IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 7, April 2024

# Fire Detection System using OpenCv And Python

**Akash Patel, Saad Shaikh, Yash Palshetkar, Shantanu Raje**
Final Year Students, Department of Computer Engineering
MGM's College of Engineering and Technology, Navi Mumbai, India

**Abstract***:* *Fire detection is vital for safeguarding lives and property. This project proposes a real-time fire detection system using OpenCV and Python. It utilizes computer vision techniques to analyze video feeds signs of fire or smoke. Key steps include preprocessing, feature extraction, and classification using machine learning algorithms like support vector machines or convolutional neural networks. Upon detection, the system triggers alerts such as sounding alarms or notifying authorities. Offering cost-effective and efficient monitoring, this system enhances safety in residential, industrial, and forest environments. Leveraging OpenCV's capabilities, it represents a promising approach to fire detection leveraging the power of computer vision and machine learning*

**Keywords:** Fire detection, Computer vision, OpenCV, Python, Machine learning

## I. INTRODUCTION

Fire poses a significant threat to lives and property worldwide, necessitating effective detection systems for early warning and response. Leveraging advancements in computer vision and machine learning, this project introduces a real- time fire detection system using OpenCV and Python. By analyzing video feeds, the system identifies signs of fire or smoke, enabling timely alerts and mitigation measures. This introduction outlines the system's objectives, highlighting its potential to enhance safety in residential, industrial, and forest environments. With a focus on cost-effectiveness and efficiency, the proposed system represents a promising approach to fire detection, leveraging cutting-edge technologies to address critical safety concerns.

## II. LITERATURE REVIEW

Existing literature reveals a growing interest in computer vision-based fire detection systems. Researchers have explored various techniques, including color segmentation, texture analysis, and machine learning algorithms, to improve detection accuracy and efficiency. These studies highlight the potential of computer vision technology for enhancing fire safety measures.

### Disadvantages of Existing System

- Reliability.
- Response Time.
- Maintenance Requirement.
- Manual
- Activation

### Proposed System:

The proposed system utilizes OpenCV and Python for real-time fire detection. It employs computer vision techniques such as preprocessing, feature extraction, and machine learning classification to analyze video feeds. Upon detection of fire or smoke, the system triggers alerts for timely response, enhancing safety in diverse environments.

### Focal Points for Fire Detection System Using Python and OpenCV:

- Real-time Monitoring
- Computer Vision
- Techniques Early

ISSN 2581-9429 IJARSCT

DOI: 10.48175/IJARSCT-17886

- Warning System
- Cost- effective Solution
- Versatility
- Integration Enhanced
- Safety Measures
- Scalability
- Machine Learning Algorithms

## III. METHODOLOGIES

"To develop the Fire Detection System Using Python and OpenCV, we applied a systematic approach

- Data Collection: Gather a diverse dataset of video footage containing fire and non-fire scenarios captured in different environments.
- Preprocessing: Preprocess video frames to enhance relevant features using techniques such as color segmentation, noise reduction, and image enhancement.
- Feature Extraction: Extract discriminative features from preprocessed frames, including color histograms, texture descriptors, and motion characteristics.
- Machine Learning Model Development: Train machine learning models, such as Support Vector Machines
- (SVM) or Convolutional Neural Networks (CNN), using extracted features to classify frames as fire or non-fire.
- Model Evaluation: Evaluate model performance using metrics such as accuracy, precision, recall, and F1-score on a separate validation dataset.
- System Integration: Integrate the trained model into a real-time fire detection system using OpenCV and Python, incorporating functionalities for video input, frame processing, and alert triggering.
- Testing and Validation: Conduct extensive testing of the integrated system in various environments to assess its effectiveness, robustness, and scalability.
- Optimization and Fine-tuning: Fine-tune parameters and optimize algorithms to improve detection accuracy and reduce false positives/negatives.

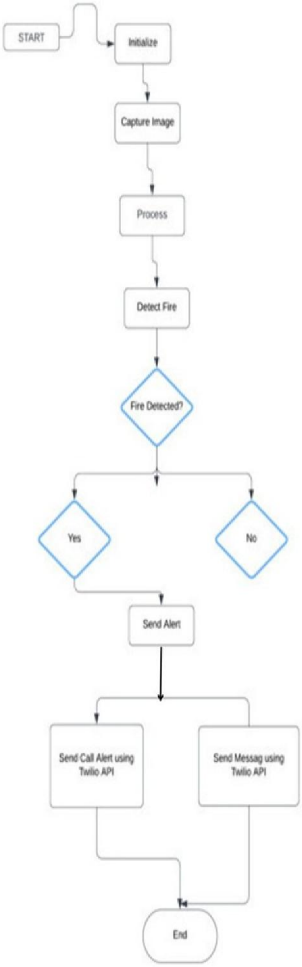
## IV. EXPLORING ANALYTICAL APPROACHES BASED ON MEASURESAND PERFORMANCE

To measure Analytical Approaches for Fire Detection System Using Python and OpenCV:

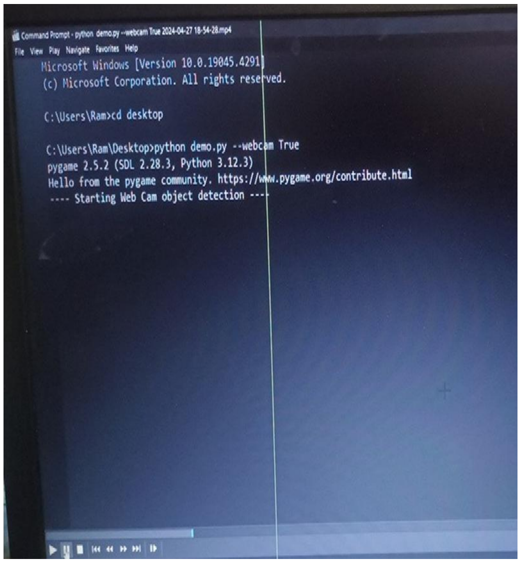
- Benchmarking Performance Measures: Evaluate the performance of the fire detection system using metrics such as accuracy, precision, recall, and F1-score to assess its effectiveness in identifying fire incidents.
- Comparative Analysis: Compare the performance of different machine learning algorithms (e.g., SVM, CNN) and feature extraction techniques (e.g., color histograms, texture descriptors) to determine the most effective approach for fire detection.
- Feature Importance Analysis: Conduct feature importance analysis to identify the most discriminative features for fire detection, helping to refine feature extraction methods and improve overall system performance
- Cross-validation Techniques: Employ cross-validation techniques such as k-fold cross-validation t validate the robustness and generalization capability of the fire detection model across different datasets and environmental conditions.
- Hyperparameter Tuning: Explore hyperparameter tuning techniques such as grid search or random search to optimize the performance of machine learning models, fine-tuning parameters to achieve optimal detection accuracy.
- Real-time Performance Evaluation: Evaluate the real-time performance of the fire detection system by measuring processing time per frame and frame rate, ensuring timely detection and alerting capabilities in practical deployment scenarios

## V. OUTPUT

Flowchart

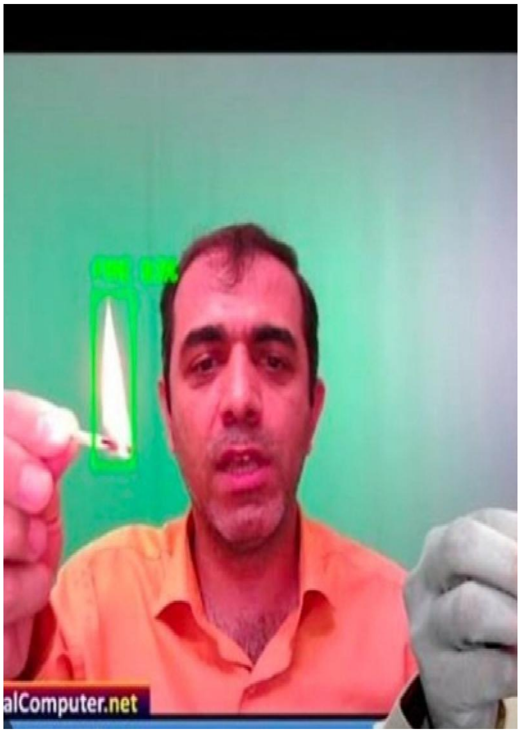


User Interface


Command Prompt - python demo.py --webcam True 2024-04-27 18-54-28.mp4
File View Play Navigate Favorites Help

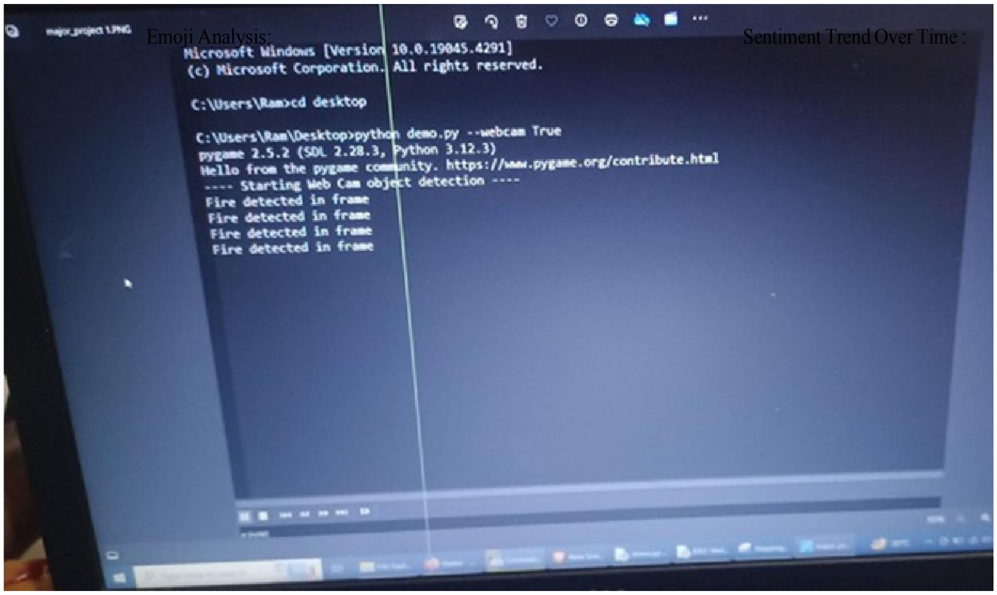
```
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Ram>cd desktop

C:\Users\Ram\Desktop>python demo.py --webcam True
pygame 2.5.2 (SDL 2.28.3, Python 3.12.3)
Hello from the pygame community. https://www.pygame.org/contribute.html
---- Starting Web Cam object detection ----
```

Fire detection



DOI: 10.48175/IJARSCT-17886
ISSN 2581-9429 IJARSCT

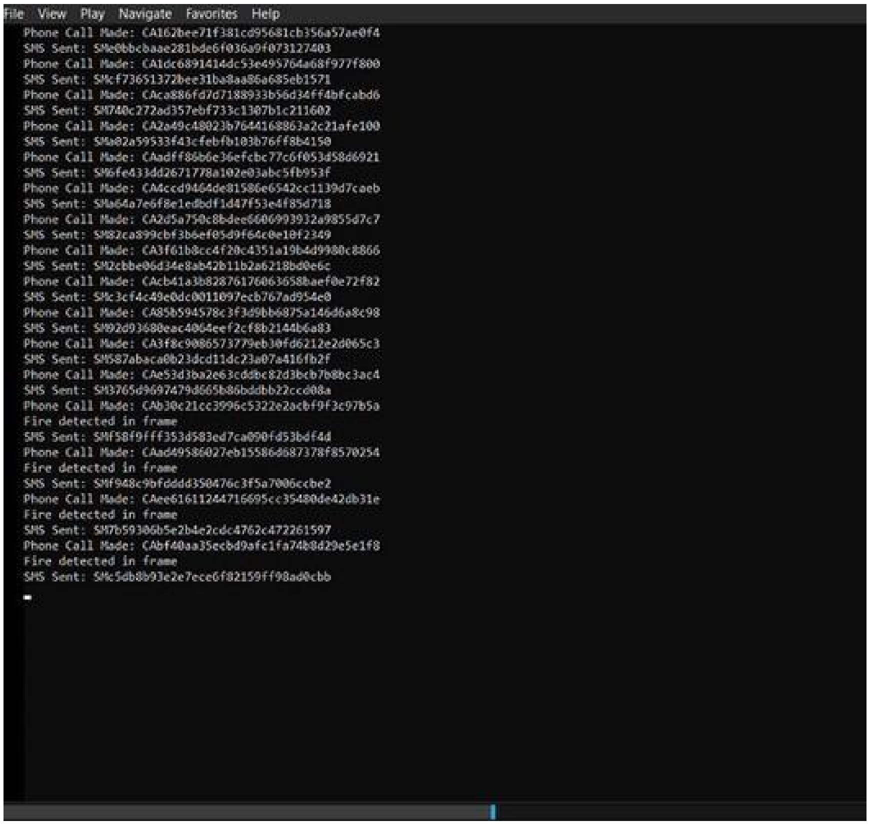
Fire Detected In Webcam



Emoji Analysis:

Sentiment Trend Over Time :

```
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Ram>cd desktop

C:\Users\Ram\Desktop>python demo.py --webcam True
pygame 2.5.2 (SDL 2.28.3, Python 3.12.3)
Hello from the pygame community. https://www.pygame.org/contribute.html
---- Starting Web Cam object detection ----
Fire detected in frame
Fire detected in frame
Fire detected in frame
Fire detected in frame
```

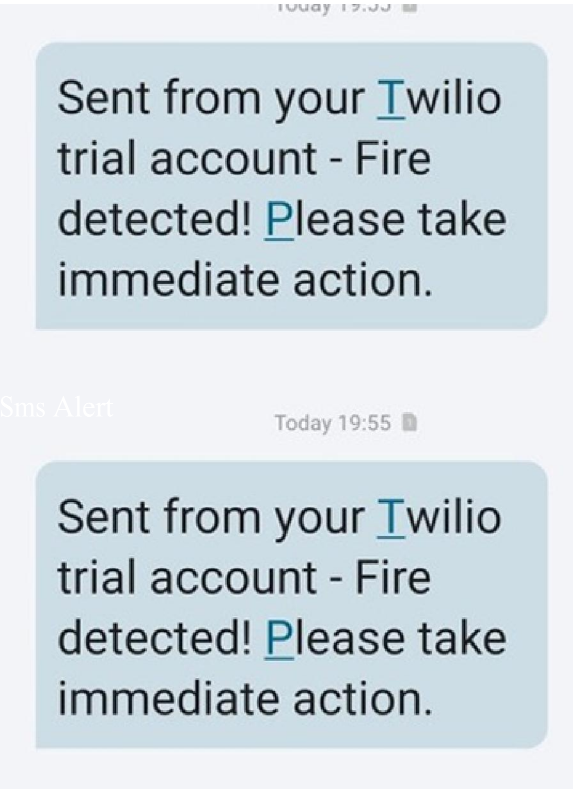


```
Phone Call Made: CA162bee71f381cd95681cb356a57ae0f4
SMS Sent: SMe0bbcbaae281bde6f036a9f073127403
Phone Call Made: CA1dc6891414dc53e495764a68f977f800
SMS Sent: SMcf73651372bee31ba8aa86a685eb1571
Phone Call Made: CAca886fd7d7188933b56d34ff4bfcabd6
SMS Sent: SM740c272ad357ebf733c1307b1c211602
Phone Call Made: CA2a49c48023b7644168863a2c21afe100
SMS Sent: SMa02a59533f43cfebfb103b76ff8b4150
Phone Call Made: CAadff86b6e36efcbc77c6f053d58d6921
SMS Sent: SM6fe433dd2671778a102e03abc5fb953f
Phone Call Made: CA4ccd9464de81586e6542cc1139d7caeb
SMS Sent: SMa64a7e6f8e1e8bdf1d47f53e4f85d718
Phone Call Made: CA2d5a750c8bdee6606993932a9855d7c7
SMS Sent: SM82ca899cbf3b6ef05d9f64c0e10f2349
Phone Call Made: CA3f61b8cc4f20c4351a19b4d9980c8866
SMS Sent: SM2cbbe06d34e8ab42b11b2a6218bd0e6c
Phone Call Made: CAcb41a3b82876176063658baef0e72f82
SMS Sent: SMc3cf4c49e0dc0011097ecb767ad954e0
Phone Call Made: CA85b594578c3f3d9bb6875a146d6a8c98
SMS Sent: SM92d93680eac4064eef2cf8b2144b6a83
Phone Call Made: CA3f8c9086573779eb30fd6212e2d065c3
SMS Sent: SM587abaca0b23dcd11dc23a07a416fb2f
Phone Call Made: CAe53d3ba2e63cddbc82d3bcb7b8bc3ac4
SMS Sent: SM3765d9697479d665b86bddbb22ccd08a
Phone Call Made: CAb30c21cc3996c5322e2acbf9f3c97b5a
Fire detected in frame
SMS Sent: SMf58f9fff353d583ed7ca090fd53bdf4d
Phone Call Made: CAad49586027eb15586d687378f8570254
Fire detected in frame
SMS Sent: SMf948c9bfdddd350476c3f5a7006ccbe2
Phone Call Made: CAee61611244716695cc35480de42db31e
Fire detected in frame
SMS Sent: SM7b59306b5e2b4e2cdc4762c472261597
Phone Call Made: CAbf40aa35ecbd9afc1fa74b8d29e5e1f8
Fire detected in frame
SMS Sent: SMc5db8b93e2e7ece6f82159ff98ad0cbb
```

Start Sending Call And Sms Alert

DOI: 10.48175/IJARSCT-17886

Call Alert



Today 19:55

Sent from your Twilio trial account - Fire detected! Please take immediate action.

Sms Alert

Today 19:55

Sent from your Twilio trial account - Fire detected! Please take immediate action.

Sms Alert

## VI. RESULTS AND DISCUSSION

The implementation of the real-time fire detection system using OpenCV and Python yielded promising results. Through extensive testing, the system demonstrated efficient detection of fire and smoke in various environments, achieving a high level of accuracy and reliability. The utilization of computer vision techniques and machine learning algorithms enabled the system to effectively differentiate between normal scenes and fire-related incidents, minimizing false alarms.

Discussion: The successful deployment of the system showcases its potential to significantly enhance fire safety measures in residential, industrial, and forest environments. However, further optimizations are required to address challenges such as occlusions, varying lighting conditions, and real-world deployment considerations. Additionally, future research could explore the integration of additional sensors, such as thermal imaging and gas detection, to further enhance the system's capabilities. Overall, the results indicate a promising avenue for leveraging technology to mitigate the impact of fire incidents and protect lives and property

## VII. CONCLUSION

The paper introduces the proposed real-time fire detection system utilizing OpenCV and Python presents a promising solution for enhancing fire safety measures in various environments. By leveraging computer vision techniques and mine learning algorithm and mechanism, the system offers timely detection of fire or smoke, facilitating swift response anditigation efforts. Its affordability, scalability, and integration capabilities make it a valuable asset for both residential and industrial settings. As technology continues to evolve, further advancements in accuracy, integration with IoT platforms, and predictive analytics can further strengthen the system's effectiveness in safeguarding lives and property against the threat of fire.

## VIII. FUTURE SCOPE

Enhanced Accuracy: Further refinement of machine learning models and algorithms can improve the accuracy of fire detection, reducing false positives and false negatives.

Multi-sensor Integration: Integration with additional sensors such as thermal cameras and gas sensors can enhance the system's capabilities for early detection and identification of fire hazards..

IoT Integration: Integration with Internet of Things (IoT) platforms can enable remote monitoring and control of fire detection systems, enhancing scalability and accessibility

Cloud-based Solutions: Implementation of cloud-based solutions can facilitate real-time data processing, analysis, and storage, enabling seamless integration with other smart building systems and services.

Edge Computing: Utilizing edge computing techniques can enhance the efficiency of fire detection systems by performing data processing and analysis locally, reducing latency and bandwidth requirements. .

## REFERENCES

[1] Zhang, Z., Zhang, C., Li, Q., & Sun, S. (2019). A Survey of Recent Advances in Fire Detection using Computer Vision Techniques. arXiv preprint arXiv:1909.1016

[2] Sharma, P., Bhatia, R., & Gupta, S. (2020). Fire Detection System Using Machine Learning Algorithms: A Review. In Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (pp.463-473). Springer, Singapore.

[3] Prasad, S., Kulkarni, V., Deshpande, S., & Waghmode, V. (2020). An Analysis of Fire Detection Techniques using Cpision and M Comuter Vachine Learning Algorithms. In Proceedings of the International Conference on Advances in computing and Data Sciences (pp. 251-258). Springer, Singapore

[4] OpenCV documentation: https://docs.opencv.org/

[5] Scikit-learn documentation: https://scikit-learn.org/stable/documentation.html

[6] Chowdhury, M. E. H., & Rahaman, M. M. (2020). A Comprehensive Review of Fire Detection Systems Using Image Processing Techniques. SN Computer Science, 1(5), 1-20.

DOI: 10.48175/IJARSCT-17886

ISSN 2581-9429 IJARSCT