IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, May 2026
Impact Factor: 8.2

Forest Fire Detection and Classification Using Deep Learning Techniques

Mr. Omkar Anil Gunjal, Mr. Tejas Ramesh Gangurde, Mr. Nikhil Bahusaheb Amabre
Ms. Aditi Balasaheb Kakade, Prof. A. R. Nehe
Department of Computer Engineering
Amrutvahini College of Engineering Sangamner, India
omkargunjal27@gmail.com, t84219@gmail.com, nikhilambre392@gmail.com
kakadeaditi72@gmail.com, ashwini.nehe@avcoe.org

Abstract: *Forest fires are one of the major environmental threats that cause severe damage to wildlife, vegetation, natural resources, and human life. Early detection and rapid response are essential to minimize the spread and impact of such disasters. Traditional forest fire monitoring methods often rely on manual observation and sensor-based systems, which may be slow, expensive, or less effective in remote areas. To overcome these limitations, this project presents an AI-powered Forest Fire Detection System using Convolutional Neural Networks (CNN) and computer vision techniques.*
The proposed system uses a customized MobileNetV2 deep learning model trained on a large dataset of fire and non-fire images to accurately identify forest fire incidents in real-time. The system continuously processes live video streams from cameras and detects the presence of fire with high confidence. Once a fire is detected, the system automatically sends instant WhatsApp alerts using the Twilio API, enabling quick emergency response and monitoring..

Keywords: Forest Fire Detection, Deep Learning, Convolutional Neural Network (CNN), MobileNetV2, Computer Vision, Real-Time Monitoring, Artificial Intelligence, Flask, Tkinter, Twilio API, WhatsApp Alert System, Image Processing, Wildfire Detection, Transfer Learning, Environmental Monitoring

I. INTRODUCTION

Forest fires are among the most dangerous natural disasters affecting ecosystems, wildlife, human life, and environmental sustainability across the world [1]. Every year, millions of hectares of forest land are destroyed due to uncontrolled wildfires, leading to severe ecological imbalance, air pollution, and economic losses [2]. Climate change, rising temperatures, dry weather conditions, and human negligence have significantly increased the frequency and intensity of forest fires in recent years [3]. Therefore, early detection and rapid response are critical for minimizing damage and preventing large-scale disasters.

Traditional forest fire detection methods mainly depend on human surveillance, watchtowers, satellite monitoring, and sensor-based systems [4]. Although these methods are useful, they often suffer from limitations such as delayed detection, high operational costs, limited coverage, and reduced efficiency in remote forest regions [5]. In many cases, smoke and fire spread rapidly before authorities can take preventive action. As a result, there is a growing demand for intelligent and automated systems capable of detecting forest fires in real-time with higher accuracy and reliability [6].

Recent advancements in Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) have transformed the field of image processing and object detection [7]. Deep learning models, especially Convolutional Neural Networks (CNNs), have shown remarkable performance in image classification and real-time video analysis applications [8]. CNN-based systems can automatically extract important visual features from images and accurately distinguish fire and non-fire patterns without manual feature engineering [9]. Due to their high accuracy and fast processing capability, CNN models are increasingly being used in wildfire detection and environmental monitoring systems [10].

This project presents an AI-powered Forest Fire Detection System using a customized MobileNetV2 CNN architecture and computer vision techniques. The proposed system continuously monitors live video streams and analyzes frames to identify the presence of fire in real-time [11]. MobileNetV2 is selected because of its lightweight architecture, low computational requirements, and high efficiency, making it suitable for real-time applications [12]. The system uses transfer learning to improve detection accuracy while reducing training time and resource consumption.

II. PROBLEM STATEMENT

Forest fires have become a major environmental and safety concern due to their rapid spread and destructive impact on forests, wildlife, human life, and natural resources. Traditional fire detection methods such as manual surveillance, watchtowers, and sensor-based monitoring systems often fail to provide early and accurate detection, especially in remote forest areas. These methods are time-consuming, costly, and highly dependent on human observation, which can lead to delayed responses and increased damage. Existing systems also face challenges such as false alarms, limited monitoring coverage, poor real-time performance, and lack of automated emergency communication. In many situations, authorities receive fire information only after the fire has spread significantly, making it difficult to control the disaster effectively. To overcome these limitations, there is a need for an intelligent, automated, and real-time forest fire detection system capable of accurately identifying fire from live video streams and images. The system should provide fast detection, reduce false alarms, support continuous monitoring, and automatically notify concerned authorities for immediate action.

III. OBJECTIVES

- To develop an AI-based system for real-time forest fire detection using Convolutional Neural Networks (CNN).
- To use the MobileNetV2 deep learning model for accurate fire and non-fire image classification.
- To monitor live video streams and detect fire automatically using computer vision techniques.
- To send instant WhatsApp alerts using the Twilio API when fire is detected with high confidence.
- To provide user-friendly web and desktop interfaces for monitoring, analysis, and system management.

IV. PROPOSED METHODOLOGY

The proposed Forest Fire Detection System uses deep learning and computer vision techniques to identify fire incidents in real-time from live video streams and images. The methodology combines image preprocessing, feature extraction, CNN-based classification, and automated alert generation to provide accurate and efficient wildfire detection.

1. Data Collection

A large dataset containing fire and non-fire images is collected from various sources. The dataset includes different forest environments, flame patterns, smoke conditions, and lighting variations to improve model performance and detection accuracy.

2. Data Preprocessing

The collected images are preprocessed before training the model. In this stage, corrupt images are removed, images are resized to a fixed dimension, and normalization is applied to improve image quality and maintain consistency. The dataset is then divided into training and validation sets.

3. Model Selection

The MobileNetV2 Convolutional Neural Network model is selected for the proposed system due to its lightweight architecture, faster processing speed, and efficient performance in real-time applications. Transfer learning is used to improve accuracy and reduce training time.

4. Model Training

The preprocessed dataset is provided to the MobileNetV2 model for training. During training, the model learns important fire-related features such as flame color, smoke patterns, brightness, and texture differences between fire and non-fire images. Multiple training epochs are performed to improve classification accuracy.

5. Fire Detection Process

The trained model is integrated with OpenCV to process live camera feeds and uploaded images. Each video frame is analyzed in real-time, and the system predicts whether fire is present based on confidence scores generated by the CNN model.

6. Alert Generation

If the prediction confidence exceeds the predefined threshold, the system classifies the image as “Fire” and automatically sends a WhatsApp alert message using the Twilio API. This helps notify authorities and users immediately for quick response.

7. User Interface Integration

The proposed system includes both a Flask-based web dashboard and a Tkinter desktop GUI. These interfaces allow users to monitor live video feeds, upload images, view prediction results, and manage system settings easily.

8. Performance Evaluation

The performance of the system is evaluated using accuracy graphs, loss graphs, confusion matrix analysis, and prediction results. The evaluation confirms the effectiveness and reliability of the proposed forest fire detection system in real-time monitoring applications.

V. WORKING OF SYSTEM

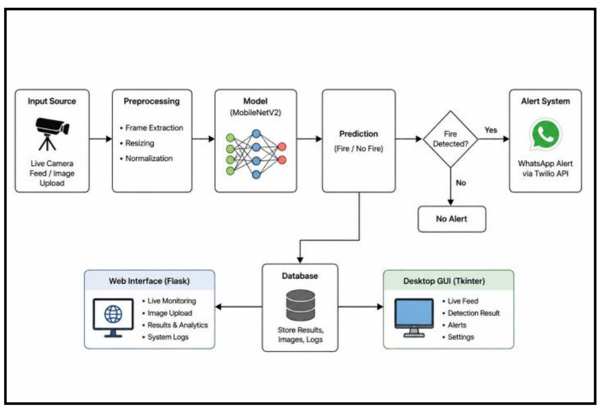

Fig 1: Block Diagram

1. Input Source

The system begins by collecting input through a live camera feed or uploaded images. The camera continuously captures forest area footage, which is used for real-time fire monitoring and analysis.

2. Image Preprocessing

The captured frames are preprocessed before being sent to the deep learning model. In this stage, image resizing and normalization are performed to improve image quality and ensure compatibility with the trained model.

3. Fire Detection Model

The preprocessed images are passed to the MobileNetV2 deep learning model. The model analyzes visual patterns such as flames, smoke, and fire color features to classify the image as fire or non-fire.

4. Prediction and Classification

After analysis, the system generates prediction results based on confidence scores. If fire-related features are detected with high confidence, the image is classified as “Fire”; otherwise, it is classified as “No Fire.”

5. Alert Generation

When fire is detected, the system automatically triggers an alert notification. A WhatsApp alert message is sent using the Twilio API to inform forest authorities or responsible users for immediate action.

6. No Alert Condition

If the system does not detect any fire, no alert is generated. The monitoring process continues continuously for further observation and analysis.

7. Database Storage

The system stores detection results, images, logs, and monitoring data in the database. This helps maintain records for future analysis, performance evaluation, and reporting purposes.

8. Web Interface (Flask)

The Flask-based web dashboard allows users to monitor live video feeds, upload images manually, view detection results, and access system analytics and logs through a browser interface.

9. Desktop GUI (Tkinter)

The Tkinter desktop application provides local system monitoring features such as live feed viewing, fire detection status, alert notifications, and system settings management in a user-friendly interface.

VI. SYSTEM DESIGN

1. System Overview

The proposed system is designed to automatically detect forest fires using deep learning and computer vision techniques. It processes both uploaded images and real-time video streams to identify the presence of fire and classify them into two categories: "Fire" and "No Fire." The system integrates image preprocessing, deep learning models, and automated alert mechanisms to ensure accurate and efficient fire detection, making it suitable for forest monitoring and disaster management applications.

The system uses the MobileNetV2 Convolutional Neural Network (CNN) model for real-time fire classification. When fire is detected with high confidence, the system automatically sends WhatsApp alert notifications using the Twilio API to help authorities take immediate action. The project also includes a Flask-based web interface and a Tkinter desktop application for live monitoring, image upload analysis, result visualization, and system management.

1. Input Acquisition Module

This module captures input data from live camera feeds or uploaded forest images. It continuously collects visual data for fire monitoring and sends it to the preprocessing stage for further analysis.

2. Image Preprocessing Module

The preprocessing module improves image quality before detection. It performs operations such as image resizing, normalization, frame extraction, and enhancement to prepare the input data for the deep learning model.

3. Fire Detection Module

This module uses the MobileNetV2 CNN model to analyze the processed images. It extracts important visual features such as flames, smoke patterns, and fire color intensity to classify images as "Fire" or "No Fire."

4. Prediction and Decision Module

The prediction module evaluates the confidence score generated by the trained model. Based on the prediction result, the system decides whether a fire is detected and initiates the next operation accordingly.

5. Alert Notification Module

If fire is detected, this module automatically sends WhatsApp alert messages using the Twilio API. The alert includes fire detection information for quick response and emergency management.

6. Database Management Module

This module stores detection results, uploaded images, logs, confidence scores, and monitoring data. It helps maintain records for future analysis and performance evaluation.

7. Web Dashboard Module

The Flask-based web dashboard allows users to monitor live video feeds, upload images for testing, view prediction results, and access system analytics through a browser interface.

8. Desktop GUI Module

The Tkinter desktop application provides a user-friendly interface for local monitoring. It displays live camera feeds, fire detection status, alert notifications, and system settings for easy management.

VII. MATHEMATICAL EQUATIONS

1. Convolution Operation

The convolution operation is used in CNN to extract important image features such as fire patterns, smoke texture, and color intensity.

$$(I * K)(x, y) = \sum m \sum n I(m, n) K(x - m, y - n)$$

Where:

I = Input image

K = Kernel or filter

x, y = Pixel coordinates

2. ReLU Activation Function

The Rectified Linear Unit (ReLU) activation function introduces non-linearity into the CNN model.

$$f(x) = \max(0, x)$$

Where:

x = Input value

$f(x)$ = Output after activation

3. Softmax Function

The Softmax function is used in the output layer for classification of "Fire" and "No Fire."

$P = \sum e^x e^x$

Where:

P = Predicted probability

e^x = Exponential output value

$\sum e^x$=Sum of all exponential values

4. Cross-Entropy Loss Function

The loss function measures the difference between actual and predicted outputs during training.

$$L = -\sum i = 1nyilog(yi^\wedge)$$

Where:

L = Loss value

yi = Actual class label

$yi^\wedge$ = Predicted probability

5. Accuracy Calculation

Accuracy is used to evaluate the performance of the forest fire detection model.

$$\text{Accuracy} = \frac{(TP + TN)}{TP + TN + FN + FP} \times 100$$

Where:

TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative

6. Precision Calculation

Precision measures the correctness of positive fire predictions.

$$\text{Precision} = \frac{TP}{TP + FP}$$

7. Recall Calculation

Recall measures the ability of the model to correctly detect fire images.

$$Recall = \frac{TP}{TP+FN}$$

VIII. RESULTS

The proposed Forest Fire Detection System was successfully trained and tested using the MobileNetV2 deep learning model. The system achieved high accuracy in distinguishing fire and non-fire images from real-time video streams and uploaded images

1. Training and Validation Accuracy

The training and validation accuracy graph represents the learning performance of the proposed MobileNetV2 model during the training process. Accuracy is an important performance metric that indicates how correctly the model classifies fire and non-fire images. During training, the model continuously learns important visual features such as flame patterns, smoke intensity, color variations, and environmental textures from the dataset.

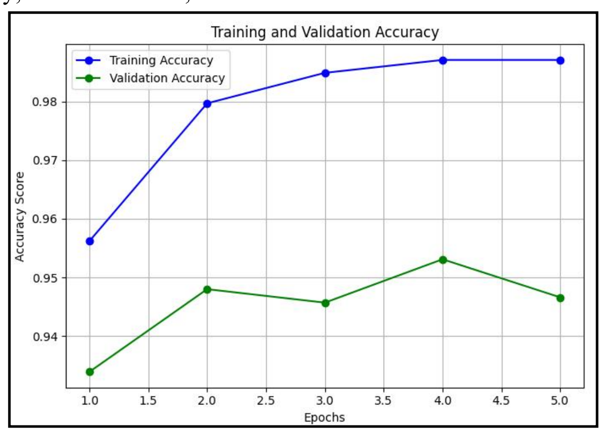

Fig.2 Training vs Validation accuracy

The training and validation accuracy graph shows the performance improvement of the model during the training process. The training accuracy increased steadily with each epoch and reached approximately 98%, while the validation accuracy remained around 94%–95%. This indicates that the model learned the fire detection patterns effectively and provided stable prediction performance on unseen data.

High training accuracy indicates successful learning of fire-related features.

Stable validation accuracy shows good generalization capability.

The model demonstrates reliable performance for real-time forest fire detection.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, May 2026
Impact Factor: 8.2

2. Training and Validation Loss

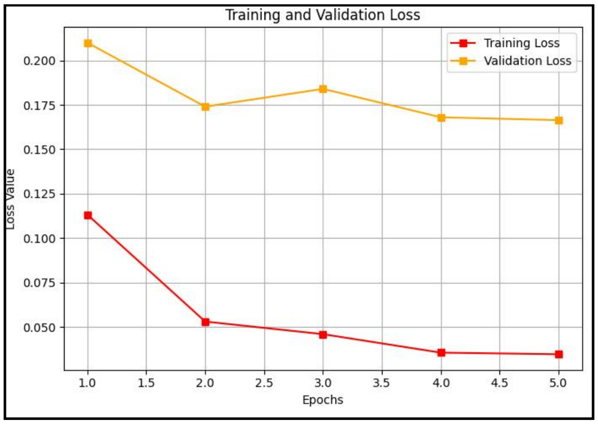

Fig.3 Training and Validation Loss

The loss graph represents the reduction in prediction error during training. The training loss continuously decreased as the epochs increased, indicating that the model optimization process was successful. The validation loss remained comparatively stable, which shows that the model avoided major overfitting problems.

Decreasing training loss indicates improved model learning.

Stable validation loss shows balanced model performance.

The model achieved efficient convergence during training.

3. Confusion Matrix Analysis

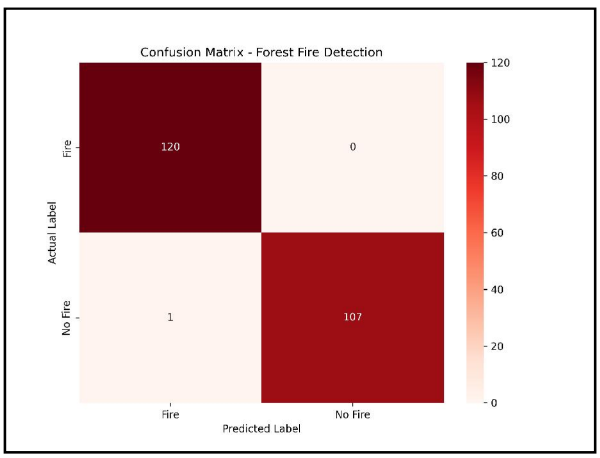

Fig.4.Confusion Matrix

The confusion matrix was used to evaluate the classification performance of the proposed system. The matrix shows the number of correctly and incorrectly classified fire and non-fire images.

True Positive (Fire correctly detected): 120

True Negative (No Fire correctly detected): 107

False Positive: 1

False Negative: 0

The confusion matrix demonstrates that the system achieved very high detection accuracy with minimal false predictions. The model successfully identified most fire incidents while reducing incorrect alerts.

4. Overall System Performance

The proposed system achieved excellent performance in real-time forest fire detection using deep learning and computer vision techniques. The integration of MobileNetV2, Flask dashboard, Tkinter GUI, and Twilio alert notifications improved the efficiency and usability of the system for practical environmental monitoring applications.

VIII. CONCLUSION

The proposed Forest Fire Detection System successfully demonstrates the use of deep learning and computer vision techniques for accurate and real-time wildfire detection. The system uses the MobileNetV2 Convolutional Neural Network model to classify fire and non-fire images with high accuracy and stable performance. By processing live camera feeds and uploaded images, the system can continuously monitor forest environments and identify fire incidents at an early stage.

The integration of automated WhatsApp alerts using the Twilio API enables quick communication with authorities, helping improve emergency response and reduce the spread of forest fires. The developed Flask web dashboard and Tkinter desktop GUI provide user-friendly interfaces for monitoring, image analysis, result visualization, and system management.

Experimental results, including accuracy graphs, loss analysis, and confusion matrix evaluation, show that the proposed system achieved excellent classification performance with minimal false predictions. The model demonstrated high training and validation accuracy, indicating effective learning and reliable real-time detection capability.

Overall, the developed system offers a cost-effective, scalable, and efficient solution for forest fire monitoring and disaster management. The project contributes toward environmental protection by enabling early fire detection, reducing wildfire damage, and supporting faster decision-making during emergency situations.

IX. FUTURE SCOPE

In the future, the proposed Forest Fire Detection System can be enhanced by integrating advanced technologies such as drones, IoT sensors, GPS, and cloud computing for more efficient and large-scale forest monitoring. Drone-based surveillance can help detect fires in remote and inaccessible areas, while IoT sensors can improve accuracy by monitoring temperature, smoke, and gas levels. The system can also be connected to cloud platforms for centralized monitoring and real-time data access from multiple locations. Advanced deep learning models such as YOLO and EfficientNet may further improve detection speed and accuracy. Mobile applications can be developed to provide instant alerts and live monitoring on smartphones. Additionally, the system can be expanded to predict fire spread, identify smoke intensity, and support smart city and industrial fire safety applications, making it more reliable and effective for future disaster management systems.

REFERENCES

[1]. Krizhevsky, A., Sutskever, I., & Hinton, G. E., "ImageNet Classification with Deep Convolutional Neural Networks," Advances in Neural Information Processing Systems, vol. 25, pp. 1097–1105, 2012.

[2]. Howard, A. G., Zhu, M., Chen, B., et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv preprint arXiv:1704.04861, 2017.

[3]. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C., "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4510–4520, 2018.

[4]. Muhammad, K., Ahmad, J., Baik, S. W., "Efficient Deep CNN-Based Fire Detection and Localization in Video Surveillance Applications," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 49, no. 7, pp. 1419–1429, 2019.

[5]. Yuan, F., "A Fast Accumulative Motion Orientation Model Based on Integral Image for Video Smoke Detection," Pattern Recognition Letters, vol. 29, no. 7, pp. 925–932, 2008.

[6]. Zhang, Q., Xu, J., Xu, L., & Guo, H., "Deep Convolutional Neural Networks for Forest Fire Detection," International Journal of Intelligent Systems and Applications, vol. 8, no. 12, pp. 57–65, 2016.

[7]. Li, Z., Wang, C., Emrich, C. T., & Guo, D., "A Novel Fire Detection System Based on Transfer Learning and Deep Learning," Fire Safety Journal, vol. 120, pp. 103–112, 2020.

[8]. Rao, S., Singh, P., Sharma, M., & Dhar, Y., "Early Detection of Forest Fire Using Fine-Tuned MobileNetV2: A Lightweight Deep Learning Approach," Lecture Notes in Networks and Systems, Springer, vol. 1479, pp. 45–54, 2025.

[9]. Tsalera, E., Papadakis, A., Voyiatzis, I., & Samarakou, M., "CNN-Based, Contextualized, Real-Time Fire Detection in Computational Resource-Constrained Environments," Energy Reports, vol. 9, pp. 442–450, 2023.

[10]. Dewangan, A., Pande, Y., Braun, H. W., et al., "FIgLib & SmokeyNet: Dataset and Deep Learning Model for Real-Time Wildland Fire Smoke Detection," arXiv preprint arXiv:2112.08598, 2021.

[11]. Huot, F., Hu, R. L., Ihme, M., et al., "Deep Learning Models for Predicting Wildfires from Historical Remote-Sensing Data," arXiv preprint arXiv:2010.07445, 2020.

[12]. Martin, V., Venable, K. B., & Morgan, D., "Development and Application of a Sentinel-2 Satellite Imagery Dataset for Deep-Learning Driven Forest Wildfire Detection," arXiv preprint arXiv:2409.16380, 2024.

[13]. Haq, I. U., Husnain, G., Iqbal, A., & Alzahrani, A. S., "Attention-Enhanced MobileNetV2 Models for Robust Forest Fire Detection and Classification," Scientific Reports, vol. 16, 2026.

[14]. Chen, T., Liu, X., & Wang, Y., "Forest Fire Detection Based on Deep Learning and Computer Vision," Journal of Physics: Conference Series, vol. 1237, no. 2, pp. 022–034, 2019.

[15]. Martínez-de-Dios, J. R., Merino, L., Caballero, F., & Ollero, A., "Automatic Forest-Fire Measuring Using Ground Stations and Unmanned Aerial Systems," Sensors, vol. 11, no. 6, pp. 6328–6353, 2011.

[16]. Celik, T., & Demirel, H., "Fire Detection in Video Sequences Using a Generic Color Model," Fire Safety Journal, vol. 44, no. 2, pp. 147–158, 2009.

[17]. Toreyin, B. U., Dedeoglu, Y., & Cetin, A. E., "Contour Based Smoke Detection in Video Using Wavelets," in European Signal Processing Conference, pp. 1–5, 2006.

[18]. Ko, B. C., "A Real-Time Fire Detection Method Using Machine Vision," Fire Safety Journal, vol. 44, no. 4, pp. 589–596, 2009.

[19]. Sharma, A., Gupta, R., & Kumar, S., "Forest Fire Detection Using Deep Learning Techniques," International Journal of Advanced Computer Science and Applications, vol. 12, no. 5, pp. 315–321, 2021.

[20]. Singh, D., Sharma, V., & Kaur, M., "Real-Time Forest Fire Detection Using Convolutional Neural Networks and Image Processing," International Journal of Computer Applications, vol. 183, no. 12, pp. 20–26, 2022