



ARTICLE

Fusion-Based Deep Learning Model for Automated Forest Fire Detection

Mesfer Al Duhayyim¹, Majdy M. Eltahir², Ola Abdelgney Omer Ali³, Amani Abdulrahman Albraikan⁴, Fahd N. Al-Wesabi², Anwer Mustafa Hilal^{5,*}, Manar Ahmed Hamza⁵ and Mohammed Rizwanullah⁵

¹Department of Natural and Applied Sciences, College of Community-Aflaj, Prince Sattam bin Abdulaziz University, Al-Kharj, 16278, Saudi Arabia

²Department of Computer Science, College of Science & Art at Mahayil, King Khalid University, Muhayel Aseer, 62529, Saudi Arabia

³Department of Information Technology, College of Computer, Qassim University, Al-Bukairiyah, 52571, Saudi Arabia

⁴Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O.Box 84428, Riyadh, 11671, Saudi Arabia

⁵Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University, Al-Kharj, 16278, Saudi Arabia

*Corresponding Author: Anwer Mustafa Hilal. Email: a.hilal@psau.edu.sa

Received: 08 October 2021 Accepted: 09 November 2021 Published: 31 October 2023

ABSTRACT

Earth resource and environmental monitoring are essential areas that can be used to investigate the environmental conditions and natural resources supporting sustainable policy development, regulatory measures, and their implementation elevating the environment. Large-scale forest fire is considered a major harmful hazard that affects climate change and life over the globe. Therefore, the early identification of forest fires using automated tools is essential to avoid the spread of fire to a large extent. Therefore, this paper focuses on the design of automated forest fire detection using a fusion-based deep learning (AFFD-FDL) model for environmental monitoring. The AFFD-FDL technique involves the design of an entropy-based fusion model for feature extraction. The combination of the handcrafted features using histogram of gradients (HOG) with deep features using SqueezeNet and Inception v3 models. Besides, an optimal extreme learning machine (ELM) based classifier is used to identify the existence of fire or not. In order to properly tune the parameters of the ELM model, the oppositional glowworm swarm optimization (OGSO) algorithm is employed and thereby improves the forest fire detection performance. A wide range of simulation analyses takes place on a benchmark dataset and the results are inspected under several aspects. The experimental results highlighted the betterment of the AFFD-FDL technique over the recent state of art techniques.

KEYWORDS

Environment monitoring; remote sensing; forest fire detection; deep learning; machine learning; fusion model

1 Introduction

In recent years, human-caused factors and climate changes had a major influence on the environment. Few events include droughts, heat waves, floods, dust storms, wildfires, and hurricanes [1].



Wildfires have serious effects on global and local ecosystems and cause severe damage to infrastructures, losses, and injuries in human life; hence, accurate monitoring and fire detection of the disturbance size, type, and effect on a wide range has become more prominent [2]. Eventually, stronger actions were taken to mitigate/avoid this consequence through earlier fire risk mapping/fire detection. Conventionally, a forest fire was primarily recognized by human surveillance from a fire lookout tower and only involved primitive tools, like Osborne fire Finder; but, this method is ineffective since it is prone to fatigue and human errors. At the same time, traditional sensor nodes for detecting smoke, heat, gas, and flame are usually time-consuming for the particle to attain the point of sensor nodes and make them active. Additionally, the range of this sensor is very smaller, therefore, a huge number of sensors needs to be mounted to cover wide ranges [3,4].

Current developments in remote sensing, computer vision, and machine learning methods offer new tools to detect and monitor forest fires, whereas the advances in novel microelectronics and materials have enabled sensor nodes to be highly effective in detecting active forest fires [5]. Different from other fire detection survey reports which have concentrated on several sensing techniques, on smoke/video flame methods in InfraRed (IR)/visible range, on different airborne systems, and environments. Based on the acquisition levels, 3 major classes of broadly employed system which is capable of detecting or monitoring active smoke/fire incident in real or near realtime are detected and deliberated, like satellite, terrestrial, and aerial [6]. This system is generally armed with IR, and visible/multispectral sensors where the information is treated using ML method. This method is based on the extraction of hand-crafted features or on a robust deep learning (DL) network to detect earlier forest fires and also for modeling smoke/fire behavior. Lastly, they introduce the weaknesses and strengths of the above-mentioned sensor methods, and also future trends in the field of detecting earlier fires [7].

The usage of an automated forest fire detection scheme, targeted to its primary stage, will be the desired solution for mitigating such problems. This method can enhance and complement the present works of human observers on detecting fires [8]. The usage of automated observation technology is a very promising choice, backed up with a considerable amount of work existing in [9]. Amongst the accessible fire detection techniques, they emphasize the usage of video cameras as input sources for automated detecting systems. The optical system might stimulate the human eye to look for obvious signs of fires. However the forest does not provide a uniform landscape scenario which makes fire detecting an extremely difficult task. Dynamic phenomena like fog, cloud shadow reflections, and the existence of human activities/apparatus translate to the most challenging problem and increase the chance of false alarms [10]. Considering this problem, the current study has been performed for implementing low-cost solutions capable of tackling this inherent challenge of detecting forest fires through a video camera feeding ML algorithm.

This paper proposes an automated forest fire detection using a fusion-based deep learning (AFFD-FDL) model for environmental monitoring. The AFFD-FDL technique involves the design of an entropy-based fusion model for feature extraction including three feature extractors namely histogram of gradients (HOG), SqueezeNet, and Inception v3 models. In addition, an optimal extreme learning machine (ELM) based classifier is used to identify the existence of fire or not. For effectively tuning the parameters of the ELM model, the oppositional glowworm swarm optimization (OGSO) algorithm is applied and thus improves the forest fire detection performance. An extensive set of simulations were performed on benchmark images and the results are examined under different dimensions.

2 Existing Works on Forest Fire Detection

Numerous forest fire detection using ML and DL models are available in the literature. To automatically identify fire, a forest fire image detection method based on CNN was presented in [11]. Traditional image processing and CNN techniques are integrated, besides an adoptive pooling method is presented. The fire areas could be divided also the features might be learned through this model in advance. Simultaneously, the loss of sight in the conventional feature extraction processes are evaded, and also the learning of in-valid feature in the CNNs are evaded. Jiao et al. [12] proposed a forest fire-detecting approach through YOLOv3 to UAV-based aerial images. First, proposed UAV platforms with the aim of forest fire recognition. Next, based on the presented computational power of onboard hardware, a small scaling of CNN is executed using YOLOv3. Experimental result shows that the detection rate is around 83%, and the frame detection rates could attain over 3.2 fps.

In [13], a comparison of different ML methods like regression, SVM, DT, NN, and so on, was made to predict forest fire. The presented method shows that how regression works better for detecting forest fires with higher precision by separating the datasets. Faster detection of forest fire is made in this study takes less time than other ML methods. Kinaneva et al. [14] proposed an infrastructure that employs UAVs that continually patrol across possibly threatened fire areas. Also, The UAV utilizes the benefits of AI and is armed with onboard processing abilities. This permits to utilization of CV approaches for the detection and recognition of fire/smoke, according to image or video input from the drone camera. In [15], new ensemble learning methods are presented for detecting forest fires in various situations. First, 2 individual learners EfficientDet and Yolov5 were incorporated to achieve the fire-detecting method. Next, for other individual learners, EfficientNet is accountable for learning global data to avoid false positives. Lastly, the detection result is performed according to the decision of 3 learners.

Chen et al. [16] proposed a UAV image-based forest fire detecting method. First, the SVM classifiers and LBP feature extraction are employed to detect smoke for making primary discrimination of forest fire. To precisely detect the earlier phase of the fire, based on CNN, it has the features of decreasing the amount of variables and improving the trained performances via pooling, local receptive domain, and weight sharing. Wu et al. [17] focused on 3 challenges which surround real-time, false detection, forest fire detection, and early fire detection. Initially, employ traditional objective detecting methods to identify forest fire: YOLO (tiny-yolo-voc 1, tiny-yolo-voc, YOLOv3, and Yolo-voc. 2.0), SSD, and Fast RCNN, amongst other SSD have improved real-time properties, early fire detection, and high detection accuracy capability. Sinha et al. [18] represented a new method that detects the HA region (near the epicenter of fire) in the forest and transmits each sensed information to BS via wireless transmission as possible. The fire office takes required actions to prevent the spread of fire. For these purposes, sensors are placed in forest areas for sensing distinct information that is needed to detect forest fires and divide them into various clusters. A semi-supervised rule-based classification model is presented for detecting whether its region is HA, MA, or LA clusters in the forest.

3 Materials and Methods

3.1 System Pipeline

In this study, a new AFFD-FDL technique is derived for automated forest fire detection. The proposed model begins with the feature extraction process using HOG, SqueezeNet, and Inception v3 models. Then, the features are fused together by the use of an entropy-based fusion model. Following by, the optimal ELM model is utilized to perform the classification process where the

parameters involved in the ELM model are optimally adjusted by the use of the OGSO algorithm. Fig. 1 demonstrates the overall working process of the proposed AFFD-FDL model. The detailed working of these processes is elaborated in the subsequent sections.

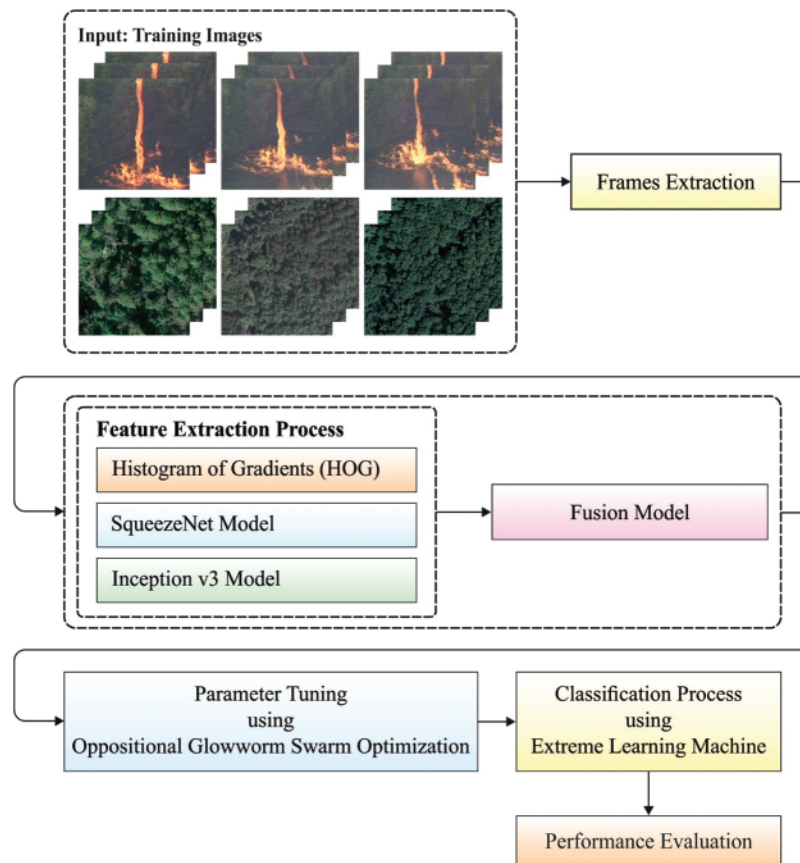


Figure 1: Overall process of AFFD-FDL model

3.2 Design of Fusion-Based Feature Extraction Model

In this study, an entropy-based fusion model is employed to derive a useful subset of features. It encompasses three models namely HOG, Inception v3, and SqueezeNet. A detailed overview of these models is offered in the following sections.

3.2.1 Overview of the HOG Model

The main feature of the HOG feature is able to hold the local appearance of objects and account for the in-variance of object conversion and illumination status as data and edge about gradient are estimated by using a multiple coordinate HOG feature vector. At first, an N gradient operator was used to determine the gradient measures. The gradient points of mammogram images are presented as G also the image frame is displayed as I. A common equation used in computing gradient points is pointed out below:

$$G_x = N * I(x, y) \text{ and } G_y = N^T * I(x, y) \quad (1)$$

The image-detecting window undergoes classification as different spatial areas are known as cells. Thus, the magnitude gradient of the pixel is implemented by an edge location [19]. As a result, the magnitude of gradients (x, y) is implied in Eq. (2).

$$G_x(x, y) = \sqrt{G_x(x, y)^2} + \sqrt{G_y(x, y)^2} \quad (2)$$

The edge locations of points (x, y) are provided below:

$$\theta(x, y) = \frac{G_y(x, y)}{G_x(x, y)} \quad (3)$$

whereas G_x represents the horizontal direction of the gradient and G_y denotes the vertical direction of the gradient. In case of enhanced illumination as well as noise, a normalization task is processed once completing the histogram measures. The determination of normalization is used in contrast and local histograms can be validated. In various coordinates HOG, 4 different methods of standardization can be employed as L2-Hys, L1-norm, L1-Sqrt, and L2-norm. When compared to this standardization, L2-norm provides an optimal function in cancer predictions. The segment of standardization in HOG is expressed as follows:

$$L2 - norm: f = \frac{h}{\sqrt{\|h\|_2^2 + e^2}} \quad (4)$$

whereas e denotes smaller positive values employed in standardization, f can be represented feature vector, h shows the non-standardized vector, also $\|h\|_2^2$ is named as the 2-norm of HOG standardization.

3.2.2 Structure of SqueezeNet Model

The structure block of SqueezeNet is named as fire component shown in Fig. 2. It has 2 layers: the squeeze and expands layers. The SqueezeNet stacks a group of these fire components and some pooling layers. The squeeze layer reduces the size of the feature map, but the expanding layer improves it again. Thus the outcome, the similar feature map size has been continued. Another form is improving depth, but diminishing the feature map size to obtain a higher-level abstract. These are accomplished by enhancing the amount of filters and utilizing the stride of 2 in the convolution layers. During this effort, it can be present the solution to an image multi-labeling issue dependent upon fine-tuning a higher version of pre-training SqueezeNet CNN. Especially, it can be planned 2 enhanced SqueezeNet techniques to be utilized with fine-tuning method. It also forms a base method or Model 1 which utilizes the pre-training SqueezeNet as a feature extractor (for instance, their weights are fixed) and improves a fully connected (FC) layer to adapt to the provided dataset.

Then another 2 increased techniques, mentioned as Model 2 and Model 3 correspondingly. In Model 2, it can be fine-tuning Model 1, by creating every weight of SqueezeNet layers trainable. In Model 3, it can get rid of the extra FC layer of Model 1 and reset the previous convolution layer of the SqueezeNet method to arbitrary weights. Eliminating the FC layer is the benefit of decreasing the weights of the deep network which therefore reduces computational cost. Rearranging the weight of the final convolution layer enhances the generalized ability and affords a superior chance for the network to adapt to a novel dataset. Also, during every model above, it can be changed the ReLU activation functions of the hidden layer with the novel weight with further advanced LeakyReLU activation functions [20]. It can insert a Batch Normalization (BN) layer then to fight over-fit and enhance the generalization ability of networks. It could not be present in the original SqueezeNet

method. The BN layer was established back in 2014 in the 2nd form of GoogLeNet. The BN has been same as the model of standardization (create the data have 0 means and sometimes also unit SD) of network input that is a general thing to do earlier utilizing a CNN on some data. The BN does a similar thing inside CNN. Especially, it can enhance a normalization layer later the convolution layer that creates the data in the present trained batch (data is separated in batches to ensure it fits in computer RAM) follows a usual distribution. This way the technique efforts to learn the pattern from the data and is not misled by huge values in it. It tries to use opposing over-fitted and improve the generalized capability of networks. Other benefits of BN approaches are that they permit the technique to converge quicker in trained and therefore permit us to utilize a superior learning rate.

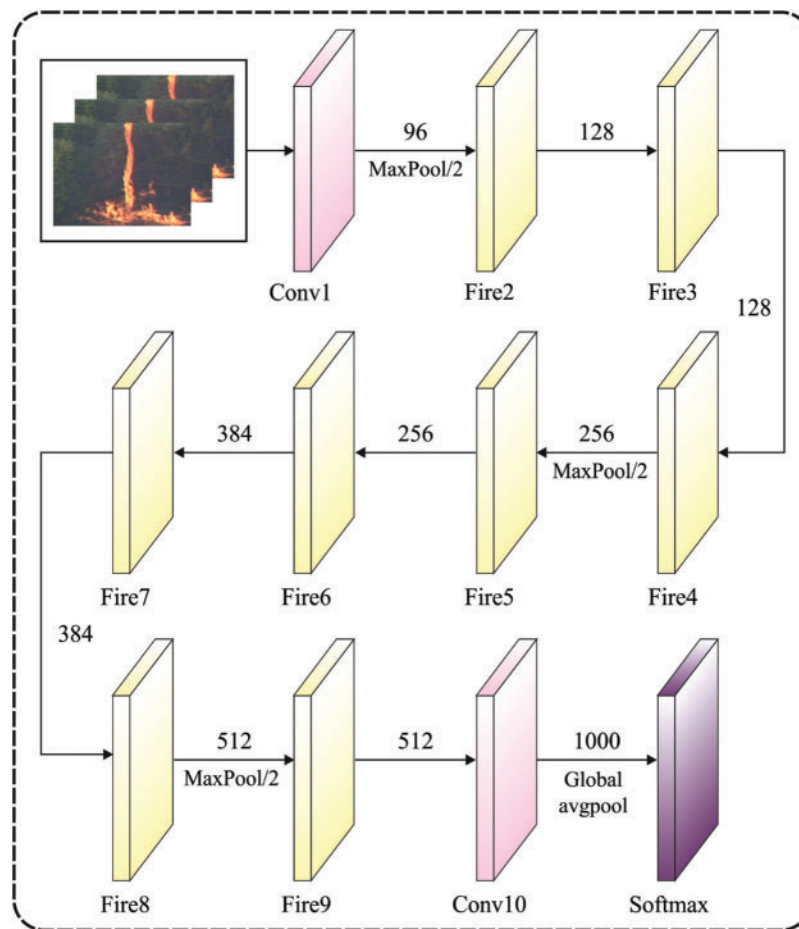


Figure 2: Architecture of SqueezeNet

The presented solution fights over-fitted by transporting the knowledge from CNN pre-training on a huge auxiliary dataset (such as ImageNet), rather than training it with arbitrary primary weight. Also, it combats over-fitted by decreasing the amount of free weight and therefore decreasing the flexibility of the network that it does not learn the small particulars or noise from the dataset. Lastly, the novel BN approach presented inside the network enhances the generalized capability of the network and fights over-fitted as it forces every feature value inside the network that exists in a similar small range (0–1) rather than taking the flexibility for utilizing feature value as huge it requires for fitting the trained data.

3.2.3 Structure of Inception v3 Model

As the primary goal of GoogLeNet networks is an Inception network framework, the GoogLeNet methods are known as Inception networks [21]. It has of maximal GoogLeNet version that is categorized into distinct forms of Inception ResNet, Inceptions v1, v2, v3, and v4. Therefore, Inception commonly consists of 3 various sizes of maximum as well as convolution pooling. The results of the network in prior layers are determined by the network gathered when finishing convolutional tasks, and a non-linear fusion has been performed. Likewise, the expression function of these networks and applicable for many scales that are improved, and eliminate the overfitting problems. Inception v3 refers to a network framework proposed by Keras that is pretrained in Image Net. The primary image input sizes are $299 * 299$ using 3 networks. In contrast to Inception v1 & v2, Inception v3 network structures employ a convolutional kernel splitting method for dividing huge volumes of integral to minimum convolution. E.g., a $3 * 3$ convolutional was separated into $3 * 1$ & $1 * 3$ convolution. With these splitting models, the amounts of attributes might be limited; therefore, the network's training speed can be triggered while extracting spatial features in an efficient way. Concurrently, Inception v3 optimizes the Inception networks framework through 3 different sized grids such as $35 * 35$, $17 * 17$, & $8 * 8$.

3.2.4 Fusion Process

Feature fusion is an important operation that integrates maximal feature vectors. The proposed model is based on feature fusion using entropy. Additionally, the attained feature is integrated into one vector. It consists of 3 vectors which have been calculated in the following:

$$f_{HOG \times p} = \{HOG_{1 \times 1}, HOG_{1 \times 2}, HOG_{1 \times 3}, \dots, HOG_{1 \times n}\} \quad (5)$$

$$f_{SqueezeNet \times p} = \{SqueezeNet_{1 \times 1}, SqueezeNet_{1 \times 2}, SqueezeNet_{1 \times 3}, \dots, SqueezeNet_{1 \times n}\} \quad (6)$$

$$f_{Inception_{v3} \times p} = \{Inception_{v3}_{1 \times 1}, Inception_{v3}_{1 \times 2}, Inception_{v3}_{1 \times 3}, \dots, Inception_{v3}_{1 \times n}\} \quad (7)$$

Later, extracted features are integrated as one vector.

$$Fused \ (feature \ vector)_{1 \times q} = \sum_{i=1}^3 \{fHOG_{1 \times p}, fSqueezeNet_{1 \times p}, fInceptionV3_{1 \times p}\} \quad (8)$$

whereas f denotes a fused vector. The entropy is executed on feature vectors only for elected features which are based on values.

$$B_{He} = -NHe_b \sum_{i=1}^n p(f_i) \quad (9)$$

$$F_{select} = B_{He}((f_i, 1186)) \quad (10)$$

In Eqs. (9) and (10), p represents the likelihood feature as well as He denotes entropy. Lastly, the elected feature is given to the classification model for discriminating the forest fire.

3.3 Design of Optimal ELM-Based Classification Model

During forest fire classification, the ELM model receives the fused feature vectors as input and performs the classification process. ELM is developed by Professor Huang, and utilizes a single hidden layer FFNN structure. It is provided by N trained instances and comprises L typical SLFN technique

is written as:

$$\sum_{i=1}^L \beta_i G(a_i \times x_j + b_i) = O_j, j = 1, \dots, N \quad (11)$$

Here $G(x)$ refers to the activation function of the Hidden layer node, β_i represents the resultant weights amongst the initial i node and linked to the resultant neuron, (a_i, b_i) defines the input and the bias of the input neuron, O_j signifies the actual outcome value of resultant neurons. At this point, (a_i, b_i) and β_i generate $\sum_{j=1}^L \|O_j - t_j\| = 0$ set up and create the SLFN technique is zero error estimate instance set $\{(x_i, t_i), i = 1, \dots, N\}$, i.e.,

$$\sum_{i=1}^L \beta_i G(a_i \times x_j + b_i) = t_j, j = 1, \dots, N \quad (12)$$

Eq. (2) has been written as $H\beta = T$, now H implies the hidden layer resultant matrix.

The ELM does not require adjusting the primary weight values under the procedure of training, only a requirement for working out the weights of output least-square norm [22]. Therefore, it requires making a compromise between minimizing the resultant weight and minimizing the error. The equation has been generated as:

$$\text{Minimize : } \|\beta\| \quad (13)$$

$$\text{Minimize : } \sum_{i=1}^N \|\beta \times h(x_i) - t_i\| \quad (14)$$

Same as the least squares SVM, the optimized issue is written as follows:

$$\text{Minimize : } L = \frac{1}{2} \|\beta\|^2 + C \frac{1}{2} \sum_{i=1}^N \|\zeta_i\|^2 \quad (15)$$

$$\text{s.t. : } h(x_i) \beta = t_i^T - \zeta_i^T, i = 1, \dots, N$$

At this point, $\zeta_i = [\zeta_{i,1}, \dots, \zeta_{i,m}]^T$ defines the error vector on trained instances amongst the resultant values and true value of primary m resultant node. And C defines the regularization parameter that is the penalty factor utilized for weighting empirical as well as structural risks and is controlling on appropriate by setting the parameter, for achieving optimum classifier outcomes. With the Moore-Penrose generalization inverse matrix H^T of hidden layer resultant matrix H , it is obtained the solutions $\hat{\beta} = H^T T$, where $T = [t_1, \dots, t_N]$.

The ELM is faster training speed and optimum generalized efficiency, however in referring to data with imbalanced class distribution it could not attain optimum outcomes so there is the need for introducing the weighted idea in ELM, and it could generate the WELM technique. For optimal adjustment of the parameters involved in the ELM model, the OGSO algorithm is used with the aim of improving the detection accuracy. GSO is considered an intelligent swarm optimization method employed in speeding up the luminescent feature of Firefly. In the GSO method, the glowworm swarm is shared in space solutions and the FF of each glowworm position [23]. The strong glowworm has maximal brightness and optimum location is made up of maximal FF rate. Glowworm is made up of a dynamic line of sight, i.e., called decision domains, that has the range of density for neighboring nodes. On the other hand, the decision radius was constrained when glowworm traveled toward the same kind of robust fluorescence in decision domains. Achieving the high value of iteration, all glowworms will be located in an optimal position. It is limited to 5 stages as follows:

- Fluorescence in concentration
- Neighbouring set
- Decision domain radius

- Moving possibility
- Glowworm location

The fluorescence in the concentration upgrading methods is expressed in the following:

$$l_i(t) = (1 - \alpha) l_i(r - 1) + \beta f(x_i(t)) \tag{16}$$

whereas $l_i(f)$ represents the fluorescence in the concentration of i th glowworms in time f , α means the fluorescence in volatilization coefficients, β indicates fluoresce in development factors, $f(x)$ signifies FF and $x_i(r)$ represent the position of glowworms i in f time is given below:

$$N_i(t) = \{j : \|x_j(r) - x_i(t)\| < r_d^i; l_i(t) < l_j(t)\} \tag{17}$$

whereas $N_i(f)$ denotes the neighbor set of i th glowworms in time r and $r_d^i(r)$ denotes the radius of the decision domain of the i th glowworms in time f as follows:

$$r_d^i(t + 1) = \min \{r_s, \max \{r_d^i(t) + \gamma (n_i - |N_i(t)|)\}\} \tag{18}$$

whereas r_s signifies the achieved radius of glowworms, γ indicates the values of the decision domain, and n_i displays the neighboring threshold. The moving probability of an upgraded method is given below:

$$p_{ij}(r) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i} l_k(t) - l_i(r)} \tag{19}$$

whereas $p_{ij}(t)$ denotes the possibility in which glowworms i travel to the glowworms j in r time as follows:

$$x_i(t + 1) = x_i(t) + s \left(\frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|} \right) \tag{20}$$

The OBL is a primary goal in the efficient optimization method for improving the convergence speed of different heuristic-enhancing models. The efficient implementation of OBL contributes approximation of the opposite and current populations in the same generation for identifying optimum candidate solutions of a given problem. The OBL models have been efficiently used in different Meta heuristics employed for improving convergent speed. The models of the opposite amount should be described in OBL.

Consider $N \in N[x, y]$ to denote real numbers. The opposite numbers N^o are given by

$$N^o = x + y - N \tag{21}$$

In the d -dimension searching region, the depiction may be extended as follows:

$$N_i^o = x_i + y_i - N_i \tag{22}$$

whereas (N_1, N_2, \dots, N_d) indicates d -dimension searching region and $N_i[x_i, y_i]$, $i = 1, 2, \dots, d$. From the OBO, the approach of OBL is employed in this initiation procedure of the GSO method and for all iterations in the application of jump rate.

3.4 Flame Area Estimation Process

The flame area estimate component is generated on the earlier established CAFE scheme [24]. For achieving further effectual detection and, concurrently, decreasing the amount of false positive

conditions, before utilizing only the FFDI color index, the CAFE manner establishes the pre-processing modules, during the Lab color space that offer optimum outcomes. These modules, generally, purpose for estimating the area of noticeable flame in an image. As the classifier component does not require that the fire be identified with smoke, with flames, or both, during this component, such omission is resolved with the following rule: when the flames are detected from the image, afterward the image has comprised of smoke and flames; when no flame is detected later it can be supposed that the component is facing an image in which the fire signal was just smoke.

4 Experimental Validation

This section examines the forest fire detection performance of the proposed model on a set of benchmark videos [25]. The proposed model converts the video into a set of frames and then the detection process is carried out. A few sample images as demonstrated in Fig. 3.

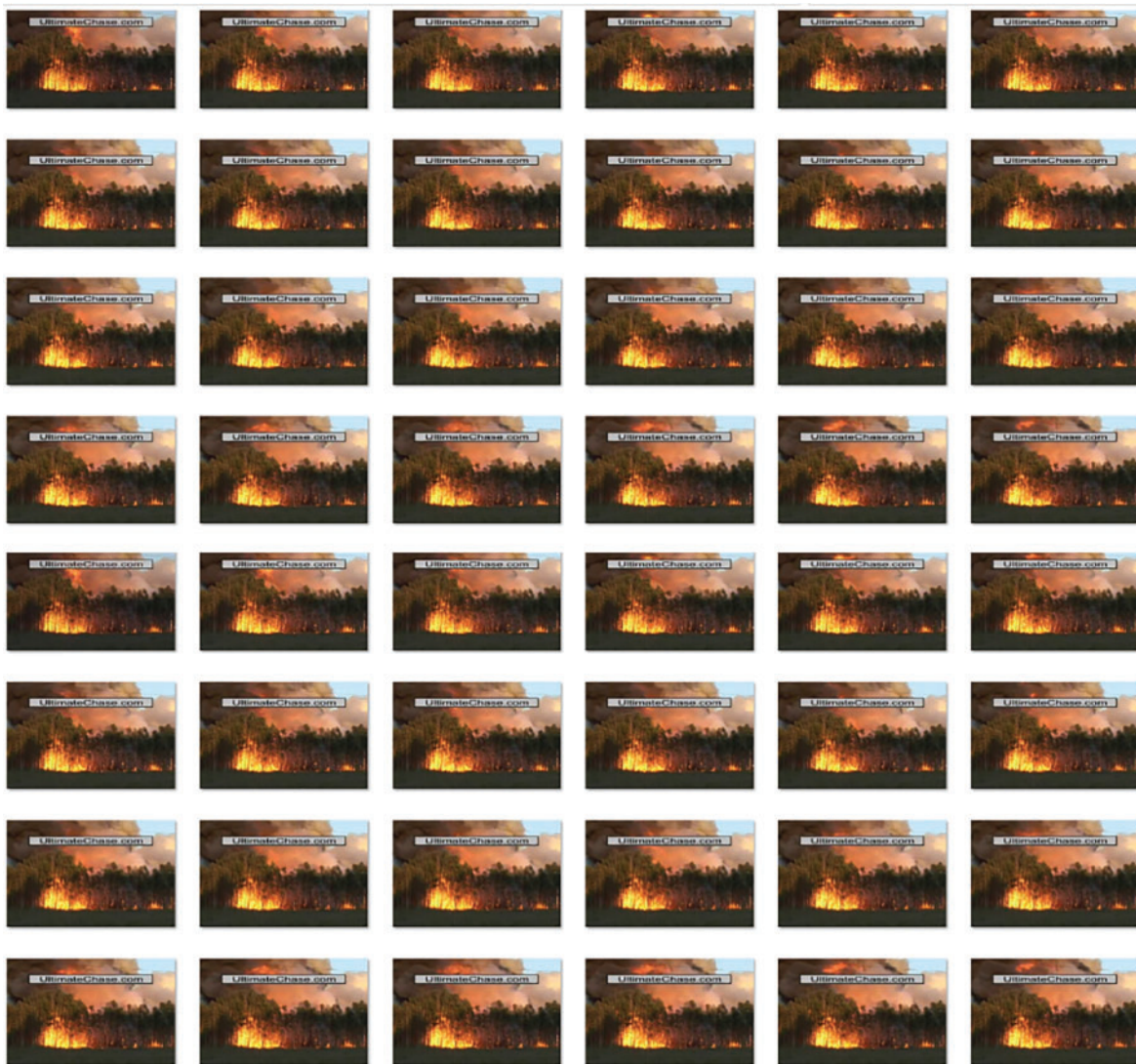


Figure 3: Sample images

A sample visualization results analysis of the AFFD-FDL technique on the detection of forest fire on the applied test images is shown in Fig. 4. The figure demonstrated that the AFFD-FDL technique has proficiently detected the fire and non-fire images correctly.

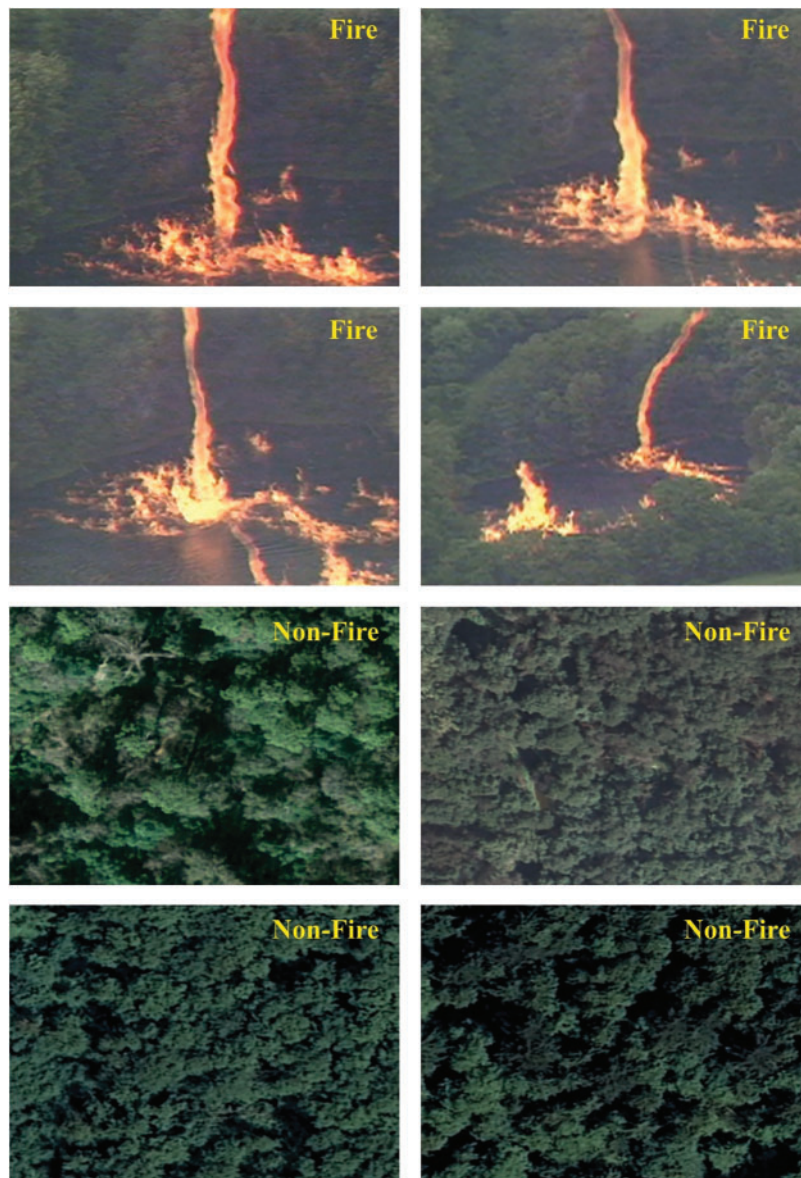


Figure 4: Visualization results analysis of AFFD-FDL technique

A set of five confusion matrices generated by the AFFD-FDL technique on the applied five distinct sets is given in Fig. 5. The figure portrays that the AFFD-FDL technique has the ability to classify the images into Fire and Non-Fire classes. For instance, on set-1, the AFFD-FDL technique has effectively categorized a set of 155 images into the Fire class and 185 images into the Non-Fire class. In line with, set-2, the AFFD-FDL manner has efficiently categorized a set of 140 images into the Fire class and 202 images into the Non-Fire class. Additionally, on set-3, the AFFD-FDL approach

has effectually categorized a set of 161 images into the Fire class and 183 images into the Non-Fire class.

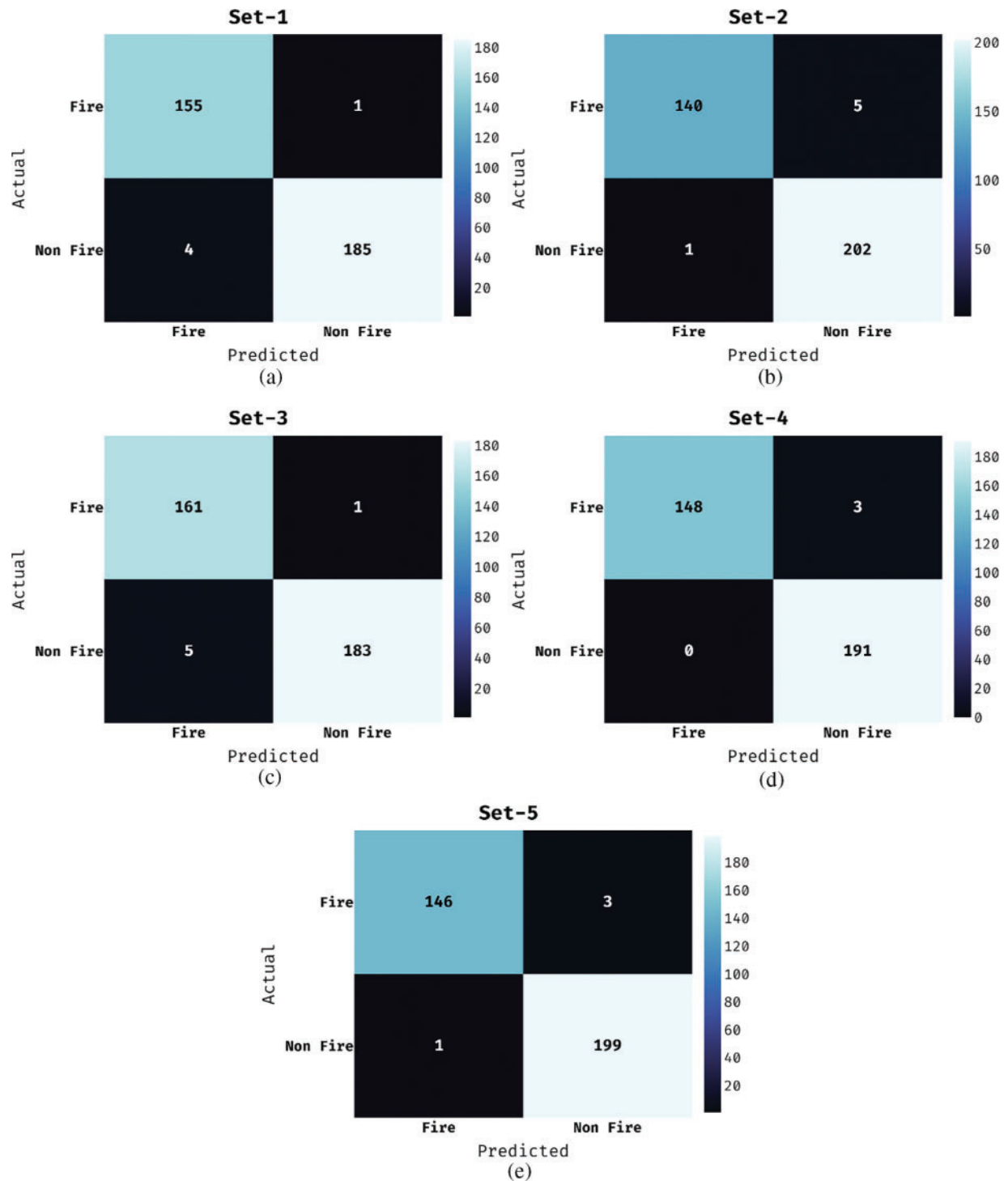


Figure 5: Confusion matrix of AFFD-FDL model

Moreover, on set-4, the AFFD-FDL method has effectively categorized a set of 148 images into the Fire class and 191 images into the Non-Fire class. Eventually, on set-5, the AFFD-FDL algorithm has efficiently categorized a set of 146 images into the Fire class and 199 images into the Non-Fire class.

Table 1 and Fig. 6 investigate the forest fire classification results analysis of the AFFD-FDL technique on five sets. The results reported the effective forest fire detection performance of the AFFD-FDL technique. For instance, on set-1, the AFFD-FDL technique has accomplished a precision of 0.986, recall of 0.994, accuracy of 0.986, F-score of 0.984, and MCC of 0.971. In line with, on set-2, the AFFD-FDL method has accomplished a precision of 0.993, recall of 0.966, accuracy of 0.983, F-score of 0.982, and MCC of 0.979. Along with that, on set-3, the AFFD-FDL manner has accomplished a precision of 0.970, recall of 0.994, accuracy of 0.983, F-score of 0.982, and MCC of 0.966. Followed by, on set-4, the AFFD-FDL algorithm has accomplished a precision of 1.000, recall of 0.980, accuracy of 0.991, F-score of 0.990, and MCC of 0.982. Lastly, on set-5, the AFFD-FDL methodology has accomplished a precision of 0.993, recall of 0.980, accuracy of 0.989, F-score of 0.987, and MCC of 0.977. Overall, the AFFD-FDL technique has demonstrated proficient performance with a maximum average precision of 0.986, recall of 0.983, accuracy of 0.986, F-score of 0.984, and MCC of 0.972.

Table 1: Result analysis of AFFD-FDL model with different sets

Dataset	Precision	Recall	Accuracy	F-score	MCC
Set-1	0.975	0.994	0.986	0.984	0.971
Set-2	0.993	0.966	0.983	0.979	0.965
Set-3	0.970	0.994	0.983	0.982	0.966
Set-4	1.000	0.980	0.991	0.990	0.982
Set-5	0.993	0.980	0.989	0.987	0.977
Average	0.986	0.983	0.986	0.984	0.972

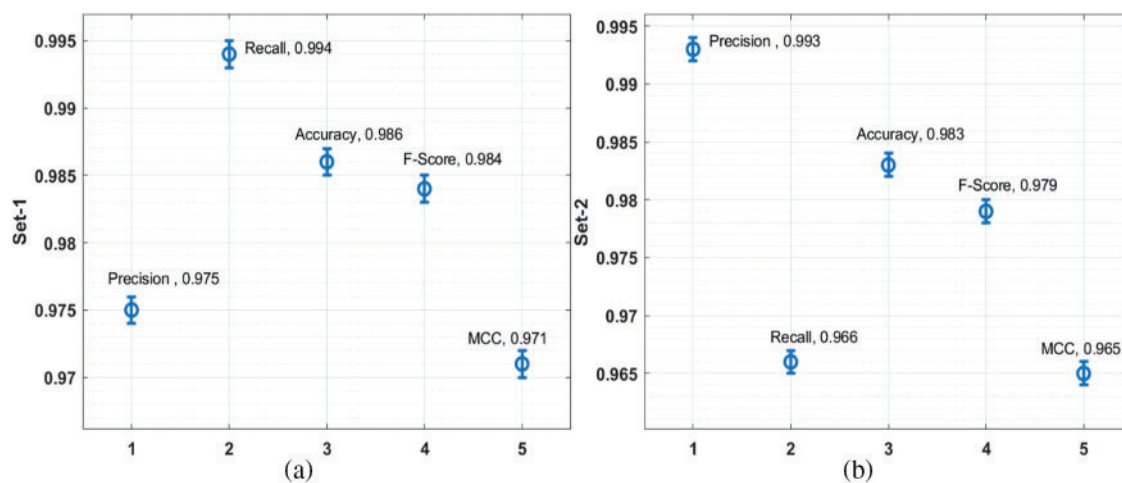


Figure 6: (Continued)

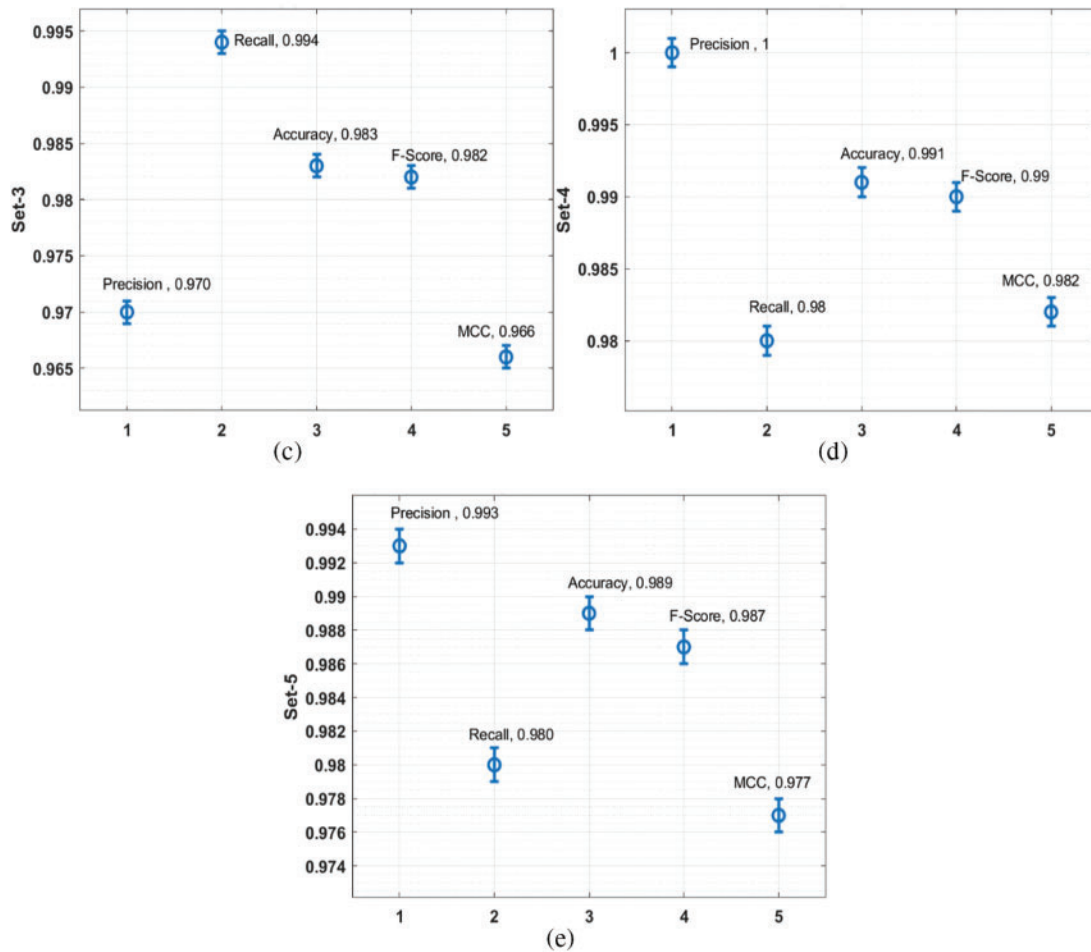


Figure 6: Result analysis of AFFD-FDL model with distinct sets

Finally, a comparative results analysis of the AFFD-FDL technique with existing techniques takes place in Table 2 and Fig. 7 [26]. The experimental results demonstrated that the AFFD-FDL technique has demonstrated superior performance over the other methods. On examining the results in terms of recall, the AFFD-FDL technique has offered an increased performance with a recall of 98.30% whereas the RF, DT, LR, and SVM models have attained a reduced performance with a recall of 98%, 54%, 62%, and 54%, respectively. Also, on investigating the outcomes with respect to precision, the AFFD-FDL manner has obtainable a higher performance with a precision of 98.60% whereas the RF, DT, LR, and SVM techniques have gained a lower performance with a precision of 98%, 52%, 51%, and 57%, correspondingly.

Table 2: Comparative analysis of AFFD-FDL model with existing approaches

Methods	Recall (%)	Precision (%)	F-score (%)	Accuracy (%)
AFFD-FDL	98.30	98.60	98.40	98.60
Random forest	98.00	98.00	98.00	97.00

(Continued)

Table 2 (continued)

Methods	Recall (%)	Precision (%)	F-score (%)	Accuracy (%)
Decision tree	54.00	52.00	53.00	50.00
Logistic regression	62.00	51.00	56.00	49.00
SVM	54.00	57.00	56.00	55.00

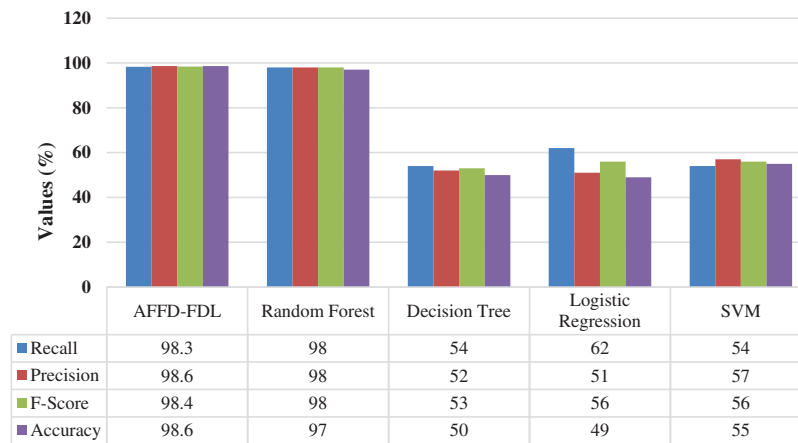


Figure 7: Comparative analysis of the AFFD-FDL model with present techniques

Besides, on inspecting the outcomes in terms of F-score, the AFFD-FDL technique has accessible an enhanced performance with the F-score of 98.40% whereas the RF, DT, LR, and SVM methodologies have achieved a minimum performance with the F-score of 98%, 53%, 56%, and 56%, respectively. Lastly, on observing the results with respect to accuracy, the AFFD-FDL technique has accessible a maximum performance with an accuracy of 98.60% whereas the RF, DT, LR, and SVM algorithms have obtained a lesser performance with an accuracy of 97%, 50%, 49%, and 55% correspondingly. From the above-mentioned results analysis, it is apparent that the AFFD-FDL technique has been found to be an effective forest fire detection model in a real-time environment.

5 Conclusion

In this study, a new AFFD-FDL technique is derived for automated forest fire detection. The AFFD-FDL technique involves a 2-stage process namely fusion-based feature extraction and classification. An entropy-based fusion model is employed to derive a useful subset of features. It encompasses three models namely HOG, Inception v3, and SqueezeNet. Moreover, the optimal ELM using the GSO algorithm is applied to fine-tune the parameters involved in the ELM model and thereby boost the classifier results. An extensive set of simulations were performed on benchmark images and the results are examined under different dimensions. The experimental results highlighted the betterment of the AFFD-FDL technique over the recent state of art techniques. In the future, the presented AFFD-FDL technique can be realized in an Internet of Things (IoT) based environment to assist real-time monitoring.

Acknowledgement: None.

Funding Statement: The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work under Grant Number (RGP.1/172/42). Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2023R191), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. This study is supported via funding from Prince Sattam bin Abdulaziz University Project Number (PSAU/2023/R/1444).

Authors Contributions: Conceptualization, M.A. and M.M.E.; methodology, M.A.; software, O.A.O.; validation, M.A.; formal analysis A.A.A.; investigation, F.N.A.; resources, M.A.; data curation, A.A.A.; writing—Original draft preparation, M.A.; A.A.A.; M.M.E. and F.N.A.; writing—Review and editing, A.M.H.; M.A.H.; and M.R.; visualization, M.R.; supervision, M.A.; project administration, A.M.H.; funding acquisition, A.A.A. and M.M.E. All authors have read and agreed to the published version of the manuscript.

Availability of Data and Materials: Data sharing is not applicable to this article as no datasets were generated during the current study.

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

References

- [1] P. Barmpoutis, P. Papaioannou, K. Dimitropoulos and N. Grammalidis, “A review on early forest fire detection systems using optical remote sensing,” *Sensors*, vol. 20, no. 22, pp. 6442, 2020.
- [2] M. A. Tanase, C. Aponte, S. Mermoz, A. Bouvet, T. Le Toan *et al.*, “Detection of windthrows and insect outbreaks by L-band SAR: A case study in the bavarian forest national park,” *Remote Sensing of Environment*, vol. 209, pp. 700–711, 2018.
- [3] K. Bouabdellah, H. Noureddine and S. Larbi, “Using wireless sensor networks for reliable forest fires detection,” *Procedia Computer Science*, vol. 19, pp. 794–801, 2013.
- [4] A. Gaur, A. Singh, A. Kumar, A. Kumar and K. Kapoor, “Video flame and smoke based fire detection algorithms: A literature review,” *Fire Technology*, vol. 56, no. 5, pp. 1943–1980, 2020.
- [5] S. E. Memane and V. S. Kulkarni, “A review on flame and smoke detection techniques in video’s,” *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 4, pp. 885–889, 2015.
- [6] S. Garg and A. A. Verma, “Review survey on smoke detection,” *Imperial Journal of Interdisciplinary Research*, vol. 2, no. 9, pp. 935–939, 2016.
- [7] F. Bu and M. S. Gharajeh, “Intelligent and vision-based fire detection systems: A survey,” *Image and Vision Computing*, vol. 91, pp. 103803, 2019.
- [8] J. Alves, C. Soares, J. M. Torres, P. Sobral and R. S. Moreira, “Automatic forest fire detection based on a machine learning and image analysis pipeline,” in *WorldCIST’19 2019: New Knowledge in Information Systems and Technologies*, New York, NY, USA, pp. 240–251, 2019.
- [9] M. Li, W. Xu, K. E. Xu, J. Fan and D. Hou, “Review of fire detection technologies based on video image,” *Journal of Theoretical and Applied Information Technology*, vol. 49, no. 2, pp. 700–707, 2013.
- [10] A. A. A. Alkhatib, “A review on forest fire detection techniques,” *International Journal of Distributed Sensor Networks*, vol. 10, no. 3, pp. 597368, 2014.
- [11] Y. Wang, L. Dang and J. Ren, “Forest fire image recognition based on convolutional neural network,” *Journal of Algorithms & Computational Technology*, vol. 13, 2019.

- [12] Z. Jiao, Y. Zhang, J. Xin, L. Mu, Y. Yi *et al.*, “A deep learning based forest fire detection approach using UAV and YOLOv3,” in *2019 1st Int. Conf. on Industrial Artificial Intelligence (IAI)*, Shenyang, China, pp. 1–5, 2019.
- [13] A. Kansal, Y. Singh, N. Kumar and V. Mohindru, “Detection of forest fires using machine learning technique: A perspective,” in *2015 Third Int. Conf. on Image Information Processing (ICIIP)*, Wagnaghat, India, pp. 241–245, 2015.
- [14] D. Kinaneva, G. Hristov, J. Raychev and P. Zahariev, “Early forest fire detection using drones and artificial intelligence,” in *2019 42nd Int. Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, pp. 1060–1065, 2019.
- [15] R. Xu, H. Lin, K. Lu, L. Cao and Y. Liu, “A forest fire detection system based on ensemble learning,” *Forests*, vol. 12, no. 2, pp. 217, 2021.
- [16] Y. Chen, Y. Zhang, J. Xin, G. Wang, L. Mu *et al.*, “UAV Image-based forest fire detection approach using convolutional neural network,” in *2019 14th IEEE Conf. on Industrial Electronics and Applications (ICIEA)*, Xi’an, China, pp. 2118–2123, 2019.
- [17] S. Wu and L. Zhang, “Using popular object detection methods for real time forest fire detection,” in *2018 11th Int. Symp. on Computational Intelligence and Design (ISCID)*, Hangzhou, China, pp. 280–284, 2018.
- [18] D. Sinha, R. Kumari and S. Tripathi, “Semisupervised classification based clustering approach in WSN for forest fire detection,” *Wireless Personal Communications*, vol. 109, no. 4, pp. 2561–2605, 2019.
- [19] K. Mizuno, Y. Terachi, K. Takagi, S. Izumi, H. Kawaguchi *et al.*, “Architectural study of HOG feature extraction processor for real-time object detection,” in *2012 IEEE Workshop on Signal Processing Systems*, Quebec City, QC, Canada, pp. 197–202, 2012.
- [20] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally *et al.*, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size,” arXiv preprint arXiv:1602.07360, 2016.
- [21] N. Dong, L. Zhao, C. H. Wu and J. F. Chang, “Inception v3 based cervical cell classification combined with artificially extracted features,” *Applied Soft Computing*, vol. 93, pp. 106311, 2020.
- [22] K. Hu, Z. Zhou, L. Weng, J. Liu, L. Wang *et al.*, “An optimization strategy for weighted extreme learning machine based on PSO,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 31, no. 1, pp. 1751001, 2017.
- [23] K. N. Krishnanand and D. Ghose, “Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions,” *Swarm Intelligence*, vol. 3, no. 2, pp. 87–124, 2009.
- [24] J. Alves, C. Soares, J. Torres, P. Sobral and R. S. Moreira, “Color algorithm for flame exposure (CAFE),” in *2018 13th Iberian Conf. on Information Systems and Technologies (CISTI)*, Caceres, Spain, pp. 1–7, 2018.
- [25] <http://www.ultimatechase.com/>
- [26] M. D. Molovtsev and I. S. Sineva, “Classification algorithms analysis in the forest fire detection problem,” in *2019 Int. Conf. “Quality Management, Transport and Information Security, Information Technologies” (IT&QM&IS)*, Sochi, Russia, pp. 548–553, 2019.