IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 17 December 2023, accepted 24 December 2023, date of publication 15 January 2024, date of current version 6 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3354699

RESEARCH ARTICLE

A Framework for Robust Deep Learning Models Against Adversarial Attacks Based on a Protection Layer Approach

MOHAMMED NASSER AL-ANDOLI[1], SHING CHIANG TAN[2], KOK SWEE SIM[3], (Senior Member, IEEE), PEY YUN GOH[2], (Senior Member, IEEE), AND CHEE PENG LIM[4]

[1]Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, Durian Tunggal 76100, Malaysia
[2]Faculty of Information Science and Technology, Multimedia University, Melaka 75450, Malaysia
[3]Faculty of Engineering and Technology, Multimedia University, Melaka 75450, Malaysia
[4]Institute for Intelligent Systems Research and Innovation, Deakin University, Waurn Ponds, VIC 3216, Australia

Corresponding author: Shing Chiang Tan (sctan@mmu.edu.my)

This work was supported in part by the Fundamental Research Grant Scheme (FRGS) from the Malaysia Ministry of Higher Education under Project FRGS/1/2019/ICT02/MMU/02/2, and in part by Multimedia University's Grant under Project MMUI/220154.

ABSTRACT Deep learning (DL) has demonstrated remarkable achievements in various fields. Nevertheless, DL models encounter significant challenges in detecting and defending against adversarial samples (AEs). These AEs are meticulously crafted by adversaries, introducing imperceptible perturbations to clean data to deceive DL models. Consequently, AEs pose potential risks to DL applications. In this paper, we propose an effective framework for enhancing the robustness of DL models against adversarial attacks. The framework leverages convolutional neural networks (CNNs) for feature learning, Deep Neural Networks (DNNs) with softmax for classification, and a defense mechanism to identify and exclude AEs. Evasion attacks are employed to create AEs to evade and mislead the classifier by generating malicious samples during the test phase of DL models i.e., CNN and DNN, using the Fast Gradient Sign Method (FGSM), Basic Iterative Method (BIM), Projected Gradient Descent (PGD), and Square Attack (SA). A protection layer is developed as a detection mechanism placed before the DNN classifier to identify and exclude AEs. The detection mechanism incorporates a machine learning model, which includes one of the following: Fuzzy ARTMAP, Random Forest, K-Nearest Neighbors, XGBoost, or Gradient Boosting Machine. Extensive evaluations are conducted on the MNIST, CIFAR-10, SVHN, and Fashion-MNIST data sets to assess the effectiveness of the proposed framework. The experimental results indicate the framework's ability to effectively and accurately detect AEs generated by four popular attacking methods, highlighting the potential of our developed framework in enhancing its robustness against AEs.

INDEX TERMS Deep learning, adversarial examples, security, adversarial attacks, adversarial examples detection.

I. INTRODUCTION

Deep learning (DL) models have shown great success in a wide range of applications, such as image classification [1] and segmentation [2], malware detection [3], [4], object detection and tracking [5], fault detection [6], speech recognition [7], and complex network analysis [8], [9]. The high accuracy performance of DL models is attributed to their continuous development, availability of data, and increase in computational power. However, the use of DL models in these tasks has been challenged by adversaries that aim to compromise the model's accuracy [10]. Recent studies have shown that DL models are vulnerable to adversarial attacks, where an attack can manipulate input data in a way that misleads the model's output [11]. Adversarial attacks have become a major concern in the field of DL, as they

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

can compromise the security and reliability of these models, especially in safety-critical applications such as medical diagnosis or self-driving cars [11], [12]. These attacks fall into two broad categories [13]: poisoning and evasion attacks. Poisoning attacks aim to contaminate the training data, while evasion attacks aim to generate adversarial examples (AEs) by carefully crafting small perturbations to input data during the model's inference phase to mislead its predictions.

The widespread adoption of DL for its impressive performance and complex system modeling has drawn attention to its vulnerability to misleading inputs. To address this, several adversarial attack detection methods have emerged, utilizing AE detection methods to identify potential attacks before they occur [11], [16], [17], [18], [19], [20] For example, support vector machines (SVMs) with radial basis function kernels were employed to identify AEs. Feature squeezing [18] reduces the search space for adversarial detection by merging feature vectors, thereby making Deep Neural Network (DNN) model more robust and resistant to adversarial attacks. These methods have demonstrated effectiveness in detecting AEs; however, they are limited to specific attack types and can be vulnerable to carefully crafted perturbations. Uncertainty-based methods have also emerged as a promising approach to adversarial attack detection. Bayesian uncertainty in dropout neural networks [26], DNNs with kernel density and deep gaussian processes [19], minimum uncertainty metrics [20], and selective and feature-based adversarial detection [12] are notable examples of uncertainty-based methods. These methods rely on the assumption that AEs exhibit a higher degree of uncertainty as compared to normal data. While these methods have shown promising results, they still face challenges such as inaccuracies, high computational overhead, and difficulties in detecting AEs that blend seamlessly with normal data.

Other AE detection methods include steganalysis-based techniques [28], convolutional neural network (CNN)-based approaches with feature regularization, histogram-based feature creation, and residual image reinforcement [29], a data augmentation technique-strengthened binary detector network [30], the feature autoencoder detector framework leveraging feature knowledge [31], and a softmax distribution-based method for detecting misclassified and out-of-distribution examples [32]. However, existing detectors lack robustness, efficiency, and effectiveness when facing new or unknown attacks [11], [15]. Additionally, defenses and detectors remain susceptible to meticulously crafted adversarial perturbations, which render them ineffective as robust detectors [11]. For practical security situations, it is crucial for detectors to have strong generalization and robustness capabilities. This paper addresses these challenges by introducing an effective and robust DL framework, incorporating a defense layer into the DL models. This integrated approach aims to detect adversarial examples and exclude them, thereby preventing misleading effects on DL models.

In this paper, we aim to develop an effective and practical DL framework for improving the robustness of learning models against adversarial attacks. The developed framework employs a CNN for feature learning, a defense mechanism to identify and exclude AEs, and a DNN with deep fully connected layers and softmax for the classification task. AEs are created by the adversaries using the Fast Gradient Sign Method (FGSM) [21], Basic Iterative Method (BIM) [22], Projected Gradient Descent (PGD) [23], and Square Attack (SA) [24] to evade the system through the adjustment of malicious samples during the test phase of DL models i.e., CNN and DNN. The proposed framework consists of a protection layer before the DNN classifier detects and excludes AEs. Specifically, the protection layer in the framework uses several machine learning (ML) models as a detection/defense model to identify AEs in the input data. These ML models include Fuzzy ARTMAP (FAMD), Artificial Neural Network (ANND), Random Forest (RFD), K-Nearest Neighbors (KNND), XGBoost (XGBD), and Gradient Boosting Machine (GBMD). The developed framework exhibits potential for application in a wide range of contexts, utilizing the defender within the protection layer approach. This approach offers several advantages, including efficiency, scalability (being applicable to multiple attacks, data sets, and models), effectiveness (capable of detecting various adversarial attacks), and robustness (by ensuring that the proposed framework can withstand attacks and maintain high detection accuracy, even when the input data are intentionally manipulated or perturbed).

The main contributions of this paper can be summarized as follows:

- Devise a DL-based framework utilizing a Protection Layer approach to detect AEs. The CNN is used for feature learning, while the DNN is employed for classification. Evasion attacks are employed to create AEs to evade and mislead the classifier by adjusting malicious samples during the test phase using FGSM, BIM, PGD, and SA attacks.
- Employ Fuzzy ARTMAP and five ML models (i.e., RFD, ANND, KNND, XGBD, and GBMD) in the protection layer as a detection mechanism to identify and exclude AEs.
- Evaluate the proposed framework using popular data sets (MNIST, CIFAR-10, SVHN, and Fashion-MNIST) and demonstrate that it can effectively detect a wide range of adversarial attacks.

The remaining sections of the paper are organized as follows: Section II presents the related work. The methodology is described in Section III. Results and discussions are presented in Section IV. Conclusions and suggestions for further work are presented in Section V.

II. RELATED WORK

Detection-based defense emerges as a prevalent approach to distinguishing between normal and AEs [16]. A method called Deep Neural Rejection (DNR) [25] was developed for detecting AEs. Specifically, DNR utilized SVMs with radial basis function (RBF) kernels to reject samples that exhibited

feature representations indicative of AEs in various network layers. The classification process of DNR involved rejecting samples if the maximum score was below a predefined rejection threshold θ. In [26], the Local Intrinsic Dimensionality (LID) technique was employed for AEs detection. LID calculated the distance distribution of an input sample to its neighbours to evaluate the region's space-filling capability surrounding that sample. In [17], an Adaptive Noise Reduction (ANR) method was developed for detecting adversarial images. It considered the perturbations in images as a form of noise, and introduced two classic image processing techniques, namely scalar quantization and smoothing spatial filters, to mitigate their impact. The image entropy was used as a metric to implement ANR for different images. AEs were detected by comparing classification result of a given sample with its denoised version. Similarly, a method called Feature Squeezing (FS) [18] was proposed to enhance the robustness of a DNN model in detecting AEs. FS reduced the search space available to an adversary by merging samples corresponding to diverse feature vectors in the original space into a single sample.

In [26], defenders utilized model confidence on AEs by leveraging Bayesian uncertainty estimates, which were available in dropout neural networks, to distinguish AEs from their normal and noisy counterparts. The uncertainty values were employed as features to construct a binary classifier for detection purposes. Typically, uncertainty was measured by introducing randomness to the model using a dropout technique. Recently, a new ensemble AE detector named Selective and Feature-based Adversarial Detection (SFAD) was developed in [12] based on the model's uncertainty and confidence. SFAD was an unsupervised detector that could detect AEs without having prior knowledge of AEs. In [27], the Attack on Frequency (AOF) method was introduced to enhance the transferability of 3D point cloud attacks. By focusing on the low-frequency component, it improved the transferability of adversarial samples for 3D point clouds, as well as enhanced the robustness of 3D point cloud classifiers.

Liu et al. [28] proposed the TIKI-TAKA framework to assess the robustness of DL-based Network Intrusion Detection Systems (NIDS) against adversarial manipulations. The framework incorporated three defense mechanisms: model voting ensembling, ensembling adversarial training, and query detection, which were aimed at increasing resistance to attacks employing evasion techniques. In [19], two features were introduced in DNNs for detecting AEs. Firstly, the DNNs performed kernel density estimation in the input space of the last layer. Secondly, uncertainty estimation was carried out through a deep Gaussian process of dropout DNNs to detect AEs that were in low-confidence regions of the input. Sheikholeslami et al. [20] introduced a randomized method to detect perturbations by employing the Minimum Uncertainty Metrics (MUM). The method involved sampling hidden nodes randomly layer by layer in a pre-trained DNN model during its inference stage to compute the overall uncertainty of network output by using Bayesian approach. Sampling probabilities were updated by minimizing uncertainty measures layer by layer to identify AEs. MUM detected AEs based on an idea that the distance of adversarial perturbation from the manifold of natural-data manifold could lead the DNN to estimate a high overall network uncertainty that exceeded that of clean data. These studies [19], [20], [28], underscore the potential of various approaches for detecting AEs, including DL and novel techniques like MUM. As NIDS continue to evolve, incorporating such diverse defenses will be crucial in ensuring robust and reliable network security.

A method based on steganalysis (i.e., technology for detecting steganography) was developed to detect AEs [29]. Steganalysis features were enhanced by estimating probability of modifications caused by adversarial attacks. In [30], three methods were proposed based on CNNs to detect possible AEs. The first method involved regularizing the feature vector, the second method utilized histograms to create a feature vector, which was then used as the input to an SVM classifier; and the third method was based on the residual image, where the residual image was used to reinforce certain parts of the input pattern for AEs detection. The best aspects of the three methods were combined to develop a more robust approach to detecting AEs. In [31], a binary detector network was developed based on intermediate feature representations to distinguish AEs from the original data. Dynamic Adversary Training (DAT) was used to strengthen the detector by training the classifier with AEs. In [32], a Feature Autoencoder Detector (FAD) defense framework was developed for detecting AEs. FAD leveraged feature knowledge in the detection process. In [33], a method based on the utilization of probabilities from softmax distributions was developed to detect misclassified and out-of-distribution examples. It was observed that accurately classified examples exhibited a higher maximum softmax probability in comparison with misclassified examples and those that fall outside the distribution. Goswami et al. [34] developed a framework to assess the robustness of face recognition engines based on DL. This framework was capable of detecting and mitigating adversarial attacks. The detection and mitigation processes were conducted in a scenario that mimicked real-world conditions, involving cross-database and cross-attack scenarios.

Despite the significant progress made in the field of AEs detection, there still exist research gaps that need to be addressed. Most of the existing detectors lack robustness when facing new or unknown attacks. Additionally, defenses and detectors remain susceptible to meticulously crafted adversarial perturbations, which render them ineffective as robust detectors [11]. Therefore, to address these limitations and develop more resilient defenses and detectors in the field of AEs detection, this paper presents a robust framework with DL models based on a protection layer as a detection mechanism to identify and exclude AEs.

III. METHODOLOGY

In this section, we explain our proposed framework, denoted as AEs Detection-based Protection Layer in DL models (AEDPL-DL), which is illustrated in Fig. 2. In this study, the term "DNN classifier" refers to a fully connected DNN with softmax for classification. The data samples are initially divided into training and test sets. The default CNN and a DNN classifier in AEDPL-DL are trained using Clean Examples (CEs) (Fig. 2 (a)). During the test phase, the testing samples undergo attacks from adversary attackers ((Fig. 2 (b)). AEDPL-DL utilizes a CNN model designed for feature learning, and is equipped with defense mechanisms against several types of adversarial attacks, such as FGSM [21], BIM [22], PGD [23], and SA [24]. The AEDPL-DL framework includes a protection layer inserted before the classifier to detect AEs. The protection layer uses FAMD and other ML models (RFD, KNND, XGBD, and GBMD) to identify AEs.

In AEDPL-DL, the default CNN and DNN classifier are trained using CEs. The CNN is explained in Section III-B, while the DNN classifier is introduced in Section III-D. During the test phase, the test samples are created by adversary attackers to produce AEs. AEDPL-DL is designed to primarily focus on detecting AEs in the test phase. The protection layer in AEDPL-DL is designed to identify AEs generated by adversarial attack algorithms and exclude them from proceeding to the DNN classifier (Section III-C). The key components of the AEPL-DL framework are explained in detail in Sections III-A-D.

A. GENERATING AEs

AEs are data samples (e.g. images) that are specifically designed to fool a DL model into misclassifying them [11]. They are generated by adding small, imperceptible perturbations to the original data that can cause the model to make incorrect predictions [11], [15].

Various adversarial attack techniques have been introduced to generate AEs. These attack methods are designed by either a black-box or white-box approach. The black-box methods use surrogate models to generate AEs, where attacks lack knowledge about both the target and defense models. Conversely, attacks in the white-box methods are made by referring to the knowledge of both the target and defense models [11], [13], [15].

In this work, we generate AEs of images using both white-box and black-box attack methods. We select four attacks, including FGSM, BIM, and PGD for white-box attacks, and SA for black-box attacks, in order to evaluate the robustness of the developed AEDPL-DL framework. The attack methods are as follows:

- FGSM [21] attack is a white-box attack that adds a small perturbation to each pixel in the input image in the direction of the gradient of the loss function. It utilizes a one-step gradient update algorithm to determine the perturbation direction, specifically the sign of the gradient ∇ for each pixel of the input image x to maximize the loss value l of the targeted DL model. It is expressed as follows:

$$x' = x + \epsilon \; sign\left(\nabla_x l\left(x, y\right)\right), \quad x' \in [0, 1] \tag{1}$$

  where ϵ is a parameter controlling the perturbation of the attack; x, x', and y indicate clean input, AEs of x, and clean input label, respectively.
- BIM [22] is the iterative version of FGSM that adds multiple small perturbations to the input image over several iterations. It applies FGSM attack k times, which is defined as follows:

$$x'_{i+1} = x'_i + \alpha sign\left(\nabla_x l\left(x, y\right)\right), \quad x'_0 = x, x'_{i+1} = \in [0, 1], i = 0 \; to \; k \tag{2}$$

  where α is the parameter to control the iteration of step size and it is between 0 and ϵ, and k is the FGSM times.
- PGD [23] is an iterative white-box attack similar to that of BIM, but with random noise added to the perturbation in each iteration. The perturbation is initialized with $\left\|x' - x\right\| < \epsilon$.
- SA [24] is a black-box attack that utilizes a quadratic model to determine the optimal perturbation for the input image. SA randomly selects ϵ-bounded localized squares at different positions to generate perturbations that satisfy the optimization problem. This process can be achieved using a random search strategy at each iteration of the algorithm. SA is expressed as follows:

$$\min_{x^1 \epsilon [0,1]} l\left(f\left(x'\right), y\right), \quad \|\delta\| \leq \epsilon, l\left(f\left(x'\right), y\right) = f_y\left(x'\right) - max_{k \neq y} f_c\left(x'\right). f_y\left(x'\right) \tag{3}$$

  where $f_c\left(x'\right)$ and $f_y\left(x'\right)$ refer to the prediction probability of x' for c and y classes, respectively.
- DeepFool (DF) [35] is a simple and accurate method to fool deep neural networks. It is based on an iterative linearization of the classifier to generate minimal perturbations that are sufficient to change classification labels.

B. FEATURE LEARNING-BASED CNN

The CNN is often utilized for image and video recognition tasks. It plays a crucial role in feature extraction from images, which aids a classifier in object recognition and classification. In this research, the CNN is employed to extract features from clear images and AEs. The feature extraction process takes place in two main layers of the CNN, namely convolutional and pooling layers.

The convolutional layer is responsible for performing convolution operation, which applies a set of learnable filters (also known as kernels or weights) on the input image or feature map. The output from the convolution operation is a new feature map that represents the presence of various features in the input image. The convolution operation is computed using the following equation:

$$Y\left(i, j\right) = \sum_{k=1}^{K} \sum_{l=1}^{L} X\left(i + k, j + l\right).W\left(k, l\right) + b \tag{4}$$

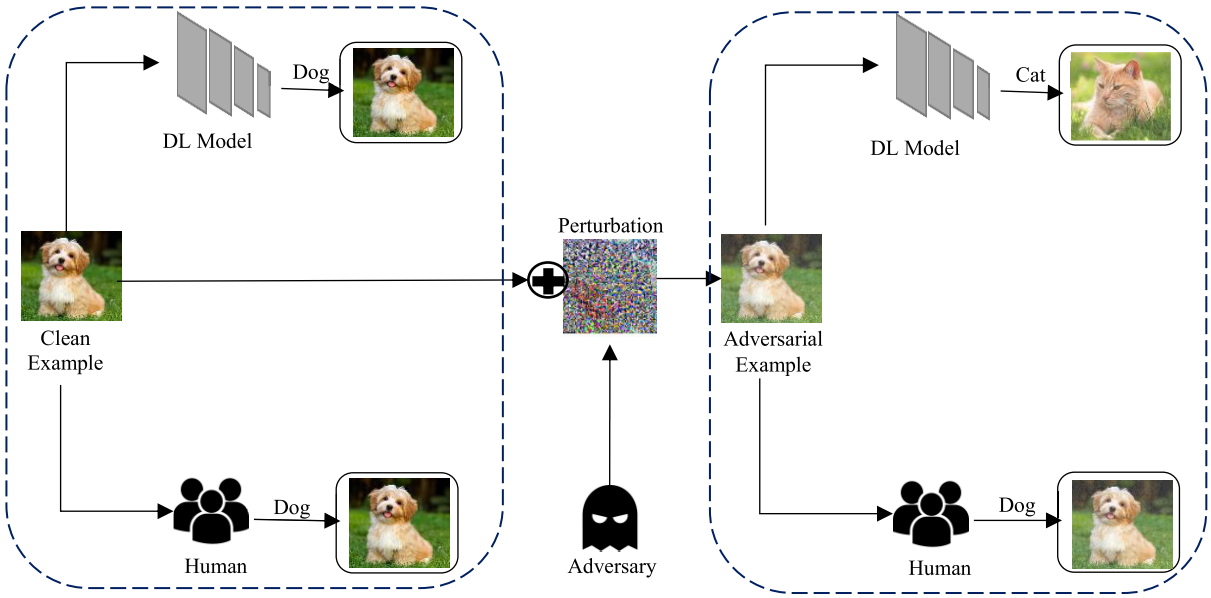

FIGURE 1. The objective of AEs is to mislead and fool DL models.

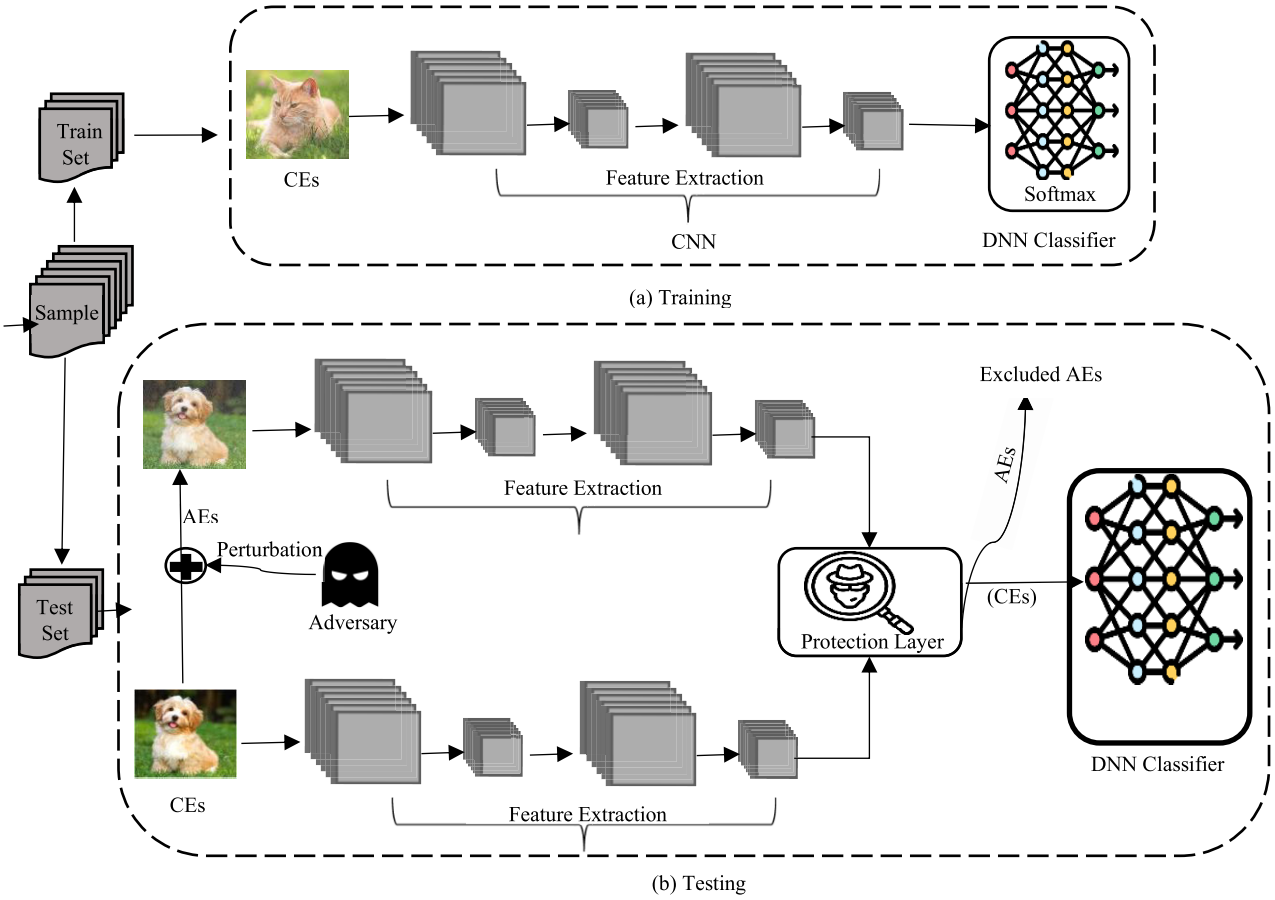

FIGURE 2. Outline of the proposed AEDPL-DL.

where X is the input image or feature map; Y is the output feature map; W is the set of learnable filters or kernels; b is the bias term; and, K and L are the dimension of the filters

Following the convolutional layers, the pooling layers in the CNN use a downsampling operation to decrease the spatial resolution of the feature maps. The most common

downsampling technique is the max-pooling operation, which identifies the maximum value in each local region of the input feature map. This operation is expressed mathematically as follows:

$$Y(i,j) = \max_{k,l} X(s_k + i, s_l + j) \tag{5}$$

where X is the input feature map; Y is the output feature map; and, $s$is the size of the pooling region.

After extracting features from an image, a flattening process is performed to re-arrange the pooled feature map into a single column. This creates a vector as the input for subsequent processing stage. In this work, the flattened data are forwarded to the protection layer, which is responsible for filtering out the AEs and allowing only CEs to proceed to the classification phase.

C. PROTECTION LAYER FOR AEs DETECTION

In AEDPL-DL, we devise a protection layer to incorporate a defender model designed to identify AEs generated by adversarial attack algorithms and exclude them from proceeding to the final classification phase. The protection layer is included in the test phase of AEDPL-DL, and is posited between the CNN and DNN classifier. The defender recognizes that AEs possess distinct features that are different from clean inputs [36]. Therefore, a defender leverages this advantage to construct a robust detector, enabling the identification and exclusion of AEs.

The protection layer functions as an intermediary between the CNN and DNN classifier. Its primary objective is to enhance the DL models, e.g. the capability of CNN and DNN classifier to detect AEs and prevent their influence on the final classification decision. By integrating the protection layer, only CEs are allowed to transfer to the final classification decision for further processing. To achieve this, the protection layer employs various defender models, including FAMD, and different ML models such as ANND, RFD, KNND, XGBD, and GBMD. The protection layer consists of one of these ML models at a time, which is the best one with high performance. The AEDPL-DL framework assesses the ML detectors and chooses the model yielding superior AE detection capabilities.

An ANND detector is a classifier that uses interconnected layers of artificial neurons to classify the input data into different categories, including AEs and CEs. It applies a rectified linear unit (ReLU) function to the weighted sum of the input X, enabling it to learn complex patterns. The classifier output is determined by adjusting weights W and biases b in $f(Wx + b)$ during training, in order to minimize the difference between its predictions and the target label, Y.

The RFD detector is an ensemble learning model that combines multiple decision trees to make accurate prediction or classification outcomes. Each tree in the forest is constructed using a random subset of the data and a random subset of the features. The final prediction is obtained by aggregating the predictions of individual trees.

The KNND detector is an ML algorithm used for classification and regression tasks. It determines the class or value of a data sample by considering its K nearest neighbors in the training data set. It relies on a distance metric to reflect the label similarity, e.g. Minkowski distance, which is computed as follows:

$$D = \left[\sum_{i=1}^{K} |x_i - x_i'|^p\right]^p \tag{6}$$

where p represents the order of the norm

GBMD and XGBD are used also as detectors in the protection layer. They are gradient boosting algorithms that iteratively combine several weak learners to create a strong predictive model. XGBD adds further optimization, such as regularization and gradient tree boosting, to enhance performance and control model complexity.

FAMD is a neural network that combines fuzzy logic and the adaptive resonance theory (ART) model for pattern recognition and classification tasks. It dynamically creates and updates recognition categories based on the input patterns using the fuzzy ART principles for mapping input samples to output classes, performing incremental learning in dynamic environments.

ANND, RFD, KNND, GBMD, XGBD, and FAMD models enable the protection layer to learn and identify patterns indicative of AEs. By leveraging the capabilities of these models, the protection layer can effectively differentiate between clean and adversarial inputs based on specific characteristics or features present in the data. These ML models are used individually rather than in combination.

By successfully identifying and excluding AEs, the protection layer ensures that the subsequent classification process operates on reliable and trustworthy data. This significantly improves the overall robustness and accuracy of AEDPL-DL, making it resilient against AEs.

D. DNN-SOFTMAX CLASSIFIER

In this phase, the data samples received from the preceding layer are processed for a classification task using a fully connected DNN with a softmax activation function, $f(z)_i = \frac{e^{z_i}}{\sum_j^C e^{z_j}}$. The objective is to combine the features into additional attributes, enhancing the predictive capabilities for data classification. The softmax function is a generalized version of the logistic function. It ensures that the predicted class probabilities p_i sum up to 1, which is closely associated with the cross-entropy function. Applying the softmax function helps evaluate the model reliability by utilizing the Cross Entropy Function as a loss function. The aim is to optimize the performance of the DNN by minimizing the cross-entropy loss (L_{CE}) as follows:

$$L_{CE} = -\log\left(\frac{e^{s_p}}{\sum_j^C e^{s_j}}\right)$$

$$= -\sum_{i}^{C} t_i \log\left(f\left(z\right)_i\right), f\left(z\right)_i$$
$$= \frac{e^{z_i}}{\sum_{j}^{C} e^{z_j}} \tag{7}$$

where t_i is the truth label, C is the number of classes, z is the output from DNN, and $f\left(z\right)_i$ is the softmax probability for the i^{th} class. Algorithm 1 depicts the steps of the developed AEDPL-DL framework.

E. TIME COMPLEXITY ANALYSIS

This section provides an overview of the time complexity analysis for AEDPL-DL, encompassing three distinct phases: feature extraction, protection layer, and DNN classifier.

The feature extraction phase involves two main tasks: convolution and max pooling. For a filter of size k, the dot product's cost is $O\left(k.d^2\right)$, where d denotes the dimension. Since the filter is applied over the input $n-k+1$ times, where n represents the length of the input or the number of input nodes, the final time complexity of a convolutional layer is $O(k.n.d^2)$. Meanwhile, the time complexity of a max pooling layer is $O(n)$. Consequently, the maximum time complexity in this phase is $O(k.n.d^2)$.

In the protection layer, various algorithms, including ANND, RFD, KNND, XGBD, GBMD, or FAMD, are employed to detect Adverse Events (AEs). The time complexity of ANND is $O(n.d)$, RFD is $O(n^2.d.t)$, KNND is $O(n.d)$, XGBD is $O(n.d.t)$, GBMD is $O(n.d.t)$, and FAMD is $O(n.d.c)$, where n is the number of data points, d is the number of features, t is the number of trees, and c is the number of codebooks in FAM. The maximum time complexity in this phase is $(n^2.d.t)$.

Moving to the DNN classifier, it requires a deep fully connected layer, where the input row vector undergoes multiplication with the weight matrix. Consequently, its computational cost is $O(n.d.l)$, where n is the number of input features, l is the number of layers, and d is the number of output features. In summary, the comprehensive time complexity of AEDPL-DL is $O(\left(k.n.d^2\right)\left(n^2.d.t\right)\left(n.d.l\right))$.

IV. IMPLEMENTATION AND RESULTS

This section presents a set of experiments to validate the performance of AEDPL-DL. The obtained results are analyzed and discussed in detail.

A. DATASETS

MNIST [37] is a data set for recognizing handwritten digits ranging from 0 to 9. It consists of 70,000 grayscale images/samples, with 60,000 for training and 10,000 for test. CIFAR-10 [38] is an image data set commonly used in computer vision applications. The images are in the RGB format, and have a resolution of 32 × 32 pixels. The data samples cover images from ten different classes: airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks. A total of 60,000 images are available, with 50,000 used for training and 10,000 for test.

The SVHN dataset [39], derived from Google Street View images of house numbers, includes 600,000 32 × 32-pixel images. These are initially of varying dimensions, but are cropped to match bounding boxes of individual digits and converted to grayscale. There are 531,131 extra digits, with 73,257 for training and 26,032 for testing.

Fashion-MNIST dataset [40], introduced by Zalando, originates from Europe's largest online fashion platform. It includes 70,000 products, each is represented as a 28 × 28 pixel grayscale image, and is categorized into 10 distinct classes.

B. EXPERIMENTAL SETTING

Two machines have been used to analyze and verify the distributed computing and parallel implementation of AEDPL-DL. The processors are Intel Core-i7, and the operating system is Ubuntu 20.04. One machine has 8 GB RAM and the other has 4 GB. The Python Ray Library [41] is utilized to implement parallelization.

In AEDPL-DL,[1] the CNN and DNN classifier models were trained with 100 iterations, a batch size of 256, and a learning rate of 0.001. The data set was split into a 4:1 training-test ratio, and the experiment was repeated for ten times to compute the average results, each time randomly shifting the data between the training and testing sets. In AEDPL-DL, both CNN and DNN models were trained using CEs. During the test phase, the test samples were subject to attacks from adversary attackers to generate AEs. As a result, the test samples contained CEs, AEs, and both clean and adversarial samples (CEs_AEs). As an example, the number of CEs from MNIST was 10,000. These CEs were subject to attacks, producing 10,000 AEs. A total of 20,000 mixture samples (i.e., CEs_AEs) were available. The same process was applied to the CIFAR-10, SVHN and MNIST_Fashion data sets. CIFAR-10 resulted in 10,000 CEs, 10,000 AEs, and 20,000 CEs_AEs samples, SVHN resulted 26,032 CEs, 26,032 AEs, and 52,064 CEs_AEs samples, while MNIST_Fashion resulted 10,000 CEs, 10,000 AEs, and 20,000 CEs_AEs samples, respectively.

Four metrics are adopted for performance evaluation and comparison, i.e., accuracy, precision, recall, and F1-score. These metrics are computed based on True Positive Rate (*TPR*), False Positive Rate (*FPR*), True Negative Rate (*TNR*), and False Negative Rate (*FNR*), as follows:

$$Accuracy = \frac{TPR + TNR}{TPR + TNR + FPR + FNR} \tag{8}$$

$$Precision = \frac{TPR}{TPR + FPR} \tag{9}$$

$$Recall = \frac{TPR}{TPR + FNR} \tag{10}$$

[1] https://github.com/MNAl-Andoli/AEDPL-DL/tree/main

Algorithm 1 AEDPL-DL

Input: Data samples, i.e., clean examples.
Output: Detection of each AEs and DL classification for CEs.
Divide the data samples into training and testing sets
a) Training process
    Train CNN and DNN classifier on training sets (i.e., clean inputs, Fig. 2(a))
b) Performing adversarial attacks on testing set (CEs): (Section III-A)
    for $i = 1$ to $size(CEs)$:
        Generate AEs using one of {FGSM, BIM, PGD, SA, and DF} attacks.
    end for
c) Testing Phase (Fig. 2(b))
    - Feature Extraction: (Section III-B)
        Foreach (i in AEs and CEs): do
            Perform convolution operation
            Conduct max-pooling
        end for
        Combine features extracted from AEs and CEs, and forward them to the protection layer
    - Protection Layer
        Select a detector from {ANND, RFD, KNND, XGBD, GBMD, or FAMD}. (Section III-C)
        Train the detector to work as a classifier for AEs and CEs
        Detect unseen AEs and exclude them
        Allow CEs to pass through the protection layer.
    - DNN classifier
        Design DNN classifier with softmax (Section III-D).
        Define and calculate loss function
        Train the DNN classifier with CEs
        Predict the labels
Return the excluded AEs and the predicted labels of CEs.

$$F1 - score = 2.\frac{Recall \times Precision}{Recall + Precision}. \tag{11}$$

The area under the receiver operating characteristic curve (AUC) [42], which is constructed using FPR and TPR, is also computed.

C. RESULTS AND DISCUSSION

In this section, the results of AEDPL-DL are analyzed and discussed using four benchmark data sets, namely MNIST, CIFAR-10, SVHN, and Fashion-MNIST.

1) MNIST

a: AEDPL-DL PERFORMANCE

AEDPL-DL has been evaluated in detecting AEs generated using four attack methods: FGSM, BIM, PGD, and SA. In the test phase, three versions of testing samples are used: CEs, AEs, and CEs_AEs. Notably, in AEDPL-DL, the CEs_AEs samples are utilized with a protection layer-based defense mechanism. This mechanism aims to identify AEs and prevent them from passing to the final classification phase. In other words, the samples of CEs and AEs are passed to AEDPL-DL. The features of CEs and AEs are then extracted by the CNN and sent to a protection layer-based defense mechanism. Next, the CEs and AEs are mixed and split into a 4:1 training-test ratio. After training the ML model in the protection layer, it is evaluated using the test set. In this set, only CEs are passed to the DNN classifier to perform the final classification. Based on this setup, we conduct two performance evaluations. First, we evaluate the overall performance of AEDPL-DL in Section IV-C.1.A, which is related to the final classification with the DNN classifier. Secondly, we evaluate the performance of the detector in Section IV-C.1.B, which assesses the effectiveness of the protection layer. This section also includes analyzing the ratio of CEs being blocked and the ratio of CEs passing to the DNN classifier

Table 1 summarizes the AEDPL-DL results with different attacks on CEs, indicating high accuracy, precision, recall, and F1-score. However, when dealing with AEs, the AEDPL-DL performance drops significantly, revealing its vulnerability to adversarial attacks. The AEDPL-DL performance with CEs_AEs shows a better performance, as compared with that from using AEs alone, but is lower than that of using CEs alone. On the other hand, the AEDPL-DL framework, when integrated with an auxiliary ML model within the protection layer (that works as a detector), exhibits a notable improvement in performance. Several ML models (ANND, RFD, KNND, XGBD, GBMD, and FAMD) are employed as detectors in the defense mechanism, resulting in enhanced performance against attacks.

Table 1(a) presents the evaluation of AEDPL-DL under the FGSM attack. To clarify the meaning of each term used and provide a better understanding of the AEDPL-DL evaluation

TABLE 1. The performance of AEDPL-DL using various detecting models (ANND, RFD, KNND, XGBD, GBMD, FAMD) in the protection layer against four attacks on the MNIST data set (CEs, AES, and CEs_AEs) with eps ϵ= 0.30.

Attacks	Measures	CEs	AEs	CEs_AEs						
		-	-	-	ANND	RFD	KNND	XGBD	GBMD	FAMD
(a) FGSM	Accuracy	99.25	36.04	67.6	97.34	98.62	97.56	96.31	94.24	87.95
	Precision	99.25	42.43	70.76	97.4	98.63	97.58	96.46	94.61	88.84
	Recall	99.25	36.04	67.6	97.34	98.62	97.56	96.31	94.24	87.95
	F1-score	99.25	36.28	68.15	97.35	98.62	97.56	96.34	94.29	88.1
(b) BIM	Accuracy	99.25	1.00	50.02	97.76	97.61	92.23	89.58	79.63	74.95
	Precision	99.25	1.40	56.67	97.86	97.62	92.41	89.82	80.4	76.93
	Recall	99.25	1.00	50.02	97.76	97.61	92.23	89.58	79.63	74.95
	F1-score	99.25	1.04	51.56	97.78	97.61	92.27	89.6	79.68	75.47
(c) PGD	Accuracy	99.25	0.61	49.9	97.91	96.74	92.48	87.15	75.58	73.88
	Precision	99.25	0.64	52.83	97.91	96.77	92.49	87.15	78.14	74.68
	Recall	99.25	0.61	49.9	97.91	96.74	92.48	87.15	75.58	73.88
	F1-score	99.25	0.6	50.6	97.91	96.75	92.48	87.12	75.83	74.04
(d) SA	Accuracy	99.25	53.3	76.26	98.84	99.06	98.08	97.96	95.62	86.93
	Precision	99.25	59.73	79.17	98.36	99.06	98.1	97.97	95.73	89.02
	Recall	99.25	53.3	76.26	98.34	99.06	98.08	97.96	95.62	86.93
	F1-score	99.25	53.65	76.72	98.34	99.06	98.08	97.96	95.63	87.32

under different conditions and defense mechanisms, the following explanation is provided:

- CEs: AEDPL-DL performance when CEs data are used.
- AEs: AEDPL-DL performance when AEs data are used without a protection layer.
- ANND: AEDPL-DL performance when both CEs and AEs are checked by an ANN model as a detector in the protection layer. Similar explanations are provided for other detection/defense models, e.g. RFD, KNND, XGBD, GBMD, or FAMD.
- (-): AEDPL-DL performance without protection layer and detectors.

The results from Table 1(a) indicate significantly higher accuracy, precision, recall, and F1-score for AEDPL-DL as compared with those without a protection layer against AEs. It is evident that AEDPL-DL incorporating a protection layer based on RFD, ANND, and KNND achieve accuracy levels comparable with those of CEs, with 98.62%, 97.34%, and 97.56%, respectively. This stands in contrast to the accuracy values of 36.04% and 67.6% for AEs and CEs_AEs, respectively. Note that the protection layer incorporating FAMD yields a slightly lower performance in detecting AEs, as compared with those from other ML models.

Table 1(b) shows the performance of AEDPL-DL under the BIM attack. Comparing with the performance of the default scenario without a protection layer using CEs_AEs, AEDPL-DL encounters a drastic drop in performance with AEs. These results imply that the DL models, i.e., the CNN and DNN classifier, are susceptible to the BIM attack. However, when AEDPL-DL incorporates auxiliary ML models in the protection layer, the performance improves significantly, as shown in Table 1(b). ANND, RFD, and KNND demonstrate higher scores in all metrics. For example, ANND achieves an accuracy rate of 97.76%, while RFD and KNND achieve accuracy rates of 97.61% and 92.23% respectively. On the other hand, XGBD and GBMD show slightly lower results, with accuracy values of 89.58% and 79.63% respectively. FAMD, similar to the findings with FGSM attacks, exhibits a relatively low performance as compared with those from other ML models, achieving an accuracy score of 74.95%.

Using the PGD attack (Table 1(c)), the performance of AEDPL-DL without the protection layer using CEs achieves accuracy, precision, recall, and F1-score of 99.25. However, the performance significantly drops on AEs generated with PGD, with an accuracy of 0.61%, precision of 0.64%, recall of 0.61%, and F1-score of 0.6%. The AEDPL-DL framework demonstrates a significant improvement in performance. For example, AEDPL-DL using ANND as a defense model in the protection layer achieves an accuracy of 97.91%, while RFD, KNND and FAMD yield 96.74%, 92.48%, and 73.88%, respectively. Under the SA attack (Table 1(d)), the performance drops on AEs generated by SA, with an accuracy of 53.3%. AEDPL-DL incorporating RFD achieves an accuracy of 99.06%, and with ANND, KNND and FAMD yield 98.84%, 98.08%, and 86.93%, respectively. These results indicate that AEDPL-DL enhances the robustness of DL models against adversarial attacks by incorporating an auxiliary ML model in the protection layer, thereby improving the overall performance.

b: DETECTORS PERFORMANCE IN THE PROTECTION LAYER

In this section, the results by employing different detectors in the protection layer are analyzed. The performance is measured based on their ability to identify AEs and prevent the progression into the classification phase of AEDPL-DL. For instance, a detection rate of 90% indicates that the detector successfully detects and excludes 90% of AEs.

Table 2 presents the performance of different detectors employed in the protection layer, as evaluated using the MNIST data set under four attacks (FGSM, BIM, PGD, and SA with an epsilon (eps ϵ) value of 0.30). Table 2(a) illustrates the accuracy, precision, recall, and F1-score values achieved by each detector with the FGSM attack. It is evident that ANND attains the highest accuracy rate of 94.32%. RFD

TABLE 2. The performance (detection rates) of various detector models in protection layer against four attacks (a-d) using the MNIST data set with eps ϵ= 0.30.

Attacks	Measures	ANND	RFD	KNND	XGBD	GBMD	FAMD
(a) FGSM	Accuracy	94.32	93.92	93.12	89.28	81.6	74.1
	Precision	94.51	93.94	93.14	89.28	81.61	74.41
	Recall	94.32	93.92	93.12	89.28	81.6	73.44
	F1-score	94.32	93.92	93.13	89.28	81.6	74.66
(b) BIM	Accuracy	94.98	92.85	92.8	89.68	80.65	76.3
	Precision	95.01	92.85	92.9	89.69	80.73	76.33
	Recall	94.98	92.85	92.8	89.68	80.65	75.82
	F1-score	94.97	92.85	92.8	89.68	80.64	77.18
(c) PGD	Accuracy	94.88	93.28	93.18	87.95	76.7	73.8
	Precision	94.89	93.28	93.23	88.04	76.81	74.16
	Recall	94.88	93.28	93.18	87.95	76.7	73.17
	F1-score	94.87	93.27	93.17	87.95	76.69	74.78
(d) SA	Accuracy	98.02	98	97.72	97.35	93.48	79.6
	Precision	98.08	98	97.82	97.36	93.63	78.28
	Recall	98.02	98	97.72	97.35	93.48	81.15
	F1-score	98.02	98	97.72	97.35	93.47	79.68

and KNND also demonstrate a competitive performance, with accuracy values of 93.92% and 93.12% respectively. On the other hand, XGBD exhibits a lower performance as compared with those of ANND, RFD, and KNND. In addition, GBMD and FAMD display even lower scores, with 81.6% and 74.1%, respectively. Table 2(b) showcases the detector performance against the BIM attack. Clearly, ANND outperforms other detector models by achieving the highest accuracy rate of 94.98%. RFD and KNND also perform well, achieving 92.85% and 92.8%, respectively. XGBD shows a slightly lower performance of 89.68%, while GBMD and FAMD exhibit lower scores overall, with 80.65% and 76.3%, respectively.

Table 2(c) presents the results against the PGD attack, ANND, RFD, and KNND report high accuracy scores of 94.88%, 93.28%, and 93.18%, respectively. XGBD demonstrates a slightly lower performance of 87.95%. GBMD and FAMD depict inferior results of 76.7% and 73.8%, respectively. Table 2(d) shows the results against the SA attacks, with ANND achieving the highest accuracy rate of 98.02%. It is followed by RFD and KNND at 98% and 97.72%, respectively. The results of XGBD, GBMD, and FAMD under SA attack are lower than those results of ANND, RFD and KNND.

In summary, ANND, RFD, and KNND demonstrate strong and consistent performance against all attacks. These models consistently yield high accuracy, precision, recall, and F1-score values. XGBD, GBMD and FAMD exhibit lower performances than those of ANND and KNN. These findings emphasize the robustness of ANND, RFD, and KNND models in detecting AEs against four attacks in the proposed AEDPL-DL framework.

c: PERFORMANCE OF AEDPL-DL AND DETECTORS UNDER VARYING EPSILON SETTINGS

The choice of epsilon (eps ϵ) in FGSM, PGD, BIM, and SA attacks has a significant impact on the magnitude of perturbations applied to the input images (CEs) and, consequently, on the effectiveness of generating AEs. It also affects the detector performance in detecting AEs. Therefore, this experimental study aims to evaluate the AEDPL-DL performance with different epsilon (eps ϵ) settings under four types of attack.

When the epsilon value is small, such as 0.05, it restricts the magnitude of perturbation applied to each pixel, resulting in a smaller change to the original image. In other words, with a small epsilon value, such as 0.05, perturbations are limited, causing minimal changes to the original image. Although some adversarial samples bypass the protection layer, the DNN classifier still performs well. This is because the changes to the image are so small that the DNN considers them normal and clean. As a result, the feature space remains within or close to the boundaries of the training data, allowing the DNN classifier to effectively handle all samples, as depicted in Fig. 3 (a to d). On the other hand, as the epsilon value increases, the performance of the DNN model decreases significantly with AEs and CEs-AEs using the FGSM, BIM, PGD, and SA attacks. For the BIM, PGD, and SA attacks, the performance of DNN with AEs decreases significantly when epsilon exceeds 0.15. This is because a larger epsilon value allows more substantial perturbation to be applied to each pixel, resulting in a larger change to the image. For larger epsilon values, the generated AEs significantly deceive the DNN without protection layer, as adversaries have more freedom to manipulate the data samples.

Despite escalating epsilon values used by attacks, the AEDPL-DL framework maintains a robust performance. This is accomplished through the integration of auxiliary ML models as detectors within the protection layer, enabling the detection and exclusion of AEs. Specifically, AEDPL-DL leverages ANND, RFD, KNND and XGBD to achieve performance levels that are closely aligned to those observed on CEs. AEDPL-DL remains unaffected by increasing epsilon values, although GBMD and FAMD models depict a slight performance reduction.

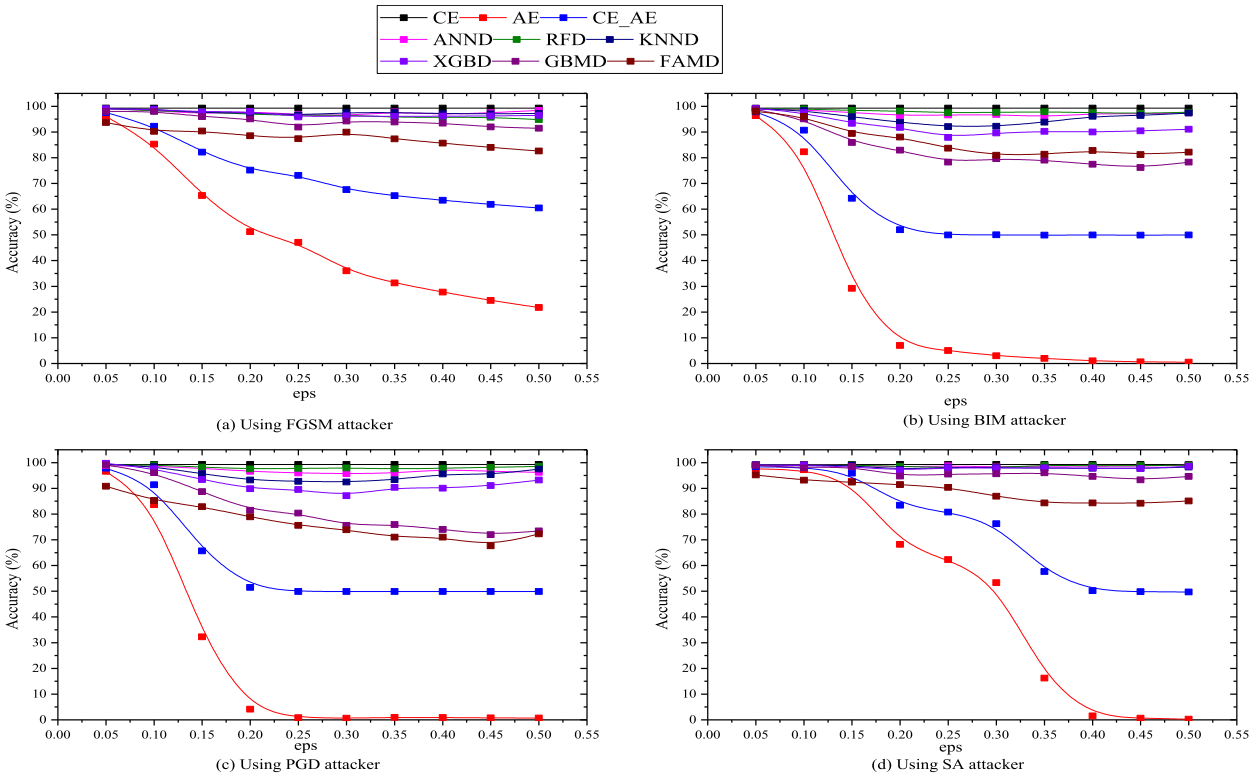

FIGURE 3. The performance of AEDPL-DL on the MNIST data set using various protection layer models against adversarial attacks with epsilon (eps ϵ) values ranging from 0.05 to 0.5.

The performance of six detectors within the protection layer against four attacks (FGSM, BIM, PGD, and SA) is summarized in Fig. 4. These attacks are assessed using various epsilon values. The results indicate that the detectors generally perform well across different epsilon values, except for a slight reduction in performance when dealing with a very small epsilon setting (e.g., 0.05). This reduction is owing to the minimal perturbation in AEs, which makes them more difficult to detect. However, with a small epsilon value, the performance of detectors still remains good. ANND and RFD demonstrate the highest performance in detecting AEs, exhibiting their robustness across various epsilon values. KNND and XGBD also perform well, albeit with slightly lower accuracy rates as compared with those of ANND and RFD. On the other hand, FAMD and GBMD exhibit a poor performance in detecting AEs, yielding lower accuracy rates as compared with those from other detectors. However, DNN models experience only small fooling rates when confronting AEs generated using small epsilon values, as illustrated in Fig. 3.

2) CIFAR-10

a: AEDPL-DL PERFORMANCE

AEDPL-DL is evaluated against AEs generated using four attack methods of FGSM, BIM, PGD, and SA on the CIFAR-10 data set. The results are presented in Table 3.

Table 3(a) presents the AEDPL-DL performance using CEs against FGSM attacks. The model demonstrates a good performance on clean data. However, when exposed to AEs generated by adversarial attacks, the model performance significantly drops, revealing its vulnerability to such attacks. The performance of AEDPL-DL with CEs_AEs, exhibits relatively a better performance as compared with that of AEs alone, but a significant decrease as compared with that of CEs alone. On the other hand, AEDPL-DL with detectors and defender models within the protection layer demonstrates a notable improvement in performance. For example, AEDPL-DL coupled with RFD and ANND achieve results close to those of CEs, with accuracy values of 84.31% and 83.61%, respectively. This is in contrast to the accuracy values of AEs and CEs_AEs, which are 10.8% and 47.24%, respectively. While AEDPL-DL with KNND XGBD, and GBMD also report good performances, their accuracy values are slightly lower than those using ANND and RFD. However, incorporating FAMD as a defense mechanism in the protection layer of AEDPL-DL yields slightly a lower performance, as compared with those from other ML models

Table 3(b) presents the performance of AEDPL-DL against the BIM attack. AEDPL-DL with ANND achieves a high accuracy rate of 85.26%, while RFD, KNND and XGBD achieve accuracy rates of 85.14%, 85.1%, and 85.14% respectively. AEDPL-DL with GBMD shows a slightly lower performance with an accuracy score of 84.87%, while with FAMD exhibits an even lower performance with an accuracy rate of 70.73%. Similarly, a significant performance drop is observed on AEs generated by the PGD attack, resulting in an accuracy rate of 2.4% in Table 3(c). However,

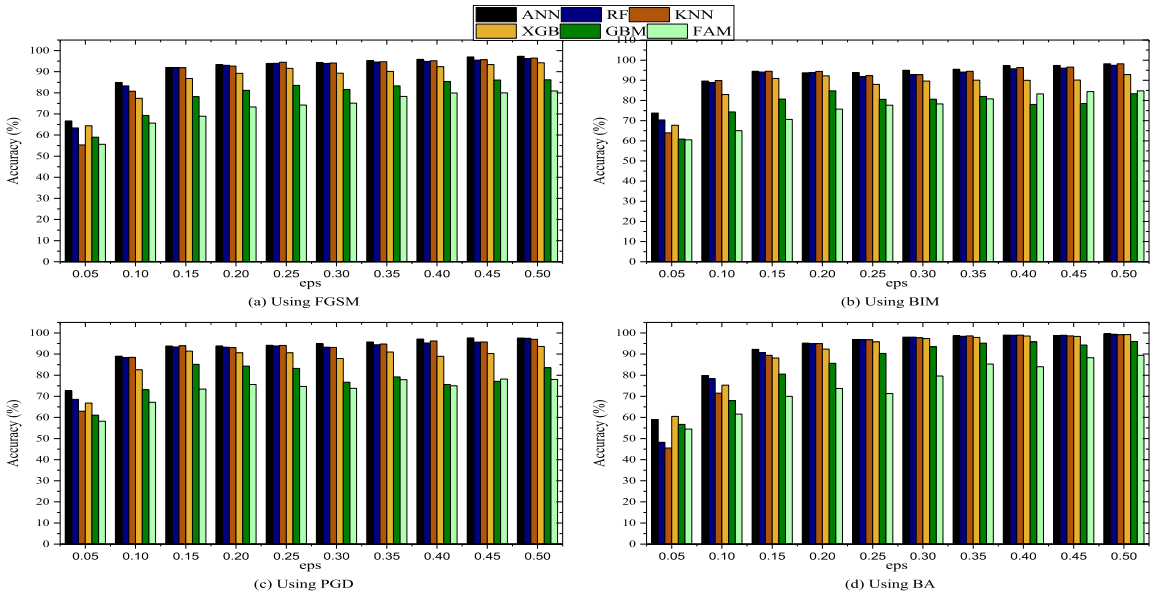

FIGURE 4. The accuracy of various detectors using four adversarial attacks with epsilon (eps ϵ) values ranging from 0.05 to 0.5, on the MNIST data set.

TABLE 3. The performance of AEDPL-DL using various detecting models (ANND, RFD KNND, XGBD, GBMD, FAMD) in the protection layer against four attacks on the CIFAR-10 data set (CEs, AES, and CEs_AEs) with eps = 0.30.

Attacks	Measures	CEs	AEs	CEs_AEs						
		-	-	-	ANND	RFD	KNND	XGBD	GBMD	FAMD
(a) FGSM	Accuracy	85.37	10.8	47.24	84.31	83.61	82.33	82.43	78.47	70.32
	Precision	85.32	18.68	72.07	84.76	83.92	82.99	83.06	81.02	72.2
	Recall	85.37	10.8	47.24	84.31	83.61	82.33	82.43	78.47	70.53
	F1-score	85.33	3.83	47.24	84.31	83.61	82.33	82.43	78.47	71.72
(b) BIM	Accuracy	85.37	5.04	45.1	85.26	85.14	85.1	85.14	84.87	70.73
	Precision	85.32	7.43	63.1	85.13	85.09	85.05	85.09	84.83	71.25
	Recall	85.37	5.04	45.1	85.14	85.14	85.1	85.14	84.87	72.46
	F1-score	85.33	4.57	50.06	85.18	85.1	85.06	85.1	84.83	71.12
(c) PGD	Accuracy	85.37	2.4	43.54	84.66	84.68	84.63	84.68	84.55	73.69
	Precision	85.32	5.9	62.43	84.67	84.69	84.64	84.69	84.57	73.07
	Recall	85.37	2.4	43.54	84.66	84.68	84.63	84.68	84.55	72.69
	F1-score	85.33	2.1	48.21	84.55	84.57	84.52	84.57	84.44	72.58
(d) SA	Accuracy	85.37	3.41	44.36	83.9	84.81	83.35	82.74	76.48	69.54
	Precision	85.32	7.53	64.37	83.95	84.8	83.24	82.57	76.89	70.47
	Recall	85.37	3.41	44.36	83.9	84.81	83.35	82.74	76.48	69.58
	F1-score	85.33	2.17	49.83	83.89	84.77	83.25	82.6	76.35	69.32

integrating ANND, RFD, KNND XGBD, and GBMD into AEDPL-DL yields higher accuracy rates, reaching up to 84.68%, while AEDPL-DL with FAMD achieves a lower accuracy rate of 73.69%. Furthermore, when evaluating the SA attack (Table 3(d)), the performance decreases, resulting in an accuracy of 3.41%. AEDPL-DL based on ANND, RFD, KNND and XGBD achieves accuracy rates of 83.9%, 84.81%, 83.35%, and 82.74% respectively, while GBMD and FAMD yield lower accuracy rates of 76.48% and 69.54% respectively. These results highlight the effectiveness of AEDPL-DL along with detectors based on auxiliary ML models in enhancing performance against the PGD, BIM, and SA attacks, leading to an improvement on AEs detection.

b: DETECTORS PERFORMANCE IN THE PROTECTION LAYER

The results of different detectors in the protection layer against FGSM, BIM, PGD, and SA attacks on the CIFAR-10 data set with an epsilon (eps) value of 0.30 are presented in Table 4. It is evident that all detectors in the protection layer perform well, with ANND achieving the highest accuracy rate of 98.6%. This is followed by KNND, RFD, and XGBD with accuracy scores of 97.9%, 97.68%, and 97.3% respectively. While GBMD and FAMD also demonstrate a good performance, their accuracy scores are lower as compared with those of other models, at 93.22% and 82.22% respectively. Similarly, Table 4(b) presents the performance of various detectors in the protection layer against the BIM attack. The results reveal that the ANND and RFD models achieve a perfect performance with 100% accuracy, precision, recall, and F1-score. The KNND and XGBD models also perform exceptionally well, with accuracy above 99.9%. GBMD demonstrates a high performance at around 99.62%. In contrast, FAMD is less effective with an accuracy rate of 88.62%.

TABLE 4. The performance of various detectors in the protection layer against four attacks (a-d) using the CIFAR-10 data set with eps $\epsilon = 0.30$.

Attacks	Measures	ANND	RFD	KNND	XGBD	GBMD	FAMD
(a) FGSM	Accuracy	98.6	97.68	97.9	97.3	93.22	82.22
	Precision	98.6	97.68	97.91	97.3	93.24	82.85
	Recall	98.6	97.68	97.9	97.3	93.22	82.42
	F1-score	98.6	97.68	97.9	97.3	93.23	82.91
(b) BIM	Accuracy	100	100	99.95	99.92	99.62	88.62
	Precision	100	100	99.95	99.93	99.63	88.13
	Recall	100	100	99.95	99.92	99.62	88.25
	F1-score	100	100	99.95	99.92	99.63	88.72
(c) PGD	Accuracy	99.98	100	99.92	99.95	99.88	87.88
	Precision	99.98	100	99.93	99.95	99.88	88.01
	Recall	99.98	100	99.92	99.95	99.88	87.95
	F1-score	99.97	100	99.92	99.95	99.88	87.95
(d) SA	Accuracy	96.28	96.48	96.1	96.28	92.85	82.85
	Precision	96.29	96.48	96.1	96.28	93.44	82.27
	Recall	96.28	96.48	96.1	96.28	92.85	82.22
	F1-score	96.27	96.47	96.1	96.27	92.82	82.22

Table 4(c) presents the performance against the PGD attack, where RFD achieves perfect accuracy. ANND, KNND XGBD, and GBMD perform well, achieving approximately 99% across all metrics. FAMD shows a lower performance with scores around 87.88%. For the SA attack (Table 4 (d)), the RFD model performs the best with an accuracy rate of 96.48%. ANND and XGBD models achieve a similar high performance with scores of 96.28%. GBMD attains an accuracy rate of 92.85%. However, FAMD shows a lower performance with an accuracy rate of 82.85%. Overall, the ANND and RFD models stand out as highly effective in detecting AEs and protection against the various attacks, while the FAMD model exhibits the least effectiveness among the evaluated models.

c: PERFORMANCE OF AEDPL-DL AND DETECTORS UNDER VARYING EPSILON SETTINGS

The AEDPL-DL framework is evaluated using different epsilon settings to simulate attacks on the CIFAR-10 data set. Fig. 5 depicts the impact of varying epsilon values (ranging from 0.05 to 0.5) on the AEDPL-DL performance with four attacks. It can be observed that there are no significant changes in performance as epsilon values increase. However, a slight decrease in performance is noticed on AEs-CEs within the epsilon range from 0.05 to 0.15, along with an improvement in the performance of AEDPL-DL with ANND, RFD, KNND XGBD, GBMD, and FAMD within the same range. AEDPL-DL performance remains consistent under FGSM, BIM, and PGD attacks (Figs. 5(a-c)). Meanwhile, when examining the SA attack (Fig. 5(d)), it is observed that the performance depicts a slight reduction within the epsilon range of 0.05 to 0.15, but maintains a consistent level of performance within the epsilon range of 0.15 to 0.5. Notably, the changes become negligible when increasing epsilon from 0.15 to 0.5 with all attacks.

The findings suggest that the CIFAR-10 data set is significantly impacted by the attacks, especially with small perturbations (e.g., epsilon = 0.05). This indicates that DL models are more susceptible to attacks when using the CIFAR-10 data set, as compared with that of the MNIST dataset, which is likely owing to the higher complexity of CIFAR-10. The results emphasize the importance of robust defense mechanisms to protect DNN models against adversarial attacks in complex data sets like CIFAR-10. However, AEDPL-DL demonstrates a robust performance against various attacks with different epsilon settings. As shown in Fig. 5 (a to d), the performance of AEDPL-DL with all detectors within the protection layer exhibits a high performance, closely approaching the performance on CEs. This achievement is attributed to the integration of auxiliary ML models in the protection layer, enabling the detection and exclusion of AEs.

Fig. 6 illustrates the performance of various detectors against four attacks (FGSM, BIM, PGD, and SA) with varying epsilon values. In general, the detectors demonstrate a high performance with small epsilon values (e.g., 0.05). Notably, ANND, RFD, KNND and XGBD exhibit the highest performance, while FAMD exhibits relatively lower effectiveness in AEs detection.

3) AUC OF AEDPL-DL

The AUC metric is utilized to further evaluate the AEDPL-DL framework. This metric provides an overall assessment of a detector's performance, and is an evaluation measure widely applicable to various classification problems.

Table 5 presents the AUC results of AEDPL-DL with the protection layer based on various detector models on two data sets: (a) MNIST and (b) CIFAR-10. The AUC values indicate the ability of AEDPL-DL to differentiate between AEs and CEs while maintaining high performance on CEs. Referring to the MNIST data set, AEDPL-DL with RFD and ANND achieves an outstanding performance across all attack types, yielding AUC scores above 98% for FGSM, BIM, PGD, and SA attacks. This indicates that RFD and

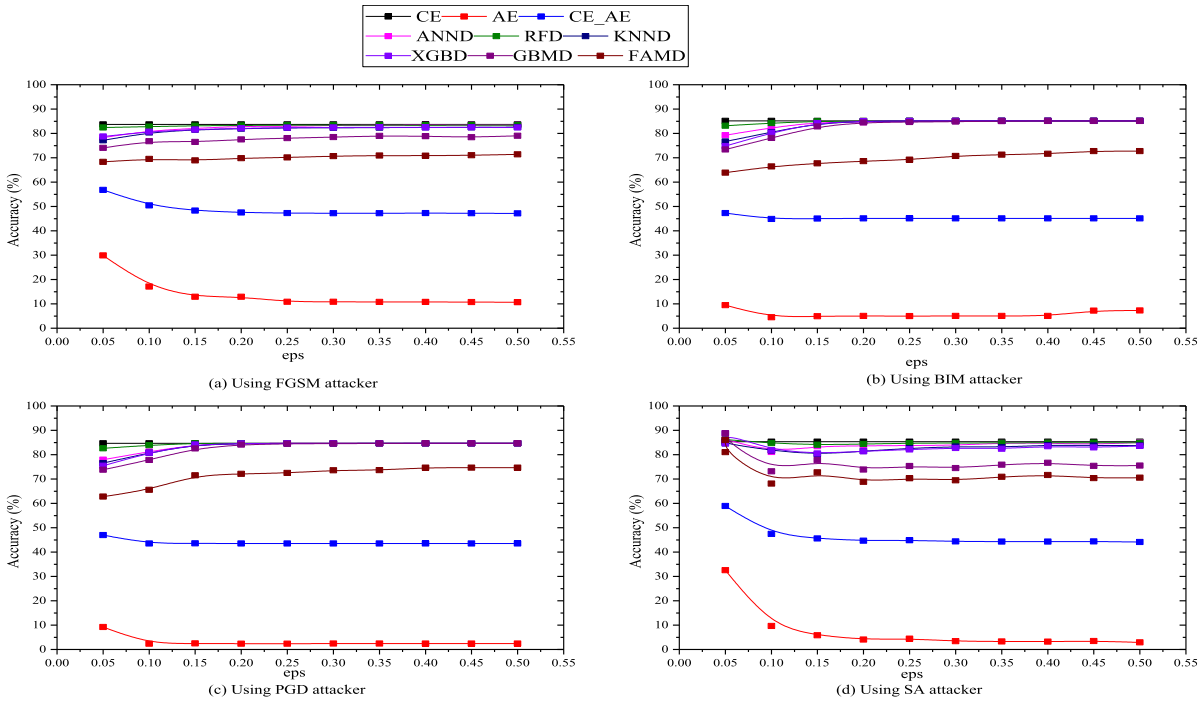

FIGURE 5. The performance of AEDPL-DL on the CIFAR-10 data set using various protection layer models against adversarial attacks with epsilon (eps ϵ) setting ranging from 0.05 to 0.5.

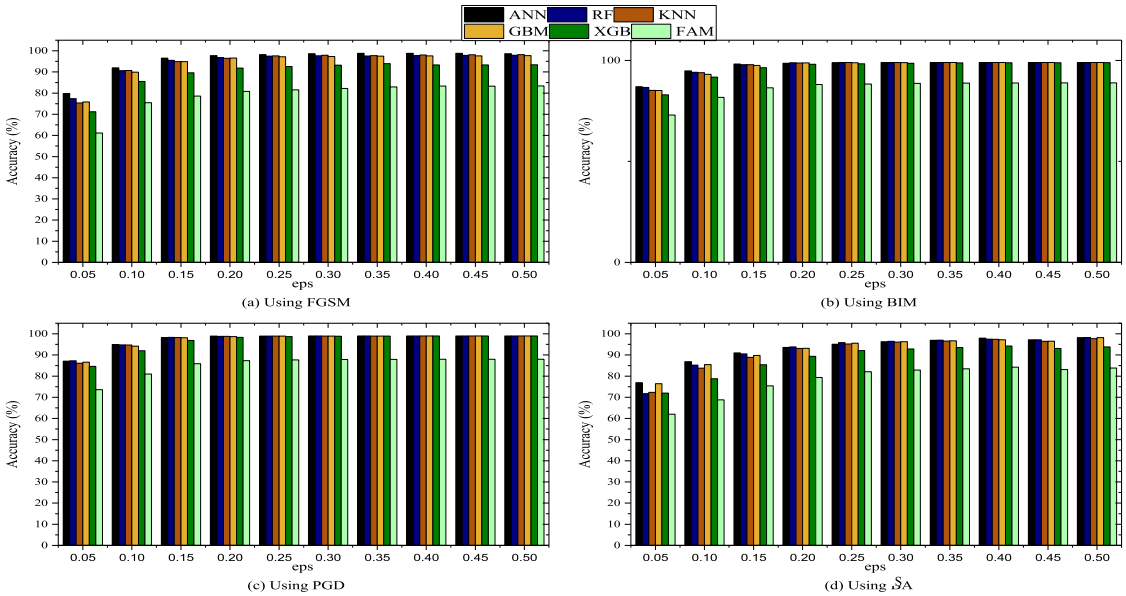

FIGURE 6. The accuracy of various detectors using four adversarial attacks with epsilon (eps ϵ) values ranging from 0.05 to 0.5, on the CIFAR-10 data set.

ANND are highly effective in detecting and filtering out AEs. Similarly, when using the CIFAR-10 data set, AEDPL-DL with ANND and RFD model performs well, achieving AUC scores above 84% for all attack types. However, the performance on CIFAR-10 is slightly lower than those for MNIST, suggesting that the higher complexity and variability in CIFAR-10 present greater challenges for adversarial detection. AEDPL-DL with ANND, KNND, and XGBD also demonstrate relatively strong performances, while those with GBMD and FAMD exhibit lower scores.

Overall, the results highlight the potential of AEDPL-DL to enhance the robustness of DL models against AEs. The incorporation of a protection layer based on detector models prior to the classifier aids in identifying and filtering out

TABLE 5. AUC measures for AEDPL-DL with protection layer based on detector of various models on (a) MNIST and (b) CIFAR-10 data sets (CEs, AEs, and CEs_AEs) with FGSM, BIM, PGD, and SA attacks.

Data sets	Attacks	CEs	AEs	CEs_AEs						
		-	-	-	ANND	RFD	KNND	XGBD	GBMD	FAMD
(a) MNIST	FGSM	99.4	58.36	78.88	98.7	99.24	98.65	98.24	96.41	85.78
	BIM	99.5	44.88	72.19	97	98.72	95.83	93.58	84.09	79.08
	PGD	99.4	44.8	72.18	97.8	98.93	96.42	93.82	84	78.69
	SA	99.4	61.99	80.76	99.08	99.37	98.64	98.37	95.87	89.14
(b) CIFAR-10	FGSM	89.47	50.99	52.58	84.31	84.36	83.04	83.51	81.78	79.54
	BIM	89.47	46.21	48.52	84.59	84.58	84.57	84.59	84.26	81.61
	PGD	89.47	46.21	48.52	84.57	84.59	84.58	84.59	84.27	80.93
	SA	89.47	46.32	50.38	83.72	84.2	83.13	82.86	82.35	81.49

adversarial inputs, thereby maintaining the high performance of the DL model on CEs. However, additional research and validation are required to optimize the selection and configuration of the protection layer models for different types of attacks.

4) COMPARISON WITH EXISTING MODELS

Table 6 provides a summary of DL models against AEs for AEDPL-DL and 13 existing methods in the literature, namely BU [19], FS [18], DNR [25], LID [26], ANR [17], FAD [32], EPM [29], SFAD [12], Regularization [30], Uncertainty [19], DAT [31], MUM [20] and DNR [25] against FGSM, PGD, and BIM attacks. The performance metrics include accuracy and AUC. The results from AEDPL-DL are selected according to the best performance of ML in the protection layer, i.e. ANND, RFD, and KNND.

By referring to the MNIST results in Table 6, AEDPL-DL achieves the highest accuracy scores, i.e., 98.62% for FGSM attacks and 97.91% for PGD attacks, as well as achieves high AUC of 99.24% for FGSM attacks and 98.72% for PGD attacks. AEDPL-DL exhibits a superior performance in detecting AEs created by these attacks as compared with those from other methods. FS also demonstrates high accuracy, with 97.96% for FGSM attacks and 97.19% for PGD attacks. On the other hand, DNR and EPM exhibit vulnerability to PGD attacks, achieving lower accuracy rates of 59.21% and 71.5%, respectively. SFAD performs well against FGSM attacks with an accuracy rate of 98.61%, but its accuracy drops considerably to 81.0% when facing PGD attacks.

AEDPL-DL also stands out with the highest accuracy rates among the compared methods of 84.31% for FGSM attacks and 84.68% for PGD attacks on the CIFAR-10 data set. FS achieves low accuracy rates of 32.5% for FGSM attacks and 4.2% for PGD attacks, while DNR achieves 30.23% for FGSM attacks and 18.23% for PGD attacks. ANR obtains promising accuracy rates of 83.2% for FGSM attacks, but its accuracy drops to 59.2% for PGD attacks. These values highlight the varying performance levels and vulnerabilities of the compared methods, further emphasizing the effectiveness of AEDPL-DL in detecting AEs.

The AUC scores are also compared with those of Uncertainty [19], DAT [31], MUM [20], and DNR [25] methods, as presented in Table 6. AEDPL-DL yields high AUC values, achieving 99.24% for FGSM attacks, 98.72% for PGD attacks, and 98.93% for BIM attacks on the MNIST data set. When handling the CIFAR-10 data set, AEDPL-DL maintains high AUC values of 84.36% for FGSM attacks, 84.58% for PGD attacks, and 84.59% for BIM attacks. The Uncertainty method also exhibits notable AUC values, particularly with 90.57% for FGSM attacks and 82.06% for BIM attacks on the MNIST data set. These values indicate its effectiveness in detecting AEs, especially against BIM attacks. While the other compared methods demonstrate varying levels of performance, AEDPL-DL consistently outperforms them, highlighting its robustness as a defense mechanism.

Overall, AEDPL-DL surpasses existing DL methods in addressing AE issues pertaining to various attacks on the MNIST and CIFAR-10 data sets. These findings highlight its potential as a highly effective defense mechanism against adversarial attacks. The high performance of AEDPL-DL as compared with those from other methods confirms its robustness and superiority in AEs detection.

5) ACCURACY OF AEDPL-DL WITH THE SHVN AND FASHION-MNIST DATA SETS

We expanded our experiment to include the SHVN and Fashion-MNIST data sets. Table 7 shows the accuracy of AEDPL-DL with a protection layer using different detectors, evaluated on SVHN and Fashion-MNIST against various attacks (FGSM, BIM, PGD, SA, and DF).

a: SVHN DATA SET

Against FGSM attacks, the model achieved an accuracy of 96.54% on CEs, yet exhibited a noticeable reduction to 19.29% against AEs. The inclusion of the protection layer, utilizing diverse detector models, markedly augmented accuracy against AEs. Specifically, AEDPL-DL with KNND yielded the highest accuracy score at 95.04%. This trend persisted across DF, BIM, PGD, and SA attacks, with AEDPL-DL featuring the protection layer consistently enhancing the model's performance against AEs, achieving

TABLE 6. AEs detection accuracy and AUC of AEDPL-DL and existing methods on MNIST and CIFAR-10 datasets.

Comparative Method	(a) MNIST			(b) CIFAR-10			
Method	FGSM	PGD	BIM	FGSM	PGD	BIM	Measures
BU [19]	82.2	77.8	-	84.0	56.5	-	Accuracy
FS [18]	97.96	97.19	-	32.5	4.2	-	Accuracy
DNR [25]	79.67	59.21	-	30.23	18.23	-	Accuracy
LID [26]	77.46	77.03	-	76.15	-	-	Accuracy
ANR [17]	92.8	74.6	-	83.2	59.2	-	Accuracy
FAD [32]	75.3	73.2	-	73.4	74.4	-	Accuracy
EPM [29]	74.3	71.5	-	61.4	66.4	-	Accuracy
SFAD[12]	**98.61**	81.0	-	80.14	41.2	-	Accuracy
Regularization [30]	-	-	87.6		-		Accuracy
Uncertainty [19]	90.57	-	82.06	72.23	-	81.05	AUC
DAT [31]	-	-	-		-	78	AUC
MUM [20]	-	-	-	81.9	-	71.2	AUC
DNR [25]	-	79	-		49		AUC
AEDPL-DL	**99.24**	**98.72**	**98.93**	**84.36**	**84.58**	**84.59**	AUC
AEDPL-DL	**98.62**	**97.91**	**97.76**	**84.31**	**84.68**	**85.26**	Accuracy

TABLE 7. Accuracy measure for AEDPL-DL with protection layer based on detector of various models on (a) SVHN and (b) Fashion-MNIST data sets (CEs, AEs, and CEs_AEs) with FGSM, BIM, PGD, SA, and DF attacks.

Data sets	Attacks	CEs	AEs	CEs_AEs						
		-	-	-	ANND	RFD	KNND	XGBD	GBMD	FAMD
(a) SVHN	FGSM	96.54	19.29	70.71	94.14	94.22	95.04	94.25	94.04	78.64
	BIM	96.54	2.67	48.34	94.03	94.31	94.86	94.16	93.89	71.90
	PGD	96.54	1.12	47.65	94.52	94.87	94.43	93.22	89.57	67.46
	SA	96.54	25.48	73.93	95.47	95.87	95.85	94.91	93.16	75.82
	DF	96.54	13.54	54.80	93.83	94.12	93.94	93.37	90.23	73.58
(b) Fashion-MNIST	FGSM	91.63	6.54	48.16	89.53	88.92	89.39	87.30	82.68	69.98
	BIM	91.63	3.87	46.61	88.47	88.76	89.15	87.27	82.30	65.12
	PGD	91.63	1.43	45.18	88.95	88.53	89.20	87.04	82.19	63.57
	SA	91.63	35.62	49.77	90.27	90.14	90.38	88.18	83.06	67.72
	DF	91.63	4.25	43.86	88.40	88.68	89.55	87.38	82.62	65.40

accuracy rates up to 93.94%, 94.86%, 94.43%, and 95.85%, respectively. The detector performance in the protection layer is up to 95.87% with the RFD.

b: FASHION-MNIST DATA SET

In the case of FGSM attacks, the model exhibited an accuracy rate of 91.63% on clean examples, but reduced by 6.54% against adversarial examples. The protection layer, employing various detector models, significantly enhanced accuracy against AEs. Specifically, AEDPL-DL with ANND achieved the highest accuracy rate at 89.53%. This trend persisted across BIM, PGD, DF, and SA attacks, demonstrating substantial enhancements in accuracy against AEs, with AEDPL-DL reporting accuracy scores up to 88.47%, 88.95%, 88.40%, and 90.27%, respectively. The detector performance in the protection layer is up to 90.38% with the KNND.

In summary, the results from the SHVN and Fashion-MNIST data sets corroborate the findings from the MNIST and CIFAR-10 data sets, i.e., accuracy of AEDPL-DL improves with the integration of a protection layer. This demonstrates the robustness of our proposed method across different data sets and against various attack types.

6) STATISTICAL SIGNIFICANCE MEASUREMENT OF AEDPL-DL

To further confirm the effectiveness of AEDPL-DL against AEs, we conducted a comprehensive analysis by using the paired t-test. This test allows a statistical significance measurement on the existence of a statistically difference in performance in test results before and after implementing the protection layer in AEDPL-DL.

Table 8 presents the computed p-values across accuracy of MNIST and CIFAR-10, SVHN and MNIST Fashion datasets. All p-values are lower than 0.05, indicating a rejection of the null hypothesis. In other words, the statistical outcomes signify that the true mean of test results differs significantly

TABLE 8. Paired t-test as statistical significance measurement on results before and after the implementation of the protection layer in AEDPL-DL.

Data sets	p-value of paired t-test					
	ANND	RFD	KNND	XGB	GBMD	FAMD
MNIST	1.03E-02	9.51E-03	8.31E-03	7.12E-03	4.72E-03	9.15E-03
CIFAR-10	3.20E-05	4.47E-05	6.00E-05	5.72E-05	1.80E-04	1.08E-04
SVHN	8.06E-04	7.76E-04	7.40E-04	7.55E-04	6.91E-04	6.60E-04
Fashion-MNIST	2.06E-03	2.05E-03	1.81E-03	2.28E-03	3.00E-03	5.43E-03

before and after the implementation of the protection layer utilizing various models as detectors. These results provide statistical evidence that AEDPL-DL enhances its effectiveness against AEs with a statistically significance difference in performance across various attack scenarios.

These results provide strong empirical evidence that AEDPL-DL substantially enhances its effectiveness against AEs with a statistically significant impact across four attack scenarios.

7) ROBUSTNESS ANALYSIS OF AEDPL-DL FRAMEWORK

This section provides a comprehensive assessment on resilience of the proposed AEDPL-DL framework against AEs. The experimental results underscore its capacity to achieve high classification accuracy across diverse data sets (MNIST, CIFAR-10, SVHN, Fashion-MNIST) (Tables 1 to 7), closely approaching the performance on clean data CEs. The analysis is supported by a range of evaluation metrics, including accuracy, precision, recall, F1-score, and AUC.

Furthermore, various adversarial attacks were employed to induce perturbations, assessing the framework's robustness. It consistently demonstrated effectiveness in withstanding adversarial manipulations. Moreover, the framework's robustness is theoretically underpinned by the introduction of bounded perturbations (ϵ), subject to FGSM, BIM, PGD, and SA attacks (Figs. 3 to 6). This indicates the framework's ability to maintain high classification accuracy amidst adversarial perturbations.

Throughout extensive experiments, AEDPL-DL consistently exhibited a superior performance. It not only accurately identified and isolated nearly all adversarial examples but also maintained classification accuracy levels akin to those achieved with clean data. Its robustness is primarily attributed to the integration of a protection layer, harnessing different ML models, such as FAMD, RFD, ANND, KNND, XGBD, and GBMD. These components collectively bolster AEDPL-DL's resilience against adversarial attacks, demonstrating its potential for real-world applications. On the other side, regarding scalability, AEDPL-DL method demonstrates scalability across different datasets with varying sizes, notably up to 100,000 samples on the SVHN dataset.

The AEDPL-DL framework demonstrates its generalizability by achieving remarkable performance in multiple experiments across four diverse image datasets: MNIST, CIFAR-10, SVHN, and Fashion-MNIST. Its ability to generalize to unseen data is further validated by a rigorous evaluation strategy that involves splitting the data into various ratios and repeating the experiments ten times with randomized data shifts between training and testing sets. This comprehensive approach provides strong evidence of the model's generalizability and strengthens confidence in its effectiveness across different data distributions.

In summary, in light of comprehensive findings, the proposed framework, AEDPL-DL, represents a significant advancement in adversarial defense. Specifically, it introduces a new integration of CNNs, DNNs, and a protection layer, enabling effective detection and exclusion of adversarial examples while maintaining high accuracy. Rigorous evaluation against diverse adversarial attacks demonstrates its useful capability in performance improvement over those from the existing methods. Indeed, comparative analyses confirm AEDPL-DL's robustness in both adversarial example detection and enhancement of DNN performance. Additionally, its computational efficiency makes it a viable tool for real-world applications.

V. CONCLUSION

In the paper, we have developed a framework for AEs detection by utilizing a protection layer mechanism in DL models. The proposed AEDPL-DL framework comprises a CNN model designed for feature learning, a DNN classifier equipped with a defense mechanism against different types of adversarial attacks, such as FGSM, BIM, PGD, and SA. The DNN classifier includes a protection layer inserted before the classifier to detect and exclude AEs. The protection layer uses ML (e.g. RFD, KNND XGBD, GBMD, and FAMD) to identify AEs. Extensive evaluation studies have been conducted using the MNIST, CIFAR-10, SVHN, and Fashion-MNIST data sets to assess the effectiveness of the proposed framework. AEDPL-DL has achieved average classification accuracy rates up to 99.06%, 85.26%, 95.87%, and 90.38% on the MNIST, CIFAR-10, SVHN, and Fashion-MNIST data sets, respectively. These results are very close to those of the DL model applied to clean data. The detection accuracy rates of the detectors used in the protection layer achieve 98.02%,100%, 98.31%, and 97.65% on MNIST, CIFAR-10, SVHN, and Fashion-MNIST, respectively. Several attacks with various settings have been employed and investigated against AEDPL-DL, demonstrating its effectiveness under different scenarios. The comparative results have indicated that AEDPL-DL is

competitive with other state-of-the-art methods. In addition, the results have demonstrated that AEDPL-DL enhances the robustness of DL models against adversarial attacks by incorporating detectors as auxiliary ML models in the protection layer, thereby improving their overall performance.

Future work will explore incorporating an additional layer to clean AEs from adversarial attacks. Furthermore, high-resolution data will be utilized to ensure the developed model functions effectively with such data. Ensemble detection techniques will be explored in the future to enhance the transferable AEs detection performance. In addition, we plan to enhance computational efficiency and scalability by implementing parallel computing.

REFERENCES

[1] K. Gupta and V. Bajaj, "Deep learning models-based CT-scan image classification for automated screening of COVID-19," *Biomed. Signal Process. Control*, vol. 80, Feb. 2023, Art. no. 104268.

[2] S.-Y. Lee, T. H. M. Le, and Y.-M. Kim, "Prediction and detection of potholes in urban roads: Machine learning and deep learning based image segmentation approaches," *Develop. Built Environ.*, vol. 13, Mar. 2023, Art. no. 100109.

[3] M. N. Al-Andoli, S. C. Tan, K. S. Sim, C. P. Lim, and P. Y. Goh, "Parallel deep learning with a hybrid BP-PSO framework for feature extraction and malware classification," *Appl. Soft Comput.*, vol. 131, Dec. 2022, Art. no. 109756.

[4] M. N. Al-Andoli, S. C. Tan, K. S. Sim, P. Y. Goh, and C. P. Lim, "An ensemble deep learning classifier stacked with fuzzy ARTMAP for malware detection," *J. Intell. Fuzzy Syst.*, vol. 44, no. 6, pp. 10477–10493, Jun. 2023.

[5] S. M. Noe, T. T. Zin, P. Tin, and I. Kobayashi, "Comparing state-of-the-art deep learning algorithms for the automated detection and tracking of black cattle," *Sensors*, vol. 23, no. 1, p. 532, Jan. 2023.

[6] M. N. Al-Andoli, S. C. Tan, K. S. Sim, M. Seera, and C. P. Lim, "A parallel ensemble learning model for fault detection and diagnosis of industrial machinery," *IEEE Access*, vol. 11, pp. 39866–39878, 2023.

[7] Z. Weng, Z. Qin, X. Tao, C. Pan, G. Liu, and G. Y. Li, "Deep learning enabled semantic communications with speech recognition and synthesis," *IEEE Trans. Wireless Commun.*, vol. 22, no. 9, pp. 6227–6240, Sep. 2023.

[8] M. N. Al-Andoli, S. C. Tan, W. P. Cheah, and S. Y. Tan, "A review on community detection in large complex networks from conventional to deep learning methods: A call for the use of parallel meta-heuristic algorithms," *IEEE Access*, vol. 9, pp. 96501–96527, 2021.

[9] M. N. Al-Andoli, S. C. Tan, and W. P. Cheah, "Distributed parallel deep learning with a hybrid backpropagation-particle swarm optimization for community detection in large complex networks," *Inf. Sci.*, vol. 600, pp. 94–117, Jul. 2022.

[10] J. Wang, G. Dong, J. Sun, X. Wang, and P. Zhang, "Adversarial sample detection for deep neural network through model mutation testing," in *Proc. IEEE/ACM 41st Int. Conf. Softw. Eng. (ICSE)*, May 2019, pp. 1245–1256.

[11] A. Aldahdooh, W. Hamidouche, S. A. Fezza, and O. Déforges, "Adversarial example detection for DNN models: A review and experimental comparison," *Artif. Intell. Rev.*, vol. 55, no. 6, pp. 4403–4462, Aug. 2022.

[12] A. Aldahdooh, W. Hamidouche, and O. Déforges, "Revisiting model's uncertainty and confidences for adversarial example detection," *Int. J. Speech Technol.*, vol. 53, no. 1, pp. 509–531, Jan. 2023.

[13] N. Pitropakis, E. Panaousis, T. Giannetsos, E. Anastasiadis, and G. Loukas, "A taxonomy and survey of attacks against machine learning," *Comput. Sci. Rev.*, vol. 34, Nov. 2019, Art. no. 100199.

[14] Y. Gong, S. Wang, X. Jiang, L. Yin, and F. Sun, "Adversarial example detection using semantic graph matching," *Appl. Soft Comput.*, vol. 141, Jul. 2023, Art. no. 110317.

[15] H. Xu, Y. Ma, H.-C. Liu, D. Deb, H. Liu, J.-L. Tang, and A. K. Jain, "Adversarial attacks and defenses in images, graphs and text: A review," *Int. J. Autom. Comput.*, vol. 17, no. 2, pp. 151–178, Apr. 2020.

[16] Z. Yin, S. Zhu, H. Su, J. Peng, W. Lyu, and B. Luo, "Adversarial examples detection with enhanced image difference features based on local histogram equalization," 2023, *arXiv:2305.04436*.

[17] B. Liang, H. Li, M. Su, X. Li, W. Shi, and X. Wang, "Detecting adversarial image examples in deep neural networks with adaptive noise reduction," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 1, pp. 72–85, Jan. 2021.

[18] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," 2017, *arXiv:1704.01155*.

[19] R. Feinman, R. R. Curtin, S. Shintre, and A. B. Gardner, "Detecting adversarial samples from artifacts," 2017, *arXiv:1703.00410*.

[20] F. Sheikholeslami, S. Jain, and G. B. Giannakis, "Minimum uncertainty based detection of adversaries in deep neural networks," 2019, *arXiv:1904.02841*.

[21] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2014, *arXiv:1412.6572*.

[22] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial Intelligence Safety and Security*. Boca Raton, FL, USA: CRC Press, 2018, pp. 99–112.

[23] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2017, *arXiv:1706.06083*.

[24] M. Andriushchenko, F. Croce, N. Flammarion, and M. Hein, "Square attack: A query-efficient black-box adversarial attack via random search," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 484–501.

[25] A. Sotgiu, A. Demontis, M. Melis, B. Biggio, G. Fumera, X. Feng, and F. Roli, "Deep neural rejection against adversarial examples," *EURASIP J. Inf. Secur.*, vol. 2020, no. 1, pp. 1–10, Dec. 2020.

[26] X. Ma, B. Li, Y. Wang, S. M. Erfani, S. Wijewickrema, G. Schoenebeck, D. Song, M. E. Houle, and J. Bailey, "Characterizing adversarial subspaces using local intrinsic dimensionality," 2018, *arXiv:1801.02613*.

[27] B. Liu, J. Zhang, and J. Zhu, "Boosting 3D adversarial attacks with attacking on frequency," *IEEE Access*, vol. 10, pp. 50974–50984, 2022.

[28] M. Liu, Z. Zhang, Y. Chen, J. Ge, and N. Zhao, "Adversarial attack and defense on deep learning for air transportation communication jamming," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 1, pp. 973–986, Jan. 2024.

[29] J. Liu, W. Zhang, Y. Zhang, D. Hou, Y. Liu, H. Zha, and N. Yu, "Detection based defense against adversarial examples from the steganalysis point of view," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4825–4834.

[30] S. Pertigkiozoglou and P. Maragos, "Detecting adversarial examples in convolutional neural networks," 2018, *arXiv:1812.03303*.

[31] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff, "On detecting adversarial perturbations," 2017, *arXiv:1702.04267*.

[32] H. Ye and X. Liu, "Feature autoencoder for detecting adversarial examples," *Int. J. Intell. Syst.*, vol. 37, no. 10, pp. 7459–7477, Oct. 2022.

[33] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," 2016, *arXiv:1610.02136*.

[34] G. Goswami, A. Agarwal, N. Ratha, R. Singh, and M. Vatsa, "Detecting and mitigating adversarial perturbations for robust face recognition," *Int. J. Comput. Vis.*, vol. 127, nos. 6–7, pp. 719–742, Jun. 2019.

[35] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2574–2582.

[36] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial examples are not bugs, they are features," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.

[37] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[38] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep. 7, 2009.

[39] N. Yuval, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn.*, 2011, pp. 1–9.

[40] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.

[41] P. Moritz, R. Nishihara, S. Wang, A. Tumanov, R. Liaw, E. Liang, M. Elibol, Z. Yang, W. Paul, and M. I. Jordan, "Ray: A distributed framework for emerging AI applications," in *Proc. 13th USENIX Symp. Operating Syst. Des. Implement.*, 2018, pp. 561–577.

[42] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006.

MOHAMMED NASSER AL-ANDOLI received the B.Sc. degree in computer information systems from Mutah University, Jordan, in 2011, the M.Sc. degree in computer science from the Jordan University of Science and Technology, Jordan, in 2016, and the Ph.D. degree in information technology from Multimedia University, Malaysia, in 2022. He is currently a Senior Lecturer with the Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM). His main research interests include malware analysis, complex network analysis, machine learning, high-performance computing, deep learning, and parallel computing.

SHING CHIANG TAN received the B.Tech. (Hons.) and M.Sc. (Eng.) degrees from Universiti Sains Malaysia, Malaysia, in 1999 and 2002, respectively, and the Ph.D. degree from Multimedia University, Malaysia, in 2008. He is currently a Professor with the Faculty of Information Science and Technology, Multimedia University. His current research interests include computational intelligence, deep learning, adversarial machine learning, and applications in data classification, malware detection, condition monitoring, fault detection and diagnosis, stroke rehabilitation, and biomedical disease classification and optimization.

KOK SWEE SIM (Senior Member, IEEE) is currently a Professor with Multimedia University, Malaysia. He is working closely with various local and overseas institutions and hospitals. He has filed more than 18 patents and 70 copyrights. He has received many International and local Awards. He was a recipient of the Japan Society for the Promotion of Science (JSPS) Fellowship, Japan, in 2018, the Top Research Scientists Malaysia (TRSM) from the Academic Science Malaysia, in 2014, the Korean Innovation and Special Awards, in 2013, 2014, and 2015, and the TM Kristal Award and International Championships of World Summit on the Information Society (WSIS) Prizes, in 2017, 2018, 2019, 2020, and 2021.

PEY YUN GOH (Senior Member, IEEE) received the B.IT. (Hons.) in business information systems, in 2004, the M.Phil. degree in management, in 2007, and the Ph.D. degree in IT, in 2018. She is currently a Lecturer with the Faculty of Information Science and Technology, Multimedia University (MMU), Malaysia. She is also attached to artificial intelligent cluster, which is one of the special interest groups within FIST. Her research field changes from management to IT in order to support her career interest in IT. Her research interests include knowledge discovery, neural networks, pattern recognition, soft computing, and adversarial machine learning.

CHEE PENG LIM received the Ph.D. degree from The University of Sheffield, U.K., in 1997. He is currently a Professor with Deakin University, Australia. He has published more than 550 technical papers in books, international journals, and conference proceedings. His research interests include computational intelligence, data analytics, pattern classification, and multi-objective optimization.