Scene Classification on Remote Sensing Images with Deep Convolutional Neural Network based on Modeling of Mayfly Optimization Algorithm

M. Rega[1], Dr. S. Sivakumar[2]

[1]Research Scholar,Department of Computer and Information Science, Annamalai University,Annamalainagar-608002, Email: rekhamohanraj00@gmail.com
[2]Assistant Professor, PG Department of Computer Science, Government Arts College, Chidambaram – 608102, Email: sivassa77@gmail.com
*Corresponding Author

Received: 15.07.2024 Revised: 19.08.2024 Accepted: 22.09.2024

ABSTRACT

Scene Classification (MFO-DCNNSC) technique on Remote Sensing Images. In the MFO-DCNNSC technique, the RS images undergo preprocessing using Wiener filtering (WF) to improve the image quality. To extract features, the MFO-DCNNSC technique applies the Inception-ResNetv2 model for learning the hierarchical depiction of the visual data. The MFO algorithm is useful for the appropriate range of the hyperparameters linked to the Inception-ResNetv2 method. With the great feature learning abilities of deep neural networks (DNNs), RSI scene classification compelled by deep learning (DL) has gained extraordinary attention and got important inventions.Scene classification is a vital study issue in RSI that has concerned numerous researchers presently. DL techniques are gaining a reputation in image feature analysis and reaching advanced performances in scene classification of RSI. This researchproposes a Mayfly Optimizer Algorithm with Deep Convolutional Neural Network. At last, the scene classification process has been executed by the usage of the deep belief network (DBN) model. To point out the improved performance of the MFO-DCNNSC approach, a huge assortment of experiments have been executed on the benchmark database. The obtained values pointed out that the MFO-DCNNSC technique outperforms the other models in terms of distinct measures.

Keywords: Scene Classification; Mayfly Optimization Algorithm; Convolutional Neural Network,Remote Sensing Image,Wiener Filtering.

1. INTRODUCTION

With the growth of Earth surveillance technology, many dissimilar types such as synthetic aperture radar and multi/hyperspectral higher-resolution imageries of the surface of Earth have been readily accessible [1]. So, it is mainly noteworthy to effectively recognize their semantic content, and more intellectual classification models of land use and land cover (LULC) are required [2]. Remote sensing image (RSI) scene classification targets to mechanically allocate an exact semantic label to every RSI scene patch as per its contents, which has become a keen study topic in the area of RSI analysis because of its key applications in LULC such as land resource organization, urban planning, traffic control, and disaster monitoring [3]. With the development of the spatial resolution of RSI, the classification of RSI slowly made 3 similar classification branches at dissimilar stages such as scene-level, object-level, and pixel-level [4]. It is worth to declare that we employ the word “RSI classification” as a common concept [5]. To be exact, in the initial works, researchers mostly concentrated on categorizing RSI at the pixel or sub-pixel level over labelling every pixel in the RSI with a semantic class, since the spatial resolution of RSI is very low. The pixel size is parallel to the object size of interest [6].

Current research has shown that data-driven deep learning (DL) methods get advanced outcomes in scene classification due to their capabilities to learn higher-level abstract features from imageries [7]. The improvements in hardware for graphic processing units (GPU) deliver the ability to procedure a huge number of data on DL structures. DL provides an architecture for feature learning techniques that contains many processing stages to absorb RSI features at dissimilar abstraction levels [8]. The Convolutional neural network (CNN) is noble at selecting local features and gradually increasing their receptive domains for additional ideas [9]. Transfer-based DL methods function on the basis that vital elements of images are similar; so, they use pre-trained approaches which are trained on larger-scale databases for RSI uses [10].

This research presents a Mayfly Optimizer Algorithm with Deep Convolutional Neural Network Scene Classification (MFO-DCNNSC) technique on Remote Sensing Images. In the MFO-DCNNSC technique, the RS images undergo preprocessing using Wiener filtering (WF) to improve the image quality. To extract features, the MFO-DCNNSC technique applies the Inception-ResNetv2 model for learning the hierarchical depiction of the visual data. Following by, the MFO algorithm is useful for the appropriate range of the hyperparameters connected to the Inception-ResNetv2 technique. At last, the scene classification procedure has been executed by the use of the deep belief network (DBN) model. To point out the superior performance of the MFO-DCNNSC model, a huge array of experiments have been executed on the benchmark database.

2. Related Works

Tian et al. [11] developed an innovative few-shot scene classification technique named hierarchical-relation network (HiReNet) system containing an embedding and relation network. Primary, Siamese architecture was employed. The technique also presents a hierarchical relation learning (HRL) method. Also, the system introduces a feature aggregation method for extraction that combines multilevel features and utilizes channel attention to re-weight these features. In [12], a lightweight modular network system was developed. Primarily, a lightweight self-compensated convolution (SCC), and an SSC technique were presented. Subsequently, a self-compensating bottleneck module (SCBM) method dependent upon the SCC was presented. Lastly, the technique employed the developed self-compensation bottleneck component for making a lightweight as well as modular self-compensation CNN (SCCNN) method. Shi et al. [13] introduced a global context feature extraction method that integrates the semantics data of various areas over a global pooling and a 3 diverse scale sub-area pooling. Similarly, a 3 branch joint feature extraction method was built. Lastly, a lightweight CNN dependent upon joint features (LCNN-JF) was made employing depth wise separable convolution, conventional convolution, joint feature extraction classifier, and component for classification.

In [14], a massive kernel CNNs into the scene classification task was presented. Similarly, the efficient data with the feature maps has been also intensely channel-based. Hence, to completely model such a channel dependency, an innovative channel separation and mixing component was developed. The integration of them makes a huge Kernel Separable Mixed Convnet (LSMNet), permitting the system to take the efficient needs of feature map at the channel and spatial sizes. Wei, Geng, and Yin [15] developed a discriminative system HCA-TSA dependent upon the head-tail global joint dual attention device. Similarly, the channel attention method spatial attention module Tail-Spatial Attention (TSA) and Head-Channel Attention (HCA) have been developed. The model employs the context data refinement capability of the GRU method.

In [16], an innovative dual-encoding technique calledmaster–the slave encoding network (MSE-Net) system was designed. The main encoder-dependent ViT removes higher-level semantic features, but the MSE dependent upon CNN, captures relatively low-level spatial structure data. Secondarily, this study further improves 2 fusion approaches. The primary scheme comprises the auxiliary enhancement units (AEUs). The second approach is an interactive perception unit that enables relations and a combination of the two encoder's representations. Shi et al. [17] introduced a feature-augmented self-distilled CNN (FASDNet) architecture. Initially, ResNet34 was employed as the backbone network for the extracting process. The subsequent one is a feature augmentation pyramid module (FAPM) was developed for removing and combing multi-level feature data. Subsequently, auxiliary branches have been built. The self-distillation algorithm was employed among the FAPM and the backbone network. In conclusion, the developed method was cooperatively supervised through cross-entropy loss, log it distillation loss, and feature distillation loss.

3. The Proposed Method

In this paper, we have proposed an MFO-DCNNSC model on RSI. The main intention of the MFO-DCNNSC model encompass different kinds of processes namely WF-based image preprocessing, Inception-ResNetv2-based feature extraction, MFO-based hyperparameter tuning, and DBN-based classification process. Fig. 1 demonstrates the entire procedure of the MFO-DCNNSC technique.

3.1. Image Preprocessing

Primarily, the MFO-DCNNSC technique on RSI undergoes preprocessing using WF to improve the image quality.WF is a signal processing model used to decrease noise from images while trying to reinstate the new and exact signal [18]. Unlike median filtering, WF takes into account both the noisy and wanted signal features, creating an adaptive filtering model. It functions in the frequency area, using the power spectral density of the noisy and preferred signals to alter the filtering parameters dynamically. WF is mainly effectual in situations where the statistical properties of the desired signal and noise are identified. This technique is commonly employed in image deconvolution, restoration, and improvement, where the objective is to increase the complete quality and clarity of images by justifying the effects of numerous kinds of noise.

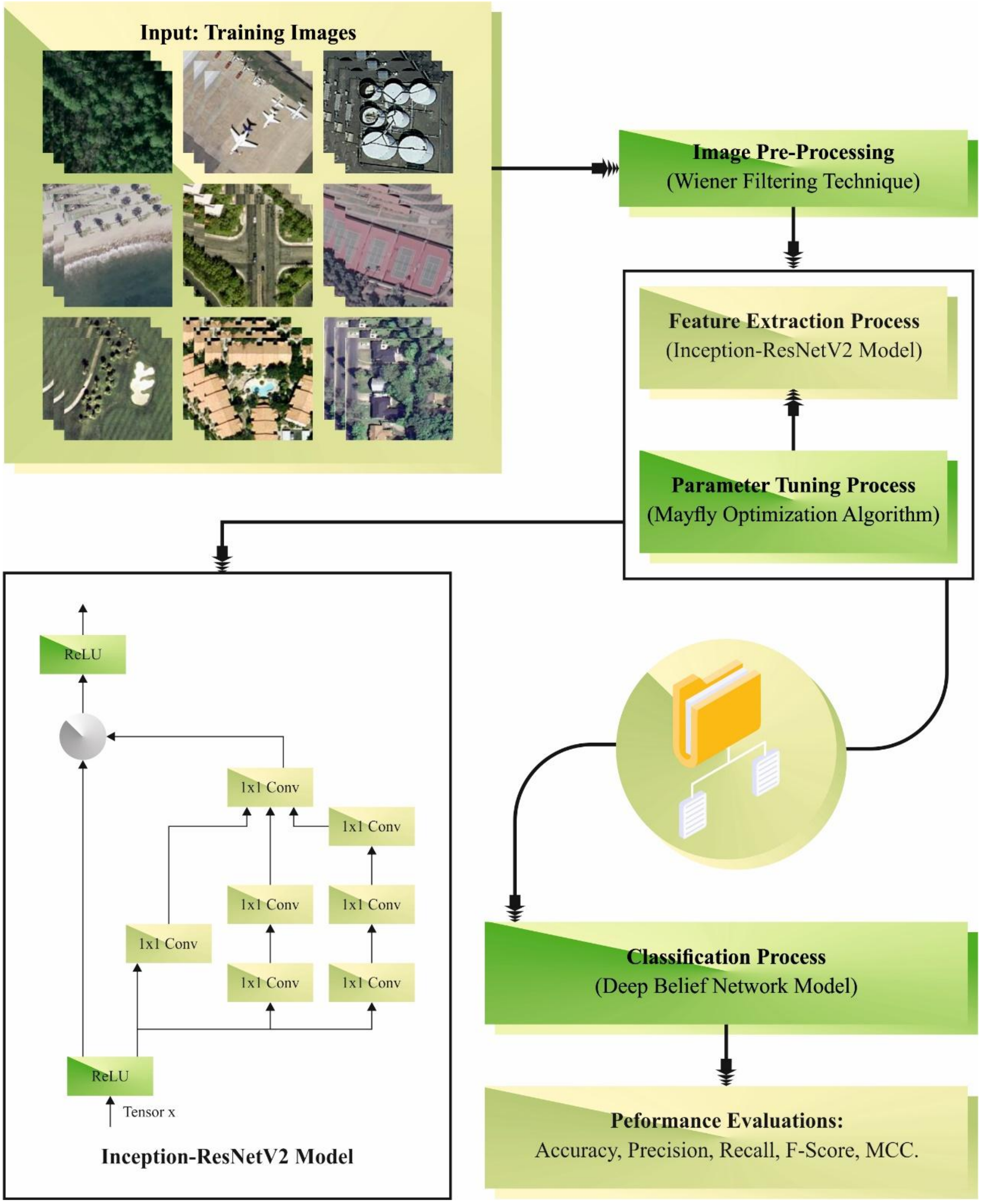

Fig. 1. Overall process of the MFO-DCNNSC technique

3.2. Inception-ResNetv2 Model

To extract features, the MFO-DCNNSC technique applies the Inception-ResNetv2 model for learning the hierarchical depiction of the visual data.The Inception-ResNetV2 architecture integrates inception modules that are basic modules that permit to extraction of multi-scale features [19]. This module consists of the parallel convolutional branch with varying filter sizes, allowing to capture of features at a receptive field. The residual connection in the Inception-ResNetV2 architecture contributes to mitigating the gradient vanishing problems, enabling effective deep network training. The Inception module extracts various features from the different sizes of convolution kernels and various scales to optimize the adaptability of the network. This can diminish the model’s complexity and reduce the specific convolution kernel. This process accelerates parameter optimization and network training which transmit the signals of numerous units to all the layers forward or backward. It is necessary to apply a 1×1 convolutional layer to decrease or enhance the dimension since the feature map y_h might be a variant in the ResNet.

$$F(y_h) = e * y_h + \beta \qquad (1)$$

$$x_h = Rl(F) + s(y) \qquad (2)$$

$$y_{h+!} = Rl(x_h) \qquad (3)$$

Here $s(y)$ refers to the simple transformation for the input, β represents the offset, y_h denotes the input, xh denotes the sum of two branches, e denotes the weight, Rl denotes the ReLU function, the convolution operation is $F(y_h)$, and y_{h+1} denotes the final output of the residual block.

$$Rl(y) = \ maxi(0, y) \qquad (4)$$

The threshold of 0 and y value is used as input and obtain the output. The objective of the ResNet learning unit is to avoid the gradient disappearing problem during the training network model, and. If the method has attained a certain capacity, then this layer might be similarly mapped and aidwith rapid and simple network training.

$$\frac{\alpha Ym}{\alpha Yj} = \frac{\alpha Yj + F(Yj, \tau j, \beta j)}{\alpha Yj} = 1 + \frac{\alpha F(Ym, \tau m. \beta m)}{\alpha Ym} \qquad (5)$$

Where the input of m^{th} unit is Ym, Yj specifies the input of j^{th} residual units, the residual function represented as $F(.)$.

3.3. Hyper parameter Tuning using MFO

In this work, the MFO algorithm is found useful for the appropriate range of the hyperparameters connected to the Inception-ResNetv2 technique.The GA,PSO, and firefly method features have been joined in this algorithm termed FA [20]. The MA is very effectual hybrid optimizer procedure that depends upon the mayflies (MFs) behaviour during mating and enhances the PSO search. This optimizer method ignores the MF's lifetime and accepts an adult directly after producing which only the sturdiest endure. Each MF's position in the solution space specifies the possibility that originated at that place. Arbitrarily produced are sets of female and male MFs. To restate, $P = \left[p_1, p_2, \dots, p_{d_{Max}}\right]^T$ the location vector signifies the search space into MFs, the agent performs the search will primarily be planted. The objective function estimates the location vector efficiency with x. Employing the vector of velocity, the MF location is studied in its movement way that is learned by its individual and social movement skills $= \left[k_1, k_2, \dots, K_{d_{Max}}\right]^T$. An MF will go up or down the search depending on its present finest location (represented by Pest) and the optimal locations gained by other mayflies in the swarm (signified by g_{best}).

Male Mayfly (MMF) Flight

The combination of MMFs into groups is a sign that their position is reviewed in light of novel info and situations. An upgraded type of MMF location is as below:

$$P_m(t + 1) = k_m(t + 1) + p_m(t) \qquad (6)$$

For the ith MF, $p_m(t)$ is its present position and $p_m(0)$ falls among x_{Min} and x_{Max} . $p_m(t + 1)$ represents the MF position and $k_m(t + 1)$ signifies the MF velocities.

The constant speed is intended as below because the MMF marital dance endures at the peak of some meters.

$$k_{md}(t + 1) = v_{md}(t) + q_1 \times \exp\left(-\varsigma^{D_p^2}\right) \times \left(pbest_{md} - p_{md}(t)\right) +$$
$$q_2 \times \exp\left(-\varsigma^{D_g^2}\right) \times \left(gbest_d - p_{md}(t)\right) \qquad (7)$$

q_1 and $q_2$denotes the attractive co-efficient that defines the significance of the psychological and social modules. When MFs are in an σ situation, they could not able to see each other. Employing Eqs. (9) and (10), we can define the distances D_p, and D_p that pi has $pbest_m$ and gbest, correspondingly. The ith agent’s speed in the dth dimension is signified by k_{md}, while its location is specified by p_{md}. d stands for dimension index, which ranges from 1 to d_{Max}, where d refers to the maximal amount of dimensions. $pbest_{md}$ denotes the finest location by the ith agent in the dth dimension and intended as below.

$$pbest_m = \begin{cases} x_m(t + 1), f\left(x_m(t + 1)\right) < f(pbest_m) \\ pbest, f\left(x_m(t + 1)\right) \geq f(pbest_m) \end{cases} \qquad (8)$$

Here, f denotes the quality define objective function. We define D_p^2 and D_g^2 which are as follows:

$$\mathrm{D}_{\mathrm{p}}^{2}=\left(\sum_{\mathrm{n}=1}^{\mathrm{d}_{\mathrm{Max}}}(\mathrm{p}_{\mathrm{md}}-\mathrm{pbest}_{\mathrm{m}})\right)^{0.5} \qquad (9)$$

$$\mathrm{D}_{\mathrm{g}}^{2}=\left(\sum_{\mathrm{d}=1}^{\mathrm{d}_{\mathrm{Max}}}(\mathrm{p}_{\mathrm{md}}-\mathrm{gbest})\right)^{0.5} \qquad (10)$$

The sturdy and healthy MFs will continue dancing vertically, defending the algorithm's optimum result. So, the healthy MFs should uphold the subsequent speed shift, which presents a component of chance in the process.

$$\mathrm{k}_{\mathrm{md}}(\mathrm{t}+1)=\mathrm{k}_{\mathrm{md}}(\mathrm{t})+\mathrm{ND}\times\boldsymbol{\varpi} \qquad (11)$$

Here,ND denotes the nuptial dance co-efficient and $\boldsymbol{\varpi}$ signifies the randomly generated number among -1 and 1.

Female Mayfly (FMF) Flight

FMFs will not group like males. They head directly for the men to mate. Utilizing$\mathrm{r}_{\mathrm{m}}(\mathrm{t})$, we can observe where the ith FMF has been positioned in the search space, and then utilize the following expression to alter our location:

$$\mathrm{r}_{\mathrm{m}}(\mathrm{t}+1)=\mathrm{k}_{\mathrm{m}}(\mathrm{t}+1)+\mathrm{r}_{\mathrm{m}}(\mathrm{t}) \qquad (12)$$

To perfect this occurrence, we accept that the most gorgeous female will be pulled to the most attractive male; the next female will be drained to the next male, and so on. By employing the below expression, we can define the velocity:

$$\mathrm{k}_{\mathrm{md}}(\mathrm{t}+1)=\begin{cases}\mathrm{k}_{\mathrm{md}}(\mathrm{t})+\mathrm{q}_{2}\times\exp\left(-\varsigma^{\mathrm{D}_{\mathrm{if}}^{2}}\right)\times\left(\mathrm{p}_{\mathrm{md}}(\mathrm{t})-\mathrm{r}_{\mathrm{md}}(\mathrm{t})\right),\\ \mathrm{f}(\mathrm{r}_{\mathrm{m}})>f(\mathrm{p}_{\mathrm{m}})\\ \mathrm{u}_{\mathrm{md}}(\mathrm{t})+\mathrm{q}_{\mathrm{w}}\times\boldsymbol{\varpi},\mathrm{f}(\mathrm{r}_{\mathrm{m}})\leq\mathrm{f}(\mathrm{p}_{\mathrm{m}})\end{cases} \qquad (13)$$

Here, $\mathrm{r}_{\mathrm{md}}(\mathrm{t})$ represents the location and $\mathrm{u}_{\mathrm{md}}(\mathrm{t})$ signifies the velocity of the ith FMF in the dth dimension at time t. MMF and FMF separate distances are signified by $\mathrm{D}_{\mathrm{if}}^{2}$, D is dual times than the original distance. The co-efficient of q_{w}, is selected randomly.

Mating Procedure

The cross-over operation is employed to model the behavior of MF mating defined under one male and female are selected from every set to be the parents, just males are appealed to precise female. Winner selection are completely dependent either on opportunity or objective function. For every cluster, the healthy female mate with the healthiest male. From the below-mentioned formulation, we can expect the offspring of the cross-over.

$$\alpha 1=\beta\times\mathrm{male}+(1-\beta)\times\mathrm{female} \qquad (14)$$

$$\alpha 2=\beta\times\mathrm{female}+(1-\beta)\times\mathrm{male} \qquad (15)$$

Here, α1 and α2 denotes the 1st and 2nd generations."β"refers therandomly generated number within a definite range. Furthermore, male and female signify biological parents. It is noticeable that a child is expected to have no preliminary speed.

The fitness function (FF) is the significant aspect prompting the MFO mode performance. The hyperparameter range procedure contains the solution encode technique to evaluate the efficacy of the candidate solution. In this paper, the MFO model reflects accuracy as the foremost standard to project the FF and conveyed as below.

$$\mathrm{Fitness}=\max(\mathrm{P}) \qquad (16)$$

$$\mathrm{P}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \qquad (17)$$

Here, FP and TP represent the false and true positive values.

3.4. DBN based Classification

At last, the scene classification procedure is executed by the application of the DBN model.As a kind of neural network (NN), DBN consists of multiple restricted Boltzmann machines (RBMs) [21]. The input layer represents the original data character, and the output layer represents the data label. In the deep architecture, the main features of data have been extracted from the input to the output layers via the abstraction layer. The input of DBN is the first RBM layer, and the output is the last hidden layer of RBM. The DBN is processed as a multilayer perceptron (MLP); if applied for classification, we add a logistic regression layer to the output. The DBN includes a few RBMs. RBM consists of hidden and visible units.The energy of RBM joint configuration comprises weight and bias. Consider $h = \{0,1\}^m$ and $v = \{0,1\}^n$ as the hidden and visible state units, correspondingly.

$$E(v, h; \theta) = -\sum_{i=1}^{n} a_i v_i - \sum_{j=1}^{m} b_j h_j - \sum_{i=1j}^{n} \sum_{=1}^{m} w_{ij} v_j h_j \qquad (18)$$

In Eq. (18), the number of visible and hidden units are represented as n and m, correspondingly. the model parameters are denoted as $\theta = \{a_j, b_j, w_{ij}\}$, the weight between i^{th} and j^{th} visible and hidden units are represented byw_{ij}; the visible and hidden units are indicated as a_i and b_j, correspondingly. Fig. 2 shows the architecture of DBN.

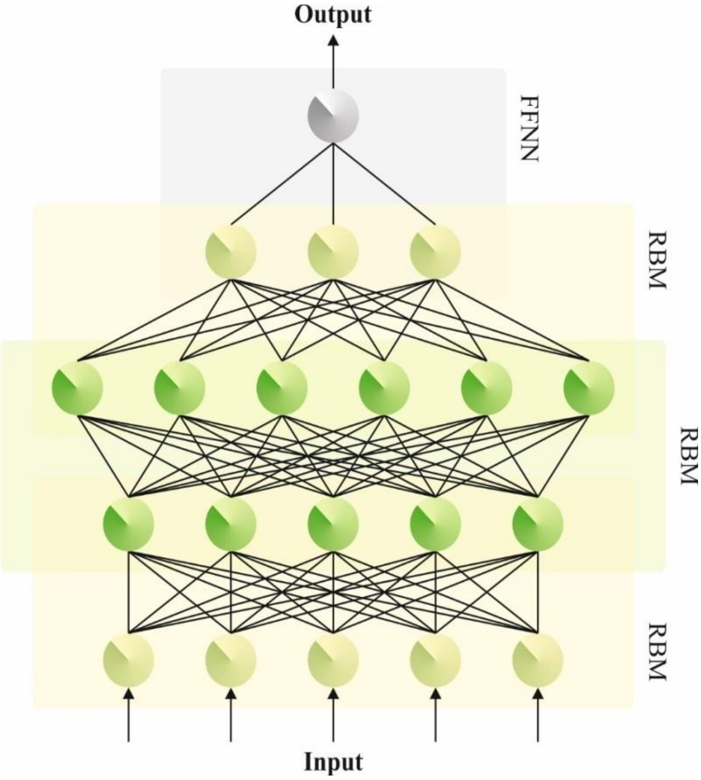

Fig 2. Architecture of DBN

The formula of joint probability for the hidden and visible vectors is:

$$p(v, h; \theta) = \frac{1}{Z(\theta)} \exp\left(-B(v, h; \theta)\right) \qquad (19)$$

Let $Z(\theta)$ be a normalization factor that can be represented by:

$$Z(\theta) = \sum_{v} \sum_{h} \exp\left(-B(v, h; \theta)\right) \qquad (20)$$

$$p\left(h_j = 1 \middle| \theta\right) = \frac{1}{1 + \exp\left(-b_j - \sum_i w_{ij} v_i\right)} \qquad (21)$$

$$p(v_i = 1|\theta) = \frac{1}{1 + \exp\left(-a_i - \sum_j w_{ij} h_j\right)} \qquad (22)$$

4. Result Analysis and Discussion

The performance outcomes of the MFO-DCNNSC method takes place employing UCM [22] and AID [23] dataset and as reported.

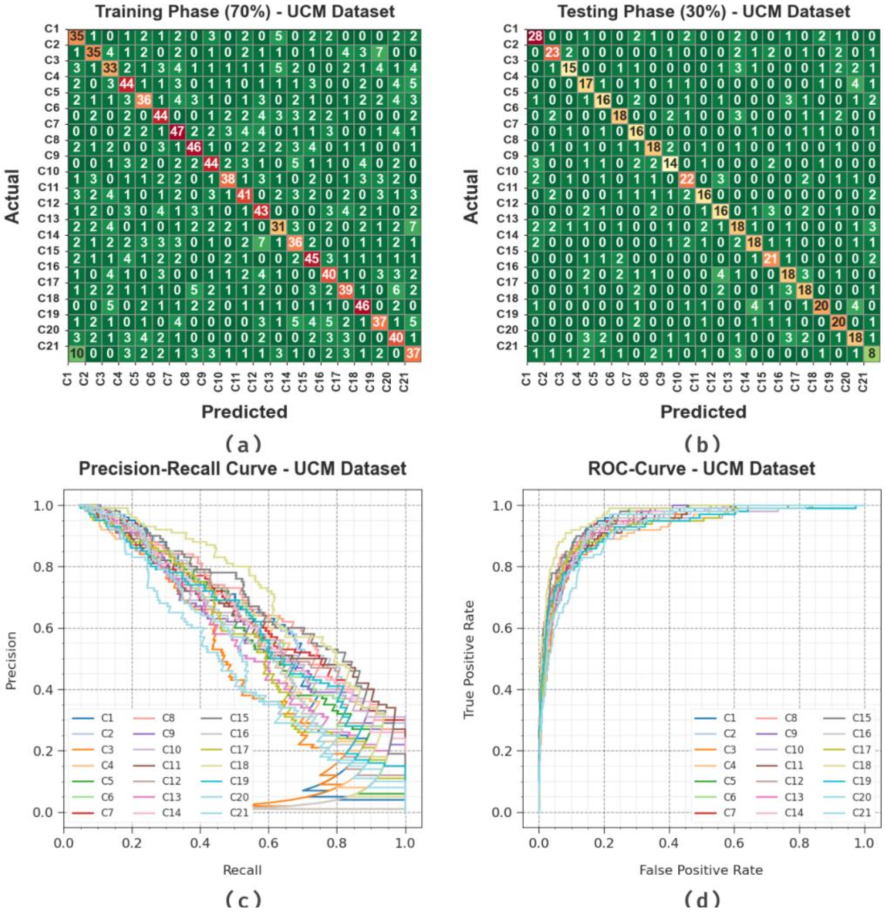

Fig 3. UCM dataset (a-b) Confusion matrices of MFO-DCNNSC model (c-d) PR-curve and ROC-curve

Fig. 3 displays the classifier results of the MFO-DCNNSC system at UCM database. Figs. 3a-3b showcases the confusion matrices accomplished by the MFO-DCNNSC technique with 70%:30% of TRAPH/TESPH. This figure point to that the MFO-DCNNSC technique can be identified and categorized with 21 class labels suitably. Concurrently, Fig. 3c indicate the PR result of the MFO-DCNNSC model. The figure described that the MFO-DCNNSC algorithm gives greater PR effectiveness with every class. Also, Fig. 3d showcases the ROC result of the MFO-DCNNSC method. This figure denoted that the MFO-DCNNSC method provides capable results with increased ROC values with diverse class labels.

A wide-ranging scene classification results provided by the MFO-DCNNSC method on the 70% of TRAPH under UCM dataset as reported in Table 1. These experimentation outcomes highlights that the MFO-DCNNSC method correctly identified all forms of scenes existed in the UCM dataset. It is also noticed that the MFO-DCNNSC algorithm has the ability of recognizing the samples with maximum classifier results.

Table 1: Scene classification of MFO-DCNNSC model with UCM dataset under 70% of TRAPH

Classes	**$\mathrm{Accu_y}$**	**$\mathrm{Prec_n}$**	**$\mathrm{Reca_l}$**	**$\mathrm{F_{score}}$**
TRAPH (70%)				
C1	95.78	48.61	58.33	53.03
C2	96.39	60.34	53.85	56.91
C3	95.03	48.53	46.48	47.48
C4	95.78	57.14	60.27	58.67
C5	95.85	58.06	50.70	54.14
C6	96.12	58.67	62.86	60.69
C7	95.44	56.63	60.26	58.39
C8	96.67	66.67	63.89	65.25
C9	96.19	61.97	60.27	61.11
C10	97.01	69.09	58.46	63.33
C11	95.85	59.42	55.41	57.34
C12	95.37	51.81	60.56	55.84

C13	95.85	53.45	47.69	50.41
C14	95.58	53.73	51.43	52.55
C15	96.12	59.21	63.38	61.22
C16	96.19	57.97	59.70	58.82
C17	95.99	60.00	54.17	56.93
C18	96.94	64.79	69.70	67.15
C19	95.58	56.06	50.68	53.24
C20	95.65	51.95	59.70	55.56
C21	94.49	46.84	48.68	47.74
Average	**95.90**	**57.19**	**56.98**	**56.94**

The overall scene classification results offered by the MFO-DCNNSC technique with 30% of TESPH at UCM datasets as displayed Table 2. These accomplished outcomes underscores that the MFO-DCNNSC system appropriately recognized all categories of scenes present in the UCM dataset. It is also observed that the MFO-DCNNSC algorithm can be the capability of recognizing the samples with boosted classifier outcomes.

Table 2: Scene classification of the MFO-DCNNSC system under UCM dataset with 30% of TESPH

Class	**$Accu_y$**	**$Prec_n$**	**$Reca_l$**	**F_{score}**
TESPH (30%)				
C1	96.19	70.00	70.00	70.00
C2	96.67	71.88	65.71	68.66
C3	96.83	71.43	51.72	60.00
C4	95.87	51.52	62.96	56.67
C5	96.51	64.00	55.17	59.26
C6	96.67	66.67	60.00	63.16
C7	96.35	48.48	72.73	58.18
C8	96.83	64.29	64.29	64.29
C9	96.19	56.00	51.85	53.85
C10	96.19	66.67	62.86	64.71
C11	96.98	64.00	61.54	62.75
C12	96.03	57.14	55.17	56.14
C13	94.44	50.00	51.43	50.70
C14	96.35	62.07	60.00	61.02
C15	96.98	65.62	72.41	68.85
C16	95.24	54.55	54.55	54.55
C17	96.19	56.25	64.29	60.00
C18	96.51	71.43	58.82	64.52
C19	96.98	62.50	74.07	67.80
C20	95.24	54.55	54.55	54.55
C21	94.76	32.00	33.33	32.65
Average	**96.19**	**60.05**	**59.88**	**59.63**

The average classifier study of the MFO-DCNNSC technique with UCM dataset is computed and described in Fig. 4. This figure demonstrated that the MFO-DCNNSC method accomplished increased results. Based on 70% of TRAPH, the MFO-DCNNSC method get average $accu_y$, $prec_n$, $reca_l$, and F_{score} of 95.90%, 57.19%, 56.98%, and 56.94%, correspondingly. Similarly, with 30% of TESPH, the MFO-DCNNSC algorithm offers average $accu_y$, $prec_n$, $reca_l$, and F_{score} of 96.19%, 60.05%, 59.88%, and 59.63%.

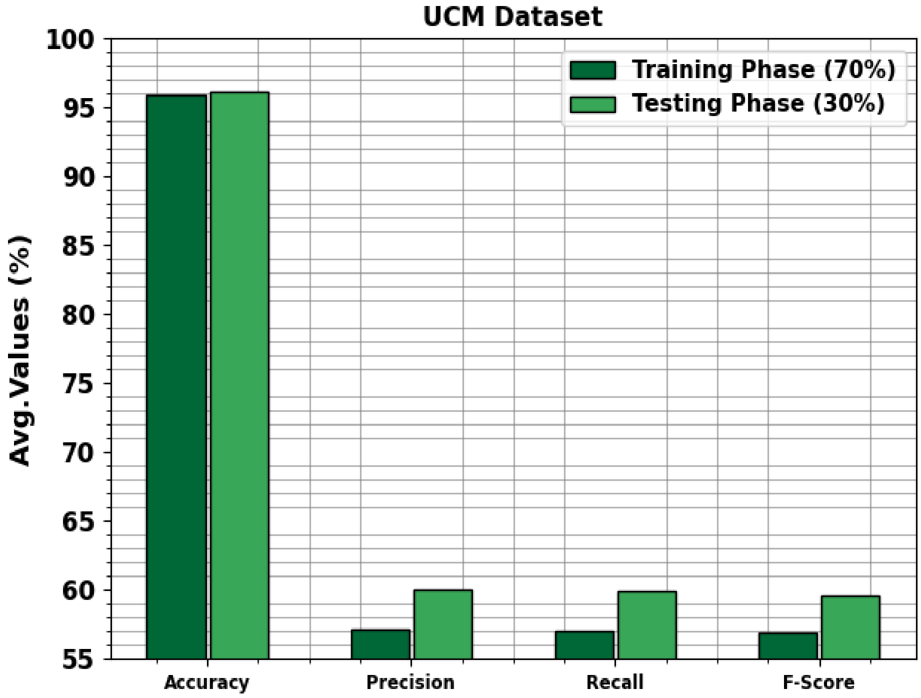

Fig 4. Average of the MFO-DCNNSC method under UCM dataset

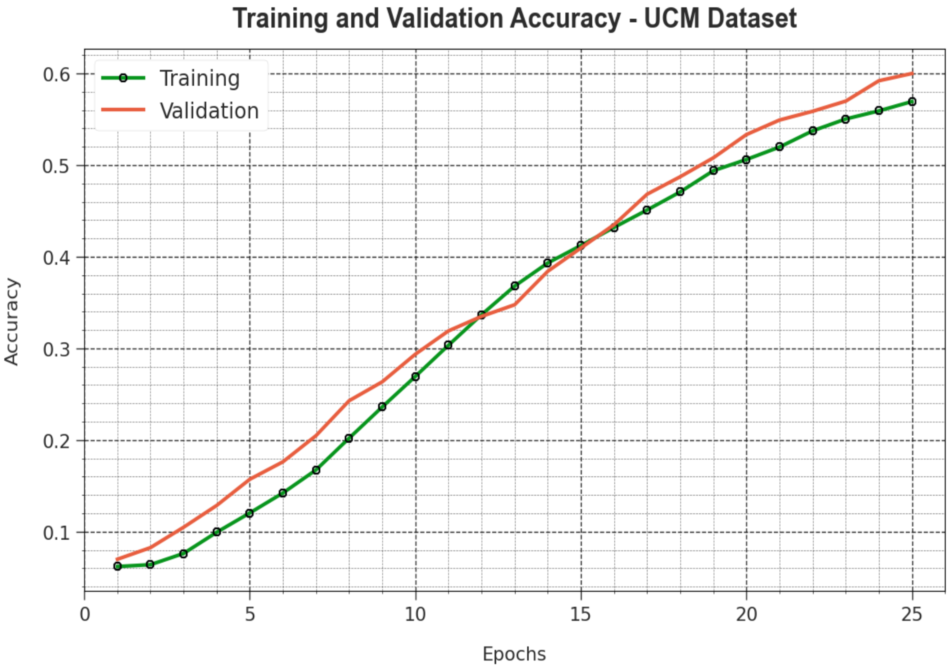

Fig 5. Accu$_y$ curve of the MFO-DCNNSC method at UCM dataset

The effectiveness of the MFO-DCNNSC algorithm with UCM dataset is clearly demonstrated in Fig. 5 in the usage of training accuracy (TRAA) and validation accuracy (VALA) curves. This figure represents useful analysis into the behaviour of the MFO-DCNNSC method over diverse epoch counts, signifying its learning process and generalization capabilities. Noticeably, the figure infer a continuous improvement in the TRAA and VALA with a growth in epochs. It make sure the adaptive nature of the MFO-DCNNSC technique with pattern recognition process under the TRA and TES data. The arising trend in VALA outlines the capability of the MFO-DCNNSC method on altering to the TRA data and also surpassing in providing correct classification on unnoticed data, pointing out the robust generalization abilities.

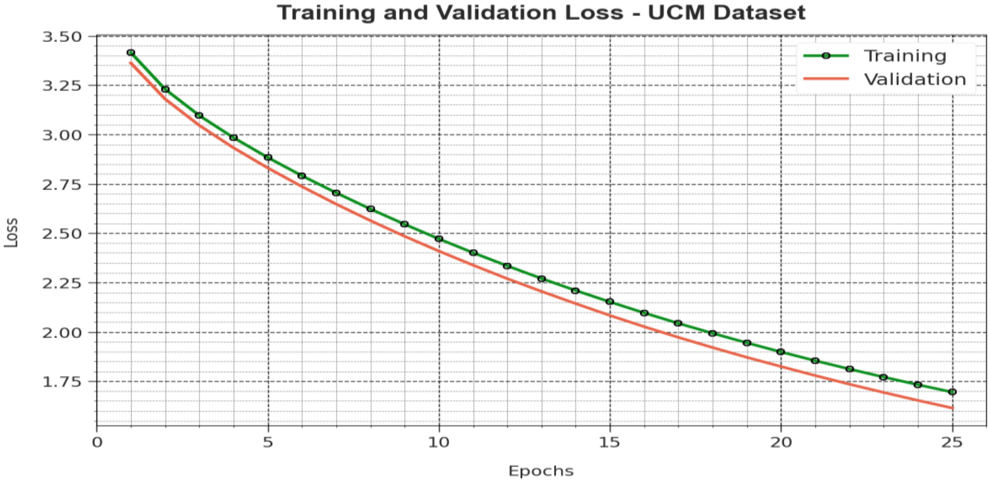

Fig 6. Loss curve of the MFO-DCNNSC technique with UCM dataset

Fig. 6 illustrates an extensive representation of the training loss (TRLA) and validation loss (VALL) results of the MFO-DCNNSC method with UCM dataset over distinct epochs. The progressive diminishes in TRLA highpoints the MFO-DCNNSC system optimizing the weights and decreasing the classification error on the TRA and TES data. The figure indicate a clear understanding into the MFO-DCNNSC model relevant to the TRA data, highlighting its proficiency in capturing patterns within both datasets. Mainly, the MFO-DCNNSC system incessantly increases its parameters in lessening the differences among the prediction and real TRA class labels.

The Accu_ycomparison study of the MFO-DCNNSC method under UCM dataset can be determined and represented in Table 3 and Fig. 7 [24, 25]. These obtained findings painted that the SC+Pooling, SG+UFL, and CCM-BOVW methods are shown minimized accu_y values of 81.67%, 86.64%, and 86.64%, correspondingly. Meanwhile, the PSR, COPD, and Dirichlet systems have reported closer accu_y values of 89.10%, 91.33%, and 92.80%. But, the MFO-DCNNSC algorithm gains higher results with increased accu_y of 96.19%.

Table 3: Accu_y outcome of the MFO-DCNNSC system with other algorithms under UCM dataset

UCM Dataset	
Method	**Accuracy (%)**
SC+Pooling	81.67
SG+UFL	86.64
CCM-BOVW	86.64
PSR Model	89.10
COPD Model	91.33
Dirichlet	92.80
MFO-DCNNSC	96.19

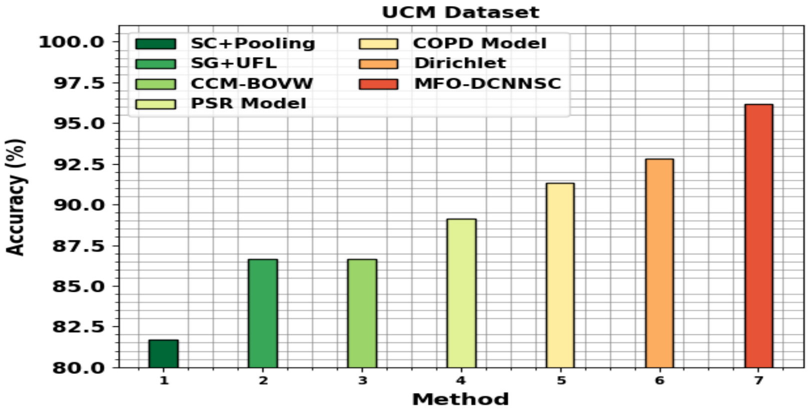

Fig 7. Accu_y outcome of the MFO-DCNNSC model with UCM dataset

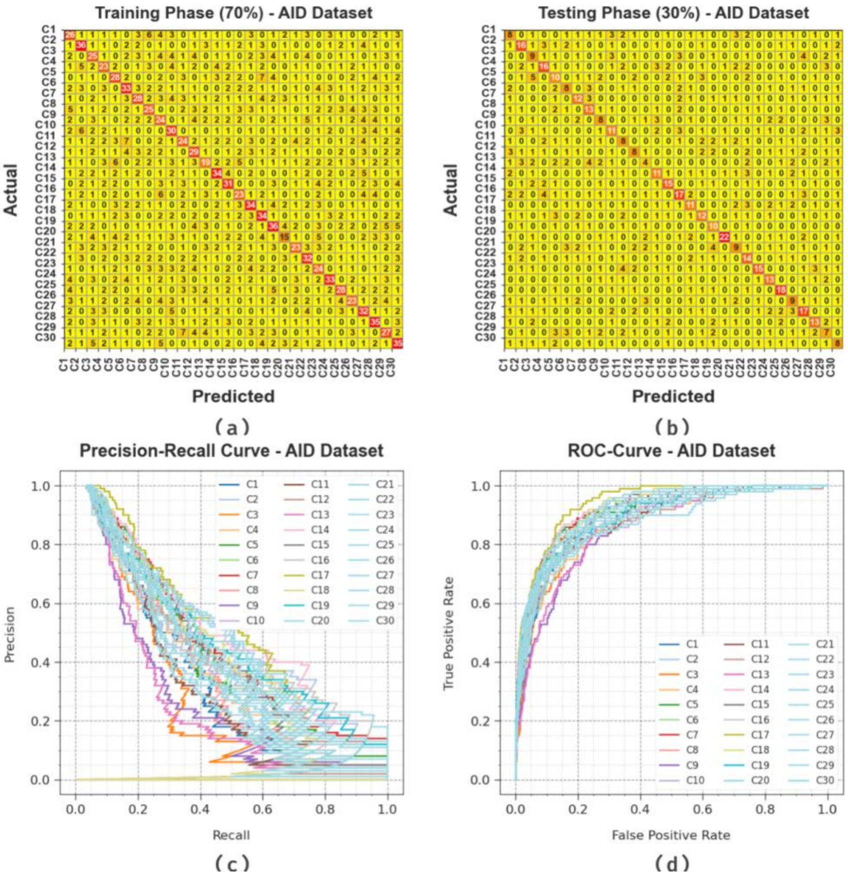

Fig 8. AID dataset (a-b) Confusion matrices of MFO-DCNNSC system (c-d) PR-curve and ROC-curve

Fig. 8 showcases the classifier results of the MFO-DCNNSC method at AID database. Figs. 8a-8b displays the confusion matrices succeeded by the MFO-DCNNSC system with 70%:30% of TRAPH/TESPH. This figure detailed that the MFO-DCNNSC algorithm can be identified and categorized with 30 class labels correctly. Meanwhile, Fig. 8c denotes the PR result of the MFO-DCNNSC model. The figure displays that the MFO-DCNNSC algorithm offers exceptional PR effectiveness with every class. In conclusion, Fig. 8d displays the ROC result of the MFO-DCNNSC technique. This figure characterized that the MFO-DCNNSC technique acquires efficient experimentation results with boosted ROC values with different class labels.

Table 4 presented the overall scene classification results accomplished by the MFO-DCNNSC technique on the 70% of TRAPH with AID dataset. These experimentation outcomes shown that the MFO-DCNNSC algorithm appropriately identified all categories of scenes existing in the UCM dataset. It is also noticed that the MFO-DCNNSC technique has the ability of recognizing the samples with maximal classifier results.

Table 4: Scene classification of MFO-DCNNSC model under AID dataset with 70% of TRAPH

Class	$\mathbf{Accu_y}$	$\mathbf{Prec_n}$	$\mathbf{Reca_l}$	$\mathbf{F_{score}}$
TRAPH (70%)				
C1	95.67	35.14	37.68	36.36
C2	96.57	46.75	53.73	50.00
C3	95.57	34.25	35.71	34.97
C4	96.19	35.94	37.10	36.51
C5	96.10	40.00	41.18	40.58
C6	95.76	41.77	43.42	42.58
C7	96.00	41.18	38.89	40.00
C8	95.95	40.98	33.78	37.04
C9	95.00	30.00	32.88	31.37
C10	96.10	40.00	44.78	42.25
C11	95.62	36.36	32.43	34.29
C12	95.71	36.71	42.03	39.19

C13	95.76	33.93	26.76	29.92
C14	96.52	47.89	48.57	48.23
C15	96.14	42.47	44.29	43.36
C16	96.71	41.07	38.98	40.00
C17	96.14	43.04	48.57	45.64
C18	96.19	43.59	48.57	45.95
C19	95.38	41.86	43.37	42.60
C20	96.57	36.59	24.59	29.41
C21	96.29	40.35	34.33	37.10
C22	96.33	43.84	47.06	45.39
C23	96.05	36.92	36.36	36.64
C24	96.52	51.56	44.00	47.48
C25	96.38	48.28	37.84	42.42
C26	95.90	38.98	31.51	34.85
C27	96.05	38.55	50.00	43.54
C28	96.05	42.68	49.30	45.75
C29	95.90	39.13	38.03	38.57
C30	95.71	41.67	46.05	43.75
Average	**96.03**	**40.38**	**40.39**	**40.19**

An extensive scene classification results succeeded by the MFO-DCNNSC algorithm with the 30% of TESPH under AID dataset and as revealed in Table 5. These experimental results underlines that the MFO-DCNNSC algorithm suitably recognized all categories of scenes existing in the AID dataset. It is also detected that the MFO-DCNNSC method can be the ability of recognizing the samples with higher classifier results.

Table 5: Scene classification of MFO-DCNNSC method with AID dataset under 30% of TESPH

Class	**Accu**$_{\mathbf{y}}$	**Prec**$_{\mathbf{n}}$	**Reca**$_{\mathbf{l}}$	**F**$_{\mathbf{score}}$
TESPH (30%)				
C1	95.00	26.67	25.81	26.23
C2	95.33	39.02	48.48	43.24
C3	95.11	28.12	30.00	29.03
C4	95.44	45.71	42.11	43.84
C5	94.89	29.41	31.25	30.30
C6	96.22	30.77	33.33	32.00
C7	96.22	40.00	42.86	41.38
C8	96.44	40.62	50.00	44.83
C9	95.89	30.77	29.63	30.19
C10	95.22	34.38	33.33	33.85
C11	95.67	27.59	30.77	29.09
C12	95.89	36.36	25.81	30.19
C13	96.00	26.67	13.79	18.18
C14	95.67	35.48	36.67	36.07
C15	96.78	51.72	50.00	50.85
C16	95.56	51.52	41.46	45.95
C17	96.00	39.29	36.67	37.93
C18	95.89	38.71	40.00	39.34
C19	97.11	34.48	58.82	43.48
C20	97.11	70.97	56.41	62.86
C21	95.33	33.33	27.27	30.00
C22	95.22	35.90	43.75	39.44
C23	97.00	65.22	44.12	52.63
C24	97.00	46.43	52.00	49.06
C25	97.00	48.65	69.23	57.14
C26	95.78	31.03	33.33	32.14
C27	95.56	44.74	47.22	45.95
C28	96.67	48.15	44.83	46.43

C29	95.00	23.33	24.14	23.73
C30	96.22	30.77	33.33	32.00
Average	**95.94**	**38.86**	**39.21**	**38.58**

In Fig. 9, the average classifier analysis of the MFO-DCNNSC system with AID dataset is revealed. This figure indicates that the MFO-DCNNSC algorithm get improved results. Based on 70% of TRAPH, the MFO-DCNNSC method get average accu$_y$, prec$_n$, reca$_l$, and F$_{score}$ of 96.03%, 40.38%, 40.39%, and 40.19%. Also, based 30% of TESPH, the MFO-DCNNSC system gives average accu$_y$, prec$_n$, reca$_l$, and F$_{score}$ of 95.94%, 38.86%, 39.21%, and 38.58%.

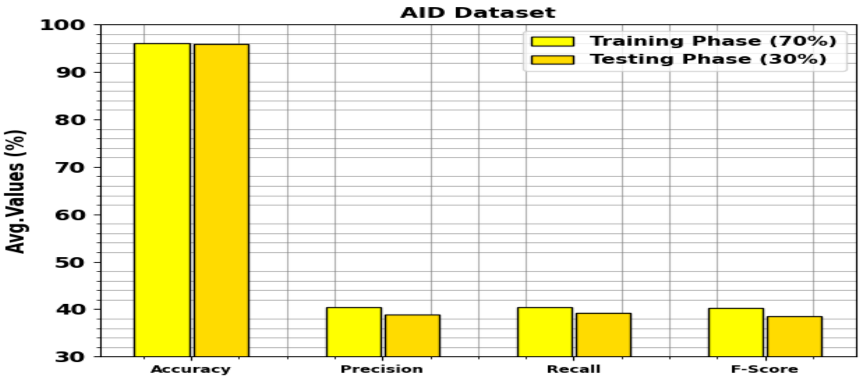

Fig 9. Average of the MFO-DCNNSC system with AID dataset

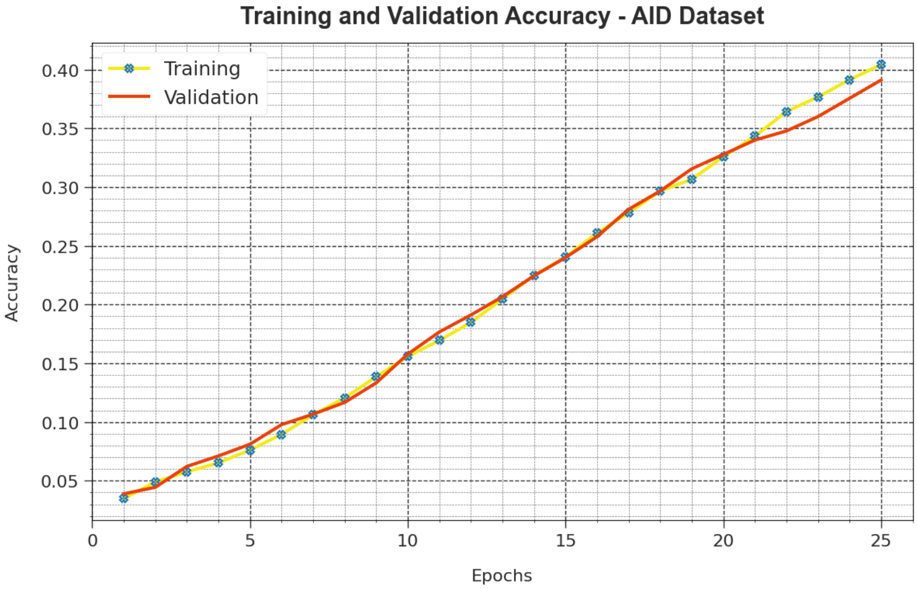

Fig 10. Accu$_y$ curve of the MFO-DCNNSC model at AID dataset

The efficiency of the MFO-DCNNSC system with AID dataset is graphically demonstrated in Fig. 10 in the form of TRAA and VALA curves. This figure exhibit useful analysis into the behaviour of the MFO-DCNNSC method over multiple epoch counts, demonstrating its learning process and generalization capabilities. Noticeably, the figure infer a continuous enhancement in the TRAA and VALA with a progress in epochs. It make sure the adaptive nature of the MFO-DCNNSC method with pattern recognition process at TRA and TES data. The arising trend in VALA outlines the capability of the MFO-DCNNSC technique on changing to the TRA data and exceling in offering particular classification on undetected data, shows the robust generalization abilities.

Fig. 11 displays an extensive representation of the TRLA and VALL results of the MFO-DCNNSC method with AID dataset over varying epochs. The progressive minimum in TRLA highpoints the MFO-DCNNSC algorithm optimizing the weights and lessening the classification error on the TRA and TES data. The figure indicate a perfect understanding into the MFO-DCNNSC model related to the TRA data, highlighting its proficiency in capturing patterns within both datasets. Significantly, the MFO-DCNNSC technique continually enhances its parameters in decreasing the variances among the prediction and real TRA class labels.

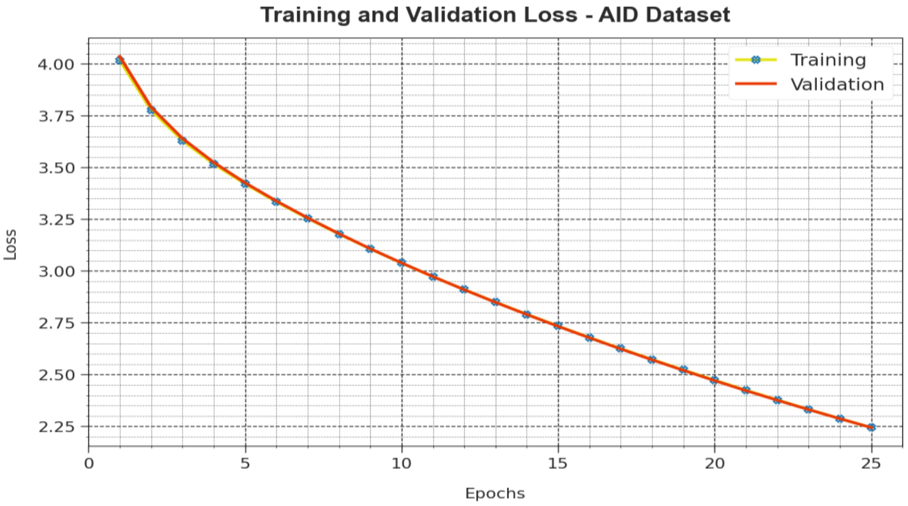

Fig 11. Loss curve of the MFO-DCNNSC method with AID dataset

An extensive $Accu_y$ comparative analysis of the MFO-DCNNSC algorithm with AID dataset is determined as exhibited in Table 6 and Fig. 12. These experimentation outcomes specify that the GoogleNet and VGG-VD-16 methods acquire lessened $accu_y$ values of 86.39%, and 89.64%. Meanwhile, the ResNet50, ResNet-50+EAM, and LCNN-BFF algorithms are achieved remarkable $accu_y$ values of 92.57%, 93.64%, and 91.66%. However, the MFO-DCNNSC method gains excellent results with improved $accu_y$ of 96.03%, respectively.

Table 6: $Accu_y$ outcome of MFO-DCNNSC model with other algorithms under AID dataset

AID Dataset	
Methods	**Accuracy (%)**
GoogleNet	86.39
VGG-VD-16	89.64
ResNet50	92.57
ResNet-50+EAM	93.64
LCNN-BFF	91.66
MFO-DCNNSC	96.03

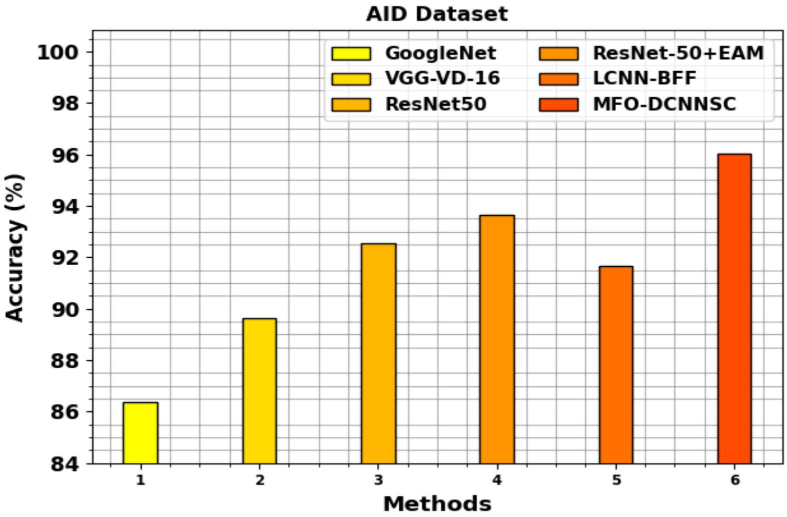

Fig 12. $Accu_y$ outcome of the MFO-DCNNSC method under AID dataset

Thus, the MFO-DCNNSC technique can be applied for automated scene classification on the RSI.

5. CONCLUSION

In this research, we have projected an MFO-DCNNSC model on RSI. The foremost intention of the MFO-DCNNSC method encompasses different kinds of processes namely WF-based image preprocessing, Inception-ResNetv2 based feature extractor, MFO-based hyperparameter tuning, and DBN-based classification process. Primarily, the MFO-DCNNSC technique on RSI undergoes preprocessing using WF to improve the image quality. To extract features, the MFO-DCNNSC technique applies the Inception-ResNetv2 model for learning the hierarchical depiction of the visual data. Following, the MFO algorithm is useful for the appropriate range of the hyperparameters connected to the Inception-ResNetv2 technique. Finally, the scene classification procedure is executed by the application of the DBN approach. To point out the enhanced performance of the MFO-DCNNSC technique, an extensive sort of experiments have been implemented on the benchmark dataset. The obtained values pointed out that the MFO-DCNNSC technique outperforms the other models in terms of distinct measures.

REFERENCES

[1] Sivasubramanian, A., Prashanth, V.R., Hari, T., Sowmya, V., Gopalakrishnan, E.A. and Ravi, V., 2024. Transformer-based convolutional neural network approach for remote sensing natural scene classification. Remote Sensing Applications: Society and Environment, 33, p.101126.

[2] Chen, Z., Yang, J., Feng, Z. and Chen, L., 2022. RSCNet: An Efficient Remote Sensing Scene Classification Model Based on Lightweight Convolution Neural Networks. Electronics, 11(22), p.3727.

[3] Karthick, V., Aiswarya, M.C., Jayavarshini, G.R. and Sruthi, R., 2023. Remote Sensing Scene Classification Using Convolutional Neural Network. International Journal of Research in Engineering, Science and Management, 6(4), pp.34-38.

[4] Yu, C.C., Chen, T.Y., Hsu, C.W. and Cheng, H.Y., 2024. Incremental Scene Classification Using Dual Knowledge Distillation and Classifier Discrepancy on Natural and Remote Sensing Images. Electronics, 13(3), p.583.

[5] Sadulla, Shaik. "Next-Generation Semiconductor Devices: Breakthroughs in Materials and Applications." Progress in Electronics and Communication Engineering 1.1 (2024): 13-18.

[6] Siddiqui, M.I., Khan, K., Fazil, A. and Zakwan, M., 2023. Snapshot ensemble-based residual network (SnapEnsemResNet) for remote sensing image scene classification. GeoInformatica, 27(2), pp.341-372.

[7] Shi, C., Zhang, X., Wang, T. and Wang, L., 2022. A Lightweight Convolutional Neural Network Based on Hierarchical-Wise Convolution Fusion for Remote-Sensing Scene Image Classification. Remote Sensing, 14(13), p.3184.

[8] Dong, Z., Lin, B. and Xie, F., 2024. Optimizing Few-Shot Remote Sensing Scene Classification Based on an Improved Data Augmentation Approach. Remote Sensing, 16(3), p.525.

[9] Hou, Y.E., Yang, K., Dang, L. and Liu, Y., 2023. Contextual Spatial-Channel Attention Network for Remote Sensing Scene Classification. IEEE Geoscience and Remote Sensing Letters.

[10] Li, D., Liu, R., Tang, Y. and Liu, Y., 2024. PSCLI-TF: Position-Sensitive Cross-Layer Interactive Transformer Model for Remote Sensing Image Scene Classification. IEEE Geoscience and Remote Sensing Letters.

[11] Zhao, Y., Liu, J., Yang, J. and Wu, Z., 2023. EMSCNet: Efficient Multisample Contrastive Network for Remote Sensing Image Scene Classification. IEEE Transactions on Geoscience and Remote Sensing, 61, pp.1-14.

[12] Tian, F., Lei, S., Zhou, Y., Cheng, J., Liang, G., Zou, Z., Li, H.C. and Shi, Z., 2024. HiReNet: Hierarchical-Relation Network for Few-Shot Remote Sensing Image Scene Classification. IEEE Transactions on Geoscience and Remote Sensing.

[13] Shi, C., Zhang, X., Sun, J. and Wang, L., 2022. Remote sensing scene image classification based on self-compensating convolution neural network. Remote Sensing, 14(3), p.545.

[14] Jajarmi, Amin, Samaneh Sadat Sajjadi, and Ahamad Hajipour. "Steam generator identification using piecewise affine model." Results in Nonlinear Analysis 2.4 (2019): 149-159.

[15] Shi, C., Zhang, X., Wang, L. and Jin, Z., 2023. A lightweight convolution neural network based on joint features for Remote Sensing scene image classification. International Journal of Remote Sensing, 44(21), pp.6615-6641.

[16] Zhang, K., Cui, T., Wu, W., Zheng, X. and Cheng, G., 2024. Large Kernel Separable Mixed ConvNet for Remote Sensing Scene Classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

[17] Wei, L., Geng, C. and Yin, Y., 2023. Remote Sensing Image Scene Classification Based on Head-Tail Global Joint Dual Attention Discrimination Network. IEEE Access.

[18] Yue, H., Qing, L., Zhang, Z., Wang, Z., Guo, L. and Peng, Y., 2024. MSE-Net: A novel master–slave encoding network for remote sensing scene classification. Engineering Applications of Artificial Intelligence, 132, p.107909.

[19] Shi, C., Ding, M., Wang, L. and Pan, H., 2023. Learn by Yourself: A Feature-Augmented Self-Distillation Convolutional Neural Network for Remote Sensing Scene Image Classification. Remote Sensing, 15(23), p.5620.

[20] Sheer, A.H. and Al-Ani, A.A., 2018, November. The effect of regularization parameter within non-blind restoration algorithm using modified iterative wiener filter for medical image. In 2018 1st Annual International Conference on Information and Sciences (AiCIS) (pp. 77-81). IEEE.

[21] Sanampudi, A. and Srinivasan, S., 2024. Local search enhanced optimal Inception-ResNet-v2 for classification of long-term lung diseases in post-COVID-19 patients. Automatika, 65(2), pp.473-482.

[22] Alqahtani, N., Alam, S., Aqeel, I., Shuaib, M., Mohsen Khormi, I., Khan, S.B. and Malibari, A.A., 2023. Deep belief networks (DBN) with IoT-based Alzheimer's disease detection and classification. Applied Sciences, 13(13), p.7833.

[23] Yan, Y., Hu, Z., Yuan, W. and Wang, J., 2023. Pipeline leak detection based on empirical mode decomposition and deep belief network. Measurement and Control, 56(1-2), pp.396-402.

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

[25] https://captain-whu.github.io/AID/

[26] Zhao, Y., Liu, J., Yang, J. and Wu, Z., 2022. Remote Sensing Image Scene Classification via Self-Supervised Learning and Knowledge Distillation. Remote Sensing, 14(19), p.4813.

[27] Rajagopal, A., Joshi, G.P., Ramachandran, A., Subhalakshmi, R.T., Khari, M., Jha, S., Shankar, K. and You, J., 2020. A deep learning model based on multi-objective particle swarm optimization for scene classification in unmanned aerial vehicles. IEEE Access, 8, pp.135383-135393.