Fire Detection System with Real-Time Alarm and Email Notification

Abstract

This project presents a real-time fire detection system using a deep learning model integrated with a web application and real-time video monitoring system. Upon detecting fire, the system triggers an alarm and sends an email alert to notify the concerned parties, ensuring timely response and safety.

1. Introduction

Fire incidents pose serious threats to life and property. Traditional fire detection systems often lack real-time responsiveness or fail to notify remotely. This project integrates computer vision, deep learning, and automation to develop a robust fire detection system that functions through both image upload and live camera feed, providing real-time alerts via sound and email notifications.

2. System Overview

The system consists of two primary modules:

1. A Flask web application where users can upload images to check for fire.
2. A Python script using OpenCV and TensorFlow for real-time fire detection via webcam, with alarm ringing and email notification.

3. Dataset & Model Training

A dataset with 'fire' and 'non_fire' categories was used to train the model. MobileNetV3Small was employed as the base model due to its efficiency. The data was preprocessed (resizing, normalization) and augmented (rotation, flipping, etc.) to improve generalization. Binary cross-entropy loss and Adam optimizer were used during training.

4. Model Architecture

The architecture is based on MobileNetV3Small pretrained on ImageNet. The top layers were customized with batch normalization, dropout, and dense layers to classify between fire and non-fire images. The model achieved satisfactory accuracy and was saved as 'fire_detection_model.h5'.

5. Web Application (Flask)

The Flask app allows users to upload images which are then processed by the trained model. If fire is detected (confidence > 0.75), the result is shown on the webpage. It includes validation of file types, size restrictions, and displays prediction confidence.

6. Real-Time Detection & Notifications

Using OpenCV, the system captures live feed from a webcam, processes each frame, and detects fire using the trained model. Upon detection, it triggers a sound alarm using winsound and sends an email alert asynchronously using the smtplib module. Fire detection logic combines deep learning prediction and HSV-based pixel thresholding for improved accuracy.

7. Results & Evaluation

The model was tested on various fire and non-fire images. It achieved over 90% accuracy on the validation set. The system performed reliably in real-time with minimal lag, and alerts (alarm and email) were sent successfully upon detection.

8. Conclusion

This fire detection system demonstrates the power of integrating AI and automation for safety-critical applications. It offers both offline (image upload) and real-time monitoring capabilities, ensuring a proactive approach to fire incidents.

9. Future Scope

- Deploy to cloud or IoT device (e.g., Raspberry Pi).
- Add SMS notifications or integration with fire department APIs.
- Improve model to detect smoke or gas leakage.
- Use thermal imaging for better detection in dark environments.

10. References

- TensorFlow Documentation
- OpenCV Library

- MobileNetV3 Research Paper
- Flask Web Framework Documentation