Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.039931
Article

Archimedes Optimization with Deep Learning Based Aerial Image Classification for Cybersecurity Enabled UAV Networks

Faris Kateb and Mahmoud Ragab*

Information Technology Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, 21589, Saudi Arabia

*Corresponding Author: Mahmoud Ragab. Email: mragab@kau.edu.sa
Received: 24 February 2023; Accepted: 22 May 2023; Published: 28 July 2023

Abstract: The recent adoption of satellite technologies, unmanned aerial vehicles (UAVs) and 5G has encouraged telecom networking to evolve into more stable service to remote areas and render higher quality. But, security concerns with drones were increasing as drone nodes have been striking targets for cyberattacks because of immensely weak inbuilt and growing poor security volumes. This study presents an Archimedes Optimization with Deep Learning based Aerial Image Classification and Intrusion Detection (AODL-AICID) technique in secure UAV networks. The presented AODL-AICID technique concentrates on two major processes: image classification and intrusion detection. For aerial image classification, the AODL-AICID technique encompasses MobileNetv2 feature extraction, Archimedes Optimization Algorithm (AOA) based hyperparameter optimizer, and backpropagation neural network (BPNN) based classifier. In addition, the AODL-AICID technique employs a stacked bi-directional long short-term memory (SBLSTM) model to accomplish intrusion detection for cybersecurity in UAV networks. At the final stage, the Nadam optimizer is utilized for parameter tuning of the SBLSTM approach. The experimental validation of the AODL-AICID technique is tested and the obtained values reported the improved performance of the AODL-AICID technique over other models.

Keywords: Aerial image classification; remote sensing; intrusion detection; cybersecurity; deep learning

1 Introduction

Nowadays, drones or Unmanned Aerial Vehicles (UAVs) [1] can be referred to as flying objects that can fly autonomously or with the help of human pilots. Drones were utilized for rescue operations, package delivery, environmental management, aerial mapping, aerial photography, irrigation, monitoring, and other critical applications [2]. Intrusion detection methods and Security schemes will be utilized for guaranteeing critical security features [3]. The aerial image classifier methods present different indispensable classes that can be achieved by operating the changes in spatial organization along with that systematic process for devising the desired method. Dissimilar to pixel categorizing

methods, scene classifier delivers localized data even at the occurrence of wider aerial imageries which have unambiguous semantic data of surfaces [4,5].

In this study, aerial scenes are distinguished into lower-level structures which often use spectral, texture, structural features, and many more [6]. Then, low-level feature vectors that are visual parameters are filtered globally or locally and implemented for defining the aerial scene images. Recently, deep learning (DL) based techniques were implemented in high-level feature extraction [7]. The orthodox DL methods deliver a potential result while processing existing problems such as speech or object analysis, processing and predicting the natural language. So, the above-discussed structure gained a reputed task and attained an existing ranking which had a focus on educational fields along with that commercial groups. Also, DL techniques relevant to hierarchical features defined several phases of extraction. Also, deep Convolutional neural networks (DCNN) were crucial components implemented in scene classification, inspecting in addition to detecting processes [8]. DL methods, particularly CNN, have attained great achievement in segmentation, image classification, and detection on numerous benchmarks [9]. CNN refers to a hierarchical network invariant to image translations. The key to success is the capability to study the progressively complicated transformations of input and for capturing invariances from large, labelled data sets [10]. But it is problematic to directly implement the CNNs to remote sensing scene classification for millions of variables to train the CNNs, which trained samples are inadequate for training.

This study presents an Archimedes Optimization with Deep Learning based Aerial Image Classification and Intrusion Detection (AODL-AICID) technique in secure UAV networks. The presented AODL-AICID technique concentrates on two major processes: image classification and intrusion detection. For aerial image classification, the AODL-AICID technique encompasses MobileNetv2 feature extraction, Archimedes Optimization Algorithm (AOA) based hyperparameter optimizer, and backpropagation neural network (BPNN) based classifier. In addition, the AODL-AICID technique employs a stacked bidirectional long short-term memory (SBLSTM) model to accomplish intrusion detection for cybersecurity in UAV networks. At the final stage, the Nadam optimizer is used for optimal hyperparameter tuning of the SBLSTM model. The experimental validation of the AODL-AICID technique is tested under different performance measures.

2 Related Works

He et al. [11] examine a conditional GAN (CGAN)-related collaborative ID technique with blockchain (BC) allowed distributed federated learning for solving these issues. This research establishes LSTM into CGAN trained for improving the outcome of generative networks. According to the extracting feature capability of LSTM networks, the created data with CGAN can be utilized as augmented data and executed in the classification and recognition of intrusion data. Alissa et al. [12] concentrated on the proposal of a Crystal Structure Optimizer with DAE-based ID (CSODAE-ID) for securing the IoD platform. The purpose of the projected CSODAE-ID technique is to detect the intrusion presence from the IoD platform. During the presented CSODAE-ID technique, a novel Modified Deer Hunting Optimized-based FS (MDHO-FS) system was executed for choosing the feature subsets. Simultaneously, the Auto-encoder (AE) technique was utilized for the intrusion classifier from the IoD platform. The CSO technique is simulated by the development of crystal frameworks dependent upon the lattice points utilized finally the hyperparameter-tuning procedure.

Tan et al. [13] introduce an ID approach dependent upon a DBN optimizer by PSO technique. At primary, the classifier method dependent upon the DBN has been created, and the PSO system was utilized for optimizing the hidden layer node count of DBN for obtaining a better DBN infrastructure.

Sedjelmaci et al. [14] concentrate on a particular case which is UAV Edge Computing (UEC) network. Addressing the security problem in the UEC network has compulsory owing to the significance of UEC services like network traffic monitor, or searching and rescuing functions. Zhang et al. [15] present a hybrid approach dependent upon either spectral traffic investigation or robust controller or observer to anomaly estimate inside a UAV network. This technique was dependent upon either Lyapunov Krasovskii functional or dynamic performance.

Whelan et al. [16] examine a novelty-based method to ID from UAVs by utilizing a one-class classifier. PCA was implemented to sensor logs for reducing dimensional, and one-class classifier techniques can be created per sensor. The count of one-class classifiers can be chosen: One-Class SVM, AE-NN, and Local Outlier Factor. In [17], a railway clearance ID technique in aerial video dependent upon CNN was presented. Initially, the rail track area has declared in aerial single frame image with linear segment recognition technique, line segment merge as well as screen. At last, the single frame detection outcome was optimised by the inter-frame connection of videos for obtaining the last outcome of railway clearance ID from aerial video.

3 The Proposed Model

In this study, we have developed a new AODL-AICID technique for aerial image classification and intrusion detection in UAV networks. The presented AODL-AICID technique concentrates on two major processes: image classification and intrusion detection. For aerial image classification, the AODL-AICID technique encompasses MobileNetv2 feature extraction, an AOA-based hyperparameter optimizer, and a BPNN-based classifier. In addition, the AODL-AICID technique employed the Nadam optimizer with the SBLSTM model for intrusion detection in the UAV networks. Fig. 1 depicts the overall procedure of the AODL-AICID approach.

3.1 Module I: Aerial Image Classification

In this study, the AODL-AICID technique encompasses MobileNetv2 feature extraction, an AOA-based hyperparameter optimizer, and a BPNN-based classifier applied for aerial image classification.

3.1.1 Feature Extraction

Here, the AODL-AICID technique involves the MobileNetv2 model to generate a collection of future vectors. MobileNet architecture alternatively referred to as MobileNetV1 applies to embedded devices and smartphones [18]. Depthwise separable convolution was designed that comprised depthwise and pointwise convolutional layers for decreasing the size of the parameter and reducing the dimension of several layers. Following, rectified linear unit (ReLU) and batch normalization (BN) layers are added afterwards depth-wise separable convolution.

While applying MobileNet for testing on the ImageNet data, MobileNetV1 had 4.2 M parameters, whereas widespread architecture GoogLeNet and VGG16 architecture had 6.8 and 138 M, correspondingly. The experiment of the GoogLeNet attained 69.8% of accuracy whereas the MobileNetV1 on the ImageNet dataset attained 70.6% of accuracy. The authors developed a MobileNetV2 by increasing invert residual, a short connection. The inverted residual was introduced to deal with the memory problem by decreasing the quantity of tensor stored in memory while processing. The linear bottleneck which is a rise in the number of feature maps namely ResNet increase the feature maps from 64 to 128, 256, and 512, correspondingly. MobileNetV2 structure could reduce the number of parameters and be faster in computational time than MobileNetV1. The experiment with

MobileNetV2 attained an accuracy of 72.0%, that is was greater than NASNet, MobileNetV1, and ShuffleNet.

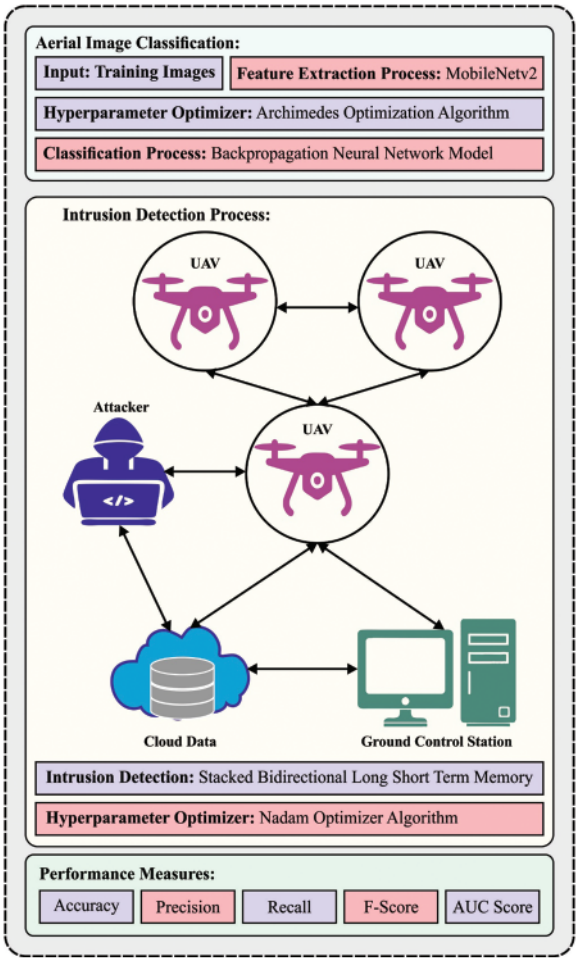

Figure 1: Overall procedure of the AODL-AICID system

To adjust the hyperparameter values of the MobielNetv2 model, the AOA algorithm is applied in this work. The AOA approach [19] was applied to resolve the optimization problem in conjunction with NR mathematical approach. The AOA technique is a metaheuristic approach used to resolve different mathematical optimized problems and is verified the ability for fetching a global solution from a limited duration. The AOA's necessary condition hinges on Archimedes' rule of buoyancy. The AOA enables various phases defining the nearby global solution, and this phase is illustrated by the following expression:

Phase 1: The initial population contains the immersed object (solution) and is regarded by acceleration, volume, and density. Every solution was initialized by the arbitrary place from the fluid as given below, following the fitness value to every solution is evaluated.

$$O_i = lb_i + rand\,(0,1) \times (ub_i - lb_i), \forall i \in \{1, 2, 3, \ldots,\ N\} \tag{1}$$

$$Den_i = rand\,(0,1) \tag{2}$$

$$Vol_i = rand\,(0, 1) \tag{3}$$

$$ACC_i = lb_i + rand\,(0, 1) \times (ub_i - lb_i), \forall i \in \{1, 2, 3, \ldots, N\} \tag{4}$$

Now O_i indicates the i^{th} solution from the population and N shows the population size. ub_i and lb_i correspondingly represent the upper and lower limitations of *the* i^{th} solution. Den_i, Vol_i, and ACC_i indicate the density, volume, and acceleration of i^{th} solutions. *rand* (0, 1) denotes the arbitrary scalar encompassing a value within [0, 1].

Phase 2: The density and volume of every solution were upgraded through the succeeding equation:

$$Den_i^{(t+1)} = Den_i^{(t)} + rand\,(0, 1) \times \left(Den_{best} - Den_i^{(t)}\right) \tag{5}$$

$$Vol_i^{(t+1)} = Vol_i^{(t)} + rand\,(0, 1) \times \left(Vol_{best} - Vol_i^{(t)}\right) \tag{6}$$

Now $Den_i^{(t)}$, and $Vol_i^{(t)}$ indicate the density and volume of i^{th} solution at t^{th} iteration. Den_{best} and Vol_{best} show the optimal density and volume of the best solution containing the best fitness value.

Phase 3: During the transfer operator and density factor, the collision among solutions remains in an equilibrium state and it can be mathematically expressed in the following:

$$TF = \exp\left\{\frac{t - t_{max}}{t_{max}}\right\} \tag{7}$$

In Eq. (7), TF represents the transfer operator that can transfer the search technique under the exploration to exploitation process. $t_{\max}$ shows the highest quantity of iterations. As well, a reducing density factor (d) can assist the AOA to find nearby global solutions.

$$d^{t+1} = \exp\left\{\frac{t - t_{\max}}{t_{\max}}\right\} - \left(\frac{t}{t_{\max}}\right) \tag{8}$$

Phase 4: During exploration, the collision between solutions takes place. Thus, if $TF \leq 0.5$, a random material (mr) has been selected where the acceleration of solution was upgraded by:

$$ACC_i^{(t+1)} = \frac{Den_{mr} + Vol_{mr} \times ACC_{mr}}{Den_i^{(t+1)} \times Vol_i^{(t+1)}} \tag{9}$$

Now n_{mr}, Vol_{mr}, and ACC_{mr} show the density, volume, and acceleration of random material.

Phase 5: During exploitation, no collision among solutions takes place. If $TF \geq 0.5$, the acceleration of the solution was upgraded by the following expression:

$$ACC_i^{(t+1)} = \frac{Den_{best} + Vol_{best} \times ACC_{best}}{Den_i^{(t+1)} \times Vol_i^{(t+1)}} \tag{10}$$

In Eq. (10), ACC_{best} represents the acceleration of the solution having an optimal fitness.

Phase 6: The Normalize acceleration is used to assess the percentage of alteration:

$$ACC_{i-norm}^{(t+1)} = g \times \frac{ACC_i^{(t+1)} - \min\{ACC\}}{\max\{ACC\} - \min\{ACC\}} + z \tag{11}$$

In Eq. (11), g, and z show the normalized range. $ACC_{i-nom}^{(r+1)}$ represent the step on the percentage that every agent is in phase.

Phase 6: during Evaluation, the fitness value of each solution is assessed, and the best solution was registered, for example, upgrading the optimal solution (x_{best}), Den_{best}, Vol_{best}, and ACC_{best}.

3.1.2 Image Classification

For classifying aerial images, the BPNN model is used. BPNN uses a monitored learning technique and the feed-forward structure of ANN [20]. The conventional BPNN involves a single hidden layer, 1 input layer, and one output layer. It contains n output neurons, one input neuron, and m hidden neurons. The data from the input layer is gone through the network by interconnecting weight to the hidden layer and later the output layer. Fig. 2 depicts the structure of BPNN. The weight interconnecting i^{th} input components to hidden neurons j^{th} are represented by W_{ji}, whereas the weighted interconnecting j^{th} hidden neurons to output neurons k^{th} are represented by W_{kj}. Every neuron evaluates the output based on the amount of simulation it obtains as an input. The neuron's net input is evaluated by the weighed amount of its inputs, and the output of the neuron depends on a sigmoid function and is based on the amount of net input.

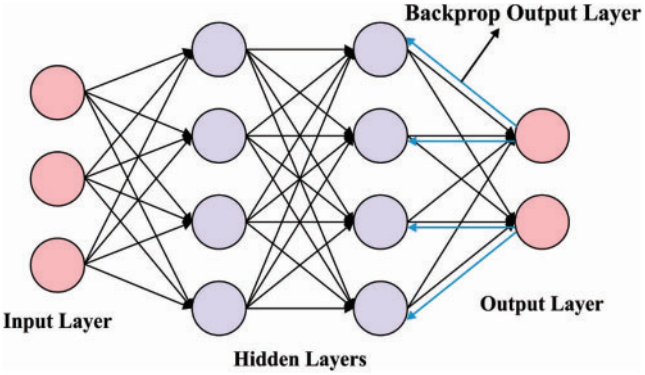

Figure 2: Architecture of BPNN

Place $I_c = (I_{c1}, I_{c2}, \ldots, I_{cl})$, $c = 1, 2, \ldots, N$ as a c^{th} patterns between N input patterns, whereby $W_j W_{kj}$ indicates the weight connecting amongst the i^{th} input and j^{th} hidden neurons, and the j^{th} hidden and k^{th} output neurons, correspondingly.

Output from the neuron in the input layer is

$$O_c = I_{ci}, i = 1, 2, \ldots I \tag{12}$$

Output from the neuron in the hidden layer is

$$0_{cj} = f\left(NET_{cj}\right) = f\left(\sum_{i=0}^{l} W_{ji} 0_{ci}\right), j = 1, 2, \ldots, m \tag{13}$$

Output from the neuron in the output layer is

$$0_{cj} = f\left(NET_{cj}\right) = f\left(\sum_{i=0}^{l} W_{ji} 0_{ci}\right), j = 1, 2, \ldots, m \tag{14}$$

where $f()$ indicates the sigmoid function provided by $(x) = 1/(1 + e^{-x})$.

3.2 Module II: Intrusion Detection

To recognize the intrusions accurately, the SBLSTM model is exploited here. The conventional architecture of LSTM successfully resolves the gradient disappearing problem and transports valuable data within the LSTM model [21]. Now, the Recurrent Neural Network (RNN) failed to capture long-term dependency among feature vectors. The LSTM cell encompasses of forget gate (f_t), input gate (i_t), and output gate (0_t). This gate controls a memory cell activation vector.

- Forget gate: this gate determines how much information from the previous layer c_{t-1} is forgotten or retained based on the present hidden layer h_{t-1} and input x_t. The output of forget gate ranges from zero and one. The forget gate is expressed as follows:

$$f_t = Sigmoid\ \left(w_{xf}x_t + w_{hf}h_{t-1} + b_f\right) \tag{15}$$

In Eq. (15), b_f, w_{xf}, and w_hf shows the bias vector, the weight matrices amongst x_t and f_t, and the weighted matrices amongst h_{t-1} and f_t, correspondingly

- Input gate: this gate determines the amount of network input x_t that should be retained in the present cell state c_t. The input gate is formulated as follows:

$$i_t = Sigmoid\ \left(w_{xi}x_t + w_{hi}h_{t-1} + b_i\right) \tag{16}$$

Now, b_f, w_{xi}, and w_{hi} correspondingly represent the bias vector, the weighted matrices amongst x_i and i_t, and the weight matrices amongst h_{t-1} and i_t.

- Output gate: this gate determines the number of data transported to the LSTM from the cell state c_t via the present output h_t. LSTM gate is a completely interconnected network where input represents a vector and output denotes a real number. The output gate can be formulated below.

$$0_t = Sigmoid\ \left(w_{xo}x_t + w_{ho}h_{t-1} + b_0\right) \tag{17}$$

whereas b_f, w_{xo}, and w_{ho} denotes the bias vectors, the weight matrices among x_t and 0_t, also the weight matrices among h_{t-1} and 0_t, correspondingly

The final output of the LSTM cell is output cell state c_t and output layer h_t, which is given as follows:

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c}_t \tag{18}$$

$$h_t = 0_t \odot tanh\left(C_t\right) \tag{19}$$

The intermediate input cell is represented by $\widetilde{c}_t$ and is expressed as follows:

$$\widetilde{c}_t = tanh\left(w_{xc}x_t + w_{hc}h_{t-1} + b_c\right) \tag{20}$$

Now b_f, w_{xc}, and w_{hc} specify the bias vector, the weight matrices amongst χ_i and $\widetilde{c}_t$, and the weight matrices amongst h_{t-1} and $\widetilde{c}_t$ correspondingly

The LSTM is exploited by fruit fly optimization for NN information for the analysis of time series. At the same time, LSTM was employed in fully diverse fields to predict the lifetime of equipment in the mechanical transmission system. A BiLSTM includes two different LSTMs that integrate data from two directions. Next, the data is accomplished as word annotation from the user information. The forward LSTM process the input left to right and evaluates the hidden layer $\vec{h}_t$ based on x_i and $\overrightarrow{h_{t-1}}$, while the backward LSTM process the input right to left and evaluates the hidden layer $\overleftarrow{h}_t$ based on x_t and $\overleftarrow{h_{t-1}}$. The forward and backward parameters in the BLSTM are mutually independent, while word embedding is distributed. Lastly, the hidden layer of BLSTM can be determined that concatenate the vector of the backward and forward directions during the t-time step.

$$h_t = \left[\vec{h}_t,\ \overleftarrow{h}_t\right] \tag{21}$$

Furthermore, the deep hierarchical technique enhances classification performance. On the other hand, the stacked BLSTM is effective in the shallow learning mechanism, thereby the proposed technique determines the stacking BLSTM to apply the local context and latent symmetry complex dataset. The output of the lower layer becomes the input of the upper layer in the Stacked BLSTM.

Finally, the Nadam optimizer is used for the hyperparameter tuning process. Nadam is called a variant of the weight update rule [22]. It can be calculated by expression of $\hat{a}_t$ and a_t as shown in Eq. (22).

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{u}_t + \varepsilon}} \left(\frac{\beta_1 a_{t-1}}{1 - \beta_1^t} + \frac{1 - \beta_1}{1 - \beta_1^t} d_t \right) \tag{22}$$

whereas $\frac{\beta_1 a_{t-1}}{1 - \beta_1^t}$ was a bias-corrected estimate of the momentum vector of the preceding time step. So, substitute it with $\hat{a}_{t-1}$

$$\Theta_{t+1} = \Theta_t - \frac{\eta}{\sqrt{\hat{u}_t + \varepsilon}} \left(\beta_1 \hat{a}_{t-1} + \frac{1 - \beta_1}{1 - \beta_1^t} d_t \right) \tag{23}$$

Thus, the Nadam update rule was presented by substituting the bias-corrected prediction of the previous momentum vector $\hat{a}$ with the bias-corrected prediction of the recent momentum vector $\hat{a}_{t-1}$ as given in Eq. (24).

$$\Theta_{t+1} = \Theta_t - \frac{\eta}{\sqrt{\hat{u}_t + \varepsilon}} \left(\beta_1 \hat{a}_t + \frac{1 - \beta_1}{1 - \beta_1^t} d_t \right) \tag{24}$$

4 Performance Validation

The experimental validation of the AODL-AICID method is tested through the UCM Land use data [23]. It is a 21-class land-use image data set meant for research purposes. There exist 100 images for all classes. Every image measures 256 × 256 pixels. The images are manually extracted from large imagery with the help of the USGS National Map Urban Area Imagery collection for various urban areas around the country. The pixel resolution of this public domain imager is 1 foot. Fig. 3 depicts the sample images from the UCM dataset.

Table 1 reports detailed classification outcomes of the AODL-AICID technique with other DL models. Fig. 4 investigates the classification results of the AODL-AICID model with existing models in terms of $accu_y$. The results demonstrated that the VGGNet and CAVGG-BiLSTM models have resulted in lower classification performance. At the same time, CAVGG-LSTM and RNet-50 models have reached somewhat improved and nearer performance. Along with that, the DLIRV2-AIC technique has gained reasonable outcomes with *an* $accu_y$ of 98%. In contrast, the AODL-AICID model highlighted its better performance on aerial image classification with an $accu_y$ of 98.60%.

Fig. 5 investigates the classification results of the AODL-AICID method with existing techniques in terms of $prec_n$, $reca_l$, and F_{score}. The results show that the VGGNet and CAVGG-BiLSTM methods have resulted in lower classification performance. Simultaneously, the CAVGG-LSTM and RNet-50 approaches have reached somewhat improved and nearer performance. Also, the DLIRV2-AIC method has obtained reasonable outcomes with $prec_n$, $reca_l$, and F_{score} of 96.94%, 98.27%, and 98.13%. In contrast, the AODL-AICID approach emphasized its better performance on aerial image classification with $prec_n$, $reca_l$, and F_{score} of 97.67%, 98.66%, and 98.52%.

To assure the enhanced performance of the AODL-AICID method, a detailed result analysis takes place on the CSE-CIC-IDS-2018 dataset [24]. In this case, only four types of attacks were considered.

Figure 3: Sample images from UCM dataset

Table 1: Classifier outcome of AODL-AICID approach with existing systems

Methods	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}
AODL-AICID	98.60	97.67	98.66	98.52
VGGNet	79.17	79.00	78.51	82.23
CAVGG-LSTM	80.78	79.55	80.68	81.95
CAVGG-BiLSTM	79.95	80.20	79.38	84.24
GNet	82.37	80.29	80.61	85.19
RNet-50	80.03	80.36	82.08	82.23
DLIRV2-AIC	98.00	96.94	98.27	98.13

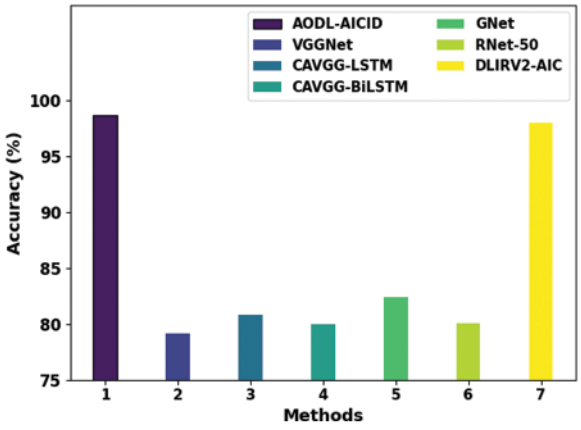

Figure 4: $Accu_y$ outcome of AODL-AICID approach with existing systems

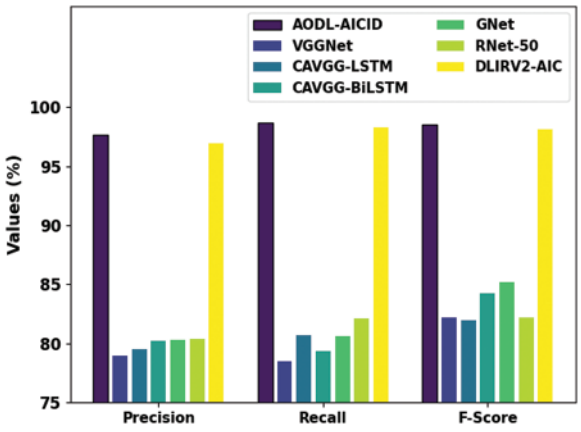

Figure 5: $Prec_n$, $reca_l$, and F_{score} outcome of AODL-AICID approach with existing systems

Table 2 and Fig. 6 investigate the attack detection results of the AODL-AICID model under botnet and DDoS attacks [5,25]. The outcomes show that the AODL-AICID technique has recognized both botnet and DDoS attacks proficiently. For instance, the AODL-AICID model has accomplished a higher $accu_y$ of 94.17% while the KNN, DT, GNB, SGD, and K-M techniques have reached lower $accu_y$ of 88.48%, 82.27%, 82.55%, 91.55%, and 86.13% respectively. Besides, the AODL-AICID technique has accomplished a higher $accu_y$ of 95.68% while the KNN, DT, GNB, SGD, and K-M approaches have reached lower $accu_y$ of 86.82%, 90.86%, 83.74%, 90.30%, and 91.45% correspondingly.

Table 2: Attack detection outcome of AODL-AICID approach under botnet and DDoS attacks

Methods	Accuracy rate	Precision	Recall	F-score	AUC score
		Botnet Attack			
AODL-AICID	94.17	95.15	95.21	96.00	96.40

(Continued)

Table 2 (continued)

Methods	Accuracy rate	Precision	Recall	F-score	AUC score
KNN	88.48	88.71	88.87	89.06	88.65
DT	82.27	82.94	82.43	82.89	82.76
GNB	82.55	82.73	82.74	82.96	84.07
SGD	91.55	90.83	90.12	90.53	91.26
K-M	86.13	85.57	86.49	86.68	86.67
DDoS Attack					
AODL-AICID	95.68	94.54	93.69	92.80	93.78
KNN	86.82	87.69	88.07	88.26	89.14
DT	90.86	90.18	90.91	91.07	90.57
GNB	83.74	84.19	83.66	83.70	84.58
SGD	90.30	89.94	89.81	89.90	90.12
K-M	91.45	92.64	92.45	91.99	92.85

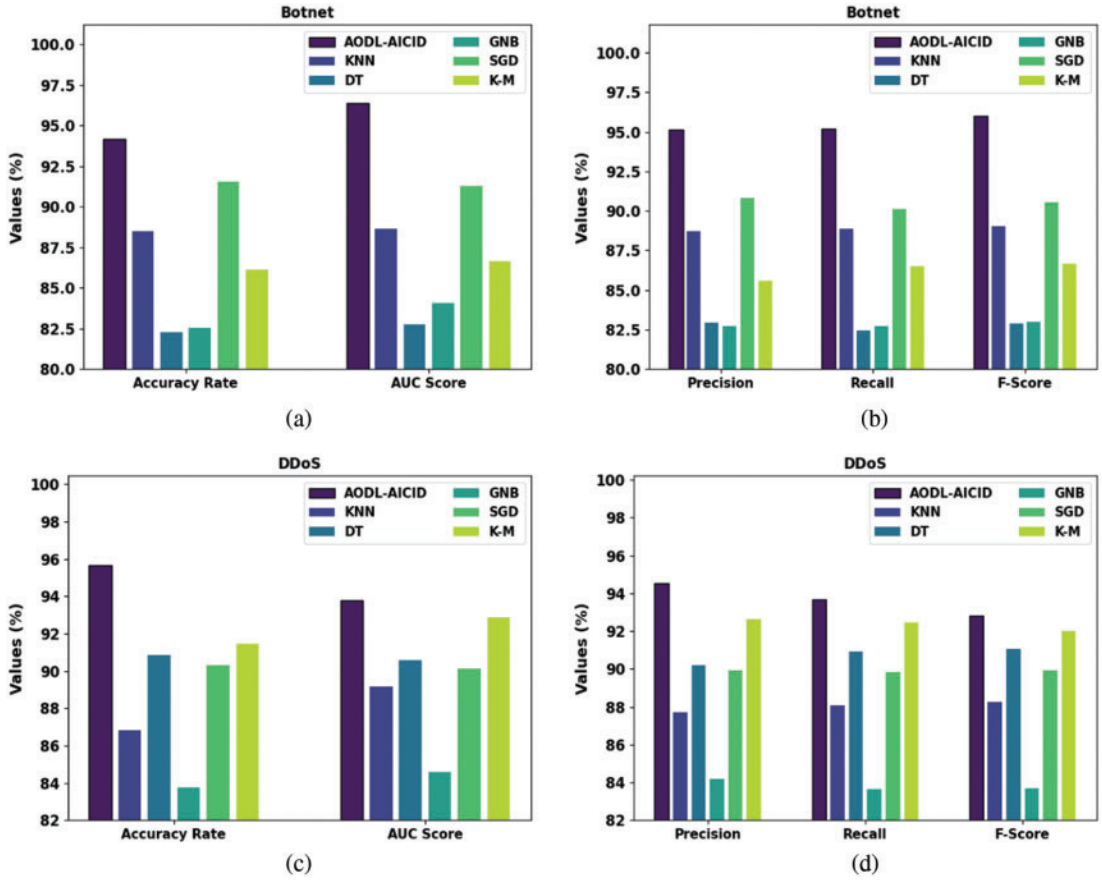

Figure 6: Attack detection of AODL-AICID approach (a,b) Botnet attacks (c,d) DDoS attack

Table 3 and Fig. 7 investigate the attack detection results of the AODL-AICID approach under web and Infiltration attacks. The results specified that the AODL-AICID method has recognized both botnet and DDoS attacks proficiently. For example, the AODL-AICID model has accomplished a higher $accu_y$ of 94.38% while the KNN, DT, GNB, SGD, and K-M techniques have reached lower $accu_y$ of 85.29%, 84.28%, 79.44%, 92.05%, and 90.10% correspondingly. Also, the AODL-AICID method has executed a higher $accu_y$ of 96.57% while the KNN, DT, GNB, SGD, and K-M approaches have gained lower $accu_y$ of 94.47%, 90.70%, 88.75%, 92.19%, and 88.39% correspondingly.

These results ensured the effectual performance of the AODL-AICID model on the UAV networks.

Table 3: Attack detection outcome of AODL-AICID approach under web and Infiltration attacks

Methods	Accuracy rate	Precision	Recall	F-score	AUC score
Web Attack					
AODL-AICID	94.38	93.24	96.56	94.16	95.08
KNN	85.29	85.77	85.58	86.50	87.22
DT	84.28	84.49	83.90	83.62	84.38
GNB	79.44	79.63	79.56	80.64	80.33
SGD	92.05	92.63	93.58	93.73	94.57
K-M	90.10	90.73	90.24	89.58	89.78
Infiltration Attack					
AODL-AICID	96.57	95.57	95.81	95.07	94.92
KNN	94.47	95.27	94.68	94.34	94.63
DT	90.70	91.53	90.92	90.76	91.60
GNB	88.75	87.99	88.44	87.77	88.25
SGD	92.19	91.77	91.85	91.27	90.70
K-M	88.39	89.04	88.63	89.58	89.22

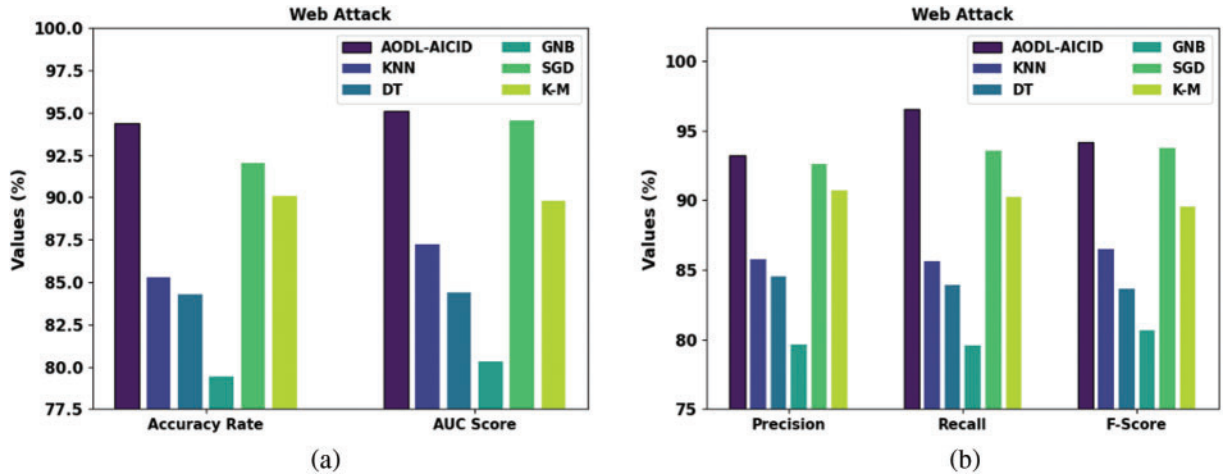

Figure 7: (Continued)

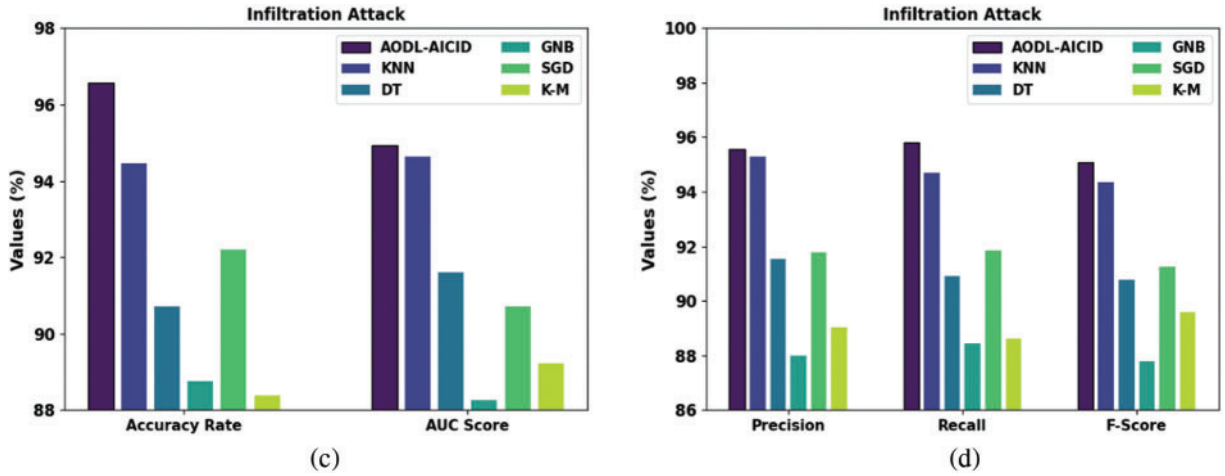

Figure 7: Attack detection of AODL-AICID approach (a,b) web attacks (c,d) infiltration attack

5 Conclusion

A new AODL-AICID technique has been introduced for aerial image classification and intrusion detection in UAV networks. The presented AODL-AICID technique concentrates on two major processes: image classification and intrusion detection. For aerial image classification, the AODL-AICID technique encompasses MobileNetv2 feature extraction, an AOA-based hyperparameter optimizer, and a BPNN-based classifier. In addition, the AODL-AICID technique employed the Nadam optimizer with the SBLSTM method for intrusion detection in the UAV networks. The result analysis of the AODL-AICID technique is tested under different performance measures. In future, deep learning-based ensemble fusion models can boost the performance of the AODL-AICID approach.

Funding Statement: This research work was funded by Institutional Fund Projects under Grant No. (IFPIP: 511-611-1443). Therefore, the authors gratefully acknowledge technical and financial support provided by the Ministry of Education and Deanship of Scientific Research (DSR), King Abdulaziz University (KAU), Jeddah, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Ouiazzane, M. Addou and F. Barramou, "A multiagent and machine learning based denial of service intrusion detection system for drone networks," in *Geospatial Intelligence, Advances in Science, Technology & Innovation Book Series*. Cham: Springer, pp. 51–65, 2022.

[2] Y. Miao, Y. Tang, B. A. Alzahrani, A. Barnawi, T. Alafif *et al.,* "Airborne LiDAR assisted obstacle recognition and intrusion detection towards unmanned aerial vehicle: Architecture, modeling and evaluation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4531–4540, 2020.

[3] E. H. Abualsauod, "A hybrid blockchain method in internet of things for privacy and security in unmanned aerial vehicles network," *Computers and Electrical Engineering*, vol. 99, pp. 107847, 2022.

[4] J. Whelan, A. Almehmadi and K. El-Khatib, "Artificial intelligence for intrusion detection systems in unmanned aerial vehicles," *Computers and Electrical Engineering*, vol. 99, pp. 107784, 2022.

[5] M. Ragab and A. Addas, "Low complexity encoder with multilabel classification and image captioning model," *CMC-Computers Materials & Continua*, vol. 72, no. 3, pp. 4323–4337, 2022.

[6] Q. Abu Al-Haija and A. Al Badawi, "High-performance intrusion detection system for networked UAVs via deep learning," *Neural Computing and Applications*, vol. 34, no. 3, pp. 1–16, 2022.

[7] G. Choudhary, V. Sharma, I. You, K. Yim, R. Chen *et al.,* "Intrusion detection systems for networked unmanned aerial vehicles: A survey," in *14th Int. Wireless Communications & Mobile Computing Conf. (IWCMC)*, Limassol, Cyprus, pp. 560–565, 2018.

[8] S. Ouiazzane, F. BarramoU and M. Addou, "Towards a multi-agent based network intrusion detection system for a fleet of drones," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 10, pp. 351–362, 2020.

[9] M. Ragab and M. F. S. Sabir, "Outlier detection with optimal hybrid deep learning enabled intrusion detection system for ubiquitous and smart environment," *Sustainable Energy Technologies and Assessments*, vol. 52, pp. 02311, 2022.

[10] M. Ragab, E. Ashary, W. Aljedaibi, I. Alzahrani, A. Kumar *et al.,* "A novel metaheuristics with adaptive neuro-fuzzy inference system for decision making on autonomous unmanned aerial vehicle systems," *ISA Transactions*, vol. 132, pp. 16–23, 2023.

[11] X. He, Q. Chen, L. Tang, W. Wang and T. Liu, "CGAN-based collaborative intrusion detection for uav networks: A blockchain empowered distributed federated learning approach," *IEEE Internet of Things Journal*, vol. 10, no. 1, pp. 120–132, 2023.

[12] K. A. Alissa, S. S. Alotaibi, F. S. Alrayes, M. Aljebreen, S. Alazwari *et al.,* "Crystal structure optimization with deep-autoencoder-based intrusion detection for secure internet of drones environment," *Drones*, vol. 6, no. 10, pp. 297, 2022.

[13] X. Tan, S. Su, Z. Zuo, X. Guo and X. Sun, "Intrusion detection of UAVs based on the deep belief network optimized by PSO," *Sensors*, vol. 19, no. 24, pp. 5529, 2019.

[14] H. Sedjelmaci, A. Boudguiga, I. B. Jemaa and S. M. Senouci, "An efficient cyber defense framework for UAV-edge computing network," *Ad Hoc Networks*, vol. 94, pp. 101970, 2019.

[15] R. Zhang, J. P. Condomines, N. Larrieu and R. Chemali, "Design of a novel network intrusion detection system for drone communications," in *IEEE/AIAA 37th Digital Avionics Systems Conf. (DASC)*, London, UK, pp. 1–10, 2018.

[16] J. Whelan, T. Sangarapillai, O. Minawi, A. Almehmadi and K. El-Khatib, "Novelty-based intrusion detection of sensor attacks on unmanned aerial vehicles," in *Proc. of the 16th ACM Symp. on QoS and Security for Wireless and Mobile Networks*, Alicante, Spain, pp. 23–28, 2020.

[17] H. Huang, L. Liang, G. Zhao, Y. Yang and K. Ou, "Railway clearance intrusion detection in aerial video based on convolutional neural network," in *Chinese Control and Decision Conf. (CCDC)*, Nanchang, China, pp. 1644–1648, 2019.

[18] B. Lin, H. Su, D. Li, A. Feng, H. Li *et al.,* "PlaneNet: An efficient local feature extraction network," *PeerJ Computer Science*, vol. 7, pp. e783, 2021.

[19] F. A. Hashim, K. Hussain, E. H. Houssein, M. S. Mabrouk and W. Al-Atabany, "Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems," *Applied Intelligence*, vol. 51, no. 3, pp. 1531–1551, 2021.

[20] N. T. Long and L. H. Chuong, "A back propagation neural network model with the synthetic minority over-sampling technique for construction company bankruptcy prediction," *International Journal of Sustainable Construction Engineering and Technology*, vol. 13, no. 3, pp. 68–79, 2022.

[21] X. Liu, S. Liu, X. Li, B. Zhang, C. Yue *et al.,* "Intelligent tool wear monitoring based on parallel residual and stacked bidirectional long short-term memory network," *Journal of Manufacturing Systems*, vol. 60, pp. 608–619, 2021.

[22] S. Bera and V. K. Shrivastava, "Analysis of various optimizers on deep convolutional neural network model in the application of hyperspectral remote sensing image classification," *International Journal of Remote Sensing*, vol. 41, no. 7, pp. 2664–2683, 2020.

[23] http://weegee.vision.ucmerced.edu/datasets/landuse.html

[24] I. Sharafaldin, A. H. Lashkari and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. of the 4th Int. Conf. on Information Systems Security and Privacy*, Funchal, Madeira, Portugal, pp. 108–116, 2018.

[25] M. S. Minu and R. A. Canessane, "Deep learning-based aerial image classification model using inception with residual network and multilayer perceptron," *Microprocessors and Microsystems*, vol. 95, pp. 104652, 2022.