\epsilon_r^{\min}$ to $\epsilon_r^{\max}$ using the following linear relationship:

$$\epsilon_r(u, v) = \epsilon_r^{\min} + (\epsilon_r^{\max} - \epsilon_r^{\min})\frac{d_{u,v}}{255} \tag{4}$$

where u and v are the row and column numbers. We set $\epsilon_r^{\min} = 1$ and $\epsilon_r^{\max} = 4$. Note that increasing ϵ_r^{max} strengthens

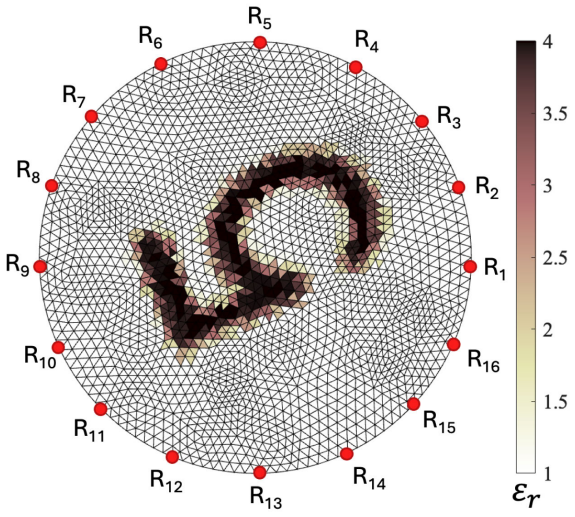

FIGURE 2. Illustration of the computational domain. 16 receiver antennae, shown with red circles, are placed uniformly on the circular boundary with a radius of $2\lambda/3$. The permittivity values of the triangles where the scatterer resides are determined via a 2D interpolation from the MNIST images using Eq. (4).

the scattering contrast. A stronger contrast produces more distinctive scattering signatures, and this makes classification easier [9]. To keep the problem more challenging and avoid trivial discrimination, we intentionally limited the $\epsilon_r^{\max}$ to 4.

The computational domain, illustrated in Fig. 2, features 16 receiver antennae, shown with red circles. These antennae are uniformly positioned at radial distances of $\rho = 2\lambda/3$. The permittivity values within the triangular elements of the scatterer are determined by 2D interpolation of randomly rotated MNIST images, providing an approximate representation of the digit's shape in the simulation environment.

The maximum length of the scatterers is limited to $\lambda/2$. Since all the dimensions are given in terms of λ and we assume lossless materials, the frequency does not matter. However, we used $f = 3$ GHz in our SI-FE simulations and stored the real and imaginary parts of the x and y components of the electric field at 16 receiver points for 12,000 scatterers created from the MNIST dataset.[1] Since E_x is proportional to $\partial H_z/\partial y$ and E_y is proportional to $\partial H_z/\partial x$, by providing these components, we are effectively giving the network local gradient features, so that the network sees edge/shape information of the scattering pattern, similar to gradient features used in classical computer vision.

In this study, receiver antennae are modeled as ideal point-like field sampling probes. Mutual coupling, radiation patterns, and antenna loading effects are neglected in order to isolate the impact of data structure and sequence length on classification performance. In practice, real receiving antennae may influence the measured scattered fields in several ways. Finite aperture size may introduce spatial averaging effects, non-ideal receiving patterns may modify angular sensitivity, polarization impurity may mix field components, and mutual coupling between closely spaced receivers may distort the measured signals. In addition, impedance mismatch, finite bandwidth, amplifier characteristics, and measurement noise may alter the amplitude and phase of the recorded data. All of these effects may alter the statistical distribution of the input features and, if not properly accounted for, affect classification accuracy.

[1]The dataset and codes implementing all the neural networks can be found at https://github.com/simsekergun/RandomlyRotated

IV. NUMERICAL RESULTS

A. HYPERPARAMETER OPTIMIZATION AND MEMORY USAGE

In our earlier study [13], we reported that NNs, SVMs, and XGB were more accurate than linear regression, k-nearest neighbor, and random forests for object classification from electromagnetic data. This is why we use NNs, SVMs, and XGB for classifying randomly oriented objects to present a comparative evaluation. To ensure a fair and well-regularized comparison among the machine learning classifiers, we first performed hyperparameter tuning as follows.

For the CNN, hyperparameter tuning was performed using Bayesian optimization, which models the validation loss as a probabilistic function of the hyperparameters and iteratively selects promising configurations for evaluation. The search space included architectural and regularization parameters, and a maximum of twenty distinct hyperparameter configurations were explored. To limit the overall computational burden, each trial was trained for a maximum of fifty epochs, and a strict early-stopping criterion based on the validation loss was applied. Training was terminated when no improvement was observed over twelve consecutive epochs, and the weights corresponding to the best validation performance were restored. Despite these measures, the total elapsed time for the Bayesian optimization process was approximately 3 hours and 41 minutes, reflecting the high computational cost associated with repeatedly training deep neural networks, even for relatively compact one-dimensional architectures.

Figure 3 illustrates the optimized CNN architecture. The input is a tensor of size 16×4, where the sequence length corresponds to the number of receiver antennae and the feature dimension represents the recorded electromagnetic field components (real and imaginary parts of the electric field's x and y components).

The first convolutional layer applies 192 one-dimensional filters with a kernel size of three to enable the network to learn local spatial correlations among neighboring receiver data. A rectified linear unit (ReLU) activation introduces nonlinearity, while ℓ_2 weight regularization with coefficient $\lambda = 1.35 \times 10^{-5}$ constrains the model complexity and mitigates overfitting. Batch normalization is subsequently applied to stabilize the training dynamics by normalizing intermediate feature distributions, followed by spatial dropout, which randomly suppresses entire feature maps to encourage robustness against channel-wise co-adaptation. A second

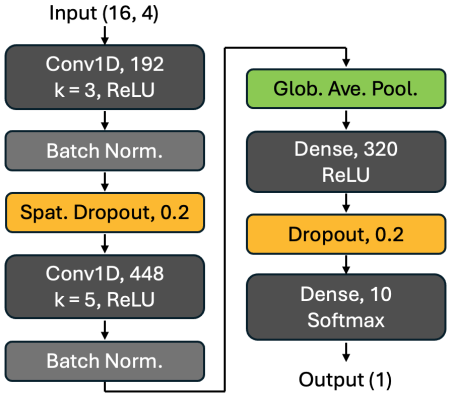

FIGURE 3. One-dimensional CNN-based neural network for electromagnetic object classification. Conv1D: one-dimensional convolution layer, Norm: normalization, Glob. Ave. Pool.: global average pooling.

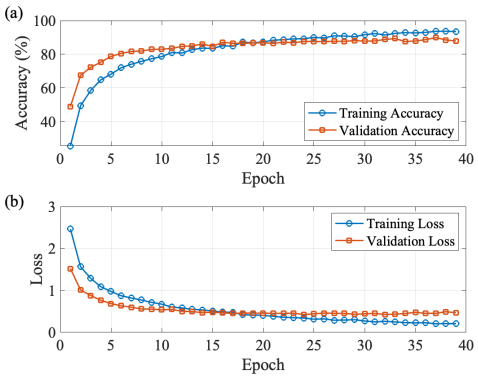

FIGURE 4. (a) Accuracy and (b) loss vs. epoch for CNN after hyperparameter tuning.

convolutional layer with 448 filters and a larger kernel size (five) further increases the network's representational capacity and captures longer-range spatial dependencies across the receiver array. As in the first layer, ReLU activation and ℓ_2 regularization are employed to balance expressiveness and generalization. Batch normalization is again used to improve convergence. The resulting feature maps are then aggregated using global average pooling, which collapses the spatial dimension by computing the mean response of each feature map. This operation reduces the number of trainable parameters, enforces translation-invariant representations, and serves as an implicit regularizer. The pooled features are passed to a fully connected layer with 320 neurons and ReLU activation, enabling nonlinear mixing of the learned convolutional features into a compact latent representation suitable for classification. Dropout is applied at this stage to further reduce overfitting by randomly deactivating a fraction of the neurons during training. The final output layer consists of a fully connected softmax classifier with ten units, corresponding to the number of object classes, and produces a normalized probability distribution over class labels. The network is trained using the Adam optimizer with a learning rate of 10^{-4}, which provides adaptive moment-based updates for efficient and stable convergence. The sparse categorical cross-entropy loss function is used to accommodate integer-valued class labels, and classification accuracy is monitored as the primary performance metric. All the codes were executed on Google Colaboratory using T4 GPU accelerators.

The memory footprint of CNN is dominated by the storage of trainable parameters and intermediate activations during training. The total number of trainable parameters, determined by the convolutional, fully connected, and batch normalization layers, is approximately 5.8×10^5. Using single-precision floating-point representation, the model parameters require approximately 2.3 MB of memory. During training, additional memory is required to store intermediate feature maps and their gradients. For the largest dataset considered, the peak memory usage during training, including activations, gradients, and optimizer states, remained on the order of tens of megabytes. While this memory requirement is higher than that of SVM and XGB, as reported below, it remains modest compared to large-scale deep learning architectures and is well within the limits of modern GPU accelerators.

To demonstrate that our CNN model indeed learns, we provide Fig. 4, where we plot accuracy and loss as a function of epoch number, where 90% of the dataset is used for training, and the remaining 10% is used for testing. We observe that the model learns extremely quickly in the first 10 epochs, jumping from 25.5% to over 80% accuracy, indicating that the architecture and learning rate are well-suited to the fundamental patterns in our dataset. However, starting around epoch 24, a gap begins to widen between the training and validation accuracies. Training accuracy continues to climb steadily toward 93.5%, whereas the validation accuracy plateaus and fluctuates between 87% and 89%. Despite this moderate overfitting, we consider an accuracy of over 87% in classifying randomly oriented objects a success.

For the SVM classifier, hyperparameter selection was performed via an exhaustive grid search over the regularization parameter C and the radial basis function kernel width parameter γ. A five-fold cross-validation scheme was used to estimate the classification accuracy for each parameter combination, and parallel computation was enabled to accelerate the search. The optimum parameters were determined to be $C = 100$ and $\gamma = 0.01$. Due to the relatively small number of tunable parameters and the convex nature of the SVM optimization problem, the total grid search time was significantly shorter, approximately 936 seconds. This substantial reduction in tuning time highlights the computational efficiency of classical kernel-based methods when the dimensionality of the hyperparameter space is limited. The memory footprint of the SVM classifier is dominated by the storage of the training data and the support vectors and scales as $\mathcal{O}(N_{\mathrm{SV}}D)$, where $D = 16 \times 4 = 64$ is the feature dimension. For the largest dataset considered in this study, the total memory usage remained below 10 MB.

TABLE 1. Time spent during hyperparameter optimization for three machine learning models.

Classifier	NN	SVM	XGB
Optimization Time (min.)	221	15.5	60
Accuracy (%)	87.17±1.1	87.42±1.2	84.08±0.9

The XGB classifier was trained using a fixed set of commonly adopted hyperparameters, including the number of trees, learning rate, tree depth, and subsampling ratios, which were selected to balance model expressiveness and overfitting. The hyperparameter optimization converged to a moderately deep gradient-boosted tree ensemble with 311 estimators, a learning rate of 0.1, and balanced regularization ($\alpha = 0.05$, $\lambda = 1.5$), indicating that strong nonlinear modeling capacity combined with controlled model complexity and stochastic subsampling ratios of 0.6 and 0.633 for rows and columns of the dataset, respectively, were essential for achieving optimal performance. The hyperparameter search time for XGB was approximately an hour, exceeding that of the SVM grid search and approaching the cost of the neural network tuning procedure. This relatively long duration is due to the sequential nature of gradient boosting, in which multiple decision trees are trained iteratively and evaluated on a validation set. The memory footprint of the XGB classifier is primarily determined by the storage of the ensemble of decision trees and the input data. For gradient-boosted trees, the model memory scales as $\mathcal{O}(T \times 2^d)$, where T is the number of trees and d is the maximum tree depth. The optimized model uses a maximum depth of $d = 6$, corresponding to at most $2^6 = 64$ nodes per tree. Each node stores a small number of floating-point values, including the split threshold, feature index, child pointers, and leaf weights, resulting in a modest per-node memory cost. Consequently, the total model memory remains on the order of a few megabytes. Including the storage required for the training and test datasets, the total memory usage of the XGB classifier for the largest dataset considered remained below 15 MB.

B. ACCURACY AND TRAINING TIME VS. TRAINING DATASET LENGTH

All models are trained and tested on the same numerically generated scattering dataset to ensure a fair comparison. The performance is assessed using classification accuracy and training time as functions of the training dataset size as follows.

Our entire dataset consists of scattered electromagnetic field data recorded at 16 uniformly distributed receivers around the scatterers. We reserve 10% of this data for testing. Hyperparameter tuning was performed using the remaining dataset. To assess the dataset size required for highly accurate classification, we increase the dataset size from 1000 to 10,000 in steps of 1000, and we record their training time and validation accuracy.

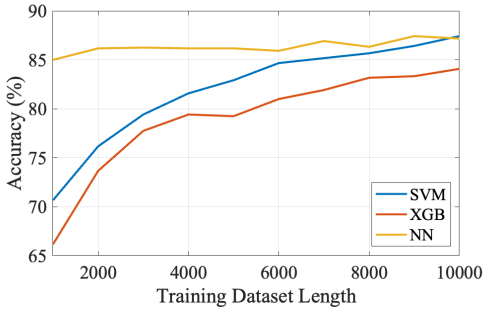

FIGURE 5. Validation accuracy vs. training dataset length.

The yellow curve in Fig. 5 shows that the CNN demonstrates stable classification performance across all dataset sizes, achieving high accuracy even in the small-data regime. For training sets containing only 1000 samples, the neural network already achieves an accuracy exceeding 85%, and this performance remains relatively consistent as the data set size increases. This behavior highlights the convolutional architecture's ability to learn physically meaningful features from limited scattering data, owing to its inductive bias toward local spatial correlations among receiver data. However, this robustness comes at the cost of increased training time, Fig. 6, which grows substantially with dataset size due to the iterative optimization of a large number of trainable parameters. Interestingly, the total training time exhibited a slight decrease at the largest dataset scales (9,000 and 10,000 samples). This can be attributed to the improved convergence stability provided by larger data batches and the early-stopping mechanism implemented, which terminated the training process in fewer epochs as the gradient updates became more representative of the underlying data distribution.

In contrast, classical machine learning methods (SVM and XGB) exhibit a stronger dependence on the availability of training data. Both classifiers show comparatively lower accuracy for small datasets, but improve steadily as the number of training samples increases. For large datasets containing $10,000$ samples, both SVM and XGB achieve classification accuracies that approach those of the neural network, differing by only a few percentage points. However, as shown in Fig. 6, these methods require significantly less training time than the neural network, even when accounting for hyperparameter tuning, making them attractive alternatives when large labeled datasets are available, and computational efficiency is a priority.

A key observation from these results is that model complexity should be matched to both the size and structure of the data. In the present study, the use of plane-wave excitation and a limited number of receiver antennae results in short, relatively simple input sequences. Under these conditions, the proposed one-dimensional CNN captures the essential scattering features without requiring more sophisticated mechanisms such as attention. Furthermore, when sufficient

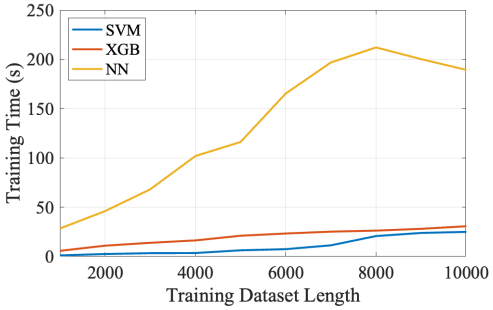

FIGURE 6. Time spent during training vs. training dataset length.

training data are available, classical methods such as SVM and XGB perform competitively with neural networks, suggesting that deep architectures are not strictly necessary in this regime. Overall, our numerical results demonstrate that while neural networks provide a clear advantage in data-limited scenarios, their benefit diminishes as the dataset size increases.

C. ATTENTION MECHANISMS AND SHORT SEQUENCES

Despite having short sequences, we also designed a hybrid convolutional–attention network as shown in Fig. 7. Initial feature extraction is performed using two consecutive 1D convolutional layers (128 filters, kernel size 3) with ReLU activation, each followed by batch normalization to stabilize training. Temporal resolution is reduced via max pooling, and dropout is applied for regularization. To capture long-range dependencies beyond local convolutional receptive fields, a multi-head self-attention block is introduced after the first convolutional stage. This block consists of layer normalization, four-head self-attention, and a residual connection, followed by a position-wise feed-forward network with an additional residual path. This structure enables adaptive weighting of sequence elements while preserving the original feature dimensionality. Following attention, a deeper convolutional block with increased channel capacity (256 filters) further refines learned representations. Global maximum pooling is then applied to aggregate temporal information into a fixed-length feature vector. The pooled features are processed by fully connected layers with ℓ_2 regularization and batch normalization. Channel-wise recalibration is achieved using a squeeze-and-excitation (SE) mechanism, which adaptively modulates feature importance. Final classification is performed using a softmax output layer. The model is trained using the AdamW optimizer with weight decay, sparse categorical cross-entropy loss, and accuracy as the evaluation metric. The training took approximately 18 minutes.

The overall accuracy of this hybrid convolutional–attention network is 87.8%, the highest among all the machine learning models implemented. Figure 8 is the confusion matrix demonstrating the strong and well-balanced classification performance across all ten object classes. The majority of samples lie along the main diagonal, indicating high

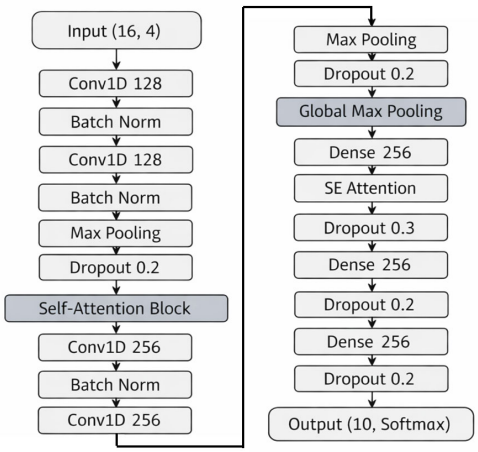

FIGURE 7. Architecture of the hybrid CNN–attention model for 1D sequence classification. Conv1D: one-dimensional convolution layer, Norm: normalization, SE: squeeze-and-excitation.

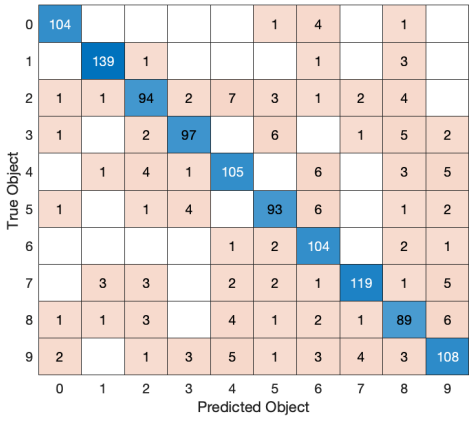

FIGURE 8. Confusion chart for the object classification performed with the hybrid CNN–attention model illustrated in Fig. 7.

per-class accuracy and effective discrimination among most scatterer types. Several classes (e.g., 0, 1, 4, 6, 7, and 9) exhibit particularly high true-positive rates with minimal confusion, suggesting that their learned feature representations are highly distinctive. Misclassifications are generally sparse and concentrated between a small number of closely related classes (e.g., classes 2–4, 3–5, and 8–9), which likely share similar scattering or temporal characteristics. Importantly, no systematic bias toward a particular incorrect class is observed, and off-diagonal errors remain low relative to the diagonal entries. This pattern indicates that the self-attention mechanism effectively captures global contextual dependencies, while the convolutional layers preserve local discriminative features.

Although the hybrid convolutional–attention network achieves a marginally higher overall accuracy (87.8%) compared to the simpler 1D CNN baseline (87.17%), the absolute improvement of 0.63 percentage points is modest. From a purely aggregate performance perspective, this gain may not, by itself, justify the increased architectural complexity,

TABLE 2. Classification accuracy under varying levels of random noise.

Noise Level	CNN (%)	Attention (%)	SVM (%)	XGB (%)
No Noise	87.17±1.1	87.80±1.0	87.42±1.2	84.08±0.9
10%	86.21±1.3	87.10±1.1	79.92±4.9	83.83±1.1
20%	83.85±2.9	85.92±1.8	78.92±5.0	81.92±1.9

computational cost, and training time associated with the attention-based model.

D. NOISE SENSITIVITY

To evaluate the robustness of the machine learning models in realistic sensing environments, we conducted a noise sensitivity study by injecting Gaussian random noise into the scattered field components. We tested two levels of degradation: 10% and 20% relative random noise. Table 2 summarizes the classification accuracies under these two noise levels. We observe that neural-network-based models exhibit superior robustness compared to classical methods. The hybrid CNN-attention model proved to be the most resilient, maintaining an accuracy of 85.92% even at a 20% noise level. The SVM classifier, while highly efficient in the "No Noise" regime, suffered the most significant performance degradation, dropping nearly 8.5 percentage points under high noise. This collapse occurs because SVMs define a rigid decision boundary in a high-dimensional kernel space; even small perturbations in the input field can shift a sample across the margin, leading to misclassification. XGB demonstrated remarkable stability, with performance remaining relatively flat between 10% and 20% noise. As an ensemble of decision trees, XGB's splitting process naturally tends to ignore features with high variance (noise) in favor of dominant scattering signatures, making it a reliable, low-cost alternative to NNs in moderately noisy environments. The 1D CNN maintains a strong balance, outperforming the classical methods in noisy conditions without the extreme training overhead of the attention model. We can conclude that for short-sequence electromagnetic data, the local feature extraction of convolutional layers provides sufficient "denoising" capability for most practical applications.

To assess the robustness of the trained classifier to variations in material contrast, we conducted a small additional test in which the relative permittivity of the targets exceeded the range used during training. Recall that the training dataset consisted of targets whose maximum relative permittivity did not exceed $\epsilon_r^{\max} = 4$. When the network was tested with targets having $\epsilon_r^{\max} = 5$, it correctly identified 7 out of 10 cases, indicating that the model can tolerate modest extrapolation beyond the permittivity range represented in the training data. However, when the permittivity contrast was increased significantly to $\epsilon_r^{\max} = 12$, the classification performance degraded substantially, with only one correct prediction among the ten test cases (corresponding to the target with the "I" shape). This behavior is expected because increasing the permittivity contrast alters the scattering amplitudes and phase distributions in a way that falls outside the statistical distribution learned during training. We can conclude that these machine learning models primarily interpolate within the permittivity range represented in the training dataset and that reliable generalization to higher-permittivity targets would require including a broader range of material contrasts during training.

V. CONCLUSION

In this work, we numerically investigated the classification of randomly oriented objects under plane-wave excitation. We demonstrated that a simple 1D CNN is sufficient to capture the dominant scattering features and achieves classification accuracy comparable to that of more complex attention-based architectures. By explicitly evaluating a hybrid convolutional–attention model, we showed that although self-attention can marginally improve overall accuracy, the absolute gain over the baseline CNN is modest (less than 1%). Hence, we conclude that attention mechanisms do not provide a decisive performance advantage for randomly oriented scatterers. Instead, they introduce additional architectural complexity and computational overhead. We further compared neural networks with classical machine learning methods, SVMs and XGB, across varying dataset sizes. Our results reveal a clear trade-off between model complexity, data availability, and computational cost. Neural networks—both convolutional and attention-based—exhibit superior performance in the small-data regime due to their ability to learn robust feature representations. However, as dataset size increases, SVM and XGB classifiers achieve accuracies comparable to neural networks while requiring substantially lower training and hyperparameter-optimization time. Furthermore, a noise sensitivity analysis reveals that while classical models suffer moderate performance degradation under realistic noise conditions, neural-network-based architectures demonstrate superior robustness, maintaining high accuracy even at 20% noise levels. We conclude that model selection for electromagnetic classification should be guided by the structure and length of the input data, the size of the available training set, and practical computational constraints. In particular, the use of attention mechanisms should be motivated by the presence of long-range correlations and complex inter-channel dependencies in the electromagnetic scattering data, rather than adopted by default. For short-sequence, limited scattering data, carefully designed convolutional architectures offer an efficient and effective solution with minimal loss in accuracy.

REFERENCES

[1] O. M. Bucci and T. Isernia, "Electromagnetic inverse scattering: Retrievable information and measurement strategies," *Radio Sci.*, vol. 32, no. 6, pp. 2123–2137, Nov. 1997, doi: 10.1029/97RS01826.

[2] Y. M. Wang and W. C. Chew, "An iterative solution of the two-dimensional electromagnetic inverse scattering problem," *Int. J. Imag. Syst. Technol.*, vol. 1, no. 1, pp. 100–108, Jun. 1989, doi: 10.1002/ima.1850010111.

[3] A. Abubakar, P. M. van den Berg, and S. Y. Semenov, "Two- and three-dimensional algorithms for microwave imaging and inverse scattering," *J. Electromagn. Waves Appl.*, vol. 17, no. 2, pp. 209–231, Jan. 2003, doi: 10.1163/156939303322235798.

[4] B. Wei, E. Şimşek, C. Yu, and Q. H. Liu, "Three-dimensional electromagnetic nonlinear inversion in layered media by a hybrid diagonal tensor approximation: Stabilized biconjugate gradient fast Fourier transform method," *Waves Random Complex Media*, vol. 17, no. 2, pp. 129–147, Apr. 2007, doi: 10.1080/17455030601016117.

[5] E. Simsek, "Determining optical constants of 2D materials with neural networks from multi-angle reflectometry data," *Mach. Learn., Sci. Technol.*, vol. 1, no. 1, Mar. 2020, Art. no. 01LT01, doi: 10.1088/2632-2153/ab6d5f.

[6] E. Simsek, "Machine learning exercises on 1-D electromagnetic inversion," *IEEE Trans. Antennas Propag.*, vol. 69, no. 10, pp. 6797–6805, Oct. 2021, doi: 10.1109/TAP.2021.3069519.

[7] Z. Wei and X. Chen, "Deep-learning schemes for full-wave nonlinear inverse scattering problems," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 1849–1860, Apr. 2019, doi: 10.1109/TGRS.2018.2869221.

[8] F. Cakoni and D. Colton, *A Qualitative Approach to Inverse Scattering Theory* (Applied Mathematical Sciences), vol. 188, 1st ed. New York, NY, USA: Springer, 2013, doi: 10.1007/978-1-4614-8827-9.

[9] E. Simsek and H. R. Manyam, "Classification with electromagnetic waves," *IET Microw., Antennas Propag.*, vol. 18, no. 12, pp. 898–910, Dec. 2024, doi: 10.1049/mia2.12522.

[10] A. Momeni and R. Fleury, "Electromagnetic wave-based extreme deep learning with nonlinear time-floquet entanglement," *Nature Commun.*, vol. 13, no. 1, p. 2651, May 2022, doi: 10.1038/s41467-022-30297-5.

[11] D. Covarrubias-Martínez, H. Lobato-Morales, J. M. Ramírez-Cortés, and G. A. Álvarez-Botero, "Classification of plastic materials using machine-learning algorithms and microwave resonant sensor," *J. Electromagn. Waves Appl.*, vol. 36, no. 12, pp. 1760–1775, Aug. 2022, doi: 10.1080/09205071.2022.2043192.

[12] T. M. Ting, N. S. Ahmad, and P. Goh, "Material classification via embedded RF antenna array and machine learning for intelligent mobile robots," *Alexandria Eng. J.*, vol. 106, pp. 60–70, Nov. 2024, doi: 10.1016/j.aej.2024.06.083.

[13] E. Simsek and H. R. Manyam, "Improving object classification accuracy from electromagnetic data using attention mechanisms," in *Proc. Int. Conf. Electromagn. Adv. Appl. (ICEAA)*, Sep. 2025, pp. 13–17, doi: 10.1109/ICEAA65662.2025.11305757.

[14] H. M. Yao, L. Jiang, and W. E. I. Sha, "Enhanced deep learning approach based on the deep convolutional encoder–decoder architecture for electromagnetic inverse scattering problems," *IEEE Antennas Wireless Propag. Lett.*, vol. 19, no. 7, pp. 1211–1215, Jul. 2020, doi: 10.1109/LAWP.2020.2995455.

[15] Y. Sanghvi, Y. Kalepu, and U. K. Khankhoje, "Embedding deep learning in inverse scattering problems," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 46–56, 2020, doi: 10.1109/TCI.2019.2915580.

[16] K. Xu, L. Wu, X. Ye, and X. Chen, "Deep learning-based inversion methods for solving inverse scattering problems with phaseless data," *IEEE Trans. Antennas Propag.*, vol. 68, no. 11, pp. 7457–7470, Nov. 2020, doi: 10.1109/TAP.2020.2998171.

[17] L.-Y. Xiao, J. Li, F. Han, W. Shao, and Q. H. Liu, "Dual-module NMM-IEM machine learning for fast electromagnetic inversion of inhomogeneous scatterers with high contrasts and large electrical dimensions," *IEEE Trans. Antennas Propag.*, vol. 68, no. 8, pp. 6245–6255, Aug. 2020, doi: 10.1109/TAP.2020.2990222.

[18] A. B. Tarokh, E. L. Miller, I. J. Won, and H. Huang, "Statistical classification of buried objects from spatially sampled time or frequency domain electromagnetic induction data," *Radio Sci.*, vol. 39, no. 4, pp. 1–11, Aug. 2004, doi: 10.1029/2003RS002951.

[19] Y. Zaky, N. Fortino, B. Miramond, and J.-Y. Dauvignac, "Shape and orientation classification of objects based on their electromagnetic signatures using convolutional neural networks," *Inverse Problems*, vol. 40, no. 4, Mar. 2024, Art. no. 045027, doi: 10.1088/1361-6420/ad2ec9.

[20] E. Simsek, J. Liu, and Q. H. Liu, "A spectral integral method and hybrid SIM/FEM for layered media," *IEEE Trans. Microw. Theory Techn.*, vol. 54, no. 11, pp. 3878–3884, Nov. 2006, doi: 10.1109/TMTT.2006.883647.

[21] E. Simsek, J. Liu, and Q. H. Liu, "A spectral integral method (SIM) for layered media," *IEEE Trans. Antennas Propag.*, vol. 54, no. 6, pp. 1742–11749, Jun. 2006, doi: 10.1109/TAP.2006.875500.

[22] Y. LeCun and C. Cortes. (2010). *MNIST Handwritten Digit Database*. [Online]. Available: http://yann.lecun.com/exdb/mnist/