Mathematics

AIMS Mathematics, 9(4): 10235–10254.
DOI: 10.3934/math.2024500
Received: 15 November 2023
Revised: 24 February 2024
Accepted: 27 February 2024
Published: 14 March 2024

http://www.aimspress.com/journal/Math

Research article

Advancements in remote sensing: Harnessing the power of artificial intelligence for scene image classification

Alaa O. Khadidos[1,2,*]

[1] Associate Professor, Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

[2] Center of Research Excellence in Artificial Intelligence and Data Science, King Abdulaziz University, Jeddah, Saudi Arabia

* **Correspondence:** Email: aokhadidos@kau.edu.sa.

Abstract: The Remote Sensing Scene Image Classification (RSSIC) procedure is involved in the categorization of the Remote Sensing Images (RSI) into sets of semantic classes depending upon the content and this procedure plays a vital role in extensive range of applications, like environment monitoring, urban planning, vegetation mapping, natural hazards' detection and geospatial object detection. The RSSIC procedure exploits Artificial Intelligence (AI) technology, mostly Machine Learning (ML) techniques, for automatic analysis and categorization of the content, present in these images. The purpose is to recognize and differentiate the land cover classes or features in the scene, namely crops, forests, buildings, water bodies, roads, and other natural and man-made structures. RSSIC, using Deep Learning (DL) techniques, has attracted a considerable attention and accomplished important breakthroughs, thanks to the great feature learning abilities of the Deep Neural Networks (DNNs). In this aspect, the current study presents the White Shark Optimizer with DL-driven RSSIC (WSODL-RSSIC) technique. The presented WSODL-RSSIC technique mainly focuses on detection and classification of the remote sensing images under various class labels. In the WSODL-RSSIC technique, the deep Convolutional Neural Network (CNN)-based ShuffleNet model is used to produce the feature vectors. Moreover, the Deep Multilayer Neural network (DMN) classifiers are utilized for recognition and classification of the remote sensing images. Furthermore, the WSO technique is used to optimally adjust the hyperparameters of the DMN classifier. The presented WSODL-RSSIC method was simulated for validation using the remote-sensing image databases. The experimental outcomes infer that the WSODL-RSSIC model achieved improved results in comparison with the current approaches under different evaluation metrics.

Keywords: remote sensing images; artificial intelligence; white shark optimizer; scene classification; deep learning
Mathematics Subject Classification: 68M11, 68M25, 68T07, 68W1

1. Introduction

Remote Sensing Image (RSI) analysis corresponds to the research and understanding of the surface semantic contents. In the recent years, a lot of RSIs have been obtained easily, especially with high-quality clarity. This phenomenon has promoted the growth of various research domains like automatic target recognition, RS Scene Image Classification (RSSIC) and geographic image retrieval [1]. In this important topic, the RSI classification process utilizes a computer to examine the ground objects in the RSI, choose the features, and label the class of the presented images [2]. Unlike the normal images, the RSIs are harder to process. For example, an RSI comprises of all types of objects that differ in position, scale, and tint [3]. Further, these objects have large interclass and intra-class differences as a result of intervention of the external factors during the RSI collection process.

Image classification refers to a step-wise process that begins with devising a method for the classification of the desirable images [4]. Then, the images are pre-processed using image enhancement, image clustering, and scaling processes. Afterwards, a desirable area of the image is selected and the initial clusters are made. Later, the method is adopted upon the images to receive the desirable classification and corrective measures and is termed as post-processing [5]. The current study focuses on the application of the deep learning models and mid-level features to construct the decision support mechanisms for RSI, smart vehicles, and so on. To acquire the terrestrial data on large scales, the RS images serve an important role while effectual land use can be attained over Earth aerial images [6].

In recent times, the growth of Deep Learning (DL) techniques is rapid in big data analytics. It has been successfully and broadly implemented in different domains like speech enhancement, Natural Language Processing (NLP), and image classification due to its exceptional performance than the conventional learning methods [7]. The inspiration behind this technique is the application of primary visual system by the human brain to capture, process, organize and gain insights out of the images captured at various levels. The DL structures are considered to be Artificial Neural Networks (ANN), typically with multiple layers [8]. The Deep Neural Networks (DNNs) use feature representations, learned from the data with their shallow counterpart. Still, these techniques do not require handcrafted features that are frequently devised, thanks to the domain knowledge [9]. This feature prevents the issue i.e., the handcrafted features have to rely upon domain-specific knowledge. In addition to these, it is impractical to produce a solution by taking all the details embedded in each type of real data through predesigned handcrafted features [10]. Instead of depending on shallow and manually-engineered features, the DL methods can routinely learn a representation of raw input datasets that possess many extraction levels.

The pursuit of optimum performance demands the relentless exploration of the advanced methodologies in the dynamic realm of remote sensing scene image classification. Hyperparameter tuning process in DL techniques is the beacon that illuminates this path and it also provides a transformative technique for enhancing the robustness and accuracy of the classification algorithms. Through meticulous adjustment of the hyperparameters, including regularization terms and learning rates, the hidden potentials of the neural networks are unlocked. This outcome enables the researchers to decipher intricate patterns within the RS scene. Such a relentless pursuit of optimization refines the prediction abilities of a model and empowers it to handle the complexity that is inherent in dynamic and diverse environmental datasets. In embracing the hyperparameter-tuned DL techniques, both practitioners as well as researchers are boosted by an unwavering commitment to push the boundaries of accuracy. This push finally contributes to a highly insightful and impactful understanding of the ever-evolving Earth from above.

The current study introduces the White Shark Optimizer with DL-driven Remote Sensing Scene Image Classification (WSODL-RSSIC) technique. The presented WSODL-RSSIC technique involves the deep convolutional neural network-based ShuffleNet model to produce the feature vectors. Moreover, a Deep Multilayer Neural Network (DMN) classifier is also used for both recognition and classification of the RSIs. Furthermore, the WSO technique is exploited to optimally adjust the hyperparameters of the DMN classifier. The integration of the WSO technique, as a hyperparameter tuning algorithm, establishes a new dimension in the synergy between optimization techniques and deep learning algorithms and it offers a promising avenue for advancing the RSI classification process. The presented WSODL-RSSIC model was simulated upon the remote-sensing image databases for validation and the results were obtained. The key contributions of the current study are summarized herewith.

- An intelligent WSODL-RSSIC technique has been proposed for remote sensing image classification and it comprises of pre-processing, ShuffleNet feature extraction, DMN classification, and WSO-based hyperparameter tuning. To the best of the researchers' knowledge, the WSODL-RSSIC technique has never been presented in the literature.
- A deep convolutional neural network has been applied based on the ShuffleNet model for extracting the feature vectors from the RS scenes. This improves the model's capability in terms of efficiently capturing the intricate spatial patterns.
- DMN has been introduced as a strong classifier and it contributes to the model's capability to accurately detect and classify a complex scene content, thus enhancing the overall discriminatory power.
- The WSO technique has been proposed to adjust the hyperparameters of the DMN classifier. This addition presents a new dimension to the study by leveraging the WSO technique for finetuning the regularization terms and learning rate, improving the performance and adaptability of the DL algorithm.

2. Related works

Sun and Zheng [11] presented a DCNN (Deep Convolutional Neural Network) based on the PSPNet and HRNet techniques for segmenting and realizing deep scene analyses and enhancing the pixel-level semantic segmentation representations of the high-resolution RSI. In this method, distinct feature vectors were exploited to fulfil the requirements, utilizing the image classification network structure. Both PSPNet and HRNet methods were utilized to examine the scene and acquire class labels for every pixel in the image. Sun et al. [12] devised a novel CNN-based network using local and global encoders for abstracting the discriminative local and global attributes for RS scene classification.

In literature [13], the authors devised the SS-RCSN (semi-supervised representation consistency Siamese network) method for RSI image scene classification. Considering the interclass similarity and intraclass diversity of the RSIs, the Involution-generative adversarial network (GAN) was exploited initially to abstract the discriminatory attributes from the RSIs through unsupervised learning methods.

Then, a Siamese network was devised for semi-supervised classification with a representation of consistency loss that targets to minimize the differences between the unlabeled and labeled data. Xu et al. [14] introduced an enhanced classification approach, involving RF with recurrent neural network (RNN) for land classification by means of satellite images that are openly available for different research purposes. The author adopted the spatial information obtained from these satellite imageries.

Cheng and Lei [15] presented a novel classification method utilizing the combined CNN–HMM method with a stacked ensemble system. A modified multi-scale CNN was devised at first to abstract the multi-scale structural attributes that possess a lightweight framework and can evade high computing complexities. Then, the author applied the HMM (hidden Markov model) to derive the context data of the mined features for the sample images. Recently, Hilal et al. [16] a DTL-related fusion method for RSI classification, named the DTLF-ERSIC method. The presented method included the entropy-related mixture of three feature extraction approaches, namely the Efficient Net, Discrete LBP (DLBP), and ResNet50 methods. Likewise, the fuzzy rule-based classifier (FRC) was implemented with rain optimization algorithm (ROA) for forecasting the class labels of the test RS imagery.

Zhang et al. [17] devised a multi-scale attention network (MSA-Network) by integrating the Channel and Position Attention (CPA) and multiscale (MS) modules to increase the efficacy of the RSI classification process. The presented module learnt multi-scale features by implementing different dimensions of the sliding windows from various receptive fields and depth layers. In literature [18], the authors tried to merge the lie group ML and CNN methods to extract a high number of features that are effective and holds differentiating capability. The study devised a new network approach named the Lie Group Regional Influence Network (LGRIN). At first, multiple space samples of the LGRIN were gained by mapping after which their attributes were derived after the operation of integral image calculation and image decomposition. Then, multi-dilation pooling was included in the CNN structure.

In literature [19], a multi-stage self-guided separation network (MGSNet) was presented for RS scene classification. Unlike the preceding work, it made use of the background data, outside the effectual target in the image. This was accomplished based on the support from a target–background separation approach that aimed to improve the distinguishability among the target similarity-background difference instances. Ragab [20] established the mayfly optimizer with a DL-based robust RSSIC (MFODL-RRSSIC) approach. The two key objectives of the projected MFODL-RRSSIC approach were scene classification and security. In the study conducted earlier [21], the foreground–background contrastive learning (FBCL) model was presented for a few-shot RSI scene classification. Huang et al. [22] examined the Evidential Combination model with Multi-color Spaces (ECMS) model to combine the complementary data of distinct color spaces for the classification process. In this ECMS model, the labeled RSIs in the RGB color space can first be changed into other color spaces and employed for CNN training approaches, correspondingly. Zhao et al. [23] presented the novel effectual multi-sample contrastive network (EMSCNet) to combine the information from multiple instances. To be specific, the authors made a dynamic dictionary with momentum upgrades to mine the negative and positive pairs from the entire database.

3. The proposed model

The current study focuses on design and development of an efficient and automated remote sensing image classification method termed as WSODL-RSSIC. The presented WSODL-RSSIC technique focuses on the effectual recognition and classification of the RSIs under distinct class labels. Figure 1 represents the workflow of the WSODL-RSSIC model. The figure shows that the WSODL-RSSIC model involves three stages, such as the ShuffleNet feature extraction, DMN classification, and WSO-based hyperparameter tuning.

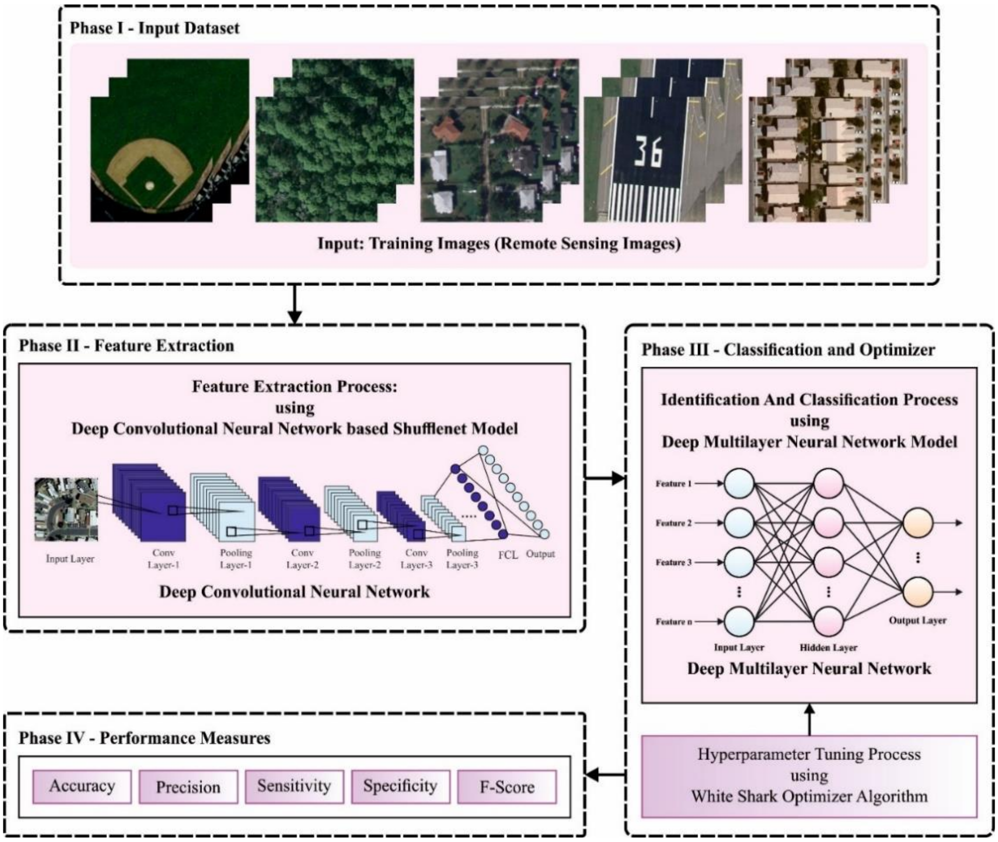
Phase I - Input Dataset
Input: Training Images (Remote Sensing Images)
Phase II - Feature Extraction
Feature Extraction Process:
using
Deep Convolutional Neural Network based Shufflenet Model
Input Layer
Conv Layer-1
Pooling Layer-1
Conv Layer-2
Pooling Layer-2
Conv Layer-3
Pooling Layer-3
FCL
Output
Deep Convolutional Neural Network
Phase III - Classification and Optimizer
Identification And Classification Process
using
Deep Multilayer Neural Network Model
Feature 1
Feature 2
Feature 3
Feature n
Input Layer
Hidden Layer
Output Layer
Deep Multilayer Neural Network
Hyperparameter Tuning Process
using
White Shark Optimizer Algorithm
Phase IV - Performance Measures
Accuracy
Precision
Sensitivity
Specificity
F-Score

Figure 1. Workflow of the WSODL-RSSIC approach.

3.1. Feature extraction using the *ShuffleNet model*

To originate an effectual set of feature vectors, the ShuffleNet model is used in this study. Deep Learning is a famous ML approach that has been extensively analyzed and applied in the recent years. It is a multi-layered approach that is utilized for both extracting and defining the features in huge volumes of data [24]. It comprises of distinct layers like pooling, convolutional, fully connected (FC), activation, and flattened with multiple tasks.

Convolution layer: It is an efficient layer that is utilized in the extraction of the features from input data. An input vector is scanned with determined filtering and the data is converted into a feature space with nearby weight sum aggregation. The first convolutional layer is directly linked to a set of images, and low-level extraction features like edges and colors are obtained.

Activation Layer (Nonlinearity layer): The activation layer is a layer in which the non-linear function is executed upon all the pixels of the image. Recently, the ReLu activation function has been frequently utilized compared to the sigmoid and hyperbolic tangent activation functions.

Pooling (downsampling) layer: This layer reduces the number of parameters and the calculations from the network, thus providing dual benefits. The primary benefit is the reduction of computation count for the following layer whereas the second is restraining the network in learning. Sum, maximum, mean, and average pooling are the processes that are generally utilized in this layer.

Flatten layer: This layer is used for preparing the input dataset for the final layer. Since the NNs consider the input data as a 1-D array, the matrix-type data in other layers too is changed into 1-D arrays. However, all the pixels of the images are defined by a single line and this procedure is termed as ‘smoothing’.

FC layers: This layer depends on every field of the preceding layer. The number of this layer tend to vary in different structures. Here, the features can be retained while the learning procedure is executed by altering the weighted and biased values. This layer is responsible for performing the actual image processing by taking the input in distinct extraction feature steps and examining the outcomes of every processing layer.

Recently, the researchers have developed a high-performance lightweight CNN ShuffleNet model. In general, a point group convolution can be used to optimize the efficacy of the convolution process while the channel shuffle function can realize data interchange between dissimilar channels that assist in encoding further data. The ShuffleNet model considerably reduces the computation cost, accomplishes a remarkable performance and ensures high computation accuracy in comparison with the rest of the network models [25]. Indeed, grouped convolution has been utilized in the AlexNet model and some effective neural network models like MobileNetv251 and Xception50 have developed depthwise convolution based on grouped convolution. Even though the amount of computation and the capability of the model are coordinated, the point-wise convolution computed in this model occupies a larger part; hence, the pixel-level group convolution is proposed for the ShuffleNet model by reducing the convolution to 1×1 size. However, the convolution operation is constrained by all the group-wise convolutions, which in turn reduces the computation difficulty. However, once the group convolution is stacked, the feature data of the output channel comes only from the smaller part of the input channel, where it is positioned. The output is related only to the input in the group whereas the data of other groups cannot be attained. Both input and output channels of the ShuffleNet model are set to a similar number so as to minimize the memory usage. Assume that the convolution of height and width is 1×1, the amount of input and output channels are $C1$ and $C2$ correspondingly, and the size of the feature map is $h \times w$.

3.2. Modeling of DMN-based scene classification

For an accurate and automated scene classification process, the DMN classifier is utilized in this study. The DMN model is efficient and popular thanks to its desirable accuracy that comes from many distinct characteristics like self-adaptive data-driven, universal approximation, and flexible nonlinear modeling processes [26]. In image classification domain, the researchers are continuously exploring for new techniques to optimize the performance, though the DMNs have been widely used these days for its high accuracy. To enhance the performance, the current study uses a distance-based cost function during the learning procedure. Likewise, based on the discrete learning-based model, many new techniques have been introduced using a deep multilayer MNN model.

The following equation shows the variable m for the DMN binary classification algorithm that involves the parameters, $y \in \{-1, +1\}$. M corresponds to the explanatory variable in X_l,

$X_2, ..., X_M \in \mathfrak{R}$, and d corresponds to the hidden layer (HLs).

$$Y_t = f\left(\sum_{i_d=0}^{H_d} \beta_{i_d}^{d} \cdot g^d \left(\sum_{i_{d-1}=0}^{H_{d-1}} \beta_{i_{d-1},i_d}^{d-l} \cdot g^{d-I} \left(\sum_{i_{d-2}=0}^{H_{d-2}} \beta_{i_{d-2},i_{d-1}}^{d-2} \dots \dots g^2 \times \left(\sum_{i_1=0}^{H_1} \beta_{i_1,i_2}^{I} \cdot g^I\left(\sum_{i_0=0}^{M} \beta_{i_0,i_1}^{0} . X_{t,i_0}\right)\right)\right)\right)\right) \quad t = I, 2, 3, \dots, N. \tag{1}$$

In Eq (1), m shows the amount of input nodes and N indicates the sample size. $\beta_{i_d}^{d}$ denotes the weight connection of the i_d^{th} neuron in d^{th} HL to the output neuron, H^d denotes the number of hidden nodes in the d^{th} HL, $\beta_{i_{d-1},i_d}^{d-1}$ indicates the weight connection of the i_{d-I}^{th} neuron in $d-1^{th}$ HL's i_d^{th} neuron in the d^{th} HL, and g^d and f are the corresponding d^{th} HL and output layer activation function.

The common learning methods to estimate the unknown parameters and weights of the DMN model use gradient descent optimization in which the sum of the classification errors can be minimized. The presented model is based on a continuous form though the classification function is discrete. Thus, a discrete learning-based model is implemented and a deep MNN is proposed in this study to produce a potential learning process. The primary objective is to increase a discrete matching function of fitted and actual values. Therefore, the procedure for evaluating the biases and the unknown weights of the DMN model, during the discrete learning-based method, is shown below.

$$\text{Max} \sum_{t=1}^{N} Match\ (y_r, \hat{y}). \tag{2}$$

In Eq (2), $Match(y_r, \hat{y})$ denotes the matching function of $(y_\zeta \in \{-1, +1\})$, and fitted $(\hat{y} \in \mathfrak{R})$ at time t is given in the binary form below.

$$Match(y_t, \hat{y}_\text{t}) = \begin{cases} +1 & if\ (y_t)(\hat{y}_\text{t}^\text{s}) \geq 0, \\ -1 & if\ (y_t)(\hat{y}_\text{t}^\text{s}) < 0. \end{cases} \tag{3}$$

Where $\hat{y} = srcl.\ (\hat{y})$ indicates the normalized value of $\hat{y}$ at time t that is evaluated by Eq (4):

$$\hat{y}_\text{t}^\text{s} = \frac{\hat{y} - \bar{\hat{y}}}{\hat{y}^{\max} - \hat{y}^{Min}}. \tag{4}$$

In Eq (4), $\bar{\hat{y}}, \hat{y}$, and $\hat{y}^{Max}$ denote the mean, maximum and minimum values of $\hat{y}$. Here, Eq (2) is modified by the Sign function:

$$Max \sum_{t=1}^{N} \text{Sign}\ [(y_t)(\hat{y}_t^s)]. \tag{5}$$

This is further simplified as follows

$$Max \sum_{t=1}^{N} \text{Sign}(y_t) \cdot Sign(\hat{y}_t^s)]. \tag{6}$$

Thus, we have

$$Max \sum_{t=1}^{N} (y_t) \cdot Sign\left(std.\left(f\left(\sum_{i_d=0}^{H_d} \beta_{i_d}^{d} \cdot g^d \left(\sum_{i_{d-1}=0}^{H_{d-1}} \beta_{i_{d-1},i_d}^{d-l} \cdot g^{d-I} \left(\sum_{i_{d-2}=0}^{H_{d-2}} \beta_{i_{d-2},i_{d-1}}^{d-2} \dots \dots g^2 \times \left(\sum_{i_1=0}^{H_1} \beta_{i_1,i_2}^{I} \cdot g^I\left(\sum_{i_0=0}^{M} \beta_{i_0,i_1}^{0} . X_{t,i_0} \right) \right) \right) \right) \right) \right) \right). \tag{7}$$

3.3. Hyperparameter tuning using the *WSO algorithm*

Lastly, the WSO technique is applied for hyperparameter tuning of the DMN classification algorithm. The WSO algorithm is inspired from the dynamic behavior of great white sharks that have excellent smell and hearing senses during foraging and navigating for food [27]. White shark is a magnificent and highly-adapted hunter with strong muscles and hunts seals, shellfish, small whales, dolphins, and seabirds, which characterize the prey. The hunting approach for great white sharks, while catching the prey, begins with rushing the prey via surprise strategies during which an enormous fatal strike is produced. Figure 2 demonstrates the flowchart of the WSO model.

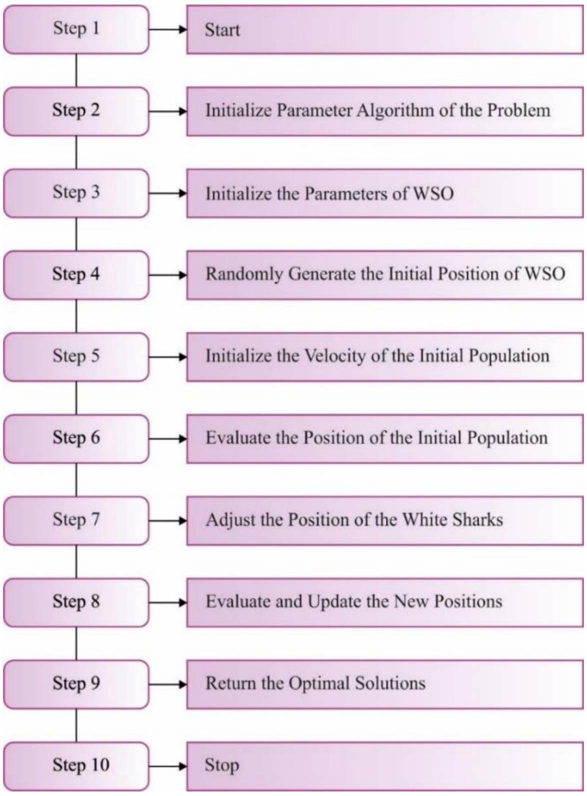
Step 1
Start
Step 2
Initialize Parameter Algorithm of the Problem
Step 3
Initialize the Parameters of WSO
Step 4
Randomly Generate the Initial Position of WSO
Step 5
Initialize the Velocity of the Initial Population
Step 6
Evaluate the Position of the Initial Population
Step 7
Adjust the Position of the White Sharks
Step 8
Evaluate and Update the New Positions
Step 9
Return the Optimal Solutions
Step 10
Stop

Figure 2. Steps involved in the WSO algorithm.

Devouring the food source (prey) involves three different behaviors as given herewith i.e., the movement towards the target using its wave hesitancy generated by the prey's motion, random searching for the prey in the ocean, and finding an adjacent prey. These steps might assist the great white shark to update its position and reach a better solution. The WSO is modeled by initializing the population matrix sized $N \times d$in which d denotes the problem dimension and N corresponds to the population size:

$$w = \begin{bmatrix} w_1^1 & w_2^1 & \dots & w_d^1 \\ w_1^2 & w_2^2 & \dots & w_d^2 \\ \vdots & \vdots & \vdots & \vdots \\ w_1^n & w_2^n & \dots & w_d^n \end{bmatrix}. \tag{8}$$

In Eq (8), w_j^i signifies the i^{th} white shark's location in j^{th} dimension. It is evaluated by using the upper (ub) and lower (lb) limits of the search space at the j^{th} dimension as given below

$$w_j^i = lb_j + rand \times \left(ub_j - lb_j\right). \tag{9}$$

In Eq (9), $rand$ is a randomly generated value in the range of $[0\, and\, 1]$ interval. The preliminary fitness value is evaluated for the initial solution provided in Eq (8). Later, the updating process is positioned, if the new location is better than the previous one. The great white shark observes the prey's position using its wave pattern and the hesitation. Then, it approaches the prey while undulating the movement with velocity as given below.

$$v_{k+1}^i = \mu\left(v_k^i + p_1\left[w_{gbest} - w_k^i\right] \times c_1 + p_2\left[w_{best}^{v_k^i} - w_k^i\right] \times c_2\right). \tag{10}$$

In Eq (10), w_k^i represents the location of the i^{th} white shark at the k^{th} iteration; c_1 and c_2 show randomly generated numbers in the range of $[0,1]$; v_{k+1}^i and v_k^i denote the updated and existing velocities of the ith white shark at the $k+1$ and k iterations, correspondingly; w_{gbest_k} indicates the global optimal position at the k^{th} iteration; v_k^i denotes the index vector number i, $w_{best}^{v_k^i}$ shows the ith better-known position to swarm at the k^{th} iteration and is described by Eq (11):

$$v = [n \times rand(1, n)] + 1. \tag{11}$$

Both p_1 and p_2 parameters denote the forces of the great white sharks which control the w_{gbesi_k} and $w_{best}^{v_k^i}$ effects on w_k^i, and they are calculated using the Eqs (12) and (13):

$$p_1 = p_{\max} + (p_{\max} - p_{\min}) \times e^{-(4k/k)^2}, \tag{12}$$

$$p_2 = p_{\min} + (p_{\max} - p_{\min}) \times e^{-(4k/k)^2}. \tag{13}$$

Here $\mathrm{p}_{\min}$ and $\mathrm{p}_{\max}$ denote the primary and secondary velocities to attain the best movement for a great white shark while the respective values are $\mathrm{p}_{\max} = 1.5$ and $\mathrm{p}_{\min} = 0.5$. K signifies the maximal iteration. The term μ represents the correction factor that is utilized for analyzing the convergence speed of WSO using the formula given below.

$$\mu = \frac{2}{|2 - t - \sqrt{t^2 - 4t}|}. \tag{14}$$

In Eq (14), t denotes the acceleration factor.

As mentioned earlier, the great white shark spends time searching for its prey. Subsequently, it changes the position either when they approach the prey by smelling their prey's scents or by hearing the waves generated by the prey's movement. In such situations, the great white shark moves toward a random position looking for the prey as follows.

$$w_{k+1}^i = \begin{cases} w_k^i \bullet \neg \oplus w_0 + ub \bullet a + lb \bullet b & if\ r\ and < mv, \\ w_k^i + \frac{v_k^i}{f} & if\ r\ and \geq mv'. \end{cases} \tag{15}$$

In Eq (15), $\neg$ shows the operator of negation, f indicates the frequency of wavy motion, a and

b denote the binary vectors and w_o characterizes the logical vector which is computed by using the following equation.

$$a = sgn\left(w_k^i - ub\right) > 0, \tag{16}$$

$$b = sgn\left(w_k^i - lb\right) < 0, \tag{17}$$

$$w_o = \oplus\ (a, b), \tag{18}$$

$$f = f_{\min} + \frac{f_{\max} - f_{\min}}{f_{\max} + f_{\min}} \tag{19}$$

The mv parameter represents the movement force of the great white shark, which is increased by the iteration process. $f_{\max}$ and $f_{\min}$denote the maximal and minimal undulating frequencies of the undulating motion of the great white shark correspondingly, as given below.

$$mv = \frac{1}{a_0 + e^{\left(\frac{0.5K-5}{a_1}\right)}}. \tag{20}$$

In Eq (20), the a_0 and a_1 parameters are used to deal with exploration or exploitation strategies respectively. The term mv assists in accelerating the search range and strengthening the features of exploration and exploitation processes. This feature encourages the authors of the current study to use this model in resolving the problems. The movement towards for a better great white shark that converges towards the prey is given below

$$w_{k+1}^i = w_{gbest_k} + r_1 \vec{D}_w \times sgn(r_2 - 0.5)\ if\ r_3 < S_s. \tag{21}$$

In Eq (21), the term $sgn(r_2 - 0.5)$ accounts for changing search directions, as it provides 1 or -1, w_{k+1}^i is the newest location of the i^{th} great white shark; r_1, r_2, and r_3 denote the randomly generated numbers in the interval $[0,1]$; and $\vec{D}_w$ denotes the distance between a white shark and its prey which is formulated as given below.

$$\vec{D}_w = \left|rand \times \left(w_{gbest_k} - w_k^i\right)\right|. \tag{22}$$

The S_s parameter in Eq (21) is used to describe the visual strength and olfactory senses of the great white shark, while closely following the prey.

$$S_s = \left|1 - e^{\frac{-a_2 k}{K}}\right|. \tag{23}$$

In Eq (23), the a_2 parameter is used for controlling the exploitation or exploration behaviors.

The WSO approach progresses a Fitness Function (FF) for managing the enhanced classification performance. It resolves a positive integer to denote a better solution for the candidate outcomes. In this case, a decline in the classifier error rate is regarded as the FF.

$$fitness(x_i) = \frac{no.of\ misclassified\ instances}{Total\ no.of\ instances} \times 100. \tag{24}$$

4. Performance validation

The proposed model was simulated using Python 3.8.5 in a PC with specifications such as i5-8600k, GeForce 1050Ti 4GB, 16GB RAM, 250GB SSD, and 1TB HDD. The performance of the WSODL-RSSIC method was validated using the UCM Landuse dataset [28]. It has a total of 2,100 samples under 21 class labels as shown in Table 1. In addition to this, each class comprises of 100 images sized at 256 × 256 pixels. Figure 3 showcases some of the sample images. For experimental validation, 70% of the training dataset and 30% of testing dataset were used.

Table 1. Details on the database.

Classes	No. of Samples
Agricultural	100
Airplane	100
baseball diamond	100
Beach	100
Buildings	100
Chaparral	100
Dense Residential	100
Forest	100
Freeway	100
Golf Course	100
Harbor	100
Intersection	100
Medium Residential	100
Mobile Home Park	100
Overpass	100
Parking Lot	100
River	100
Runway	100
Sparse Residential	100
Storage Tanks	100
Tennis Court	100
Total Samples	2100

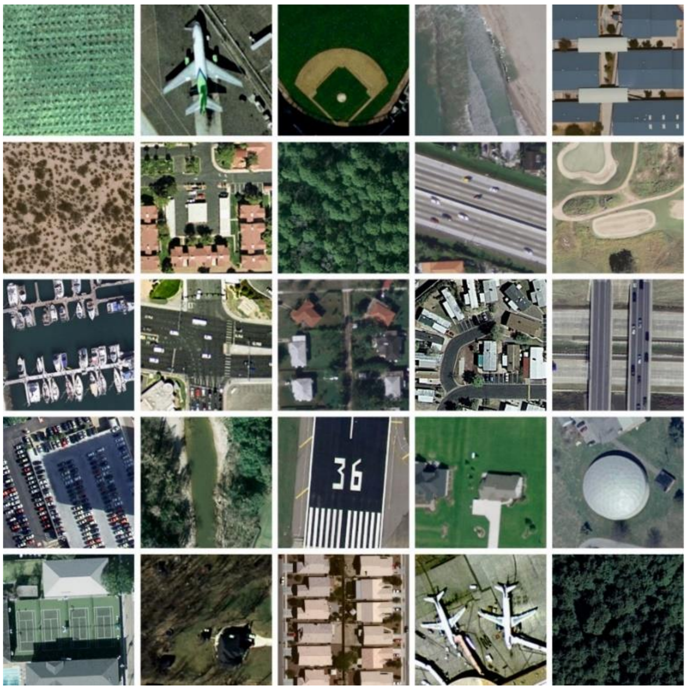

Figure 3. Sample images.

Figure 4 demonstrates the classifier outcomes of the WSODL-RSSIC method for the test dataset. Figures 4a and b depict the confusion matrices generated by the WSODL-RSSIC method on 70:30 of the TRP/TSP. The figure depicts that the WSODL-RSSIC system recognized and categorized all the 21 class labels precisely. Similarly, Figure 4c indicates the PR analysis outcomes of the WSODL-RSSIC method. The figures infer that the WSODL-RSSIC model attained the maximum PR performance for all the 21 classes. Finally, Figure 4d demonstrates the ROC examination outcomes achieved by the WSODL-RSSIC method. The figure depicts that the WSODL-RSSIC method produced effective outcomes with the highest ROC values for all the 21 class labels.

In Table 2 and Figure 5, the detailed image classification outcomes of the WSODL-RSSIC technique on 70:30 of the TRS/TSS are shown. The outcomes imply that the WSODL-RSSIC model gained effective outcomes under all the class labels. For example, on 70% of TRP, the WSODL-RSSIC method gained the average $accu_y$, $prec_n$, $sens_y$, $spec_y$, and F_{score} values such as 99.95%, 99.47%, 99.45%, 99.97%, and 99.46%, respectively. Afterward, on 30% of TSP, the WSODL-RSSIC technique achieved the average $accu_y$, $prec_n$, $sens_y$, $spec_y$, and F_{score} values such as 99.97%, 99.66%, 99.70%, 99.98%, and 99.67%, correspondingly.

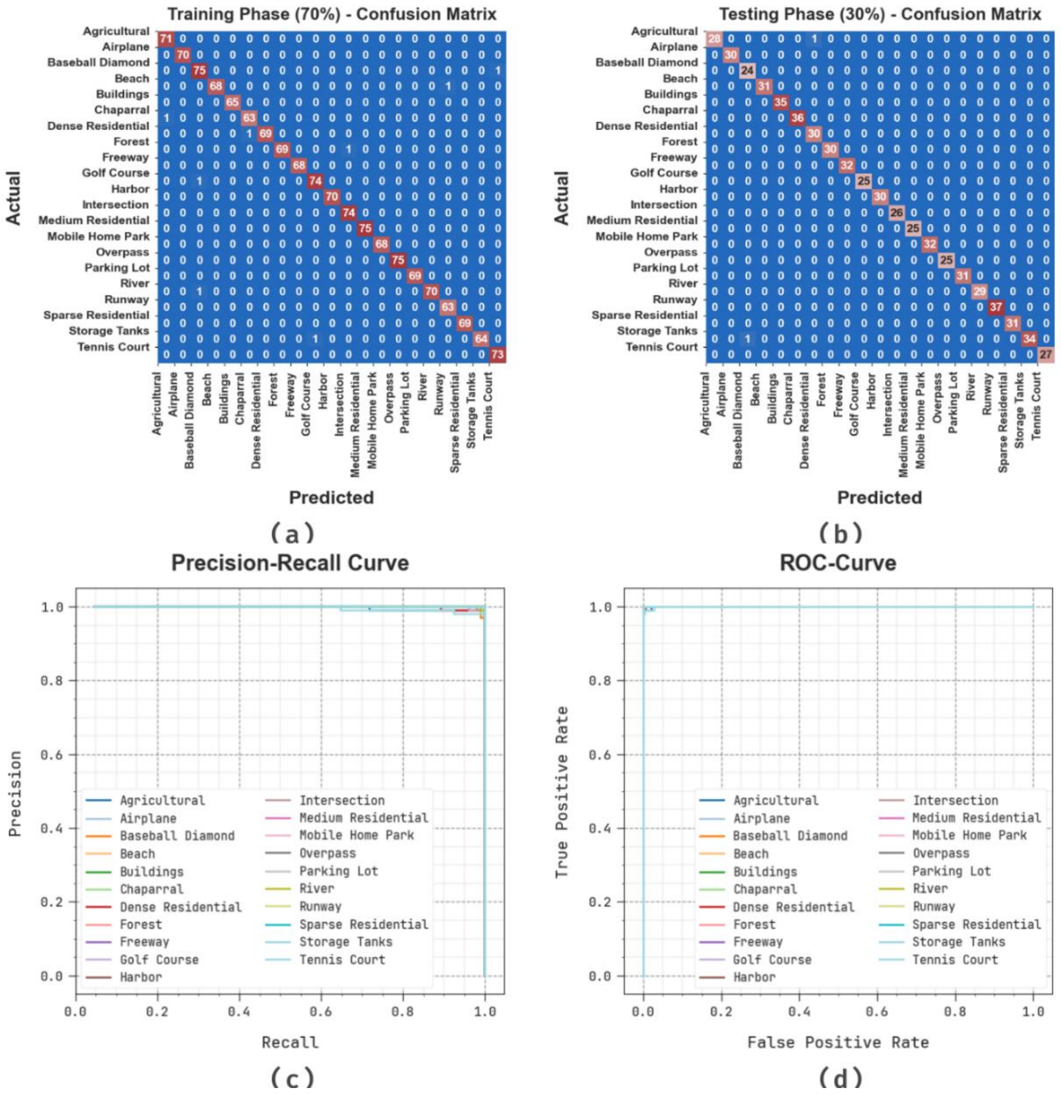

Figure 4. Classifier outcomes of (a and b) confusion matrices, (c) PR-curve, and (d) ROC-curve.

Table 2. Image classification outcomes of the WSODL-RSSIC approach on 70:30 of TRP/TSP.

Class	$Accu_y$	$Prec_n$	$Sens_y$	$Spec_y$	F_{score}
Training Phase (70%)					
Agricultural	99.93	98.61	100.00	99.93	99.30
Airplane	100.00	100.00	100.00	100.00	100.00
Baseball Diamond	99.80	97.40	98.68	99.86	98.04
Beach	99.93	100.00	98.55	100.00	99.27
Buildings	100.00	100.00	100.00	100.00	100.00
Chaparral	99.86	98.44	98.44	99.93	98.44
Dense Residential	99.93	100.00	98.57	100.00	99.28
Forest	99.93	100.00	98.57	100.00	99.28
Freeway	100.00	100.00	100.00	100.00	100.00
Golf Course	99.86	98.67	98.67	99.93	98.67

Continued on next page

Class	$Accu_y$	$Prec_n$	$Sens_y$	$Spec_y$	F_{score}
Training Phase (70%)					
Harbor	100.00	100.00	100.00	100.00	100.00
Intersection	99.93	98.67	100.00	99.93	99.33
Medium Residential	100.00	100.00	100.00	100.00	100.00
Mobile Home Park	100.00	100.00	100.00	100.00	100.00
Overpass	100.00	100.00	100.00	100.00	100.00
Parking Lot	100.00	100.00	100.00	100.00	100.00
River	99.93	100.00	98.59	100.00	99.29
Runway	99.93	98.44	100.00	99.93	99.21
Sparse Residential	100.00	100.00	100.00	100.00	100.00
Storage Tanks	99.93	100.00	98.46	100.00	99.22
Tennis Court	99.93	98.65	100.00	99.93	99.32
Average	99.95	99.47	99.45	99.97	99.46
Testing Phase (30%)					
Agricultural	99.84	100.00	96.55	100.00	98.25
Airplane	100.00	100.00	100.00	100.00	100.00
Baseball Diamond	99.84	96.00	100.00	99.83	97.96
Beach	100.00	100.00	100.00	100.00	100.00
Buildings	100.00	100.00	100.00	100.00	100.00
Chaparral	100.00	100.00	100.00	100.00	100.00
Dense Residential	99.84	96.77	100.00	99.83	98.36
Forest	100.00	100.00	100.00	100.00	100.00
Freeway	100.00	100.00	100.00	100.00	100.00
Golf Course	100.00	100.00	100.00	100.00	100.00
Harbor	100.00	100.00	100.00	100.00	100.00
Intersection	100.00	100.00	100.00	100.00	100.00
Medium Residential	100.00	100.00	100.00	100.00	100.00
Mobile Home Park	100.00	100.00	100.00	100.00	100.00
Overpass	100.00	100.00	100.00	100.00	100.00
Parking Lot	100.00	100.00	100.00	100.00	100.00
River	100.00	100.00	100.00	100.00	100.00
Runway	100.00	100.00	100.00	100.00	100.00
Sparse Residential	100.00	100.00	100.00	100.00	100.00
Storage Tanks	99.84	100.00	97.14	100.00	98.55
Tennis Court	100.00	100.00	100.00	100.00	100.00
Average	99.97	99.66	99.70	99.98	99.67

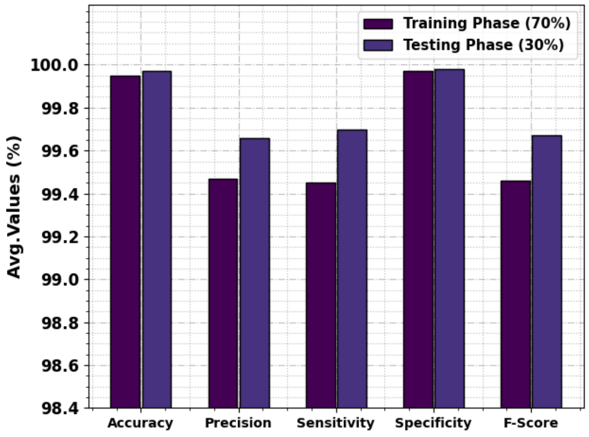

Figure 5. Average outcomes of the WSODL-RSSIC system on 70:30 of TRP/TSP.

Figure 6 shows the accuracy values achieved by the WSODL-RSSIC technique at the time of training and validation processes on the test database. The figure indicates that the WSODL-RSSIC model attained the maximum accuracy values over the highest number of epochs. Moreover, the maximum validation accuracy values over the training accuracy values display that the WSODL-RSSIC approach has learnt proficiently on the test database.

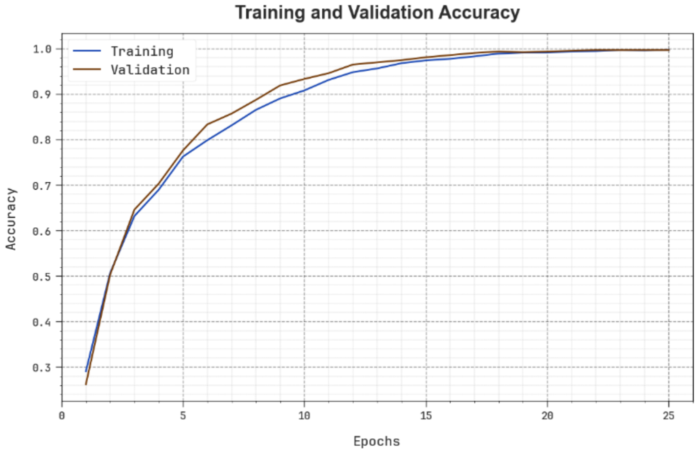

Figure 6. Accuracy curve of the WSODL-RSSIC approach.

The loss analysis outcomes of the WSODL-RSSIC method during training and validation procedures using the test database are depicted in Figure 7. The outcomes show that the WSODL-RSSIC model gained closer values for both training and validation losses. From this outcome, it can be inferred that the WSODL-RSSIC methodology has learnt proficiently from the test database.

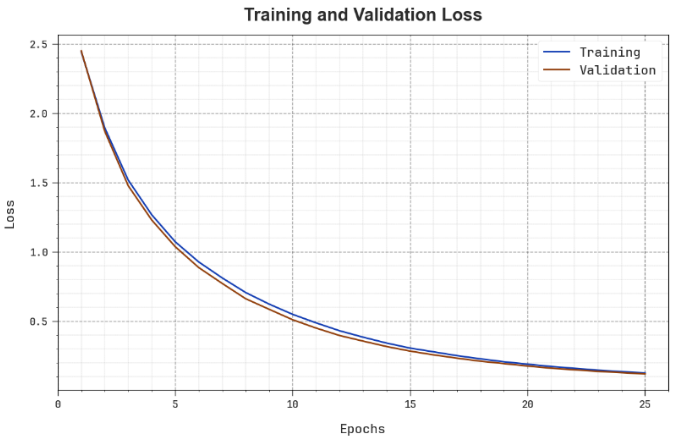

Figure 7. Loss curve of the WSODL-RSSIC method.

Finally, the superior classification performance of the WSODL-RSSIC methodology was compared with recent DL models and the results are shown in Table 3 and Figure 8 [16]. The outcomes demonstrate that the WSODL-RSSIC technique exhibited improved outcomes over other models. In terms of $accu_y$, the WSODL-RSSIC model obtained a maximum $accu_y$ of 99.97%, while the FBA, TS-Fusion, IV3-CapsNet, Bi-MobileNetv2, MVFLN+VGG, and the DTLF-ERSIC models accomplish the least $accu_y$ values such as 97.40%, 98%, 99.10%, 99.30%, 99.50%, and 99.70%, respectively.

Table 3. Comparative outcomes of the WSODL-RSSIC approach with recent DL approaches.

Methods	Precision	Sensitivity	Specificity	Accuracy
FBA Model	95.04	95.03	98.96	97.40
TS-Fusion	96.00	95.71	98.16	98.00
IV3-CapsNet	96.86	96.50	98.00	99.10
Bi-MobileNetv2	96.94	96.09	98.03	99.30
MVFLN+VGG	95.95	95.31	98.65	99.50
DTLF-ERSIC	96.80	96.70	99.80	99.70
WSODL-RSSIC	99.66	99.70	99.98	99.97

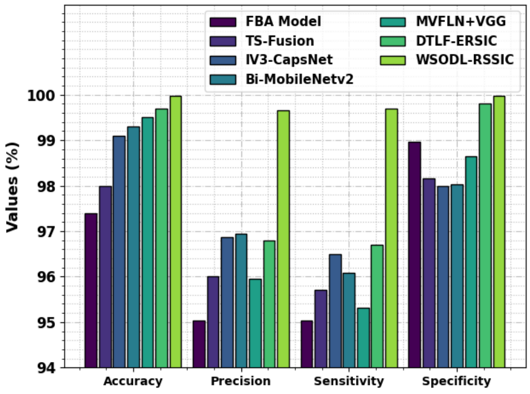

Figure 8. Comparative outcomes of the WSODL-RSSIC method with recent DL approaches.

In terms of $prec_n$, the WSODL-RSSIC technique obtained a maximum $prec_n$ of 99.66%, while the FBA, TS-Fusion, IV3-CapsNet, Bi-MobileNetv2, MVFLN+VGG and the DTLF-ERSIC models accomplished the least $prec_n$ values such as 95.01%, 96%, 96.86%, 96.94%, 95.95%, and 96.80% respectively.

Concurrently, in terms of $sens_y$, the WSODL-RSSIC technique achieved an increase in the $sens_y$ up to 99.70%, while the FBA, TS-Fusion, IV3-CapsNet, Bi-MobileNetv2, MVFLN+VGG, and the DTLF-ERSIC models accomplished the least $sens_y$ values such as 95.03%, 95.71%, 96.50%, 96.09%, 95.31%, and 96.70%, correspondingly. With regards to $spec_y$, the WSODL-RSSIC technique attained a maximum $spec_y$ of 99.98%, whereas the FBA, TS-Fusion, IV3-CapsNet, Bi-MobileNetv2, MVFLN+VGG, and the DTLF-ERSIC methods accomplished the least $spec_y$ values such as 98.96%, 98.16%, 98.0%, 98.03%, 98.65%, and 99.80%, correspondingly.

Table 4 and Figure 9 show the comparative $accu_y$ analysis outcomes attained by the WSODL-RSSIC technique and other approaches under the AID database. The simulation values imply that the TS-Fusion technique achieved the worst performance with a minimal $accu_y$ of 83.30%. Moreover, the FT-VGGNet-16 and IV3-CapsNet approaches produced slightly higher results with $accu_y$ values being 90.50% and 92.60%, respectively. In addition to this, the FDP-RN, CNN-MLP, and DTLF-ERSIC methods demonstrated moderate and reasonable results with $accu_y$ values such as 95.50%, 97.40%, and 99.80%, correspondingly. But, the WSODL-RSSIC technique yielded a better performance with a maximum $accu_y$ of 99.92%.

Table 4. $Accu_y$ analysis outcomes of the WSODL-RSSIC and other existing models under the AID dataset.

Methods	Accuracy (%)
FT-VGGNet-16	90.50
TS-Fusion	83.30
IV3-CapsNet	92.60
FDP-RN	95.50
CNN-MLP	97.40
DTLF-ERSIC	99.80
WSODL-RSSIC	99.92

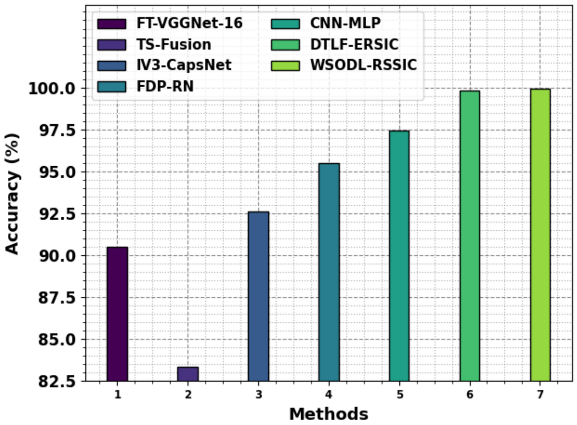

Figure 9. $Accu_y$ analysis outcomes of the WSODL-RSSIC under the AID dataset.

These results highlight the superior classification outcomes of the WSODL-RSSIC method proposed in this study.

5. Conclusions

The key focus of the current study is to design and develop an efficient and automated remote sensing image classification method termed as WSODL-RSSIC. The presented WSODL-RSSIC technique focuses on effective recognition and classification of the remote-sensing images into distinct class labels. The WSODL-RSSIC method has a three-stage procedure involving the ShuffleNet feature extraction, DMN classification, and WSO-based hyperparameter tuning. The design of the WSO technique effectually chooses the hyperparameters of the DMN classifier and this feature improved the classification performance. The proposed WSODL-RSSIC method was simulated using the remote sensing image databases. The experimental outcomes demonstrate the superior performance of the WSODL-RSSIC technique than the recent state-of-the-art approaches under diverse evaluation metrics. In the future, an ensemble learning process can be included to increase the performance of the WSODL-RSSIC technique.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The author gratefully acknowledges with thanks technical support provided by the Center of Research Excellence in Artificial Intelligence and Data Science, King Abdulaziz University (KAU), Jeddah, Saudi Arabia.

Conflict of interest

The author declares that there is no conflict of interest. The manuscript was written through the contributions of all authors. The author has given approval to the final version of the manuscript.

References

1. S. Thirumaladevi, K. V. Swamy, M. Sailaja, Remote sensing image scene classification by transfer learning to augment the accuracy, *Meas. Sens.*, **25** (2023), 100645. https://doi.org/10.1016/j.measen.2022.100645
2. M. Ragab, Multi-label scene classification on remote sensing imagery using modified Dingo optimizer with deep learning, *IEEE Access*, **12** (2023), 11879–11886. https://doi.org/10.1109/ACCESS.2023.3344773
3. P. Wang, H. Zhao, Z. Yang, Q. Jin, Y. Wu, P. Xia, et al., Fast tailings pond mapping exploiting large scene remote sensing images by coupling scene classification and sematic segmentation models, *Remote Sens.*, **15** (2023), 327. https://doi.org/10.3390/rs15020327

4. M. Ye, N. Ruiwen, Z. Chang, G. He, H. Tianli, L. Shijun, et al., A lightweight model of VGG-16 for remote sensing image classification, *IEEE J-STARS*, **14** (2021), 6916–6922. https://doi.org/10.1109/JSTARS.2021.3090085
5. M. Ragab, H. A. Abdushkour, A. O. Khadidos, A. M. Alshareef, K. H. Alyoubi, A. O. Khadidos, Improved deep learning-based vehicle detection for urban applications using remote sensing imagery, *Remote Sens.*, **15** (2023), 4747. https://doi.org/10.3390/rs15194747
6. A. A. Aljabri, A. Alshanqiti, A. B. Alkhodre, A. Alzahem, A. Hagag, Extracting feature fusion and co-saliency clusters using transfer learning techniques for improving remote sensing scene classification, *Optik*, **273** (2023), 170408. https://doi.org/10.1016/j.ijleo.2022.170408
7. X. Tang, W. Lin, J. Ma, X. Zhang, F Liu, L. Jiao, Class-level prototype guided multiscale feature learning for remote sensing scene classification with limited labels, *IEEE T. Geosci. Remote Sens.*, **60** (2022), 5622315. https://doi.org10.1109/TGRS.2022.3169835
8. M. N. Akhtar, E. Ansari, S. S. N. Alhady, E. A. Bakar, Leveraging on advanced remote sensing- and artificial intelligence-based technologies to manage palm oil plantation for current global scenario: A review, *Agriculture*, **13** (2023), 504. https://doi.org/10.3390/agriculture13020504
9. C. Peng, Y. Li, R. Shang, L. Jiao, RSBNet: One-shot neural architecture search for a backbone network in remote sensing image recognition, *Neurocomputing*, **537** (2023), 110–127. https://doi.org/10.1016/j.neucom.2023.03.046
10. X. Huang, Y. Sun, S. Feng, Y. Ye, X. Li, Better visual interpretation for remote sensing scene classification, *IEEE Geosci. Remote Sens. Lett.*, **19** (2022), 6504305. https://doi.org/10.1109/LGRS.2021.3132920
11. Y. Sun, W. Zheng, HRNet-and PSPNet-based multiband semantic segmentation of remote sensing images, *Neural Comput. Appl.*, **35** (2023), 8667–8675. https://doi.org/10.1007/s00521-022-07737-w
12. H. Sun, Y. Lin, Q. Zou, S. Song, J. Fang, H. Yu, Convolutional neural networks based remote sensing scene classification under clear and cloudy environments, In: *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2021, 713–720. https://doi.org/10.1109/ICCVW54120.2021.00085
13. W. Miao, J. Geng, W. Jiang, Semi-supervised remote-sensing image scene classification using representation consistency Siamese network, *IEEE T. Geosci. Remote Sens.*, **60** (2022), 5616614. https://doi.org/10.1109/TGRS.2022.3140485
14. X. Xu, Y. Chen, J. Zhang, Y. Chen, P. Anandhan, A. Manickam, A novel approach for scene classification from remote sensing images using deep learning methods, *Eur. J. Remote Sens.*, **54** (2021), 383–395. https://doi.org/10.1080/22797254.2020.1790995
15. X. Cheng, H. Lei, Remote sensing scene image classification based on mmsCNN–HMM with stacking ensemble model, *Remote Sens.*, **14** (2022), 4423. https://doi.org/10.3390/rs14174423
16. A. M. Hilal, F. N. Al-Wesabi, K. J. Alzahrani, M. Al Duhayyim, M. A. Hamza, M. Rizwanullah, et al., Deep transfer learning based fusion model for environmental remote sensing image classification model, *Eur. J. Remote Sens.*, **55** (2022), 12–23. https://doi.org/10.1080/22797254.2021.2017799
17. G. Zhang, W. Xu, W. Zhao, C. Huang, E. N. Yk, Y. Chen, et al., A multiscale attention network for remote sensing scene images classification, *IEEE J-STARS*, **14** (2021), 9530–9545. https://doi.org/10.1109/JSTARS.2021.3109661

18. C. Xu, G. Zhu, J. Shu, A lightweight and robust lie group-convolutional neural networks joint representation for remote sensing scene classification, *IEEE T. Geoscie. Remote Sens.*, **60** (2022), 5501415. https://doi.org/10.1109/TGRS.2020.3048024
19. J. Wang, W. Li, M. Zhang, R. Tao, J. Chanussot, Remote sensing scene classification via multi-stage self-guided separation network, *IEEE T. Geosci. Remote Sens.*, **61** (2023), 5615312. https://doi.org/10.1109/TGRS.2023.3295797
20. M. Ragab, Leveraging mayfly optimization with deep learning for secure remote sensing scene image classification, *Comput. Electr. Eng.*, **108** (2023), 108672. https://doi.org/10.1016/j.compeleceng.2023.108672
21. J. Geng, B. Xue, W. Jiang, Foreground-background contrastive learning for few-shot remote sensing image scene classification, *IEEE T. Geosci. Remote Sens.*, **61** (2023), 5614112. https://doi.org/10.1109/TGRS.2023.3290794
22. L. Huang, W. Zhao, A. W. C. Liew, Y. You, An evidential combination method with multi-color spaces for remote sensing image scene classification, *Inform. Fusion*, **93** (2023), 209–226. https://doi.org/10.1016/j.inffus.2022.12.025
23. Y. Zhao, J. Liu, J. Yang, Z. Wu, EMSCNet: Efficient multisample contrastive network for remote sensing image scene classification, *IEEE T. Geosci. Remote Sens.*, **61** (2023), 5605814. https://doi.org/10.1109/TGRS.2023.3262840
24. D. Singh, Y. S. Taspinar, R. Kursun, I. Cinar, M. Koklu, I. A. Ozkan, et al., Classification and analysis of pistachio species with pre-trained deep learning models, *Electronics*, **11** (2022), 981. https://doi.org/10.3390/electronics11070981
25. X. Zhang, X. Zhou, M. Lin, J. Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, 6848–6856. https://doi.org/10.1109/CVPR.2018.00716
26. M. Khashei, F. Chahkoutahi, N. Bakhtiarvand, A novel discrete deep learning-based intelligent methodology for energy consumption classification, *Energy Rep.*, **9** (2023), 4861–4871. https://doi.org/10.1016/j.egyr.2023.04.006
27. A. Fathy, D. Yousri, A. G. Alharbi, M. A. A bdelkareem, A new hybrid white shark and whale optimization approach for estimating the Li-Ion battery model parameters, *Sustainability*, **15** (2023), 5667. https://doi.org/10.3390/su15075667
28. UC Merced land use dataset, Available from: http://weegee.vision.ucmerced.edu/datasets/landuse.html.