



AI Fire Detection & Alarm System



Project Overview

An intelligent fire detection system powered by **Python** and **OpenCV** that provides real-time monitoring through computer vision. The system uses HSV color thresholding to identify fire patterns and triggers immediate visual and audio alarms when fire is detected.



Dual Display Mode

Live camera feed with detection overlay + dedicated alarm status window



Audio Alerts

Looping alarm sound with configurable on/off control



Real-Time Processing

Fast HSV color-based detection with FPS monitoring



Multi-Camera Support

Automatic camera detection and selection interface



System Architecture



Camera Input

Captures video frames at 640×480



Fire Detector

HSV color analysis & filtering



Alarm Manager

State management & audio control



Display Renderers

Dual window visualization



How Fire Detection Works

1

Frame Capture & Conversion

Captures video frame from camera and converts from BGR to HSV color space for better color detection

2

Multi-Range Color Filtering

Applies three HSV color ranges: Orange-red (H: 0-15), Yellow-orange (H: 15-35), Deep red (H: 170-180) with high saturation (100-255) and value (200-255)

3

Mask Processing & Cleanup

Morphological operations (close, open) remove noise, Gaussian blur smooths edges, creating clean binary mask

4

Contour Analysis

Finds contours in mask, filters by minimum area (500 pixels), calculates confidence based on total fire area detected

5

Consecutive Frame Confirmation

Requires fire detection in consecutive frames before triggering alarm to reduce false positives

🌟 Key Features & Capabilities

- ▶ Real-time fire detection using computer vision
- ▶ Dual display windows (feed + alarm)
- ▶ Confidence meter display
- ▶ FPS performance monitoring
- ▶ Configurable detection sensitivity
- ▶ Consecutive frame validation
- ▶ HSV color-based detection algorithm
- ▶ Visual bounding boxes around fire regions
- ▶ Looping audio alarm (M4A/WAV support)
- ▶ Multi-camera auto-detection
- ▶ Keyboard controls for interaction
- ▶ Three-state alarm system (Normal/Warning/Alarm)

🚀 Quick Setup & Installation

📦 Prerequisites

Python 3.7+

Webcam

⚙️ Required Dependencies

opencv-python

numpy

pygame (optional)

💻 Installation Steps

```
# Install dependencies
pip install -r requirements.txt

# Or install manually
pip install opencv-python numpy
pygame
```

▶ Running the Application

```
python fire_alarm_system.py
```

The system will scan for available cameras and present a selection menu. Select your camera and the application will start monitoring for fire.

🎮 Keyboard Controls Reference

Key	Action	Description
Q	Quit Application	Cleanly exits the fire detection system and closes all windows
S	Toggle Alarm Sound	Enables or disables the audio alarm (visual alerts remain active)
R	Reset Detection	Stops current alarm, clears detection history, resets to normal state



Testing & Configuration Tips



Testing Fire Detection

Display fire/flame images on your phone screen and show to the camera. The system detects orange, red, and yellow fire colors.



Tuning Sensitivity

Adjust `FIRE_MIN_AREA` (minimum pixels) and `CONSECUTIVE_FRAMES` in the code for your environment.



Technology Stack

Python 3.7+

Core programming language for the application

OpenCV 4.5+

Computer vision library for image processing

NumPy

Array operations and numerical computations

Pygame

Audio playback for alarm sounds

Made with ❤️ by

Amarjeet Sharma



@amarjeet.creator