e-ISSN : 0976-5166
p-ISSN : 2231-3850

INTEGRATION OF LOGICAL FEATURES WITH NEURAL NETWORKS FOR CONTROLLED VS UNCONTROLLED FIRE CLASSIFICATION: A COMPARATIVE STUDY

Omkar Bhosale

Computer Engineering Dept., CAiLMD, Pune Institute of Computer Technology,
Pune, 411043, India
omkarbhosale277@gmail.com

Aryan Dande

Computer Engineering Dept., CAiLMD, Pune Institute of Computer Technology,
Pune, 411043, India
aryan.dande@gmail.com

Sagar Abhyankar

Computer Engineering Dept., CAiLMD, Pune Institute of Computer Technology,
Pune, 411043, India
sagarabhyankar18@gmail.com

Sarang Joshi

Computer Engineering Dept., CAiLMD, Pune Institute of Computer Technology,
Pune, 411043, India
sajoshi@pict.edu

Abstract

This paper presents an analysis of the performance of a convolutional neural network (CNN) for the classification of controlled and uncontrolled fires. The study focuses on the incorporation of custom features such as standard deviation, spikes, fall, vertical intensity arrays (VIA), and arc length to improve the accuracy of the model. These features were individually concatenated with the features selected by the neural network to test the cumulative performance. The paper also puts forth the comparison between a logical (decision tree) classifier and a black box (neural net) classifier and the corresponding performance analysis.

***Keywords*: Computer Vision, Image Processing, Feature Engineering, Fire Classification, Fusion Models**

1. Introduction

Controlled and uncontrolled fires have a significant impact on our environment, economy, and society. Uncontrolled fires can cause severe damage to properties, wildlife habitats, and human lives. On the other hand, controlled fires can cause false alarms in the fire detection system. Therefore, accurately distinguishing between controlled and uncontrolled fires is crucial for effective management and mitigation strategies. In recent years, machine learning techniques, specifically deep learning, have shown great promise in automatic controlled fire and uncontrolled fire classification. Convolutional Neural Networks (CNNs), a type of black box model, have been successfully used for this task, achieving high accuracy and speed. However, these models lack interpretability, which makes it difficult to understand how they arrive at their predictions.

To address this issue, this research explores the use of a logical classifier for the classification of controlled vs uncontrolled fires using features derived from the input image. These features have been designed to mimic the way humans analyse images, providing a more intuitive approach to the problem. By combining these human-crafted features with the neural network's learned features, this research aims to create a hybrid model that can improve the accuracy of fire classification while maintaining interpretability. This study analyses the performance and interpretability of black box models for controlled vs uncontrolled fire classification and compare it with the performance of the hybrid model that fuses human-crafted features with the neural net.

Despite their interpretability advantages, logical classifiers can face certain challenges in practice. One of the primary challenges is the need for domain expertise and knowledge to create the rules and features that the model relies on. These rules may be complex and require significant effort to develop and refine, which can make the model less accessible to non-experts. Another concern with pure neural net models is that they tend to overfit the training dataset and perform poorly on new data. To overcome these challenges, careful selection and refinement of features, as well as techniques such as cross-validation, regularization, and ensemble methods, may be required. This research aims to improve the accuracy of fire classification and provide more interpretable models, ultimately contributing to the effective management and mitigation of fires.

2. Components

2.1. *Masking*

Masking is a critical process in image processing, used to isolate specific regions of interest while minimizing the effects of other sources of light and color in the image. In the context of fire, masking is essential in accurately extracting the features of controlled and uncontrolled fire from images for various applications, including fire detection, monitoring, and analysis. This paper discusses a customized masking technique that effectively captures the fire features from images, even in the presence of varying brightness levels and low intensity. Our technique uses a combination of HSB color space, percentile values, and a custom *inRange* function to achieve accurate and reliable fire feature extraction.

2.2. *BlackBox*

Black box models, such as neural networks, have become increasingly popular in various fields due to their ability to automatically learn complex patterns from data without requiring explicit feature engineering. They are known for their ability to handle high-dimensional and noisy data, making them suitable for a wide range of applications. Additionally, black box models have been proven to achieve state-of-the-art performance in a variety of applications ranging from cybersecurity to autonomous driving. However, these models lack transparency, which makes it challenging to understand how they make their predictions. Nonetheless, with proper training and tuning, black box models can provide accurate and robust predictions, making them a powerful tool in machine learning.

2.3. *Fusion Models*

Fusion models, also known as hybrid models, combine the strengths of multiple machine learning techniques to improve the accuracy and robustness of predictions. These models are especially useful when dealing with complex tasks that require different approaches to be tackled effectively. For instance, a fusion model may use a logical classifier to provide human-understandable insights while also using a black box model to leverage the power of deep learning. By combining the advantages of multiple models, fusion models can outperform individual models and provide a more comprehensive understanding of the problem at hand. However, developing a fusion model can be challenging as it requires expertise in multiple machine learning techniques and careful integration of their outputs.

3. Methodology

3.1. *Dataset*

This research used two datasets of fire images for our experiments: the Flame (Controlled Fire) dataset and the Fire dataset. The Flame dataset contains 482 labelled images of candle flames, and can be found at the following link: https://github.com/MartinRobomaze/candle-flame-dataset/tree/master/yolo-labels. The Fire dataset contains 755 labelled images of various types of fires and can be found on the following link: https://www.kaggle.com/datasets/phylake1337/fire-dataset. All the images have been resized to have a height and width of 150 pixels for training our models. The size of 150*150 was chosen due to computational limitations as well as to check the feasibility of the model with low-resolution cameras. Overall, the datasets provided a diverse range of images that allowed us to explore different types of fires and flames in our experiments.

3.2. *Masking Script*

To accurately extract the fire features from images, it is important to reduce noise and isolate the region of interest. This process involves masking the image to only capture the fire part while minimizing the effect of other sources of light and color in the image.

The procedure for the script was as follows:

1. Convert the input image to the HSB color space.
2. Compute the 75th percentile value of the B channel and assign it to a variable X.
3. Define the lower and upper bounds for each value of the HSB color space as follows:
    - Lower bound = [0, 70, X]
    - Upper bound = [35, 255, 255]
4. Create a binary mask for the input image using a custom *inRange* function that checks if the Hue, Saturation, and Brightness of each pixel falls within the established bounds. Set the mask value to 1 for pixels that meet this condition, and 0 for those that don't.
5. Apply a bitwise AND operation between the binary mask and the HSB image to obtain an output image that captures only the fire features while minimizing the effects of other sources of light and color in the image.

The custom *inRange* function was implemented to address the limitations of the standard OpenCV *inRange* function in capturing the whites inside the fire. The modified *inRange* function utilizes the lower and upper bounds for each value of the HSB channel established in the previous step, as follows:

$$\begin{aligned} &Mask[i][j] = lower\_bound\_hue < Hue < upper\_bound \, \& \\ &lower\_bound\_saturation < Saturation < upper\_bound\_saturation \, \& \\ &lower\_bound\_value < Value < upper\_bound\_value \end{aligned} \quad (1)$$

Additionally, an extra condition is added to preserve the whites inside the fire, whereby if the Value of the pixel is greater than or equal to 250 and the Hue is less than or equal to 60, the mask value is set to 1. This ensures that the binary mask captures all the necessary features of the fire, including the whites inside it, to achieve accurate feature extraction.

For the lower bound's B value, the value of the 75th percentile of all B values in the image is found. This is to ensure that features from images with differing brightness are extracted properly. It also helps capture low intensity which the model may miss.

3.3. *Logical Features Constructed*

3.3.1. Vertical intensity arrays (via)

A novel approach called the Vertical Intensity Arrays (VIA) is developed to classify images as controlled fire or uncontrolled fire. The VIA method was inspired by how humans visually categorize fires based on factors such as size, spread, and color. Our goal was to create a mathematical representation of the visual characteristics of fire.

For each image in our dataset the steps taken to calculate VIA were

1. Generated an array of shape 1 x n, where n is the horizontal length of the image where each value in the array is the sum of the number of orange and yellow pixels in the image for that position along the horizontal axis.
2. For every x-coordinate the total number of pixels with yellow or orange HSV values is summed. For example, if the shape of the image is 5 x 5, its VIA is of size 1 x 5.

Finally, to visualize the intensity profile of the fire, the VIA with x-coordinates of the image on the x-axis and pixel counts on the y-axis is plotted. It is observed that the plot has a shape similar to that of fire.

3.3.2. Standard deviation-spike-fall (ssf)

In addition to the Vertical Intensity Array (VIA), this research developed a simplified version called the SD-Spike-Fall (SSF) array to further classify images as controlled or uncontrolled fire. The SSF array captures three values:

1. The standard deviation of the VIA
2. The number of spikes (abrupt rises) in the VIA
3. The number of falls (abrupt drops) in the VIA.

3.3.3. Total arc length (tal)

Another approach to classifying controlled and uncontrolled Fire is by differentiating by the number of contours they produce. Usually, uncontrolled Fires are made up of many smaller individual fire and smoke particles which in turn increases the number of contours. The Total Arclength is computed by adding up the perimeters of all contours.

3.4. *Logical Classifier*

In this study, a threshold-based logical classifier for detecting controlled and uncontrolled fire has been developed. The classifier employs a set of rules based on statistical parameters. The primary criterion for detection is the standard deviation of the VIA, which must be less than 2500. If this condition is not met, the output of the classifier is set to "fire detected". If the standard deviation is less than 2500, the presence or absence of spikes and falls in the VIA is checked. This acts as a primary condition for detection of uncontrolled fire. If the number of spikes and falls are both equal to zero, then the image is classified as a fire. Otherwise, the difference in the number of spikes and falls is calculated, and if it exceeds twice the minimum of the two, the classifier checks the Arc length parameter generated for that image. If it is less than 642, the output is set to "Controlled Fire detected". Otherwise, the output is set to "Uncontrolled Fire detected". If the difference in the number of spikes and falls does not exceed twice the minimum of the two, the Arc length parameter is checked, and if it is greater than 1883, the output is set to "Uncontrolled Fire detected". Otherwise, the output is set to "Controlled Fire detected". The classifier has been tested on the dataset, and the results have shown a good amount of accuracy in classification.

3.5. *Black Box Model*

As the Black box model, two convolutional neural nets were trained and tested on the same dataset to find out the classification accuracy of each. The first neural net consists of two convolutional layers, each with 64 filters of size (3,3) and ReLU activation, followed by a max-pooling layer with pool size (2,2). The purpose of the convolutional layers is to extract meaningful features from the input images, while the max-pooling layers reduce the spatial dimensionality of the features and help prevent overfitting.

After the convolutional layers, a flattened layer is added to convert the 2D feature maps into a 1D feature vector. This vector is then passed through a fully connected layer with 128 units and ReLU activation, which acts as a classifier on top of the extracted features. Finally, a dense layer with 2 units and SoftMax activation is used to produce the final classification probabilities for the two classes.

During training, the model utilizes the Adam optimizer and sparse categorical cross-entropy loss function, while the accuracy metric is employed for evaluation purposes. The dataset used for training and validation is split into 80:20 ratio, with 80% used for training and 20% for validation. The model is trained for 15 epochs, during which the weights are updated to minimize the loss function and maximize the accuracy. The second neural net is similar to the first one but consists of 3 layers, containing 128 neurons each. The 64x2 model configuration performed better as compared to 128x3, hence this model has been used to concatenate logical features and to compare the results against the logical classifier.

Following is the summary model of the neural net.

Model: Sequential		
Layer (type)	**Output Shape**	**Param #**
Conv2D (Conv2D)	(None,148,148,64)	1792
MaxPooling2D (MaxPooling2D)	(None, 74, 74, 64)	0
Conv2D_1 (Conv2D)	(None, 72, 72, 64)	36,928
MaxPooling2D_1 (MaxPooling2D)	(None, 36, 36, 64)	0
Flatten (Flatten)	(None, 82944)	0
Dense (Dense)	(None, 128)	10,616,960
Dense_1 (Dense)	(None, 2)	258

Table 1. Summary of BlackBox Model

3.6. *Fusion Impure Model*

The impure model is a fusion of traditional convolutional neural networks with external features supplied by us. For each logical feature viz. SSF, VIA and Total Arc Length the 64x2 neural network is fed with an external feature set representing the logical feature and hence a merged model is formed.

First, the architecture of the image model has been defined. It has an input layer that takes images of shape (150,150,3), where 3 corresponds to the RGB channels of each pixel in the image. This input layer is followed by two convolutional layers with 64 filters of size (3,3) each, and ReLU activation function. A max-pooling layer is then applied with pool size (2,2) to down sample the feature maps. The same pattern is repeated with another set of convolutional and max-pooling layers, followed by a Flatten layer that converts the output tensor into a 1D tensor, which is then passed to the next layer.

Next, the architecture of the secondary input model has been defined. It has an input layer that takes in a 3-dimensional tensor with shape (3,), which is used to pass in some additional information related to the images. This input layer is followed by a dense layer with 128 units and ReLU activation function.

The output of the image model and the secondary input model are then concatenated together using the concatenate layer. This merged output is then passed through two dense layers, each with 128 units and a ReLU activation function. The output is passed through a dense layer with 2 units and SoftMax activation function, which gives us the probability of the image belonging to each of the two classes.

Finally, the model has been compiled using the Adam optimizer. The model utilizes a sparse categorical cross-entropy loss function for integer labels, and accuracy is employed as the evaluation metric. The model has been trained using the fit method by providing both the image data and secondary input data as inputs, along with the corresponding labels. Following is the summary table of the Fusion Model.

Model: Fusion Model			
Layer (type)	**Output Shape**	**Param #**	**Connected to**
input_1 (Input Layer)	[(None, 150, 150, 3)]	0	
conv2d (Conv2D)	(None, 148, 148, 64)	1792	input_1[0][0]
max_pooling2d (MaxPooling2D)	(None, 74, 74, 64)	0	conv2d [0][0]
conv2d_1 (Conv2D)	(None, 72, 72, 64)	36928	max_pooling2d [0][0]
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 64)	0	conv2d_1[0][0]
flatten (Flatten)	(None, 82944)	0	max_pooling2d_1[0][0]
input_2 (InputLayer)	[(None, 3)]	0	
dense (Dense)	(None, 128)	10616960	flatten [0][0]
dense_1 (Dense)	(None, 128)	512	input_2[0][0]
concatenate (Concatenate)	(None, 256)	0	dense [0][0] dense_1[0][0]
dense_2 (Dense)	(None, 128)	32896	concatenate [0][0]
dense_3 (Dense)	(None, 2)	258	dense_2[0][0]

Table 2. Summary of Fusion Model

4. Results:

4.1. Models

The performance of all the models was compared based on F1 score as there was an imbalance between the dataset size of Uncontrolled Fire and Controlled Fire. The following table summarizes the F1 score.

Model Name	**F1 Score (performance)**
64*2 Neural Net (unmasked)	0.91
64*2 Neural Net (masked)	0.97
128*3 Neural Net (unmasked)	0.91
Fusion Model (SSF,64*2)	0.76
Fusion Model (VIA,64*2)	0.51
Fusion Model (TAL,64*2)	0.95
Pure Logical Classifier	0.79
Stand-alone Logical Feature (TAL)	0.68
Stand-alone Logical Feature (SSF)	0.75

e-ISSN : 0976-5166
p-ISSN : 2231-3850

Table 3. f1 scores of all Models

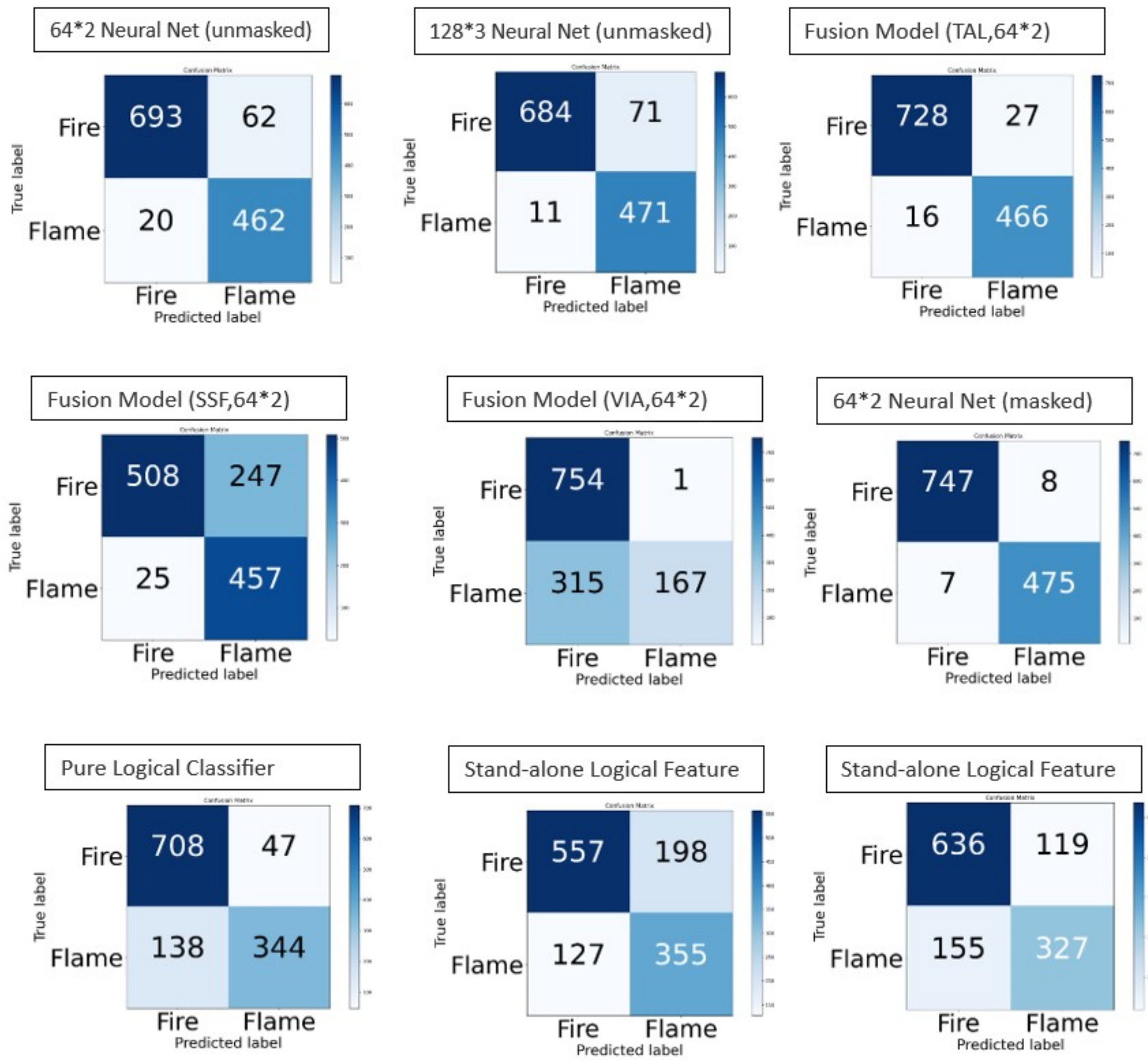


Fig (1)

As seen in the performance table, the neural net trained on masked images performed with highest f1 score (0.97) followed by the Fusion Model with Total Arc Length as the concatenated logical feature at 0.95. This shows that by masking, that is, on removing the noisy background, the neural net adapts to extract more meaningful features to classifying an image as controlled or uncontrolled fire. Being a black box, it is hard to point out which exact features are at play to achieve that performance. By analyzing a small set of test images containing standard, noisy and outlier images this research speculates that the features extracted by the neural net were related to containment of the fire base and the uniformity of flame/intensity spread throughout the image.

Overall, the Neural nets performed much better than stand-alone logical features, but when combined to form Fusion Models, their performance shows improvement. The key finding here was that even when trained on unmasked images the Fusion Model (TAL) was at par with the neural net trained on masked images. This suggests that the provided Total Arc length feature was an excellent logical baseline for the model when it encountered outliers while classifying.

The Fusion Models constructed with other logical features like SSF and VIA did not fuse well with the neural net, and hence can be seen as nothing more than noisy classifiers.

4.2. Features

The results and observations for individual logical features are as follows:

Figure (2.1)

Figure (2.2)

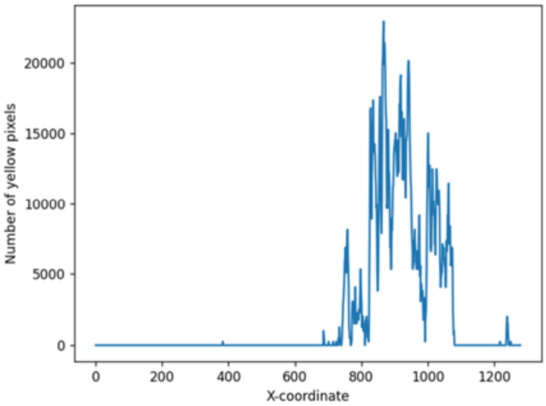


Figure (2.3)

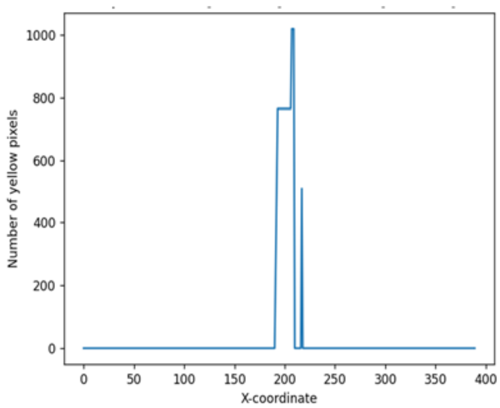


Figure (2.4)

4.4.1. Vertical intensity array

By plotting the VIA with x-coordinates of the image on the x-axis and pixel counts on the y-axis, observation has been made that the plot had a shape similar to that of fire. A comparison of VIAs has been conducted on multiple images of uncontrolled and controlled fires. Additionally, the standard deviation of the pixel counts has been calculated. The comparative study of these images revealed that images with perceptible uncontrolled fire had a greater deviation in pixel values compared to those with controlled fire. Additionally, it is observed that images with controlled fire had a steep increase and decrease in pixel counts instead of a tapered one.

4.4.2. Standard deviation spike fall

Through observation, it is found that images containing Uncontrolled fire typically had a higher standard deviation and either no spikes or falls, or a vague number of spikes and falls. In contrast, images with controlled fires tended to have a lower standard deviation and an equal number of spikes and falls. The equal number of spikes and falls was due to the symmetric shape of the controlled fires. As seen in Figure 2.2 The spike, fall count, and standard deviation were 2,2 and 170.3 respectively whereas Figure 2.1 had its spike, fall count, and standard deviation 0,0 and 4600.4 respectively.

4.4.3. Total arc length

Through our observation, it is found that images containing uncontrolled Fire mostly had a higher Total Arc Length than Controlled Fire. Total Arc Length is highly dependent on image resolution. Higher image resolutions increase Total Arc Length classifying power. Figure 2.1, and Figure 2.2 have TAL values 3987.62, 680.12 respectively.

The accuracy of the individual logical features depends heavily on selecting the right threshold value, which is a right fit for a large dataset, proper threshold value can be found out either by estimation or brute force techniques which imitates the working of a neural net.

5. Conclusion

This paper has presented the comparative performance analysis of different neural networks, fusion models, and logical models for classifying images of controlled vs uncontrolled fires.
Special Emphasis was given on finding the feasibility of combining logical features with neural nets to form Fusion Models and then analysing their performance for our use case. This research concludes that training the neural net on masked images significantly improves its performance, although the lack of interpretability in black box models is still a problem. This can be countered using fusion models where the provided logical feature acts as an ancillary for both interpretability, and reliability. It was also found that neural nets tend to overfit the dataset more, as compared to fusion models.

In conclusion, our study showed that combining logical features with a neural network improves the performance and robustness of controlled and uncontrolled fire classification models. This approach also provides a safety net in case the neural network struggles with outlier inputs. Our evaluation of the black box model used in the study helped it demonstrate its relatively strong performance and robustness against outliers, indicating its potential for real-world applications.

6. Limitations and Future Scope

The addition of external logical features to a Fusion model for Controlled vs Uncontrolled Fire classification may not always increase its accuracy. Choosing the correct features to extract from the Controlled or Uncontrolled Fire and ensuring that their addition doesn't confuse the black box, remains a challenge and a time taking process. Therefore, there is a need to carefully evaluate the impact of adding external features to the Fusion model and select only those that significantly enhance its performance.
Another concern is the lack of a proper data set that covers fires in all possible contexts (controlled and uncontrolled) and situations.

To address the limitations of the current approach, future research can focus on developing more sophisticated feature selection algorithms that can identify the most informative features for fire classification. Additionally, the study of feature interaction and feature fusion can be explored to determine the optimal combination of features that work together with the black box for improved accuracy.

Funding
No funding is provided for the preparation of manuscript.

Conflicts of Interest
The authors have no conflicts of interest to declare.

7. References:

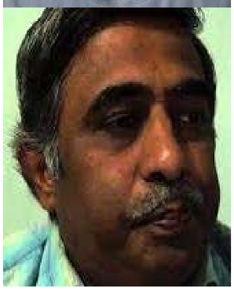
[1] Adnan Khalil; et al. Fire Detection Using Multi Color Space and Background Modelling 2020
[2] Di Wu; et al. Forest Fire Recognition Based on Feature Extraction from Multi-View Images 2021
[3] Huibai Wang; et al. Facial Expression Recognition based on The Fusion of CNN and SIFT Features 2020
[4] Jeremy Patch; et al. Opening the Black Box: The Promise and Limitations of Explainable Machine Learning in Cardiology 2022
[5] Krizhevsky; et al. ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 2012
[6] Laith Alzubaidi; et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions 2021
[7] Li Huang; et al. Multi-Scale Feature Fusion Convolutional Neural Network for Indoor Small Target Detection 2022
[8] Xudie Ren; et al. A Novel Image Classification Method with CNN-XGBoost Model 2021
[9] Yongmei Ren; et al. Multi-Feature Fusion with Convolutional Neural Network for Ship Classification in Optical Images 2021

8. Authors Profile

Omkar Bhosale
Currently pursuing BE in Computer Engineering [20-24] from Pune Institute of Computer Technology.
He is part of the research group CAiLMD.
His Research Interests are Computer Vision, Deep Learning, Causal AI.

Aryan Dande
Currently pursuing BE in Computer Engineering [20-24] from Pune Institute of Computer Technology.
He is part of the research group CAiLMD.
His Research Interests are Computer Vision, Deep Learning, Machine Learning.

Sagar Abhyankar
Currently pursuing BE in Computer Engineering [20-24] from Pune Institute of Computer Technology.
He is part of the research group Centre For Analytics With Intelligent Learning For Multimedia Data (CAiLMD). CAILMD research group is actively involved in problem solving for the emerging needs of industry and academia in the field of Data Analytics, Intelligent leaning using AL/ML/DL, GPU computing and Natural language Processing
His Research Interests are Computer Vision, Machine Learning, Cyber Security, Distributed Systems.



Dr. Sarang Achyut Joshi
He is a Professor at PICT, Savitribai Phule Pune University, Pune,Maharashtra, India. He completed his Ph.D. in Computer Science and Engineering from Bharati Vidyapeeth, Pune, India. He completed Masters in Computer Engineering and Bachelors in Computer Engineering from the University of Pune, India. He works as a Professor in Computer Engineering at PICT, SPPU, Pune, Maharashtra, India for the last 31 years. He was the Chairman of the Board of Studies of Computer Engineering at Savitribai Phule Pune University.