FOREST FIRE DETECTION USING CONVOLUTIONAL NEURAL NETWORKS (CNN)

Dara Swetha[1] , Dr.Nazimunisa[2]

[1]PG Scholar, Department Of Computer Science And Engineering, Sree Dattha Institute Of Engineering And Science Sheeriguda, Ibrahimpatnam Hyderabad , Telangana,India.

[2]Associate Professor, Department Of Computer Science And Engineering ,Sree Dattha Institute Of Engineering And Science Sheeriguda, Ibrahimpatnam Hyderabad , Telangana,India.

Abstract**:** Forests are an important natural resource for humanity, providing a variety of direct and indirect benefits. Global warming and the persistence of life on Earth are significantly impacted by natural calamities such as forest fires. Automatic detection of forest fires is thus an essential subject of study in order to reduce disasters. Early fire detection can also assist decision-makers prepare mitigation and fighting strategies. In this study, AI-based computer vision techniques are used to investigate the identification of fire and smoke from photographs. Convolutional Neural Networks (CNN) are a sort of Artificial Intelligence (AI) approach that has been shown to beat cutting-edge methods in image classification and other computer vision tasks; however, their training time can be prohibitively long. Furthermore, a pretrained CNN may underperform if there is insufficient data provided. To address this problem, transfer learning is used on pre-trained models. Nevertheless, the classification capabilities of the equations on the initial datasets might be compromised during the implementation of transfer learning. In order to address this issue, we employ learning without forgetting (LwF), which teaches the network a new task while preserving its prior knowledge.

Keywords**:** Convolutional Neural Networks (CNN), Artificial Intelligence (AI), Forest, Fire.

1. INTRODUCTION

Forest fires are a prevalent global phenomenon caused by climate change, leading to significant economic losses and ecological devastation [1]. Debris and other biomes, together with human carelessness, can start summertime forest fires, which can be man-made or natural. While wildfires can have positive effects on local vegetation, wildlife, and ecosystems, they can also result in significant harm to property and human life. Over the past few years, the occurrence of forest fire incidents has been steadily rising. Therefore, there is more passion for the establishment of mechanisms that automatically watch for and report forest fires as a way to keep trees from being destroyed.

Several traditional and innovative methods for detecting fire and smoke have been suggested to minimise the impact of fire disasters. Sensor-based and vision-based smoke detection systems have sparked significant interest in the research community among these methodologies. The fire detection technique is categorised into five fundamental classes based on sensor types and applications: smoke-sensitive, light-sensitive, gas-sensitive, temperature-sensitive, and composite [2]. Temperature and smoke sensors are commonly employed for this objective. The sensor-based technique is subject to notable constraints in terms of both detection range and detection speed [3]. Since fire spreads swiftly, it's critical to minimise the amount of time that passes. Subsequently, with the emergence of video surveillance equipment, researchers collected fire photos and utilised their colour attributes to detect flames. The most prevalent visual depictions of fire in films and photographs are typically characterised by flames that oscillate horizontally and exhibit shades of orange or yellow. Smoke often contains a mixture of white, grey, and black plumes, which are composed of soot or burnt particles. Detecting smoke in movies and photos presents unique challenges.

In order to be efficient, a system must possess the capability to distinguish between images that genuinely depict fire and those that merely give the illusion of flames. The utilisation of basic colour characteristics for fire detection results in elevated frequencies of false alarms [4]. To capture the qualities of a fire, such as colour, shape, flickering, frequency, and dynamic textures, image processing-based methods were created. These methods employ the RGB, YUV, YCbCr, and CIELabcolour spaces to identify fire.

In addition to colour information, motion data has been included as well. Methods covered in [5] have led to an increase in the reliance on fire detection technologies. However, the utilisation of security camera images has presented a novel challenge in the field of image processing. Video cameras generate a continuous sequence of images that necessitates storage and processing, resulting in higher costs. Consequently, several kinds of methods and systems for detecting fires have been introduced in order to maximise the system's accuracy and autonomy. As video surveillance technology evolved in recent years, so did image processing in machine vision [6], allowing for faster transmission and sensing. As a result, computer vision-based fire and smoke detection technology has evolved, allowing for a wider range of fire detection methodologies. Using video surveillance to capture and extract features from images of fire and smoke, an artificial vision-based smoke and fireplace detection system can construct a detection model that relies on these images. This model can then be used to identify fires and smokes. Therefore, when trying to determine the existence of fire and smoke in images, conventional machine learning and deep learning-based computer vision methods have been recommended.


(a) Fire (b) No Fire (c) Smoke (d) Smoke Fire

Figure 1. Fire Classification

Machine learning has been applied for a variety of applications, including forest fire prediction and detection. Offer a comprehensive survey of the application of machine learning methods in forest fire detection. Algorithms for detecting fires using machine learning depend on manually collecting visible information from photos. These features only highlight the flame's surface features, which could cause data loss during manual extraction. In contrast to machine learning methods, deep learning [7] can extract and learn complex feature representations automatically. The achievements of CNN in image classification and the rapid advancement of deep learning in computer vision have made fire detection a very promising field of study. CNN-based approaches utilise frames captured by surveillance systems as input, with the prediction outcome being transmitted to an alarm system. Several CNN variations have been used for fire detection tasks, including Inception.

Classifying images of fire and smoke has previously proven problematic due to the wide parameter space employed by off-the-shelf deep architectures such as VGG16, DenseNet, Inception, and Xception, among others. When confronted with huge parameter spaces, transfer learning may be a feasible alternative.

Then, information acquired in one area can be applied to another with less data. Deep architectures utilising pre-trained models can be constructed with as little as two or three photos [8]. According to [9], deep learning models perform better when trained on a large number of samples. Overfitting and sliding into a local optimum can happen when training samples are insufficient [10]. Transfer learning can assist us in handling such issues. Many computer vision tasks, such as object detection and face recognition, have recently experienced success with deep learning, but their utility in fire detection has been limited. The study on fire detection may be limited due to a scarcity of data available for training deep learning models. As a result, we are now driven to prioritise the acquisition of a substantial number of fire/smoke photos from various sources. Furthermore, even if a pre-trained CNN classifier is trained to categorise specific types of tasks using transfer learning, the model can perform well on tasks for which it has been prepared, but underperforms when a new, but similar task is presented. In the field of machine learning, this is referred to as "the catastrophic forgetting phenomenon." This behaviour encouraged us even more to investigate the idea of LwF for identifying smoke and forest fire photographs from a fresh dataset.

2. LITERATURE SURVEY

2.1 Computer Vision (CV)

Understanding and interpreting digital images is known as computer vision [11]. Digital picture interpretation and comprehension have several practical uses, particularly in the domains of automated inspection and machine vision. In the subject of automatic inspection, computer vision can be used to inspect product quality as it is manufactured. Machine vision uses computer vision to "see" the world and direct robots or other machines. Computer vision has a number of applications, such as medical diagnosis, video surveillance, and 3D reconstruction. In each of these applications, computer vision can be used to interpret and comprehend digital images in order to accomplish a desired outcome. Recently, the utilisation of

Computer vision has made significant advancements through the use of deep learning. Compared to typical machine learning approaches, deep learning has the benefit of learning several layers of data representations, which can better capture the complex structure of data and increase pattern recognition performance. Computer vision encompasses various areas of research, such as classification, segmentation, and object identification.

2.1.1 Categorization (CLA)

Traditionally, image categorization was performed by human experts who inspected photos and identified which category they fell into. Machines currently conduct picture categorization, and they can learn to recognise patterns in photos better than people [12]. Deep learning networks can be trained to identify and classify different objects in photos through image classification. A deep learning network may be trained to identify the characteristics of a dog, such as its fur, eyes, and ears. After undergoing training, a deep learning network can be utilised to categorise photos into several classifications. An image classification task, such as distinguishing between images of dogs and cats, can be accomplished using a deep learning network [13]. Deep learning for image classification has the benefit of acquiring the ability to identify intricate patterns that surpass human detection capabilities. Deep learning networks sometimes surpass classic picture categorization methods in performance [14].

2.1.2 Segmentation (SEG) refers to the process of dividing a larger entity into smaller, distinct parts or segments

Deep learning has been used to segment images into regions that represent different objects or classes of objects. A deep neural network for image segmentation would often contain multiple layers, each of which is in charge of incrementally refining the picture segmentation [15]. The initial layer of a deep neural network (particularly CNN) for image segmentation is often a convolutional layer that learns to detect picture features.

These qualities can range from basic features like edges or corners to more intricate ones like the forms of objects [16]. The convolutional layer generates a collection of feature maps, which are subsequently transmitted to the subsequent layer. The following layer is usually a pooling layer, which decreases the dimensionality of the input by averaging the values in a small region of the input feature maps. Subsequently, there are completely connected layers (one or more) that acquire object recognition skills. The network's ultimate layer will produce a collection of labels indicating the specific category or categories of object(s) that are present in the image. While there are many other picture segmentation techniques, the majority fall into one of three categories: instance, panoptic, or semantic.

a) Semantic segmentation: this method separates objects in an image from their background. Typically, this task is achieved by identifying and classifying every object in the image based on a predefined set of labels. A segmentation algorithm can be trained to identify several types of vehicles, including autos, trucks, and buses.

b) Instance segmentation: is a method used to accurately identify and separate individual items inside a picture. This is commonly achieved by identifying the individual pixels that form an object and subsequently grouping them again.

A segmentation algorithm may be trained to identify distinct regions of the human body, including the head, chest, and legs [17] as in Figure 2. Panoptic segmentation is a method used to create a three-dimensional depiction of an item using only one photograph. This is achieved by projecting the surface of the object into a three-dimensional grid and subsequently reconstructing the object by extrapolating the information from the grid cells. Often, this technique is employed to find items that are too small or challenging to find with other techniques.

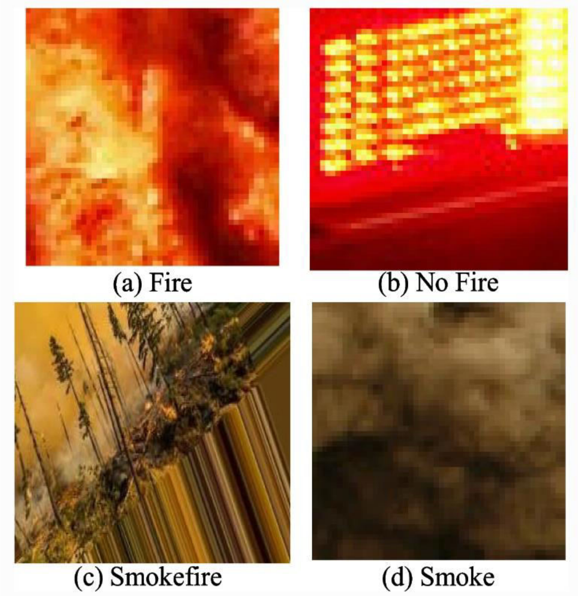


Figure 2. Forest fire datasets

2.1.3 Object Detection (OD)

Object detection in computer vision refers to the process of precisely identifying and 4ptimize4o a particular object inside an image or video sequence. Object detection can be used to find a single object or a collection of objects [18]. Many object identification methods, particularly deep learning-based object detectors, have emerged in recent years. Deep learning-based object detectors have demonstrated cutting-edge performance on a range of object detection benchmarks. Among the most widely used deep learning-based object detectors are the YOLO, SSD, and Faster R-CNN algorithms. The YOLO method is a rapid and effective object detector that identifies objects across various sizes. The SSD method is a high-speed and precise object detector that can detect things in immediate time.

3. PROPSOED METHODOLOGY

The dataset for the proposed study was created using geostationary weather satellites MODIS, VIIRS, Copernicus Sentinel-2, and Landsat-8. These satellites are utilised for fire detection all over the world because of their high temporal precision and capacity to identify flames in remote regions. A compilation of the forest fire's satellite imagery has also been made. The imagery were manually labelled by hand with symbols that mean "Fire," "No Fire," "Smoke," and "Smoke Fire." It has 4800 pictures in the dataset that was collected. To increase the number of images, image augmentation techniques such shifting, flipping, rotating, scaling, blurring, padding, cropping, translation, and affine modification were used. After augmentation, 6,911 photos make up the collection. After that, the datasets were split into three categories: testing, validation, and training. Ten half of the dataset was used for testing, and the rest, or eighty percent, was used to train the classifier. The image distribution within the training, testing, and validation datasets in Figure 3.

	File	Label
0	../input/fire-dataset/fire_dataset/non_fire_im...	nofire
1	../input/fire-dataset/fire_dataset/fire_images...	fire
2	../input/fire-dataset/fire_dataset/fire_images...	fire
3	../input/fire-dataset/fire_dataset/non_fire_im...	nofire
4	../input/fire-dataset/fire_dataset/fire_images...	fire

Figure 3. Fire Datasets

Furthermore, we evaluated the efficacy of the proposed models in applying the insights acquired from the classification of forest fire and smoke images to the BoWFire dataset, which was contributed to the compilation. The BoWFire dataset, available at http://bitbucket.org/gbdi/ bowfire-dataset/downloads/, comprises 240 photographs that are categorized into four groups: fire images, no-fire images, smoky fire images, and light photographs. Although small in size, this dataset poses substantial difficulties because of the inclusion of fire-like sunset and sunrise scenarios, fire-colored objects, and architectural lighting.

CNN:

Complex vision problems have been efficiently solved making use of various types of CNN basic designs. Convolution and pooling are the two primary operations in a Convolutional Neural Network (CNN). The capacity to extract features from images via convolution with various filters allows for the retention of the corresponding spatial information. CNNs are employed as feature extractors and classifiers in image processing applications, notwithstanding their usefulness in image processing and classification. Instead of exclusively using stacked convolutional layers like LeNet, AlexNet, and VGG, modern network architectures such as ResNet, Inception, and Xception are investigating novel methods to construct convolutional layers in an effort to improve learning efficiency. VGG is a commonly utilised convolutional neural network (CNN) architecture due to its simplicity. This project involves training the VGG16, InceptionV3, and Xception models to accurately classify photos of fire.

The VGG16 (Visual Geometry Group) CNN architecture is widely employed and is specifically implemented in ImageNet, a massive visual database initiative. VGG16 is commonly used in several types of deep learning image categorization algorithms due to its ease of implementation. Despite being introduced in 2014, it still stands as one of the most exceptional vision architectures up to now. VGG employs 1 × 1 convolutional layers to modify the decision function without changing the receptive fields, resulting in a reduction in linearity. Due to the small size of the convolution filters, VGG can have a lot of the load layers. Having more layers makes the algorithm work better, of course.

InceptionV3 is a convolutional neural network (CNN) architecture that is a variant of the Inception family. It incorporates various modifications, like smoothed-label and batch normalisation, to enhance its performance. InceptionV3 primarily emphasisesoptimising computational resources by modifying the previous Inception architectures to enhance their efficiency. It has been discovered that Inception networks are more computationally efficient than VGGNet. As a result of this efficiency, Inception networks generate fewer parameters and use less resources than previous generations. We employed dimension reduction, factorised convolutions, regularisation, and parallel calculations to enhance the efficiency of InceptionV3 for the project.

Excessive Inception is a variant of the Inception module. InceptionV3 is surpassed by Xception on the ImageNet dataset and by a wide margin on a larger dataset with 17,000 classes. Depthwise Separable convolutions necessitate less computing than separable convolutions. Therefore, Xception necessitates a smaller parameter count in comparison to other convolutional neural network variations. However, it is worth noting that depth-wise 2D convolutions can be slower than regular 2D convolutions in terms of computational speed, despite their advantage of requiring less memory. Crucially, it possesses an equivalent amount of model parameters as Inception, leading to enhanced computational efficiency. Xception and Inception differ in yet another manner. Following the initial procedure, the presence or lack of non-linearity is determined. The Inception model incorporates non-linearity by applying filtering and compression techniques to the input space, but the Xception model does not.

The VGG16 model, originally designed as a deep convolutional neural network (CNN), outperforms ImageNet in several tasks and datasets. The purpose is to decrease the degree of parameters in convolution layers and expedite the training duration. VGG16 is widely regarded as one of the most popular models for image recognition. InceptionV3 significantly reduces processing expenses while maintaining high speed and accuracy. InceptionV3 employs directed acyclic graphs for powerful processing.

On the ImageNet dataset and in most conventional classification challenges, the Xception architecture fared better than VGG16, ResNet, and InceptionV3. Conventional network architectures, such VGG16, consist solely of stacked convolutional layers. However, more recent network architectures, such as InceptionV3 and Xception, aim to develop new and inventive methods for constructing convolutional layers as a way to enhance learning efficiency. Thus, our work has employed the VGG16, InceptionV3, and Xception CNN architectures.

1. Variables
   Shared parameters $\rightarrow P_S$ (Network parameters updated for original forest fire dataset)
   Task-specific parameters for original forest fire dataset $\rightarrow P_O$
   Task-specific parameters for Bow Fire dataset $\rightarrow P_n$
   (X_n, Y_n) ◊ Training data and class label for the Bow Fire dataset
2. Procedure
   $Y_o =$ Pre-trained CNN $(X_n, P_S, P_O) \rightarrow$ find Y_o for each image in the Bow Fire dataset.
   Add nodes in the output layer for each class in the Bow Fire dataset.
   Initialize P_n with random weights.
   Train the network with Bow Fire dataset images.
   Compute $Yo\wedge$ =Pre-trained CNN $(X_n, Ps\wedge, Po\wedge)$
   Compute $Yn\wedge$ =Pre-trained CNN $(X_n, Ps\wedge, Pn\wedge)$

Compute loss functions for images in the original and Bow Fire dataset and update P_S, P_O, and P_n.

Repeat from step 4 till convergence

Transfer learning

The process of applying data from a source domain (such as ImageNet) to a target domain with a significantly smaller sample size is known as transfer learning. Usually, this entails starting a model with weights that have already been trained from VGGNet, Inception, or another source, and then utilising it to extract features or fine-tuning the last few layers on a fresh dataset. Transfer learning enables us to repurpose these models for other applications, such object recognition for self-driving cars or caption creation for videos. This study involves the extraction of features and the fine-tuning of a pre-trained model. A concise comment on customisation is provided thereafter.

Extractor of attributes:

This method extracts important features from new samples by using previously learned representations. We have developed a novel classifier that utilises the feature mappings obtained from the previous dataset (ImageNet) and applies them to the pre-trained model. Retraining the entire model using this method is nonessential. The fundamental convolutional network possesses inherent characteristics that can be employed to identify images in a broad sense. The pre-trained model's final classification layer is unique to ImageNet, but our layers are proprietary to the classes the model is customised for.

Appraiser:

Using this strategy, we train the newly added classifications and the unfrozen levels of the models after unfreezing a couple of their top layers. The model's feature representations can be modified, if needed, to enhance their relevance to the unique dataset under analysis. Furthermore, the weights of multiple upper levels of the convolution base will undergo retraining during the fine-tuning phase, in addition to the classification layers. This will be accomplished as a method to attain maximum efficiency. Because the early convolution layers have relatively generic traits, as we go through the network, the layers learn increasingly task-specific characteristics. Therefore, early layers are retained frozen while upper layers are retrained during fine-tuning. By using fine-tuning, we may leverage pre-trained networks to classify distinct categories in datasets that have not been previously trained on. Retraining the weights of the top layers on a new dataset results in fine-tuning, which produces higher accuracy than feature extraction-based transfer learning.

4. RESULT ANALYSIS

To fully utilise deep learning models, it is essential to select the right hyperparameters. A more objective approach would be to search for different hyperparameter values and select the subset that performs the best on a given dataset. This process is commonly known as hyperparameter optimisation or tuning. Any optimisation process begins with the definition of the search space. When looking for something, Bayesian optimisation, random search, and query grid are the easiest and most common ways to do it. In this study, Bayesian optimisation is employed to select optimal values for hyperparameters. The models are executed several times using various sets of hyperparameter values. However, earlier model knowledge is taken into account when determining the subsequent model. The most accurate models can supposedly be reached faster using the Bayesian Optimisation approach. As a result, we employed this search strategy to identify the hyperparameters' ideal values. According to the literature survey, the learning rate, optimizer, activation function, batch size, number of epochs, and number of neurons have all been tweaked in various research endeavours. Therefore, in the suggested study, the aforementioned hyperparameters have been fine-tuned using Bayesian Optimisation.

Table 1: Search space.

Parameter	Search space
Optimizer	Adam, RMSProp, SGD, Adagrad, Adadelta
Number of neurons in customized layers	64, 128, 256, 512, 1024
Activation function	Relu, Elu, LeakyRelu and Tanh
Learning rate	1e−3, 1e−4, 1e−5, 1e−6
Number of epochs	100, 125, 150, 200
Batch size	32, 64, 128

Table 2: Parameters and their values.

Hyperparameters	VGG16	InceptionV3	Xception
Optimizer	Adam	Adam	Adam
Learning rate	1e−01	1e−05	1e−03
Activation function	Elu	Relu	Relu
Number of neurons in customized layers	512	256	256
Number of epochs	100	70	75
Batch size	128	64	64

We conducted tests to assess the efficacy of pre-trained models using feature extraction, fine-tuning, and learning without forgetting. Our more thoroughly models were trained with GPU-enabled Kaggle kernels. Tensorflow and Keras frameworks are utilised to train the models. The models have undergone training using the hyperparameters given in Table 2. Table 3 displays the optimised values of hyperparameters that produced the most favourable outcomes during the training process. The models were executed for 100 epochs, however, we terminated them prematurely. Early stopping is a technique in which the model is trained for an arbitrary number of epochs and then terminated when there is no progress in validation accuracy or reduction in validation loss. As previously stated, we conducted two distinct sets of tests. We removed the classifier from these models and replaced it with a customised classifier to allow us to conduct these tests. We incorporated two fully linked layers and a softmax layer into the VGG16 model. An additional completely connected and softmax layer was successfully used in both InceptionV3 and Xception. We have retrained the fifth, eighth, and seventh top layers of VGG16, InceptionV3, and Xception, respectively, while fine-tuning the models.

Table 3: Accuracy and validation of models.

Models		Validation accuracy (%)	Testing accuracy (%)
VGG16	Feature extractor	95.18	94.38
	Fine tuner	96.32	95.46
InceptionV3	Feature extractor	93.93	92.04
	Fine tuner	97.87	97.01
Xception	Feature extractor	98.27	97.77
	Fine tuner	99.12	98.72

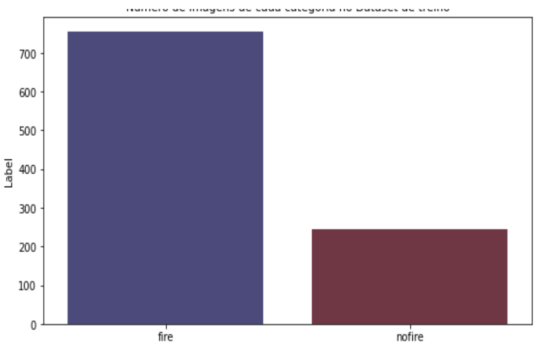
```
import pandas as pd
import numpy as np
import datetime as dt
import os
import os.path
from pathlib import Path
import glob
import cv2
```

The above are the various packages being used for performing the detecting the forest fire along with CNN.

```
filepaths = pd.Series(png_filepaths, name = 'File').astype(str)
labels = pd.Series(labels, name = 'Label')

# Concatenando...
df = pd.concat([filepaths, labels], axis=1)

# Mudando os nomes...
df['Label'].replace({"non_fire_images":"nofire","fire_images":"fire"}, inplace=True)
```



The above figure makes the label of fire and nofire classification based on the various images being used in the datasets.

Training dataset:

Number of images: 899

Number of images with fire: 684

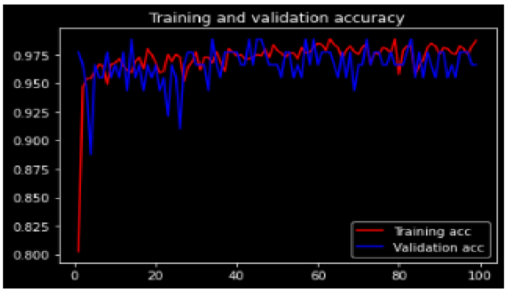
Number of images without fire: 215

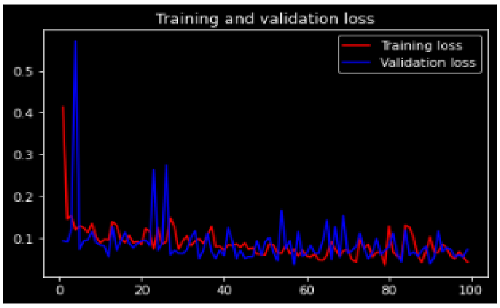
Test dataset:

Number of images in the test dataset: 100

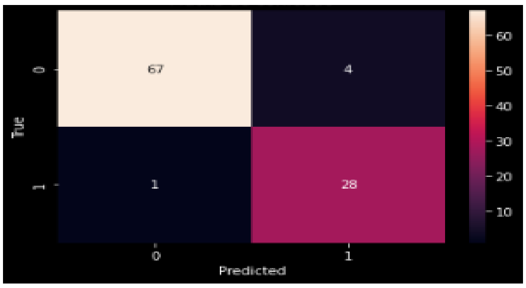
Number of images with fire: 71

Number of images without fire: 29





	precision	recall	f1-score	support
0	0.99	0.94	0.96	71
1	0.88	0.97	0.92	29
accuracy			0.95	100
macro avg	0.93	0.95	0.94	100
weighted avg	0.95	0.95	0.95	100



CONCLUSION

It is essential to accurately and quickly identify active flames in their early stages in order to lessen the devastating effects of wildfires. There is a scarcity of studies that specifically address the real-time monitoring of active flames using deep learning techniques. In this study, we looked into how pre-trained models for smoke and forest fire detection could learn from each other. We utilised the models to extract distinctive characteristics and refine them through a process of careful adjustment. The findings demonstrate that the Xception-based model had superior performance compared to the other models, with an accuracy rate of 98.72%. In order to maintain the distinctive attributes of the original dataset, we utilised the Learning without Forgetting (LwF) technique and discovered that it surpasses feature extraction in terms of performance. Interestingly, while fine-tuning the new task with LwF, it performed well in comparison to the original dataset when utilising fine-tuned parameters. Recent research shows that it is vital to identify fires fast and correctly in their early stages to prevent them from spreading. Therefore, we aim to further pursue our research in this domain and augment our discoveries. In order to quickly detect fire incidents with a low percentage of false positives, we want to use the most recent CNN models. In addition, we are interested in delving more into the concepts of LwF (Learning without Forgetting) and multitask learning.

REFERENCES:

[1]. Abid, F. 2021. A survey of machine learning algorithms based forest fires prediction and detection systems. Fire Technology 57 (2): 559–590.

[2]. Al-Duryi, M. H. A. 2022. Design and analysis of forest fire detection system using image processing technique. AltınbaşÜniversitesi/LisansüstüEğitim Enstitüsü.

[3]. Anh, N. D., P. Van Thanh, D. T. Lap, N. T. Khai, T. Van An, T. D. Tan, N. H. An, and D. N. Dinh. 2022. Efficient forest fire detection using rule-based multi-color space and correlation coefficient for application in unmanned aerial vehicles. KSII Transactions on Internet and Information Systems (TIIS) 16 (2): 381–404.

[4]. Arif, M., K. Alghamdi, S. Sahel, S. Alosaimi, M. Alsahaft, M. Alharthi, and M. Arif. 2021. Role of machine learning algorithms in forest fire management: a literature review. J Robotics Autom 5 (1): 212–226.

[5]. Bari, A., T. Saini, and A. Kumar: Fire detection using deep transfer learning on surveillance videos, in Editor (Ed.)^(Eds.): 'Book Fire detection using deep transfer learning on surveillance videos' (IEEE. 2021. edn.), pp. 1061–1067.

[6]. Best, N., J. Ott, and E. J. Linstead. 2020. Exploring the efficacy of transfer learning in mining image-based software artifacts. Journal of Big Data 7 (1): 1–10.

[7]. Bot, K., and J. G. Borges. 2022. A systematic review of applications of machine learning techniques for Wildfire Management decision support, Inventions, 7 (1): 15.

[8]. Bouguettaya, A., H. Zarzour, A. M. Taberkit, and A. Kechida. 2022. A review on early wildfire detection from unmanned aerial vehicles using deep learning-based computer vision algorithms. Signal Processing 190: 108309.

[9]. Castelli, M., L. Vanneschi, and A. Popovič. 2015. Predicting burned areas of forest fires: an artificial intelligence approach. Fire ecology 11 (1): 106–118.

[10]. Cheng, X. 2021. Research on application of the feature transfer method based on fast R-CNN in smoke image recognition, Advances in Multimedia. 2021.

[11]. Chollet, F. 2017. Xception: deep learning with depthwise separable convolutions, in Editor (Ed.)^(Eds.): 'Book Xception: Deep learning with depthwise separable convolutions'. edn.), 1251–1258.

[12]. Dampage, U., L. Bandaranayake, R. Wanasinghe, K. Kottahachchi, and B. Jayasanka. 2022. Forest fire detection system using wireless sensor networks and machine learning. Scientific reports 12 (1): 1–11.

[13]. de Venâncio, P. V. A., A. C. Lisboa, and A. V. Barbosa. 2022. An automatic fire detection system based on deep convolutional neural networks for low-power, resource-constrained devices, Neural Computing and Applications, 1–20.

[14]. Dogan, S., P. D. Barua, H. Kutlu, M. Baygin, H. Fujita, T. Tuncer, and U. R. Acharya. 2022. Automated accurate fire detection system using ensemble pretrained residual network. Expert Systems with Applications 203: 117407.

[15]. Fang, Q., Z. Peng, and P. Yan. 2022. Fire detection and localization method based on Deep Learning in Video Surveillance. In)^(eds.): 'Book Fire detection and localization method based on Deep Learning in Video Surveillance', ed. Editor (, 012024. IOP Publishing. edn).

[16]. Fouda, M. M., S. Sakib, Z. M. Fadlullah, N. Nasser, and M. Guizani. 2022. A lightweight hierarchical AI model for UAV-enabled edge computing with forest-fire detection use-case. IEEE Network.

[17]. Friggens, M. M., R. A. Loehman, C. I. Constan, and R. R. Kneifel. 2021. Predicting wildfire impacts on the prehistoric archaeological record of the Jemez Mountains, New Mexico, USA, Fire Ecology. 17 (1): 1–19.

[18]. Guede-Fernández, F., L. Martins, R. V. de Almeida, H. Gamboa, and P. Vieira. 2021. A deep learning based object identification system for forest fire detection. Fire 4 (4): 75.