Development of Image Processing and AI Model for Drone Based Environmental Monitoring System

[1]Cuddapah Anitha, [2]Shivali Devi, [3]Vinay Kumar Nassa, [4]Mahaveerakannan R, [5]Kingshuk Das Baksi and [6]Suganthi D
[1]Department of Computer Science and Engineering, School of Computing, Mohan Babu University, (Erstwhile Sree Vidyanikethan Engineering College, Tirupati-517102, Andhra Pradesh, India.
[2]Department of Computer Science and Engineering, Chandigarh University, Kerala, India.
[3]Department of Information Communication Technology, Tecnia Institute of Advanced Studies, New Delhi, India.
[4]Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences (SIMATS), Chennai -602105, India.
[5]Department of Computer Science and Engineering, Ramgovind Institute of Technology, Koderma, Jharkhand, 825410, India.
[6]Department of Computer Science, Saveetha College of Liberal Arts and Sciences, Saveetha Institute of Medical and Technical Sciences (SIMATS), Chennai-602105, India.
[1]anithacuddapah@vidyanikethan.edu, [2]shivali.sd7@gmail.com, [3]vn.nassa@gmail.com, [4]mahaveerakannanr.sse@saveetha.com, [5]kingshukdasbakshi@gmail.com, [6]suganthiphd@gmail.com

Correspondence should be addressed to Mahaveerakannan R : mahaveerakannanr.sse@saveetha.com.

Article Info
Journal of Machine and Computing (http://anapub.co.ke/journals/jmc/jmc.html)
Doi: https://doi.org/10.53759/7669/jmc202404021
Received 02 June 2023; Revised from 25 August 2023; Accepted 30 November 2023.
Available online 05 January 2024.

Abstract – Data from environmental monitoring can be used to identify possible risks or adjustments to ecological patterns. Early detection reduces risks and lessens the effects on the environment and public health by allowing for prompt responses to ecological imbalances, pollution incidents, and natural disasters. Decision-making and analysis can be done in real time when Artificial Intelligence (AI) is integrated with Unmanned Aerial Vehicles (UAV) technology. With the help of these technologies, environmental monitoring is made possible with a more complete and effective set of tools for assessment, analysis, and reaction to changing environmental conditions. Multiple studies have shown that forest fires in India have been happening more often recently. Lightning, extremely hot weather, and dry conditions are the three main elements that might spontaneously ignite a forest fire. Both natural and man-made ecosystems are affected by forest fires. Forest fire photos are pre-processed using the Sobel and Canny filter. A Convolutional Neural Network (CNN)–based Forest Fire Image Classification Network (DFNet) using the publicly accessible Kaggle dataset is proposed in this study. The suggested DFNet classifier's hyperparameters are fine-tuned with the help of Spotted Hyena Optimizer (SHO). With a performance level of 99.4 percent, the suggested DFNet model outperformed the state-of-the-art models, providing substantial backing for environmental monitoring.

Keywords – Forest Fire Detection, Sobel Filter, Canny Filter, Deep Learning, Spotted Hyena Optimizer.

I. INTRODUCTION

Monitoring the environment is essential to comprehending and protecting the planet's health. It gives us vital information about the state of the air, water, soil, and biodiversity [1]. With this information, we can detect changes in the environment, evaluate the effects of human activity, and reduce risks [2]. The environment is greatly impacted by forest fires. They damage biodiversity, cause significant emissions of carbon dioxide into the atmosphere, and contribute to deforestation. Early detection enables more rapid containment and control, which helps to lessen these effects [3,4].

Forest fires put lives and property at serious risk to neighboring communities. Due to damage to timber, loss of wildlife habitat, destruction of property, and effects on tourism, forest fires can result in significant economic losses [5]. UAVs and AI together have the potential to industries because of their complementary advantages [6]. AI-capable UAVs provide unmatched efficiency for a variety of applications, including delivery services, agriculture, disaster relief, and surveillance and reconnaissance [7]. Their capacity to gather enormous volumes of real-time data from isolated or unreachable locations offers priceless information for making decisions [8].

AI enhances UAV performance through autonomous flight, accurate navigation, and data analysis, facilitating quick, accurate, and economical operations. important part in the detection of forest fires because of their advanced capabilities, speed, and agility [9]. In large and remote forested areas, these technologies allow for the premature and precise credentials of possible fire outbreaks [10]. UAVs with thermal imaging and AI-powered sensors can quickly cover large areas, spotting minute variations in temperature and smoke patterns that could point to possible fire hotspots [11]. These drones gather real-time data, which AI algorithms can process to quickly analyze and precisely pinpoint potential fire locations [12]. This early detection is crucial as it allows for immediate intervention, enabling firefighting teams to respond swiftly and effectively, minimizing the spread of wildfires, reducing environmental damage, and ultimately saving lives and valuable natural resources [13].

Main Contributions

- Forest Fire Image Pre-processing: Sobel and Canny filters are used to pre-process images taken during forest fires.
- DFNet: Deep Learning-Based Forest Fire Classification Network: Convolutional named DFNet is proposed and implemented in use for the purpose of classifying forest fire images using the Kaggle dataset, which is made available to the public.
- Optimization with Spotted Hyena Optimizer (SHO): Utilizing SHO to tune the DFNet classifier's hyperparameters.
- Performance Evaluation: Findings show that the suggested DFNet model outperforms baseline models, demonstrating its important support for environmental monitoring initiatives.

Organization of the Work

Remaining units with reference to the paper are arranged based on: In The second section, the essential literatures are reviewed, and in Section 3, the projected model is briefly discussed. The results and an overview of the validation process are included in Section 4. A conclusion is provided in Section 5 to finish.

II. RELATED WORKS

This study contributed to understanding of computer vision by evaluating a system for smoke and wildfire detection that use ensembles of several CNN architectures to sequentially address two separate objectives [14]. The proposed design integrated two YOLO weights with a voting ensemble CNN. The operation of the pipeline was divided into two parts. If the CNN detected an irregularity in the frame, the YOLO construction pinpointed the source of the smoke or fire. The tasks that were addressed in the provided technique were detection and classification. The acquired model's weights performed admirably throughout testing and training. A sensitivity level of 0.98, to the accuracy of 0.99, In addition, an F1 score of 0.95 were the results produced by the classification model. This is the model used a learning that can be transferred approach regard to the classification position. That model of the detector assessment yielded a powerful performance. The smoke detection model achieved an average precision score of 0.85 at the threshold value of 0.5 (mAP at 0.5), while model that incorporates both achieved 0.76 milliamperes. The F1-score for an example of the smoke detection model was also 0.93. The offered deep learning pipeline demonstrated promising experimental results with practical application potential, despite training-related challenges such as a dearth of high-quality real-world UAV-captured images of fire and smoke.

In [15] provided a comprehensive review of several methods, including transfer learning, deep convolutional neural networks (CNNs), with CNNs that are not heavy. Out of all the representations evaluated, using focus-based typical detected forest fires better than any of the others. Reiterating the efficacy of the EfficientNetB0-based model in spotting wildfires, the precision, recall, accuracy, and F1-score of the test all stood at 92.02%. The model was also shown to be applicable for wildfire detection utilizing UAVs with limited hardware, as the network's parameter size was less than the evaluated networks.

The purpose of the study by [16] was to look into and assess how well UAVs and deep learning have worked together to monitor and identify forest fires. Drones equipped with certain sensors and cameras allowed for immediate surveillance in the early stages of a fire at a low cost. This study presents a comprehensive review of recent developments in deep learning object recognition, focusing on their potential applications in forest fire monitoring. These developments include YOLO and variants of these algorithms. The findings of the trials showed promising outcomes in several parameters, turning it into a useful fire tool monitoring and detection.

A CNN based method for forest fire identification was introduced in a study by [17] using a new dataset for fire detection. Importantly, the method used standard convolution layers and separable convolution layers to identify fires instantly; this allowed it to run on less computing power and was therefore suitable for usage in real-time settings. The outcome of the experiments showed that the approach, after trained on the dataset, could accurately detect forest fires in photos with a 98.00% F1 Score, 80% Kappa, and 97.63% accuracy. Consequently, this method of identification has the potential to be a useful tool for detecting fire breakouts in real-world scenarios, allowing take precautions to minimize damage.

A use MS-FRCNN to minor identify potential forest fires was introduced by [18] as an improvement on the standard Faster RCNN target detection model. By replacing VGG-16 with ResNet50 serving as the support system for the Faster RCNN, the MS-FRCNN model was able to reduce the impact of VGG-16's gradient dispersion or explosion during feature extraction. After that, ResNet50's feature map was fed directly into the FPN (Feature Pyramid Network). The MS-FRCNN was able to collect more thorough feature data because to the benefit of extracting features from many scales for FPN. In the Regional Proposal Network (RPN), Concurrently, the MS-FRCNN made advantage of a new attention module named PAM. By simultaneously using channel attention and spatial attention, this module helped reduce the effect of complex visual backdrops. Because of this, the RPN was able to zero in on the precise semantics and geographic locations of localized forest fires. In addition, by utilizing a soft-NMS method instead of an NMS technique, the MS-FRCNN model was able to decrease clearing out the highlighted frames' mistakes. Results from experiments showed that the proposed MS-FRCNN outperformed the baseline model by 5.7% when it came to tracking down specific forest fires. It shown that the MS-FRCNN model's parallel attention mechanism and the multi-scale image feature extraction method could greatly make smaller-scale forest fire suppression efforts more effective finding by suppressing interference information.

The [19] used the following stages as their proposed YOLOv5 optimization strategy: 1) implementing ShuffleNetV2 as the backbone network; 2) coordinating the pruning of the Head and Neck network with the backbone baseline; 3) carrying out the model-pruning technique through sparse training; 4) restoring detection accuracy through fine-tuning of the pruned network; and 5) increasing the algorithm's inference speed through hardware acceleration by overclocking the RPi4B. When compared to the state-of-the-art, which was running on a Raspberry Pi single board computer, the suggested method lowered CPU utilization by 35%, temperature by 25%, and power consumption by 10%. It also achieved 50% greater inference speed, 9 FPS. The accuracy (92.5%) was only marginally degraded. These results corroborate the suggested forest fire detection system's trial findings. And lastly, the accuracy of the projected algorithm was unaffected by changes to the bird's-eye view angle.

Research Gap

A notable research lacuna evident in all the models showcased is the requirement for a greater variety and caliber of real-world datasets, especially those obtained through unmanned aerial vehicles. The models exhibit remarkable results, proving the effectiveness of different deep learning architectures, including CNN, Efficient Net, and YOLO, as well as customized versions like MS-FRCNN and YOLOv5. Nevertheless, the robustness and generalizability of these models in real-world scenarios may be impacted because they were trained and assessed on datasets that may not have been as diverse and authentic as those captured by UAVs. Closing this gap could improve the models' accuracy and adaptability, as well as their reliability in real-time fire detection scenarios. This could be achieved by curating comprehensive datasets with UAV-captured images of varying environmental conditions, fire intensities, and landscapes. Improved generalizing, better handling of real-world complexities, and greater accuracy in identifying and localizing fires in diverse environmental conditions are some of the benefits of models that use diverse datasets captured by UAVs. Ultimately, these benefits would bolster the effectiveness of forest fire detection systems.

III. PROPOSED METHODOLOGY

This research suggests for soil data classification. **Fig 1** depicts the flow of the suggested approach's steps. The initial step is image pre-processing with Sobel and Canny filter, which is followed by DFNet classification and hyperparameter tuning using SHO.

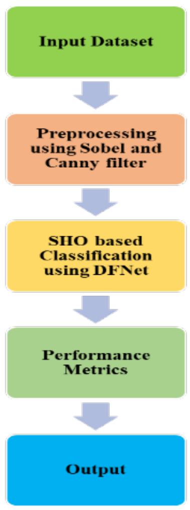

Fig 1. Shows the Model's Expected Workflow.

Description of the Dataset

No standardized dataset exists for photographs of fires. Therefore, making one is inescapable. The study used the developed UAV to capture photos, which were then combined with frames from forest fire movies found on kaggle and FLAME. There were two sections in the dataset: fire and non-fire. There are a total of 677 photographs in the non-fire dataset and 5313 images in this fire dataset. One technique to improve a model's accuracy is to add more pictures from live video streams to the dataset. This way, the dataset may grow over time, and the model's accuracy will improve with each training iteration. You may access the dataset at this link: [20].

Pre-processing using Sobel and Canny filter

Sobel Filter

For example, sharp edges can be better understood by the use of the Sobel filter, which is comprised of operators that apply gradients [21]. Two convolution layers, one of which is a modified counterpart of the other, make up a Sobel edge detector. The two layers are shown in the following way by equations (1) and (2):

$$S_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} \times A \tag{1}$$

and

$$S_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \times A \tag{2}$$

given that A is the initial picture. P_x responds to edges finished x-axis while P_y to the y-axis. By combining these two metrics in equations (3) and (4), we can determine the absolute size and direction of each position:

$$P = \sqrt{P_x^2 + P_y^2} \tag{3}$$

$$\Theta = \arctan\left(\frac{G_y}{G_x}\right) \tag{4}$$

Canny Filter

As a result of its characteristics, Canny has been one of the most researched and used filters. Here are the key points of the clever filter: (1) Reduce the amount of noise in photos from the Fire dataset by applying a Gaussian filter. The second step is to apply a threshold and see if the discovered edges are likely accurate or inflated. The default 5×5 Gaussian filter in the experiment is also the most popular:

$$C = \frac{1}{159} \times \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \times A \tag{5}$$

Classification Using DFNet

Table 1. All of the Constraints that are Part of the Suggested Dfnet Model

Layer Name Shape of the Output	Limits
Layer for Input (None,150,150,3,3)	01
Section 04 (None, 18, 18, 64)	19,496
Fifth Block (None, 9, 91, 128)	74,856
Withdrawal 1 (None, 4, 41, 128)	2
Lay Flat (None ,2048)	1
Section 01 (None,150,150,8)	234
Block02 (None ,75,75,16)	1268
Block03 (None ,37,37,32)	4740
Dense_1 (None ,512)	1,149,088
ReLu (None ,512)	0
Dense_2 (None ,4)	2052
Output: SoftMax (None ,4)	0
Non -Trainable Limits	0
Total Parameters	1,149,524

The suggested DFNet model is detailed in this section. Computer vision applications benefit greatly from the CNN assembly, which is modelled after the anatomy of the human brain. Translation, also known as spatial invariance, states that a convolutional neural network (CNN) can recognize the same feature in different pictures despite their relative placements [22–24]. In order to properly categorize skin cancer disorders, our study built a CNN-based model that is resilient. With a activation function, one dropout layer, two dense layer, the DFNet model is comprised of five convolutional blocks. Please refer to **Table 1** for the parameters and kind of layer.

Convolutional Blocks of CNN Perfect

A convolutional 2D, a pooling 2D with a max value, and a ReLU make up each convolutional block, which is the basic building block of the work that is given. To allocate kernel weights to layers, the LecunUniformV2 initializer for the kernel layer is constructed. By utilizing the activation function of ReLU, the gradient-vanishing issue is resolved, and the network is facilitated in its understanding and timely execution of tasks.

The input image has RGB channels. The convolutional layer is the first layer of our model. This layer applies filters, sometimes called the kernel, to start the process off. As seen in equation (6), the size of the kernel depends on two variables.

$$\text{Filter Size } (FS) = f_w \times f_h \tag{6}$$

where f_w represents the width of the filter and f_h signifies the filter. In our study, we set the size of the filter to 3, hence, the resulting equation is FS=3×3. These filters, which also go by the moniker "feature identifiers," let us make sense of the visual details at a basic level, such curves and edges.

Flattened Layer

Nestled between the thick and convolution layers is this particular layer. Convolution layers accept tensor data types as inputs, but dense layers require a one-dimensional architecture. In order to convert the picture representation into a one-dimensional input, the flattened layer was used.

Dropout Layer

This layer, with a 0.2 dropout value, was used by the suggested model. We included this number to make sure that our planned DFNet model wouldn't get overfit. To reduce the model's training time and difficulty, this layer was designed to turn units on and off. As a result, the model acquires knowledge of the pertinent attributes.

Dense Block of Proposed D

What follows is an explanation of how this study used two dense blocks made up of an activation function.

ReLU Function

The mathematical processes known as activation functions decide whether the output of one layer of neurons should be passed on to the next. Typically, they are responsible for turning the network nodes on and off. We utilized ReLU because of its humble and time-saving calculation, albeit DL classifiers need many activation functions. In order for ReLU to function, it must be activated by setting all negative outcomes to zero. The results from the convolutional layer were activated using this function.

Dense Layer

Using the properties of the input matrix, the dense layer produces output. Image recognition and classification take place in these levels. The model's final output, which assigns each picture to one of four functions. After a few layers, the probability-based activation function known as SoftMax is applied. In this function, the total sum of classes is equal to the sum of neurons. The total sum with 1,149,524 of them being trainable and zero of them being non-trainable.

Hyperparameter tuning using SHO

Here we go over the basics of a SHO, and then we'll take a quick look at the MO version of a spotted hyena optimizer. The basic concept of SHO is laid forth first, followed by the suggestion of the multi-objective version of SHO [25,26]. The algorithm was mostly influenced spotted hyenas hunt and interact with one another. The SHO mimics the coordinated social behavior of trustworthy spotted hyenas. There are four main parts of SHO: searching, encircling, hunting, and attacking. In order to store the best optimal solutions and guide the toward the most effective search agent, the SHO algorithm employs a group of trustworthy individuals. To simulate the spotted hyena's circling behavior, one uses the following formula:

$$D_h = \left|\vec{B} \cdot \vec{X}_{p(x)} = \vec{X}(x)\right| \tag{7}$$

$$\vec{X}(x+1) = \vec{X}_{p(x)} = \vec{E} \cdot \vec{D}_k \tag{8}$$

where $\vec{D}_h$ stands for the sepuration among both are expesented by $\vec{X}$, and X, correspondingly, and 11 is the absolute value. The vectors ane $\vec{B}$ and $\vec{E}$. The formulas used to evaluate $\vec{B}$ and $\vec{E}$ are calculated in equation (9), (10) and (11) as shadows:

$$\vec{B} = 2 \cdot \vec{d_1} \tag{9}$$
$$\vec{E} = 2\vec{\hat{h}} \cdot \overrightarrow{d_2} - \vec{h} \tag{10}$$
$$\vec{\mathrm{h}} - 5 - \left(\text{Iteration } \times \frac{5}{\text{Max}_{\text{ileretien}}} \right) \tag{11}$$

the iterative process occurs when minimum to extreme repetition. Throughout the iterations from 5 to 0, $\vec{\hat{h}}$ demonstrated gradual decreases. Arrays of random variables $\overrightarrow{h_1}$ and $\overrightarrow{h_2}$ for the values that ranged from 0 to 1. By fluctuating the values of vectoes $\vec{B}$ and $\vec{E}$, other locations could be attained comparative to the current focation. In the meantime, this system saved the best response and encouraged other search agents to refine their positions. In order to find good random search locations, we used the following equations (12) through (14) to model the hunting behavior of spotted hyenas.

$$\vec{D}_k - \left|\vec{B}^2 \cdot \vec{X}_k - \vec{X}_k\right| \tag{12}$$
$$\vec{X}_k - \vec{X}_k - \vec{E}^2 \cdot \vec{D}_k \tag{13}$$
$$\vec{C}_k - \vec{X}_{k+} + \vec{X}_{k+1} + \cdots + \vec{X}_{k+N} \tag{14}$$

where N is the sum ot repetitions, which may be computed in equation (15) as follows:

$$\vec{x}_{(x+1)=} \frac{\vec{c}_B}{N} \tag{15}$$

in where nos are the set of all solutions that are very comparable to the optimal one in the provided space of solutions, A cluster of optimum solutions is represented by Ck. M represents a zany vector between half a million and one million. In light of the fact that this task was finished successfully, X((x+1)) modifies the placement of further search agents and retrieves the N most optimal replies.

Fig 1 shows that the necessary exploration which were met by means of vector Eof arbitrary magnitudes greater than or equal to 1 require the search contenders to withdraw from the target. Another part of SHO, B is involved in exploration change [2]. It provides the prey's weight and can take on values between 0 and 5. **Fig 1** shows that the SHO approach may be utilized to attack the prey by meeting the poey's anticipated position when |E |<1.

Before optimizing anything, the approach generates a set of arbitrary responses to use since. All the search agents work together during optimization on the best search engine.

Nevertheless, when the number of repeats increases, vectors h and E continue to decrease. The clustered placements of candidate solutions are considered optimal when the termination condition is satisfied. When tuning optimization problems arise, the SHO approach can fix them. SHO outperforms the present metaheuristics because of its massive exploration capacity.

IV. RESULTS AND DISCUSSION

Experimental Setup

Keras was used to implement the proposed model. Another accomplishment was the use of Python for the scripting of methods unrelated to convolutional networks. A Windows 10 PC with an NVIDIA GPU measuring 11 GB and 32 GB of RAM was used to conduct the experiment.

Performance Metrices

This paper calculated four main analytical metrics:
$true\ positive\ (TP), true\ negative\ (TN), false\ positive\ (FP), and\ false\ negative\ (FN)$ to assess the efficacy of the classification system created using the datasets.

When evaluating the efficacy of a classification model, ACC is described as the ratio of correct assumptions to total assumptions made in equation (16):

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{16}$$

PR, also known as positive predictive value in equation (17), measures the share of correctly detected positive examples to all positive examples:

$$Precision = \frac{TP}{TP+FP} \tag{17}$$

RC, also known as the true positive rate or sensitivity, is the percentage of correctly identified positive cases among all positive instances, as shown in equation (18).

$$Recall = \frac{TP}{TP+FP} \tag{18}$$

The F1 in equation (19) is an integrated metric that incorporates PR and RC into a single numerical value:

$$F1 = \frac{Precision*Recall}{Precision+Recall} \tag{19}$$

Classification Analysis

Table 2. Classification Analysis with Existing Models and Proposed Model

Models	ACC (%)	PR (%)	RC (%)	F1 (%)
AE	90.1	90.7	91.0	90.8
VAE	92.5	91.9	92.2	92.1
VGG-16	93.3	92.8	92.5	92.2
ResNet	94.4	93.9	93.7	94.2
Proposed DFNet Classification model	95.7	94.9	95.4	95.1

A comparison of various models is shown in the **Table 2** and **Fig 2** according to performance metrics. With precision, recall, and F1 scores of 90.7%, 91.0%, and 90.8%, respectively, the "AE" model achieves an accuracy of 90.1%. After that, the "VAE" model exhibits gain in all areas, with accuracy reaching 92.5% and precision, recall, and F1 scores reaching 91.9%, 92.2%, and 92.1%, respectively. With increasingly complex models, the performance keeps getting better: "VGG-16" shows a 93.3% accuracy rate, along with 92.8%, 92.5%, and 92.2% scores for precision, recall, and F1. With accuracy of 94.4%, precision, recall, and F1 scores of 93.9%, 93.7%, and 94.2%, respectively, "ResNet" improves the outcomes even more. In the end, the "Proposed DFNet Classification model" performs better than the rest, attaining the maximum accuracy of 95.7% along with outstanding recall, precision, and F1 scores of 94.9%, 95.4%, and 95.1%, in that order. This illustrates how models are becoming more and more effective; among the models in the table, the "Proposed DFNet Classification model" performs better on all metrics.

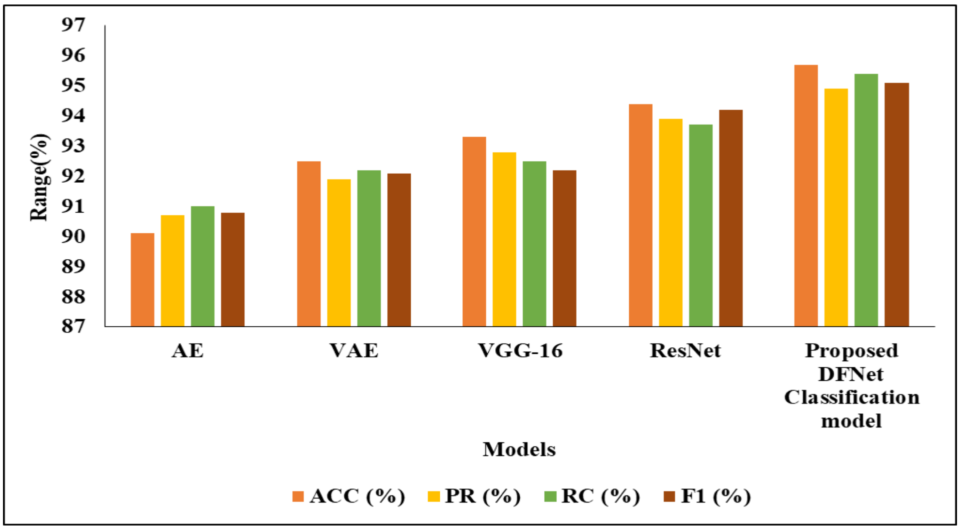

Fig 2. Classification Analysis

Table 3. Classification Analysis with Existing Models and Proposed Model

Models	ACC (%)	
	Without SHO	With SHO
AE	90.1	97.2
VAE	92.5	97.8
VGG-16	93.3	98.3
ResNet	94.4	98.7
Proposed SHO based DFNet Classification model	**95.7**	**99.4**

Table 3 and **Fig.3.** presents the relative accuracy percentages (%) of different models under two conditions: SHO and not SHO. A proposed SHO-based DFNet Classification model, Autoencoder (AE), Variational Autoencoder (VAE), VGG-16, ResNet, and other models are evaluated. The models exhibit differing levels of accuracy in the absence of SHO; AE achieves 90.1%, VAE 92.5%, VGG-16 93.3%, ResNet 94.4%, and the suggested SHO-based DFNet Classification model leads with 95.7%. All of the models' accuracy has noticeably improved, though, since SHO was integrated. ResNet achieves 98.7%, AE rises to 97.2%, VAE to 97.8%, VGG-16 to 98.3%, and the suggested SHO-based DFNet Classification model performs admirably at 99.4%. This data shows how adding SHO to these models significantly improved the classification accuracy. It also shows how well the suggested SHO-based DFNet Classification model performed, outperforming the other models evaluated with the highest accuracy in both scenarios.

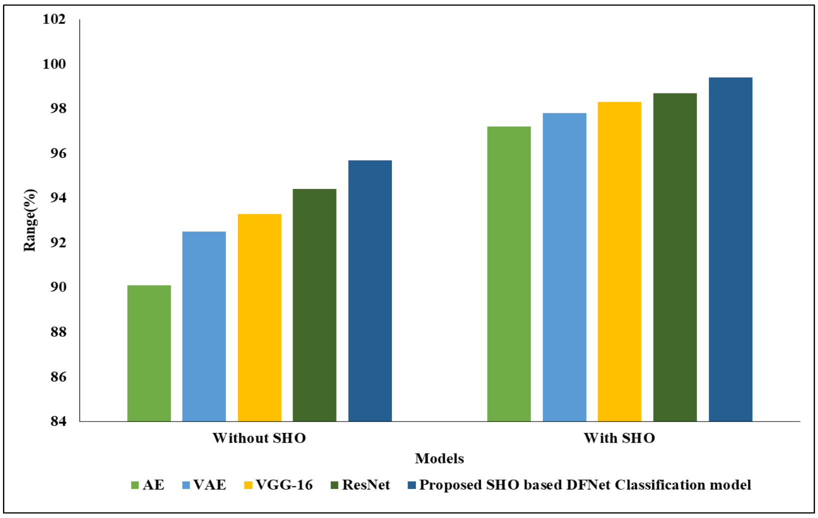

Fig 3. Accuracy Analysis

V. CONCLUSION

The proposed model emphasizes how important environmental monitoring is to identifying and mitigating natural disasters, pollution incidents, and ecological imbalances. It highlights how AI and UAV technology can be combined to provide real-time analysis and decision-making. In particular, it discusses the rising number of forest fires in India and suggests utilizing CNN to create DFNet. The efficacy of the suggested DFNet model is demonstrated by the use of SHO for hyperparameter tuning, which provides notable improvements in accuracy of 99.4% over baseline models. In the end, this study offers a viable strategy to improve environmental monitoring, especially in tackling the difficulties caused by forest fires, thereby making a substantial contribution to environmental management and protection. We plan to integrate a deep attention module with blockchain and federated learning in the future so that we can improve the accuracy of forest fire picture classification.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interests

The author(s) declare(s) that they have no conflicts of interest.

Funding

No funding agency is associated with this research.

Competing Interests

There are no competing interests.

References

[1]. J. Sherry, T. Neale, T. K. McGee, and M. Sharpe, "Rethinking the maps: A case study of knowledge incorporation in Canadian wildfire risk management and planning," Journal of Environmental Management, vol. 234, pp. 494–502, Mar. 2019, doi: 10.1016/j.jenvman.2018.12.116.
[2]. J. F. C. dos Santos et al., "Wildfires as a major challenge for natural regeneration in Atlantic Forest," Science of The Total Environment, vol. 650, pp. 809–821, Feb. 2019, doi: 10.1016/j.scitotenv.2018.09.016.
[3]. A. Badia, M. Pallares-Barbera, N. Valldeperas, and M. Gisbert, "Wildfires in the wildland-urban interface in Catalonia: Vulnerability analysis based on land use and land cover change," Science of The Total Environment, vol. 673, pp. 184–196, Jul. 2019, doi: 10.1016/j.scitotenv.2019.04.012.
[4]. Y. Peng and Y. Wang, "Real-time Forest smoke detection using hand-designed features and deep learning," Computers and Electronics in Agriculture, vol. 167, p. 105029, Dec. 2019, doi: 10.1016/j.compag.2019.105029.
[5]. K. Avazov, M. Mukhiddinov, F. Makhmudov, and Y. I. Cho, "Fire Detection Method in Smart City Environments Using a Deep-Learning-Based Approach," Electronics, vol. 11, no. 1, p. 73, Dec. 2021, doi: 10.3390/electronics11010073.
[6]. Y. Wang, L. Dang, and J. Ren, "Forest fire image recognition based on convolutional neural network," Journal of Algorithms & Computational Technology, vol. 13, p. 174830261988768, Jan. 2019, doi: 10.1177/1748302619887689.
[7]. Y. Hu et al., "Fast Forest fire smoke detection using MVMNet," Knowledge-Based Systems, vol. 241, p. 108219, Apr. 2022, doi: 10.1016/j.knosys.2022.108219.
[8]. Wahyono, A. Harjoko, A. Dharmawan, F. D. Adhinata, G. Kosala, and K.-H. Jo, "Real-Time Forest Fire Detection Framework Based on Artificial Intelligence Using Color Probability Model and Motion Feature Analysis," Fire, vol. 5, no. 1, p. 23, Feb. 2022, doi: 10.3390/fire5010023.
[9]. K. Avazov, A. E. Hyun, A. A. Sami S, A. Khaitov, A. B. Abdusalomov, and Y. I. Cho, "Forest Fire Detection and Notification Method Based on AI and IoT Approaches," Future Internet, vol. 15, no. 2, p. 61, Jan. 2023, doi: 10.3390/fi15020061.
[10]. M. P. Ferreira et al., "Individual tree detection and species classification of Amazonian palms using UAV images and deep learning," Forest Ecology and Management, vol. 475, p. 118397, Nov. 2020, doi: 10.1016/j.foreco.2020.118397.

[11]. J. Xie, A. Li, J. Zhang, and Z. Cheng, "An Integrated Wildlife Recognition Model Based on Multi-Branch Aggregation and Squeeze-And-Excitation Network," Applied Sciences, vol. 9, no. 14, p. 2794, Jul. 2019, doi: 10.3390/app9142794.
[12]. J. Liu, Q. Zhou, Y. Qiang, B. Kang, X. Wu, and B. Zheng, "FDDWNet: A Lightweight Convolutional Neural Network for Real-Time Semantic Segmentation," ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), May 2020, doi: 10.1109/icassp40776.2020.9053838.
[13]. W. Hu and Y. Guan, "Landmark-free head pose estimation using fusion inception deep neural network," Journal of Electronic Imaging, vol. 29, no. 04, Aug. 2020, doi: 10.1117/1.jei.29.4.043030.
[14]. C. Bahhar et al., "Wildfire and Smoke Detection Using Staged YOLO Model and Ensemble CNN," Electronics, vol. 12, no. 1, p. 228, Jan. 2023, doi: 10.3390/electronics12010228.
[15]. R. A. Aral, C. Zalluhoglu, and E. Akcapinar Sezer, "Lightweight and attention-based CNN architecture for wildfire detection using UAV vision data," International Journal of Remote Sensing, vol. 44, no. 18, pp. 5768–5787, Sep. 2023, doi: 10.1080/01431161.2023.2255349.
[16]. M. YANDOUZI et al., "Investigation of Combining Deep Learning Object Recognition with Drones for Forest Fire Detection and Monitoring," International Journal of Advanced Computer Science and Applications, vol. 14, no. 3, 2023, doi: 10.14569/ijacsa.2023.0140342.
[17]. A. K. Z Rasel Rahman, S. M. Nabil Sakif, N. Sikder, M. Masud, H. Aljuaid, and A. Kumar Bairagi, "Unmanned Aerial Vehicle Assisted Forest Fire Detection Using Deep Convolutional Neural Network," Intelligent Automation & Soft Computing, vol. 35, no. 3, pp. 3259–3277, 2023, doi: 10.32604/iasc.2023.030142.
[18]. L. Zhang, M. Wang, Y. Ding, and X. Bu, "MS-FRCNN: A Multi-Scale Faster RCNN Model for Small Target Forest Fire Detection," Forests, vol. 14, no. 3, p. 616, Mar. 2023, doi: 10.3390/f14030616.
[19]. J. Ye, S. Ioannou, P. Nikolaou, and M. Raspopoulos, "CNN based Real-time Forest Fire Detection System for Low-power Embedded Devices," 2023 31st Mediterranean Conference on Control and Automation (MED), Jun. 2023, doi: 10.1109/med59994.2023.10185692.
[20]. Nambur, A.; Sankalp Saxena, M.S.S.; Natarajan, M.G. Fire and Non-Fire Image Dataset. 2022. Availableonline: https://kaggle.com/datasets/f7517a19d918cae42ac1222937d07096179e663d7b8ed0a4c66deae33073b21d(accessed on 9 January 2023).
[21]. H. Shahverdi, M. Nabati, P. Fard Moshiri, R. Asvadi, and S. A. Ghorashi, "Enhancing CSI-Based Human Activity Recognition by Edge Detection Techniques," Information, vol. 14, no. 7, p. 404, Jul. 2023, doi: 10.3390/info14070404.
[22]. S. Wang, Y. Xing, L. Zhang, H. Gao, and H. Zhang, "Deep Convolutional Neural Network for Ulcer Recognition in Wireless Capsule Endoscopy: Experimental Feasibility and Optimization," Computational and Mathematical Methods in Medicine, vol. 2019, pp. 1–14, Sep. 2019, doi: 10.1155/2019/7546215.
[23]. Nature. Olympus. The Endocapsule 10 System. Olympus Homepage. 2021. Available online: https://www.olympus-europa. com/medical/en/Products--and--Solutions/Products/Product/ENDOCAPSULE-10-System.html (accessed on 2 March 2023).
[24]. H. Malik, M. S. Farooq, A. Khelifi, A. Abid, J. Nasir Qureshi, and M. Hussain, "A Comparison of Transfer Learning Performance Versus Health Experts in Disease Diagnosis from Medical Imaging," IEEE Access, vol. 8, pp. 139367–139386, 2020, doi: 10.1109/access.2020.3004766.
[25]. G. Dhiman and R. Sharma, "SHANN: an IoT and machine-learning-assisted edge cross-layered routing protocol using spotted hyena optimizer," Complex & Intelligent Systems, vol. 8, no. 5, pp. 3779–3787, Nov. 2021, doi: 10.1007/s40747-021-00578-5.
[26]. A. Naderipour et al., "Deterministic and probabilistic multi-objective placement and sizing of wind renewable energy sources using improved spotted hyena optimizer," Journal of Cleaner Production, vol. 286, p. 124941, Mar. 2021, doi: 10.1016/j.jclepro.2020.124941.