IUST
Iran University of Science and Technology

Iranian Journal of Electrical and Electronic Engineering

Journal Homepage: ijeee.iust.ac.ir

Deep Learning Approach for Forest Fire Detection: A CNN Classification Model on the DeepFire Dataset

Tara Sistani*, Seyed-Javad Kazemitabar*(C.A.)

Abstract: Forests play several vital roles in our lives and provide various resources. However, in recent years, the increasing frequency of wildfires has led to the widespread burning and destruction of many forests and wildlands. Therefore, detecting forest fires and finding suitable solutions to address this issue has become one of the critical challenges for researchers. Today, with the advancement of artificial intelligence, forest fire detection using deep learning is an important method with the aim of increasing the efficiency of forest fire detection and monitoring systems. In this article, a method based on a type of convolutional neural network called Xception is proposed for classifying forest fire images. In this method, transfer learning technique is used on the proposed neural network and a new classifier is designed for the problem. Also, various hyperparameters have been used to optimize the performance of the proposed model. The proposed method is performed on the DeepFire dataset, which contains 1900 images equally divided between fire and no-fire classes. The results obtained from the implementation of the proposed method show that this method with an accuracy of 99.47% has achieved a favorable performance in classifying forest fire images.

Keywords: Artificial Intelligence, Classification, Convolutional Neural Network, Deep Learning, Forest Fire Detection.

1 Introduction

FORESTS are essential to our ecosystem and daily lives, offering a wide range of resources and services, from clean air and water to biodiversity and climate regulation [1]. However, these vital ecosystems are increasingly under threat from various factors, with wildfires being one of the most significant [2]. In recent years, the frequency, intensity, and duration of wildfires have been escalating [3], leading to widespread destruction of forests and wildlands [1]. This growing threat underscores the urgent need for effective strategies to detect and mitigate forest fires, as preserving our forests is crucial for environmental sustainability and human well-being.

Given the immense value of forests, the early detection of forest fires and the development of strategies to minimize the damage they cause have become critical challenges for researchers. Fortunately, recent advancements in Artificial Intelligence (AI), particularly in Machine Learning (ML) and Deep Learning (DL), offer promising solutions [2]. These technologies enable the development of sophisticated systems capable of detecting forest fires quickly and accurately, providing a powerful and efficient approach to combating this escalating threat [2].

Machine learning encompasses a range of methods and algorithms that enable computers to learn from data, analyze it, and uncover hidden patterns, ultimately allowing machines to make predictions and informed decisions [4]. Among these methods, deep learning stands out as a particularly powerful subset of machine learning, renowned for its ability to process vast amounts of data with minimal human intervention. Deep

Iranian Journal of Electrical & Electronic Engineering, 2025.
Paper first received 09 Dec. 2024 and accepted 26 Jul. 2025.
* The author is with the Department of Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Mazandaran, Iran.
E-mail: tarasistani@stu.nit.ac.ir, j.kazemitabar@nit.ac.ir.

Corresponding Author: Seyed-Javad Kazemitabar.

learning algorithms are inspired by the neural networks of the human brain, allowing them to automatically learn features and patterns directly from raw data, without the need for manually designed rules [5].

In other words, unlike traditional machine learning, which often relies on handcrafted features and predefined rules, deep learning excels by leveraging large datasets to map inputs to specific labels with remarkable accuracy. This capability makes deep learning especially promising for complex tasks like forest fire detection [5]. The ability to analyze massive datasets with high precision can significantly enhance the speed and accuracy of detecting and responding to wildfires [6].

In recent years, with the explosion of available data and advancements in computational power, deep learning has been increasingly employed for various wildfire-related tasks, including classification, detection, and segmentation. Its ability to process both ground and aerial imagery has demonstrated its potential to overcome the limitations of classical machine learning methods. Furthermore, integrating deep learning into automated fire detection systems can be a game-changer, enabling the rapid and accurate identification of fires through camera feeds. This technology holds great promise for the development of AI-driven fire detection agents, which could play a crucial role in monitoring and managing wildfires more effectively [7].

Deep learning utilizes Artificial Neural Networks (ANN) to process and analyze large volumes of data through complex computations [8]. In these networks, there are many layers between the input and output [9], so that, the initial layers focus on extracting low-level features, such as edges or textures, while the deeper layers progressively extract higher-level features, such as shapes or objects, enabling the model to understand the data at a more abstract level [5].

One of the most widely used and effective deep learning algorithms is the Convolutional Neural Network (CNN) [10], which typically consists of three types of layers: convolutional layers, pooling layers, and fully connected (FC) layers [11]. Convolutional layers are responsible for scanning the input image with filters to detect features like edges or textures. Pooling layers reduce the dimensionality of the data, making the model more computationally efficient while retaining important information. Finally, fully connected layers combine these extracted features to make predictions or classifications [8].

CNNs are particularly well-suited for image analysis [12] because they treat input images as two-dimensional (2D) matrices, preserving the spatial relationships between pixels [13]. Due to this characteristic, CNNs have been extensively researched and applied in the field of image recognition and classification. One of these applications, is the classification of forest fires. In this research, we aim to provide a mechanism with an optimal accuracy for early forest fire classification. In summary, we can list the contributions of our study as follows;

(1) Comprehensive review of prior research in this field,
(2) Detailed introduction and presentation of the proposed method,
(3) Evaluation of the proposed method and a comprehensive analysis of the results obtained from its implementation,
(4) Comparative performance analysis of the proposed neural network against machine learning algorithms,
(5) Comparative performance analysis of the proposed neural network against similar architectures,
(6) Comparison of the proposed method with other studies for performance validation.

The rest of the paper is organized as follows. Section 2 details the review of the literature on deep learning-based forest fire classification methods; Section 3 covers the dataset used in this study, the proposed neural network, the proposed classifier, hyperparameter tuning, Transfer Learning (TL), and evaluation metrics; Section 4 presents the experimental results obtained from implementing the proposed method. Additionally, this section provides a performance comparison between the proposed model and other models by implementing the proposed method on comparable architectures and analyzing the results. A comparison of the proposed method with other studies and approaches is also presented in this section. Finally, Section 5 presents a general conclusion of this study.

2 Literature Review

In recent years, numerous studies have been conducted on forest fire detection systems based on deep learning in order to set up a fire detection system as accurately as possible. The main goal of most of these studies has been to detect forest fires using classification methods. One such study, presented in [14], proposes a transfer learning approach using the Inception-v3 model for fire detection. Another study [15] presents several deep learning architectures trained for early fire detection using images captured by drones. This study uses a two-step transfer learning approach on five different CNN-based models. In the first step, the feature extraction layer weights of each model are frozen, while new classifiers are trained for five epochs. In the second training phase, all layers are unfrozen, and training continues for five more epochs with a lower learning rate. In [16], the Xception network as a binary classifier model is proposed. In this paper, the model is trained from scratch on a new dataset called FLAME, which was collected and provided by the authors of the article. The authors in [17] propose the FFireNet method based on CNN for classifying forest fires. This paper uses transfer learning techniques on the MobileNet-v2 network to address the problem of data scarcity and computational power limitations. Another study [18] provides a solution for challenges such as small fire sizes, complex backgrounds, and image degradation. This paper proposes a novel deep learning method combining EfficientNet-b5 and DenseNet201 models for fire detection and classification using aerial images. In [19], two new deep feature engineering models are proposed to detect the fire accurately using images. To create deep features, four pretrained ResNets: ResNet18, ResNet50, ResNet101, and InceptionResNetV2 are used. By using the eight feature vectors generated of these networks, two ensemble models have been presented. In the first ensemble model, all generated features are concatenated, and the top 1000 features are selected using Neighborhood Component Analysis (NCA), after which these features are classified using Support Vector Machine (SVM). In the second ensemble model, Iterative Hard Majority Voting (IHMV) has been applied to the generated results. In [20], First, an improved Dynamic Convolutional Neural Network (DCNN) network model is trained in combination with transfer learning, and multiple pre-trained DCNN models are used to extract features from forest fire images. Second, Principal Component Analysis (PCA) reconstruction technology is used to convert features into a shared feature subspace. Another study [21] presents a type of convolutional neural network that is very suitable for real-time applications, as it uses separable convolutional layers. The authors in [22] propose a novel module using attention mechanism for convolution kernels, which can dynamically select and fuse feature maps from different scales of convolution kernels, termed the Dual Semantic Attention (DSA) module and on the basis of ResNet, the above-mentioned DSA module is integrated into the model**.** In [23], several methods are proposed to solve the problem of forest fire detection. In the first method, the Inception-v3, DenseNet121, ResNet50-v2, VGG19, and NASNet-Mobile models are trained from scratch for forest fire classification. In the second method, transfer learning is applied to the mentioned networks. In the third method, fine-tuning of these pre-trained networks is explored. Finally, in the last method, SVM, Random Forest (RF), Bidirectional Long Short-Term Memory (BiLSTM), Gated Recurrent Unit (GRU) algorithms, and hybrid approaches are employed to enhance the efficiency of the Inception-v3 and NASNet-Mobile models. The authors in [24] explore the potential of RGB image data for forest fire detection using the MobileNet-v2 neural network. The main focus of this paper is to address the prevalent issue of high false alarm rates in DL-based fire detection systems, and in this regard, it proposes two distinct methods: a one-step classification and a two-step MTL multi-class classification. Another study [25] uses a deep learning-based architecture that first combines three convolutional neural network architectures, namely, XceptionNet, MobileNetV2, and ResNet-50, as an ensemble. The second contribution is linked to the implementation of the fire and smoke detection model by using the YOLO architecture. The study [26] addresses the limitations of deep learning in handling limited and complex fire data by using an SVM with RBF kernel to classify fire and non-fire pixels. It employs processed data from the Corsican, FLAME, and Firefront_Gestosa datasets, and uses information-theoretic feature selection to improve classification efficiency by reducing dimensionality. In [27], the SWIFT dataset, a collection of synthetic images, videos, annotations, and environmental data for wildfire detection, is introduced. This dataset is used to train and test three deep learning models—BoucaNet, DC-Fire, and CT-Fire—which are evaluated on real wildfire images. Another study [28] proposes 3ENB2, an end-to-end deep learning model based on EfficientNetB2 with transfer learning for fire detection from images. It incorporates online data augmentation techniques like random rotation and horizontal flipping during training. The study [29] presents a wildfire prediction framework for Morocco using a newly developed localized dataset built from multisource environmental observations. It employs machine learning and deep learning algorithms to forecast next-day wildfire events. In [30], a dataset derived from JULES-INFERNO simulations, which provide global climate and wildfire data, including environmental variables and fire occurrence data, is used. This enables the construction of a graph-based representation for training the wildfire prediction model. Another study [31] proposes a UAV-based real-time forest fire detection system using deep learning. It evaluates YOLOv5n and YOLOv8n for object detection, as well as CNN-RCNN and YOLOv8 classification models. Another study [32] introduces FireNet-CNN, a deep learning model for real-time forest fire detection on resource-constrained devices. To address dataset size and class imbalance, it uses synthetic data augmentation with Stable Diffusion and two custom-augmented datasets from various video and image sources. The model also integrates explainable AI techniques like Grad-CAM and Saliency Maps to improve transparency in fire detection. The authors in [33] propose a novel fire detection framework that combines LapSRN-based super-resolution with an attention-enhanced Xception network and Adaptive Spatial Attention for better feature focus. It uses a custom high-resolution, imbalanced fire/non-fire dataset for training and evaluation. The model also compares various pretrained DNNs with attention modules on both the custom dataset and a standard benchmark.

The aforementioned methods for forest fire classification demonstrate effectiveness in addressing the classification problem. However, despite their success, these existing methods often face challenges, such as high false alarm rates and limited scalability in real-time monitoring systems, especially when dealing with complex environmental conditions. Since accurate and early detection is crucial in minimizing the damage caused by fires, enabling timely intervention, and reducing the strain on firefighting resources, it is essential to address these limitations. To reduce the number of false alarms while achieving higher accuracy, in the next section, we present a deep learning-based forest fire classification method that is expected to enhance the efficiency and reliability of forest fire detection and monitoring systems.

3 Material and Methods

In this section, we first describe the dataset used in the training and testing phase of the proposed model. Then, we introduce our proposed method for classifying forest fire images.

3.1 Dataset

Data is the main element of all deep learning-based classification algorithms, such that the efficiency of algorithms largely depends on the nature and characteristics of the dataset used [13]. In this research, we use the DeepFire dataset mentioned in [17] and [34], which consists of forest fire and no-fire images aimed at the problem of forest fire detection and classification. This dataset has been collected from various online sources and contains images from multiple viewpoints and a wide range of landscapes to better train the model in distinguishing fire images from no-fire ones. It also includes various environmental conditions such as smoke, fog, rain, sunlight glare, and shadows, which reflect real-world scenarios and enhance the robustness of the model. However, the trained model still may fail to generalize to new and unseen data. Therefore, to prevent this problem, data augmentation techniques such as resizing, flipping, shifting, zooming, and others were applied to this dataset. This dataset contains a total of 1900 images with dimensions of 250 × 250, equally divided into two classes: fire and no-fire. The fire class contains images of forests and mountains with visible flames or flames accompanied by smoke clouds, while the no-fire class contains images of lush forests and mountains from different angles [17], [34].

Here, in line with the Deepfire dataset's two-class structure, we adopted a binary classification method to address the problem. Also, to ensure compatibility with the neural network (Xception), the image dimensions were adjusted from 250×250 to 224×224 to match the network's required input size. The dataset was then divided into three parts: training, validation, and testing datasets. For this division, 10% of the total data was initially set aside for testing, and the remaining images were split in a 90:10 ratio, with 90% allocated for training and 10% for validation. This splitting is shown in Table 1.

Table 1. Dataset splitting.

Dataset	Training + Validation	Test	Total
Fire	850	100	950
No-Fire	860	90	950
Total	1710	190	1900

3.2 Proposed approach

In this paper, we propose a CNN-based method for forest fire classification. A classifier is a mathematical model designed to extract the most critical information and relevant features from labeled images in the training dataset, and then apply this knowledge to classify images in the test dataset [35]. Fig. 1 shows the structure of our proposed method. This model comprises two main components: a pre-trained base model and newly added fully connected layers.

In the first part of our model, the Xception neural network [36] is used as the base model, responsible for extracting features and useful information from the images in the training dataset. The design of Xception, as a more powerful version of the Inception architecture, assumes that channel correlations and spatial correlations in convolutional layers can be effectively separated, which enhances its feature extraction capabilities. This separation allows Xception to build upon the strengths of Inception, offering improved performance by simplifying and refining how information is processed through the network. This network is structured into three main parts: entry flow, middle flow, and exit flow. The architecture comprises 36 convolutional layers organized into 14 modules, with all but the first and last modules featuring linear residual connections [36]. Here, we applied the transfer learning technique [37], [38], [39] to the Xception neural network by utilizing pre-trained weights from the ImageNet dataset, rather than initializing the network with random weights. Our approach also allows for the fine-tuning of weights and biases across all layers, enhancing the model's ability to adapt to the specific task at hand.

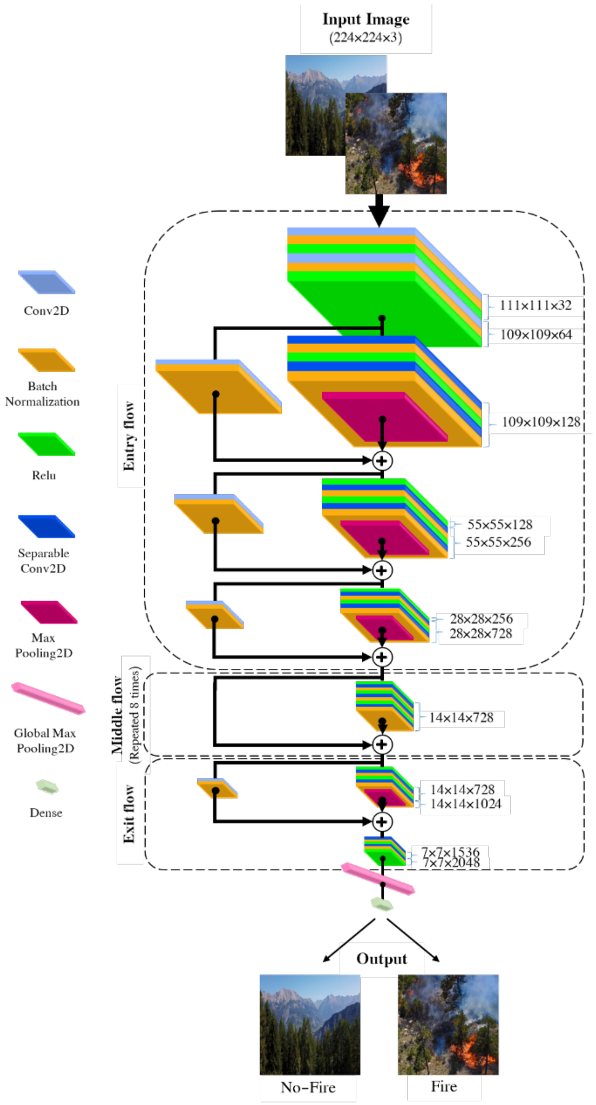

Fig 1. The architecture of proposed model.

While the original Xception model is designed for classify the images into 1000 classes, our target task is binary classification (fire vs. no-fire). To address this, in the second part of our approach, we introduce a classifier specifically tailored to the binary classification problem, optimizing the model's performance for this task. In this part, the output from the final layer of the base model, which has dimensions of 7×7 with 2048 channels, is passed through a global max pooling layer. This layer reduces the output to a vector of 2048 neurons, each representing the maximum value from its corresponding feature map (channel). This process helps to capture the most significant features from the feature maps. The resulting vector is then fed into a dense layer with a softmax activation function, which handles the classification task. This function transforms raw scores into a probability distribution across the different classes. Equation (1) expresses this function as follows:

$$S(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}} \quad (1)$$

Where z_i is the raw score for the i-th class and N is the total number of classes. In all classification algorithms, there are hyperparameters that are set before training begins and remain unchanged throughout the training process. One such hyperparameter is the loss function, for which we used the categorical cross-entropy function. This choice aligns with the softmax function used in our model, ensuring the output probabilities are effectively optimized during training. This function is expressed as follows in Eq. (2):

$$L(y.\hat{y}) = -\sum_{i=1}^{N} y_i \, log(\hat{y}_i) \quad (2)$$

Where y is the true probability distribution, $\hat{y}$ is the predicted probability distribution and N is the total number of classes. In addition to selecting the appropriate loss function, the choice of optimization algorithm is equally critical for enhancing model performance. We employed the Adam Optimizer, which, compared to other optimization algorithms, requires less memory and converges faster, particularly in the early stages of training. Adam's ability to effectively manage large and small gradients—a common challenge in deep learning problems—further supports its use in our method. Here, a fixed learning rate was not explicitly set. Instead, a minimum learning rate (min-Lr) of 1e-04 was applied. In this case, initially, the model begins with a default learning rate, which is not predetermined. As training progresses, the learning rate is gradually reduced by the learning rate scheduler until it reaches the specified minimum value. This approach allows the model to adjust its learning pace dynamically, improving training efficiency. Additionally, in this method, we determined an optimal batch size of 64, and the model was trained over 40 epochs to strike a balance between training time and performance. The hyperparameters used to optimize the performance of the proposed method are summarized in Table 2.

Table 2. Hyperparameters.

Hyperparameters	Value
Loss Function	Categorical Cross-Entropy
Optimizer	Adam
min-Lr	1e-04
Batch Size	64
Epoch	40

3.3 Performance metrics

The performance of the proposed method will be evaluated using various metrics, including accuracy, precision, recall and F1-score, in the next section. These evaluation metrics are presented in Table 3. In these relationships, a true Positive (TP) means there is a fire in the input image, and the model correctly predicts that there is a fire. A true negative (TN) means there is no fire in the input image, and the model correctly predicts that there is no fire. A false negative (FN) means there is a fire in the input image, and the model incorrectly predicts that there is no fire. A false positive (FP) means there is no fire in the input image, and the model incorrectly predicts that there is a fire.

Table 3 The performance metrics.

Metrics	Formula
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
F1-Score	$2*\frac{\text{Precision}*\text{Recall}}{\text{Precision}+\text{Recall}}$

4 Results and Discussion

4.1 Performance analysis of the proposed method

In this section, we analyze the results generated from implementing the proposed method. The process began with the training phase, where the model was trained on the dataset while simultaneously undergoing validation to monitor performance. In another word, during the training process, at each epoch, the loss on the validation dataset is monitored. If the current epoch's loss is lower than that of previous epochs, the model is saved for use in the testing phase. In this approach, the model achieved the lowest validation loss at epoch 31, and this version of the model was saved for classifying the test dataset. Figs. 2 and 3 illustrate the loss and accuracy per epoch for both the training and validation datasets, respectively, providing insight into the model's performance during each stage of the process. Also, the details of the saved model are provided in Table 4.

Table 4. Results of the proposed model on the training and validation datasets.

Epoch	Train Loss	Train Accuracy	Validation Loss	Validation Accuracy	Learning Rate
31	0.0291	0.9981	0.0338	0.9942	0.0001

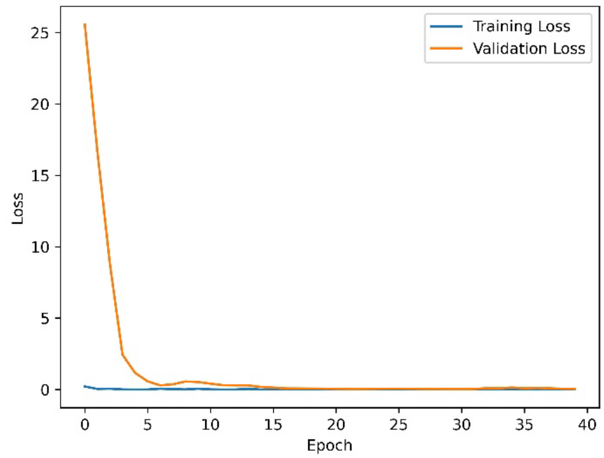

Fig 2. Loss chart for training and validation data.

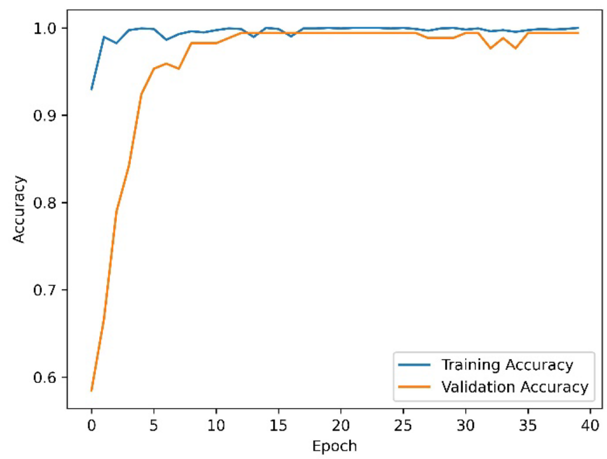

Fig 3. Accuracy chart for training and validation data.

After completing the training phase, the trained model was applied to the test dataset to classify the images and evaluate its effectiveness. Table 5 presents a comprehensive report and analysis of the model's performance on the test dataset, while Table 6 breaks down these metrics by class. Also, Fig. 4 presents the confusion matrix for the proposed method. The test dataset consists of 190 images, with 100 belonging to the fire class and 90 to the no-fire class. According to the matrix results, the model accurately classified all images in the fire class, achieving a false negative rate of zero. However, one image from the no-fire class was incorrectly classified as fire, resulting in a false positive rate of one. Additionally, Fig. 5 displays the Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) for this method.

Table 5. Results of the proposed model on the test datasets.

Loss	Accuracy	Precision	Recall	F1-Score
0.009105	0.9947	0.9901	1.0000	0.9950

Table 6. Results of the proposed model for each class in the test datasets.

Class	Precision	Recall	F1-Score
Fire	0.99	1.00	1.00
No-Fire	1.00	0.99	0.99

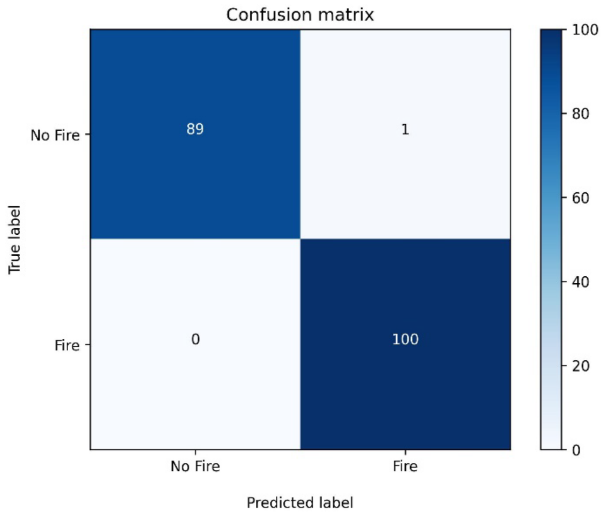

Fig 4. The confusion matrix for the proposed method.

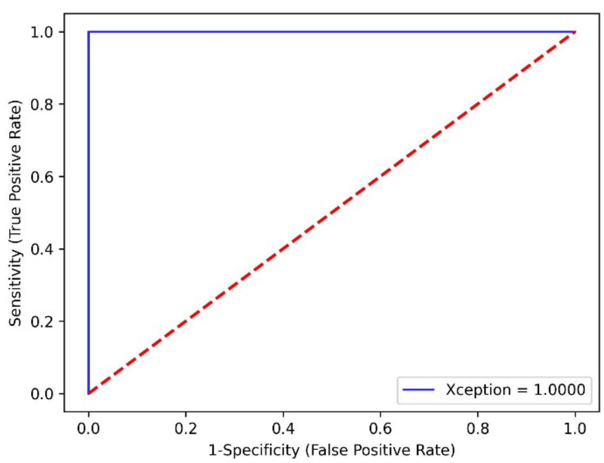

Fig 5. The ROC curve for the proposed method.

Furthermore, Figs. 6 and 7 showcase examples of the model's correct and incorrect classifications from the test dataset. Specifically, Figs. 6(a) and 6(b) display images from the fire and no-fire classes, respectively, which the model accurately classified, despite challenges like the presence of sunlight, autumn leaves, cloud and fog—elements that could potentially be mistaken for fire. In contrast, Fig. 7 presents the only image from the no-fire class that the model mistakenly classified as fire. This misclassification is likely due to the presence of a dark cloud in the image, resembling black smoke typically associated with fires.

To further evaluate the generalizability of the proposed model, we tested it on images not included in the DeepFire dataset, randomly collected from other sources. These samples cover varied geographical regions and environmental conditions such as fog, different lighting, and seasonal vegetation. Representative results are shown in Figs. 8 and 9. Fig. 8(a) and 8(b) display correctly classified images from the fire and no-fire classes, respectively. Fig. 9 shows a no-fire image misclassified as fire, likely due to fog and autumn leaves creating a visual effect similar to smoke. This example highlights the challenge of such ambiguous cases, even for human observers.

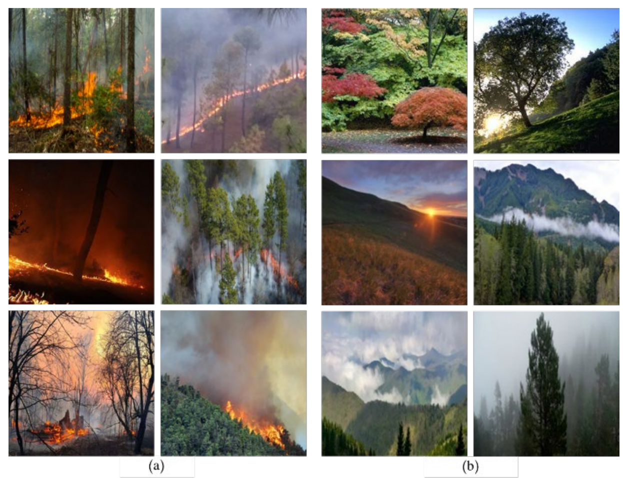

Fig 6. True classification (a) Fire (b) No-Fire.

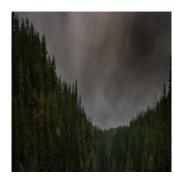

Fig 7. False classification (ground truth= No-Fire).

Fig 8. True classification (a) Fire (b) No-Fire.

Fig 9. False classification (ground truth= No-Fire).

4.2 Computational costs and resource requirements

In terms of computational requirements, the model was trained using Google Colab Pro, which provides access to high-performance GPUs such as the NVIDIA Tesla T4 or P100. The total training time for the final model was approximately 1,252 seconds (about 21 minutes). The trained model, based on the Xception architecture with a custom classifier, consists of approximately 20.87 million parameters and occupies around 79.60 MB of storage.

This level of complexity suggests that the model is suitable for deployment on systems with moderate computational resources. However, further optimization may be necessary for real-time deployment on highly constrained edge devices.

4.3 Performance comparison with machine learning algorithms

In this subsection, we compare the performance of the proposed deep learning model with a traditional machine learning method, namely the Support Vector Machine (SVM) [40]. The comparison aims to highlight the advantages of the proposed approach over non-deep learning techniques in the task of forest fire classification. Table 7 presents the details of this comparison.

Furthermore, the confusion matrices of the proposed method and the traditional SVM algorithm are shown in Fig. 10. The Xception model achieved near-perfect classification, correctly identifying 100 fire instances and 89 no-fire instances, with only 1 false positive and no false negatives. In contrast, the SVM model exhibited slightly lower performance, with 94 true positives and 87 true negatives, accompanied by 6 false negatives and 3 false positives. This clearly demonstrates that Xception is more precise and reliable in detecting both fire and no-fire cases.

Additionally, the Receiver Operating Characteristic (ROC) curves in Fig. 11 further support this finding, with the Xception model achieving an AUC of 1.0000 compared to 0.9533 for the SVM model. As evident from the figures, the Xception network significantly outperforms the SVM model in terms of classification accuracy and robustness.

4.4 Performance comparison with other CNN models

This subsection provides a performance comparison of the proposed approach on the Deepfire dataset against other CNN models, including architectures similar to Xception, such as Inception-v3 [41] and MobileNet-v2 [42]. In this evaluation, the proposed method was applied to these networks without altering any parameters, allowing for a direct comparison of their performance. To assess the networks' performance during the training and validation phases, Figs. 12 and 13 illustrate the loss and accuracy per epoch for each network on both training and validation data, respectively.

Moreover, the confusion matrices of the three networks are compared in Fig. 14. As previously mentioned, the Xception network achieved a perfect classification of all fire class images, resulting in a false negative rate of zero, but misclassified one of the 90 no-fire class images as belonging to the fire class. In contrast, the Inception-v3 network misclassified two fire class images as no-fire and additionally produced a false positive rate of two by misclassifying two no-fire class images as fire. The MobileNet-v2 network misclassified a total of six images from the test dataset, with a false negative rate of one, a false positive rate of five, a true positive rate of 99, and a true negative rate of 85. Table 8 details the results of the comparison.

Additionally, the Receiver Operating Characteristic (ROC) curves for Xception, Inception-v3, and MobileNet-v2 are shown together in Fig. 15 to facilitate comparison. According to this figure, the Xception network shows the best performance, followed by Inception-v3 and then MobileNet-v2.

Table 7. Comparative analysis of proposed approach with SVM.

Model	Accuracy	Precision	Recall	F1-Score	Time
Xception	0.9947	0.9901	1.0000	0.9950	1252 sec.
SVM	0.9526	0.9691	0.9400	0.9543	4020 sec.

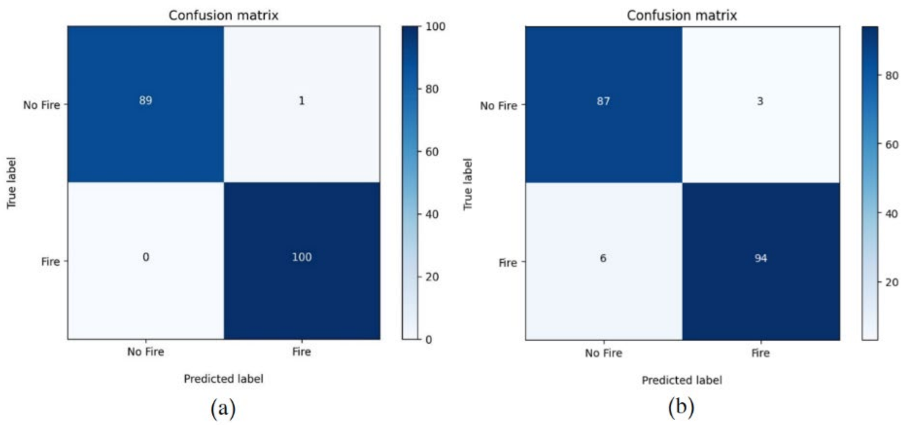

Fig 10. The confusion matrix for (a) Xception (b) SVM.

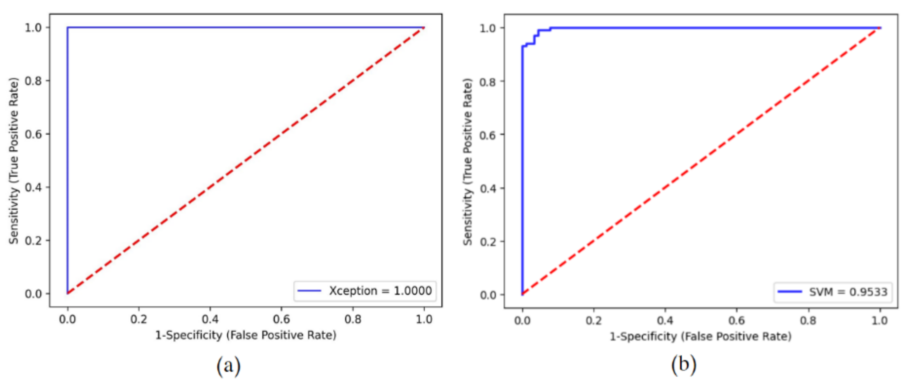

Fig 11. The ROC curves for (a) Xception (b) SVM.

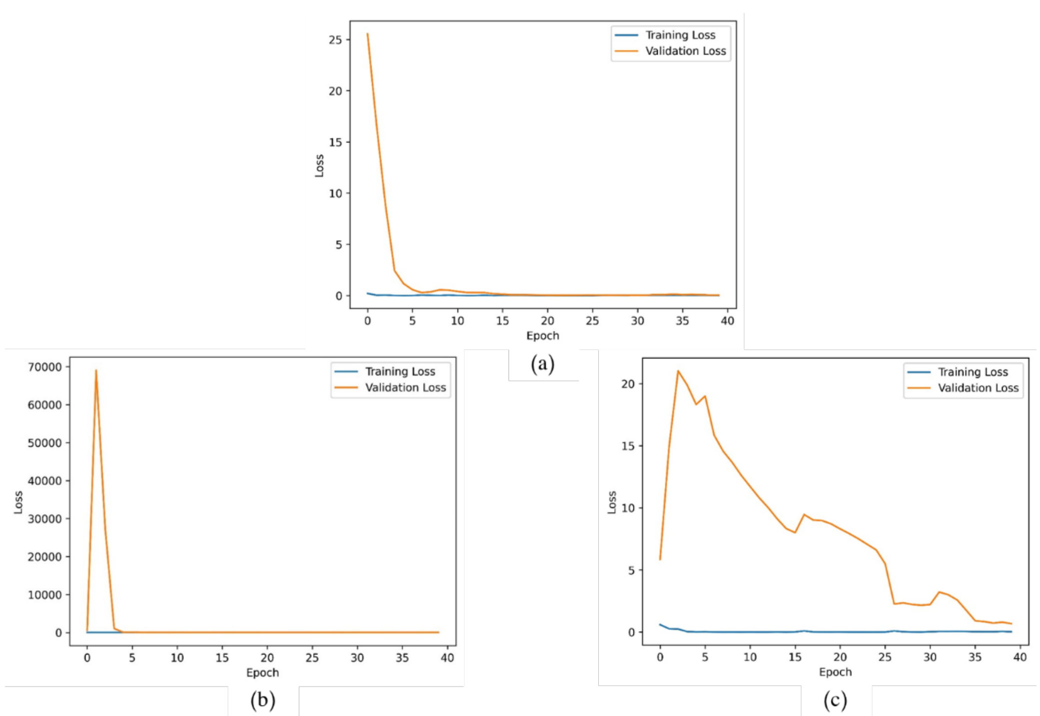

Fig 12. Loss chart for training and validation data for (a) Xception (b) Inception-v3 (c) MobileNet-v2.

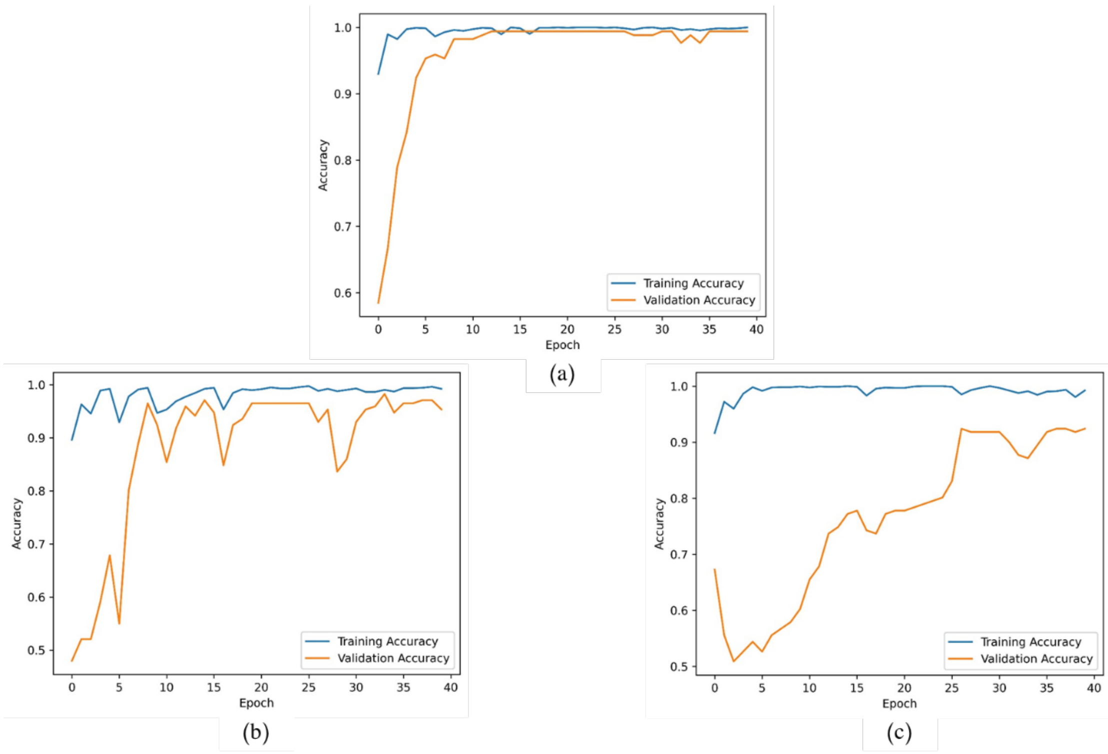

Fig 13. Accuracy chart for training and validation data for (a) Xception (b) Inception-v3 (c) MobileNet-v2.

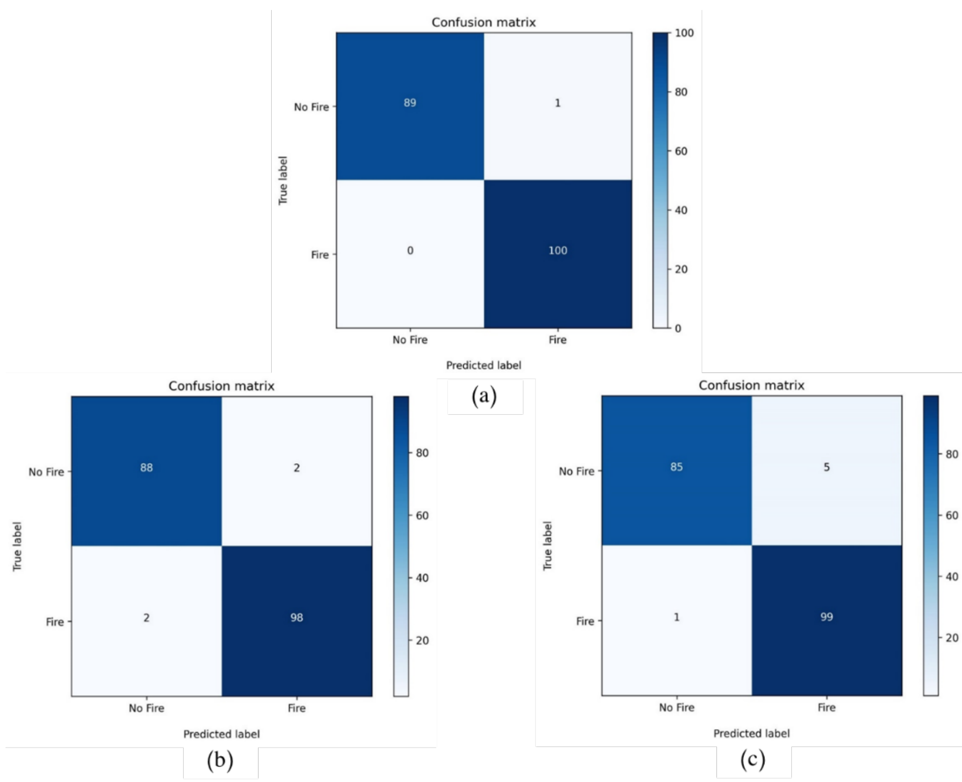

Fig 14. The confusion matrix for (a) Xception (b) Inception-v3 (c) MobileNet-v2.

Table 8. Comparative analysis of proposed approach with other CNN models.

Neural Network	Loss	Accuracy	Precision	Recall	F1-Score	Time
Xception	0.009105	0.9947	0.9901	1.0000	0.9950	1252 sec.
Inception-v3	0.06887	0.9789	0.9800	0.9800	0.9800	797 sec.
MobileNet-v2	0.2877	0.9684	0.9519	0.9900	0.9706	492 sec.

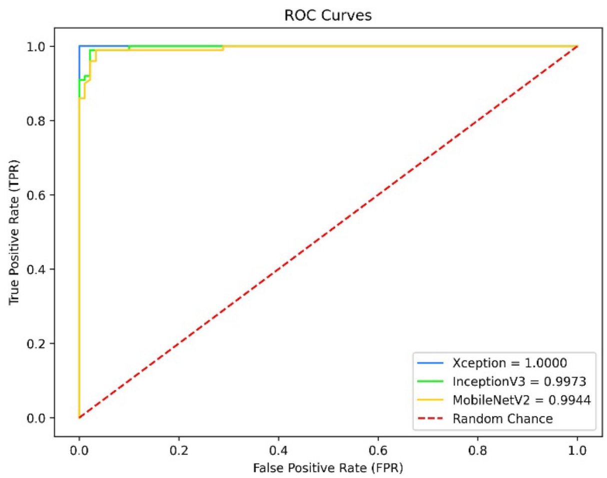

Fig 15. The ROC curves for networks Xception, Inception-v3 and MobileNet-v2.

Overall, based on the results and the presented charts, it is evident that while the Inception-v3 neural network benefits from the Inception architecture and the MobileNet-v2 neural network utilizes deep separable convolutions, both of which are advanced and efficient structures, the Xception neural network outperforms them in this specific context and dataset. This superior performance can be attributed to Xception's architecture, which extends the Inception design by more effectively separating channel-wise and spatial correlations in convolutional layers. As a result, Xception can capture more detailed features, leading to higher accuracy and better overall performance in the task of forest fire classification.

4.5 Performance comparison with other research

To further evaluate the efficiency of the proposed method, its performance was compared against other studies, as detailed in Table 9. These comparative studies were conducted on various datasets, including FLAME, Corsican Fire, and others, with some studies using private or custom datasets and each with different image quantities and characteristics. According to the table, it is observed that some of these studies have used multiple different neural networks in their proposed methods. In this case, the accuracy provided for these studies corresponds to the best result obtained by them.

Table 9. Comparative analysis of proposed approach with other classification research works.

Reference	Dataset/ Number of Images	Neural Network	Accuracy
[14]	Corsican fire/500	Inception-v3	98.60%
[15]	FLAME/47992	VGG16 VGG19 ResNet50 Inception Xception	88.00%
[16]	FLAME/47992	Xception	76.23%
[17]	DeepFire/1900	FFireNet (Based on MobileNet-v2)	98.42%
[18]	FLAME/48000	Combination of EfficientNet-b5 & DenseNet201	85.12%
[34]	DeepFire/1900	VGG19	95.00%
[19]	DeepFire & Fire/1650	Ensemble-ResNet-v1 Ensemble-ResNet-v2	99.15%
[20]	Private/3845	DCN_Fire	98.30%
[21]	DeepFire/1900	CNN	97.63%
[43]	FLAME/31501	FT-ResNet50	79.48%
[22]	FLAME/8000	DSA-ResNet	93.65%
[44]	Private/121464	ICNN (EdgeFireSmoke)	98.97%
[23]	FLAME/1452	Inception-v3 DenseNet121 ResNet50-v2 VGG19 NASNet-Mobile	99.32%
[24]	Private/2700	MobileNet-v3	90.73%
[25]	FLAME/39375	XceptionNet MobileNet-v2 ResNet50	99.30%

[26]	Corsican & FLAME & Firefront_Gestosa/7912	SVM	96.21%
[27]	SWIFT/69000	BoucaNet	93.67%
[28]	private/7977	3ENB2	99.04%
[31]	private/Not mentioned	CNN-RCNN YOLOv8	96.00%
[32]	private/Not mentioned	FireNet-CNN	99.05%
[33]	private/Not mentioned	EfficientNet	95.80%
Proposed method	DeepFire/1900	Xception	99.47%

Despite the diversity in datasets and the approaches used, the results clearly indicate that our proposed method stands out. Specifically, our approach achieved an impressive accuracy of 99.47%, surpassing the performance of other studies in this domain. This high accuracy underscores the effectiveness of our model in accurately classifying forest fire images, making it a more reliable solution compared to the methods used in previous research.

5 Conclusion

As mentioned at the outset, detecting and classifying forest fires is both critical and challenging, as the system must balance high accuracy with fast processing speed to be effective in real-time scenarios. Achieving this balance is essential for early detection, which can significantly reduce the damage caused by forest fires. In response, this research introduced a new method that not only meets but surpasses current standards, demonstrating exceptional performance compared to existing approaches in the field. The study leveraged advanced deep learning techniques, specifically utilizing transfer learning on the Xception neural network, which is known for its efficiency in feature extraction and classification tasks. By using a pre-trained model, the research was able to significantly reduce the training time while maintaining high accuracy. Additionally, a customized classifier was developed to better address the specific nuances of forest fire images, and hyperparameters were meticulously fine-tuned to further enhance the model's effectiveness. This research employed the DeepFire dataset, a specialized collection of images that includes both forest fires and non-fire scenarios, providing a robust foundation for training and evaluating the model. The evaluation metrics for the proposed method were highly impressive: accuracy of 99.47%, precision of 99.01%, recall of 100%, and an F1 score of 99.50%. These metrics highlight the model's ability to correctly identify fire images while minimizing false positives and false negatives. Moreover, the total processing time, including both training and testing phases, was recorded at 1,252 seconds, demonstrating its efficiency in handling the dataset within a reasonable timeframe. When comparing these results with those from other methods, it becomes evident that this approach not only meets but exceeds the performance of alternative solutions, showcasing superior accuracy and efficiency in forest fire classification.

Limitations

This study does not assess real-time deployment or integration with live surveillance systems, which may involve processing delays and computational constraints. The study does not include a detailed report of all hyperparameter tuning experiments, such as variations in network depth and activation functions.

Future Work

To enhance robustness and generalization, future work will involve expanding the dataset with diverse and challenging environmental conditions, using both synthetic and real-world samples. Future efforts will focus on evaluating the model’s real-time performance, including latency and resource demands, on hardware used in surveillance systems.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Tara Sistani: Conceived the research idea, conducted experiments, analyzed the data, and wrote the manuscript.

Seyed-Javad Kazemitabar: Supervised the research, provided guidance on methodology, reviewed the manuscript, and contributed to the interpretation of the results.

References

[1] A. Bouguettaya, H. Zarzour, A. M. Taberkit, and A. Kechida, “A review on early wildfire detection from unmanned aerial vehicles using deep learning-based computer vision algorithms,” *Signal Processing*, vol. 190, 2021.

[2] A. Saleh, M. A. Zulkifley, H. H. Harun, F. Gaudreault, I. Davison, and M. Spraggon, “Forest fire surveillance systems: a review of deep learning methods,” *Heliyon*, vol. 10, no. 1, 2023.

[3] A. Mohapatra and T. Trinh, “Early wildfire detection technologies in practice: a review,” *Sustainability*, vol. 14, no. 19, 2022.

[4] R. Alkhatib, W. Sahwan, A. Alkhatieb, and B. Schütt, “A brief review of machine learning algorithms in forest fires science,” *Applied*

Sciences, vol. 13, no. 14, 2023.
[5] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaria, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, 2021.
[6] M. Yandouzi, M. Grari, I. Idrissi, O. Moussaoui, M. Azizi, K. Ghoumid, and A. K. Elmiad, "Review on forest fires detection and prediction using deep learning and drones," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 12, pp. 4565–4576, 2022.
[7] R. Ghali and M. A. Akhloufi, "Deep learning approaches for wildland fires remote sensing: classification, detection, and segmentation," *Remote Sensing*, vol. 15, no. 7, 2023.
[8] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, pp. 611–629, 2018.
[9] S. Dong, P. Wang, and K. Abbas, "A survey on deep learning and its applications," *Computer Science Review*, vol. 40, 2021.
[10] S. Badillo, B. Banfai, F. Birzele, I. I. Davydov, L. Hutchinson, T. Kam-Thong, J. Siebourg-Polster, B. Steiert, and J.D. Zhang, "An introduction to Machine Learning," *Clinical Pharmacology & Therapeutics*, vol. 107, no. 4, pp. 871–885, 2020.
[11] I. H. Sarker, "Machine learning: algorithms, real-world applications and research directions," *SN Computer Science*, vol. 2, 2021.
[12] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the Recent Architectures of Deep Convolutional Neural Networks," *Artificial Intelligence Review*, vol. 53, pp. 5455–5516, 2020.
[13] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, 1st ed. Cambridge, MA, USA: MIT press, 2016.
[14] M. J. Sousa, A. Moutinho, and M. Almeida, "Wildfire detection using transfer learning on augmented datasets," *Expert Systems with Applications*, vol. 142, 2019.
[15] S. Treneska and B. R. Stojkoska, "Wildfire detection from UAV collected images using transfer learning," in *Proc. 18th Int. Conf. Informatics Inf. Technol.,* Skopje, North Macedonia, May 2021.
[16] A. Shamsoshoara, F. Afghah, A. Razi, L. Zheng, P. Z. Fulé, and E. Blasch, "Aerial imagery pile burn detection using deep learning: the FLAME dataset," *Computer Networks*, vol. 193, 2021.
[17] S. Khan and A. Khan, "FFireNet: deep learning based forest fire classification and detection in smart cities," *Symmetry*, vol. 14, no. 10, 2022.
[18] R. Ghali, M. A. Akhloufi, and W. S. Mseddi, "Deep learning and transformer approaches for UAV-based wildfire detection and segmentation," *Sensors*, vol. 22, no. 5, 2022.
[19] S. Dogan, P. D. Barua, H. Kutlu, M. Baygin, H. Fujita, T. Tuncer, and U. R. Acharya, "Automated accurate fire detection system using ensemble pretrained residual network," *Expert Systems with Applications*, vol. 203, 2022.
[20] S. Zheng, P. Gao, W. Wang, and X. Zou, "A highly accurate forest fire prediction model based on an improved dynamic convolutional neural network," *Applied Sciences*, vol. 12, no. 13, 2022.
[21] A. K. Z. R. Rahman, S. M. N. Sakif, N. Sikder, M. Masud, H. Aljuaid, and A. K. Bairagi, "Unmanned aerial vehicle assisted forest fire detection using deep convolutional neural network," *Intelligent Automation & Soft Computing*, vol. 35, no. 3, pp. 3259–3277, 2022.
[22] Z. Guan, X. Miao, Y. Mu, Q. Sun, Q. Ye, and D. Gao, "Forest fire segmentation from aerial imagery data using an improved instance segmentation model," *Remote Sensing*, vol. 14, no. 13, 2022.
[23] H. C. Reis and V. Turk, "Detection of forest fire using deep convolutional neural networks with transfer learning approach," *Applied Soft Computing*, vol. 143, 2023.
[24] I. El-Madafri, M. Peña, and N. Olmedo-Torre, "The wildfire dataset: enhancing deep learning-based forest fire detection with a diverse evolving open-source dataset focused on data representativeness and a novel multi-task learning approach," *Forests*, vol. 14, no. 9, 2023.
[25] C. Bahhar, A. Ksibi, M. Ayadi, M. M. Jamjoom, Z. Ullah, B. O. Soufiene, and H. Sakli, "Wildfire and smoke detection using staged YOLO model and ensemble CNN," *Electronics*, vol. 12, no. 1, 2023.
[26] H. Harkat, J. M. P. Nascimento, A. Bernardino, and H. F. T. Ahmed, "Fire images classification based on a handcraft approach," *Expert Systems with Applications*, vol. 212, no. 118594, 2023.
[27] L. Fernando, R. Ghali, and M. A. Akhloufi, "SWIFT: Simulated Wildfire Images for Fast Training Dataset," *Remote Sensing*, vol. 16, no. 9, 2024.
[28] E. Zia, H. Vahdat-Nejad, M. A. Zeraatkar, J. H. Joloudari, and S. A. Hoseini, "3ENB2: end-to-end EfficientNetB2 model with online data augmentation for fire detection," *Signal, Image and Video Processing*, vol. 18, no. 10, pp. 7183–7197, 2024.
[29] A. Jadouli and C. El Amrani, "Advanced wildfire prediction in Morocco: Developing a deep learning

dataset from multisource observations," *IEEE Access*, vol. 12, pp. 1–12, 2024.
[30] D. Chen, S. Cheng, J. Hu, M. Kasoar, and R. Arcucci, "Explainable global wildfire prediction models using graph neural networks," *arXiv preprint arXiv:2402.07152*, 2024.
[31] I. SHAMTA and B. E. Demir, "Development of a deep learning-based surveillance system for forest fire detection and monitoring using UAV," *PLOS ONE*, vol. 19, no. 3, 2024.
[32] G. M. I. Alam, N. Tasnia, T. Biswas, M. J. Hossen, S. A. Tanim, and M. S. U. Miah, "Real-time detection of forest fires using FireNet-CNN and explainable AI techniques," *IEEE Access*, vol. PP, 2025.
[33] M. Altaf, M. Yasir, N. Dilshad, and W. Kim, "An optimized deep-learning-based network with an attention module for efficient fire detection," *Fire*, vol. 8, no. 1, 2025.
[34] A. Khan, B. Hassan, S. Khan, R. Ahmed, and A. Abuassb, "DeepFire: A novel dataset and deep transfer learning benchmark for forest fire detection," *Mobile Information Systems*, vol. 2022, pp. 0–14, 2022.
[35] I. C. Udousoro, "Machine learning: a review," *Semiconductor Science and Information Devices*, vol. 2, no. 2, pp. 5–14, 2020.
[36] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 1251–1258.
[37] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Jornal of Big Data*, vol. 3, 2016.
[38] M. Iman, H. R. Arabnia, and K. Rasheed, "A Review of deep transfer learning and recent advancements," *Technologies*, vol. 11, no. 2, pp. 1–5, 2023.
[39] A. Hosna, E. Merry, J. Gyalmo, Z. Alom, and M. A. Azim, "Transfer learning: a friendly introduction," *Jornal of Big Data*, vol. 9, 2022.
[40] V. N. Vapnik, "Support vector machines: Theory and applications," in *Machine Learning and Its Applications, Advanced Lectures* (Lecture Notes in Computer Science, vol. 2049, pp. 249–257), Springer, 2001.
[41] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE conf. Comput. Vision Pattern Recognit. (CVPR)*, Las Vegas, USA, 2016, pp. 2818–2826.
[42] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: inverted residuals and linear bottlenecks," in *Proc. IEEE conf. Comput. Vision Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, 2018, pp.4510–4520.
[43] L. Zhang, M. Wang, Y. Fu, and Y. Ding, "A forest fire recognition method using UAV images based on transfer learning," *Forests*, vol. 13, no. 975, 2022.
[44] J. S. Almeida, C. Huang, F. G. Nogueira, S. Bhatia, and V. H. C. de Albuquerque, "EdgeFireSmoke: a novel lightweight CNN model for real-time video fire–smoke detection," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 11, pp. 7889–7898, 2022.

Biographies

Tara Sistani received her B.S. degree in Electrical Engineering from Babol Noshirvani University of Technology in 2020, followed by her M.S. degree in Electrical Engineering-Telecommunications Systems from the same institution in 2024. Her research interests include Artificial Intelligence, Deep Learning, Computer Vision and Pattern Recognition.

Seyed-Javad Kazemitabar received the B.S. degree from Sharif University of Technology, Tehran, Iran in 2003, and M.S. and Ph.D. degree from University of California, Irvine in 2005 and 2008, respectively, all in electrical engineering. His research interests include multiple antenna communications, joint routing, and scheduling for wireless networks.