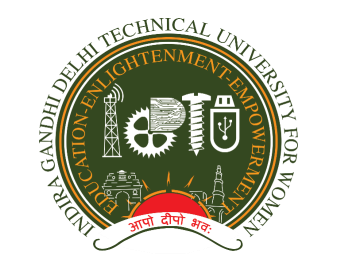
**Indira Gandhi Delhi Technical**

**University for Women**

**(Established by Govt. of Delhi vide Act 09 of 2012)**

**Kashmere Gate, Delhi–110006**

****

**SYNOPSIS**

**MAJOR PROJECT**

**MCA (IT), Semester – 4**

**TITLE OF PROJECT : REAL TIME FIRE DETECTION USING YOLO**

**Submitted By: Submitted To:**

**Ambika Solanki (00404092023) Dr. Kamal Kumar**

**Komal Pandey (03104092023)**

**Pallavi Patne (04404092023)**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S NO.** | **TOPIC** | **PAGE NO** |
| **1** | **TITLE** | **3** |
| **2** | **INTRODUCTION** | **4-6** |
| **3** | **OBJECTIVES** | **7** |
| **4** | **SCOPE OF THE PROJECT** | **8-9** |
| **5** | **EXISITING SYSTEM** | **10-11** |
| **6** | **PROPOSED SYSTEM** | **12-13** |
| **7** | **METHODOLOGY** | **14-18** |
| **8** | **HARDWARE/SOFTWARE REQUIREMENTS** | **19-20** |
| **9** | **SYSTEM ARCHITECTURE** | **21-22** |
| **10** | **MODULES** | **23** |
| **11** | **TECH STACK** | **24** |
| **12** | **EXPECTED OUTCOMES** | **25** |
| **13** | **CONCLUSION** | **26-27** |
| **14** | **FUTURE SCOPE** | **28-29** |
| **15** | **REFERENCES** | **30** |

1. **TITLE**

**REAL-TIME FIRE AND SMOKE DETECTION USING YOLO**

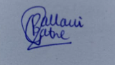
***Under the Guidance of:***

***Dr. Kamal Kumar (Mentor)***

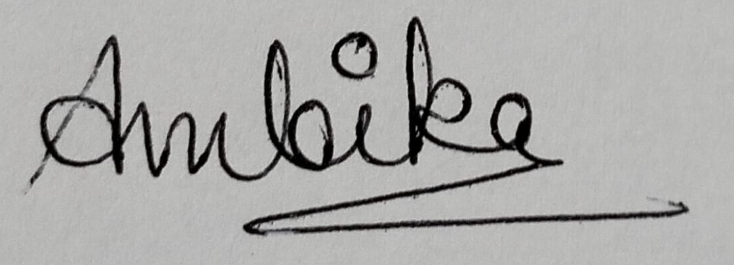
**Project Team:**

****

**Komal Pandey(03104092023)**

****

**Pallavi Patne(04404092023)**

****

**Ambika Solanki(00404092023)**

1. **INTRODUCTION**



Fire outbreaks remain one of the most devastating natural and man-made disasters, posing serious threats to human life, property, and the environment. Whether occurring in forests, industrial zones, or densely populated urban areas, fires can spread rapidly and result in catastrophic consequences. Traditional fire detection systems—such as heat sensors, gas detectors, or manual surveillance—often suffer from critical limitations. These include delayed detection, limited spatial coverage, susceptibility to false alarms, and a dependency on human intervention. Such shortcomings can severely hamper timely emergency response, leading to increased risk, greater damage, and significant financial losses.

In light of these challenges, there is an urgent need for a smarter, faster, and more reliable solution. With the rapid advancement in artificial intelligence and computer vision technologies, deep learning models have emerged as powerful tools for real-time object detection and classification tasks. Leveraging these capabilities, this project focuses on developing an intelligent and automated fire and smoke detection system using YOLO (You Only Look Once) object detection algorithms, particularly YOLOv5 and YOLOv8.

YOLO models are well-known for their high accuracy, speed, and efficiency in detecting multiple objects in real-time within a single pass of the neural network. By training these models on a diverse dataset of fire and smoke images, our system is capable of identifying potential fire hazards in both indoor and outdoor environments with high precision. Unlike conventional sensor-based systems, the YOLO-based detection approach processes visual data directly, enabling early and accurate detection even in the initial stages of a fire when it might not trigger traditional sensors.

The primary objective of this project is to create a robust early warning mechanism that can significantly enhance situational awareness, shorten response times, and ultimately save lives and property. In recent years, the increasing frequency of wildfires and industrial accidents has highlighted the urgent necessity for smarter surveillance and disaster mitigation strategies. As urbanization expands and climate change intensifies, fire hazards are becoming more unpredictable and widespread. Early detection is not just a matter of minimizing economic loss—it is also crucial for protecting biodiversity, preventing long-term environmental degradation, and ensuring public safety. Traditional systems fall short, especially in remote areas where human supervision is limited or entirely absent.

In contrast, a deep learning-based solution can operate autonomously, 24/7, analysing live video feeds from CCTV cameras or drones, and instantly flagging suspicious fire or smoke activity. This real-time functionality makes it possible to alert emergency services at the earliest possible stage, allowing for quicker containment and evacuation measures. Moreover, as YOLO models are lightweight and computationally efficient, they can be deployed on edge devices and embedded systems, making them suitable even for resource-constrained environments such as rural zones or mobile surveillance units. This level of scalability and adaptability reinforces the relevance and potential impact of our project in creating safer, more responsive communities. The integration of deep learning-based fire detection into existing surveillance infrastructure could revolutionize how fire safety is managed, offering an affordable and scalable solution for a wide range of applications—from smart cities and industrial automation to environmental monitoring and disaster management.

1. **OBJECTIVES OF THE PROJECT**

1) FIRE DETECTION USING YOLO V5

2)IMPLEMENTATION OF FIRE DETECTION

3)REVIEW AND ANALYSIS

4)PROCEDURE

5)COMPARISON AND PAPER WRITING

1. **SCOPE OF THE PROJECT**

This project focuses on developing an intelligent, real-time fire and smoke detection system using deep learning-based object detection techniques—specifically YOLOv5 and YOLOv8. The primary goal is to detect the presence of fire and smoke in visual data, enabling early intervention in potentially hazardous situations. The system is designed to work with both images and video streams, making it suitable for integration with existing surveillance infrastructure such as CCTV cameras or drone footage.

Key features within the scope include:

1) Training and fine-tuning YOLOv5 and YOLOv8 models on a curated dataset containing various fire and smoke scenarios.

2) Evaluating and comparing both models using standard performance metrics such as precision, recall, mAP@0.5, and [mAP@[0.5:0.95](mailto:mAP@[0.5:0.95)].

3) Real-time detection capability to simulate practical use cases in industrial, urban, and forest environments.

4) Visualization of detection results, including bounding boxes and confidence scores on test media.

5) A comparative performance report to determine which version of YOLO is more effective in different environmental conditions.

Features excluded from the current scope are:

1) Hardware integration for automatic suppression systems (e.g., activating sprinklers or alarms).

2) Mobile application development or remote alert notification services (e.g., SMS or email alerts).

3) Multilingual or voice-based interfaces for system interaction.

4) Detection of other hazards like gas leaks, intrusions, or structural damage.

The project is primarily academic and research-focused, aiming to demonstrate the effectiveness and feasibility of deep learning models in critical safety applications. However, the architecture is scalable and can be extended in future work to support edge deployment, IoT integration, or automated emergency response mechanisms.

1. **EXISTING SYSTEM AND LIMITATIONS**

Traditional fire detection systems currently in use include smoke detectors, thermal sensors, and CCTV surveillance with manual monitoring. While these systems are commonly installed in indoor environments such as homes, offices, and warehouses, they have several drawbacks that limit their effectiveness, particularly in large-scale or outdoor settings.

Smoke detectors operate using optical or ionization sensors to identify smoke particles in the air. Though effective in enclosed spaces, they are limited in scope and often fail to detect early signs of fire in open or ventilated areas. Additionally, they are prone to false alarms triggered by non-fire-related elements like dust or steam.

Thermal sensors detect variations in temperature or infrared radiation and are widely used in industrial settings. However, they are expensive to deploy across large areas and may not detect smoke unless accompanied by flames. Moreover, these sensors often struggle to distinguish between actual fires and harmless heat sources, leading to reduced accuracy.

CCTV systems monitored by human operators are another common approach. These systems rely heavily on human attention and judgment, which introduces the possibility of human error or fatigue. They are also not practical for monitoring expansive areas in real time, often resulting in delayed response during emergencies.

Overall, these traditional systems suffer from limitations in terms of coverage, accuracy, scalability, and response time. Their dependence on confined spaces, manual operation, or expensive hardware makes them insufficient for modern, dynamic environments like smart cities or forests.

To overcome these challenges, this project proposes an advanced solution called the Smart Fire Detection System (SFDS), which is based on deep learning using the YOLOv8 algorithm. The system leverages real-time video analysis to identify fire and smoke with high accuracy. Compared to traditional methods, SFDS offers quicker detection, fewer false alarms, and the flexibility to be deployed in diverse environments.

The proposed system is designed within the architecture of a smart city, consisting of four functional layers. The **Application Layer** focuses on practical implementations of SFDS in locations like government buildings, public areas, hospitals, and highways. The **Fog Layer** serves as a real-time processing zone between IoT devices and the cloud, reducing latency and improving data security. The **Cloud Layer** handles large-scale data storage and analysis, offering scalability and reliability. Finally, the **IoT Layer** includes sensors, actuators, and other devices that collect and transmit real-time environmental data to the system.

By combining deep learning with modern smart city infrastructure, the SFDS model aims to overcome the inefficiencies of traditional systems and provide a scalable, cost-effective, and highly responsive fire detection solution.

1. **PROPOSED SYSTEM**

The proposed system addresses the limitations of traditional fire detection methods by introducing an intelligent, real-time fire and smoke detection framework powered by deep learning. It eliminates the dependency on manual monitoring and hardware-based sensors by utilizing advanced computer vision techniques. This system is designed to operate efficiently in various environments including indoor settings, outdoor areas, and forest regions, ensuring early detection and prompt response.

The core of the system is based on two state-of-the-art object detection models—YOLOv5 and YOLOv8—which are known for their high speed and accuracy. Both models were trained and tested on a carefully curated and labeled fire and smoke dataset obtained from Roboflow. The dataset includes a diverse range of fire scenarios to ensure that the models generalize well across different conditions.

Bounding box annotations were used to help the models accurately identify and localize fire and smoke in both images and video streams. Once trained, the models were evaluated on key performance metrics such as accuracy (measured by mean Average Precision or mAP), speed (Frames Per Second or FPS), and detection quality (Precision and Recall). This comparative analysis helped determine the strengths and weaknesses of each version of YOLO in real-world fire detection tasks.

One of the key enhancements of the proposed system is its ability to generate real-time alerts and visual overlays on live video feeds, highlighting areas where fire or smoke is detected. This allows for faster situational awareness and can assist emergency personnel in responding more effectively. Unlike conventional systems, this deep learning-based approach offers a scalable and automated solution that can be integrated into smart city infrastructure, surveillance systems, or even drone-based monitoring units.

In summary, the proposed system not only improves detection speed and accuracy but also reduces false alarms and enhances adaptability across a variety of fire-prone environments.

1. **METHODOLOGY**

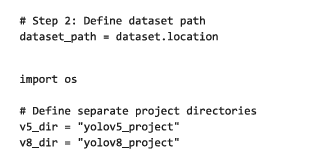
The methodology used in this project follows a structured approach for deep learning model development, incorporating iterative techniques from the Agile methodology to ensure flexibility and continuous improvement throughout the process. The project is divided into several major phases, each addressing a key component of the fire and smoke detection system.

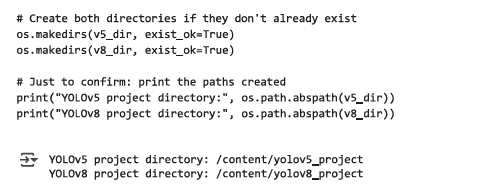
1. **Dataset Preparation:**

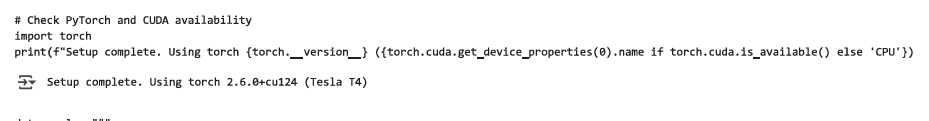
****

The initial phase involved collecting a comprehensive dataset from Roboflow, consisting of approximately 3900 annotated images of fire, smoke, and other objects. The dataset covers various scenarios such as indoor, outdoor, and forest fires, including real-world and synthetic images to enhance model robustness. The images were annotated using the YOLOv8 format and split into training, validation, and test sets to evaluate model performance effectively.

1. **Model Selection:**

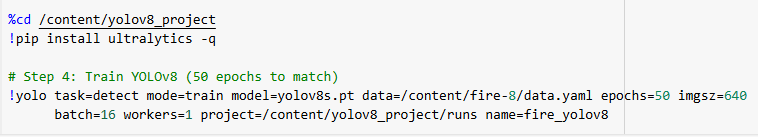
****



  
The fire and smoke detection system was developed using the PyTorch framework, which supports both YOLOv5 and YOLOv8 object detection models. These models are well-known for their speed and accuracy, making them ideal choices for real-time fire detection tasks. Pre-trained weights were utilized via transfer learning to leverage the models’ knowledge from large-scale datasets, helping the models adapt quickly to fire and smoke detection.

1. **Training:**





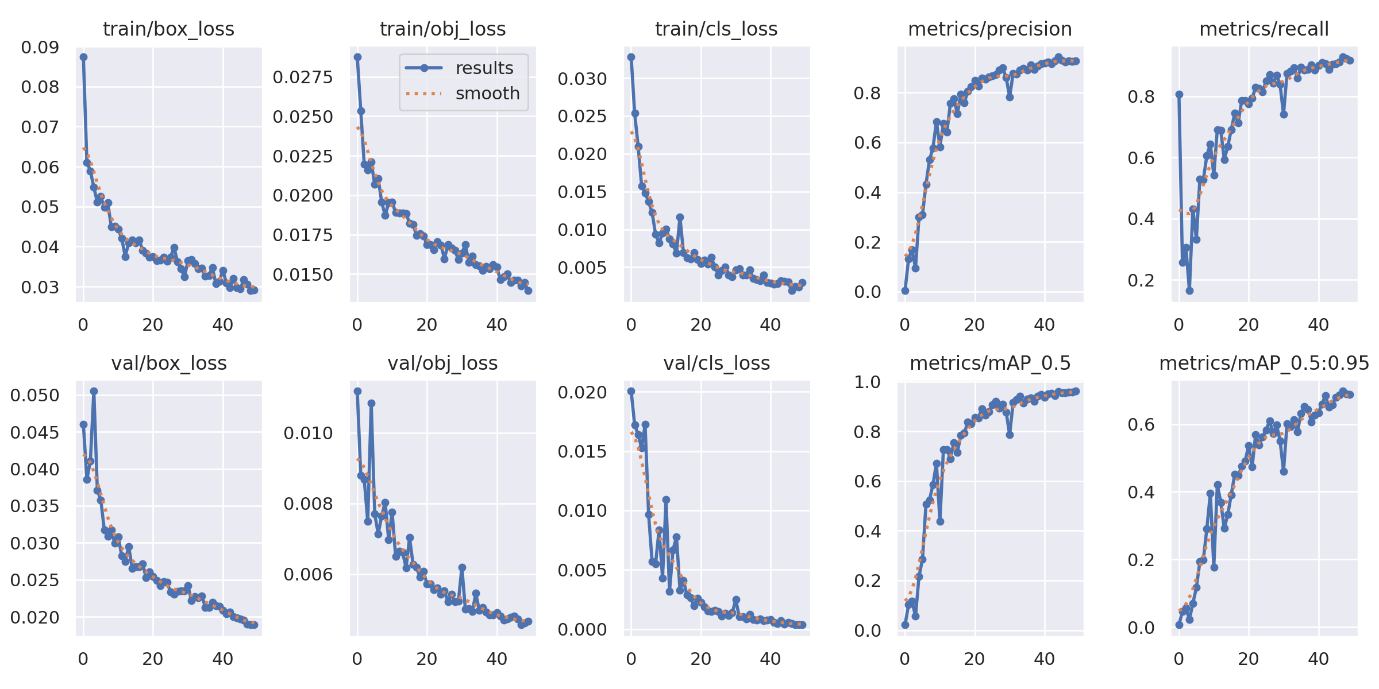
The models were trained with the following settings:

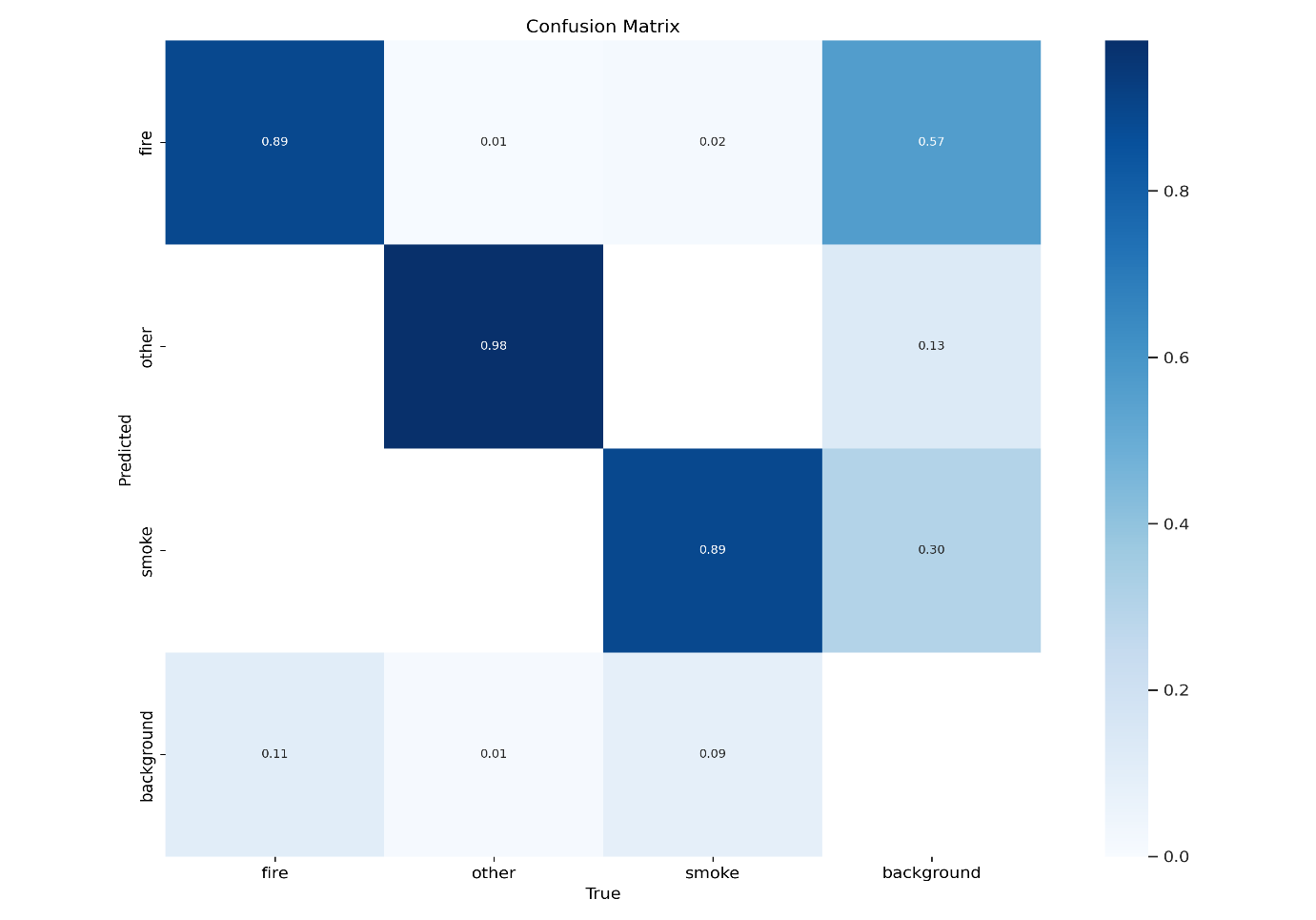
* Epochs: 3
* Batch Size: 16
* Image Size : 640×640 pixels

Data augmentation techniques were employed to enhance the variety of input data and prevent overfitting. These techniques included horizontal and vertical flipping, rotation, brightness/exposure adjustment, blur, shear, crop, and noise. The combination of these augmentations helped to generalize the model's performance in diverse conditions.

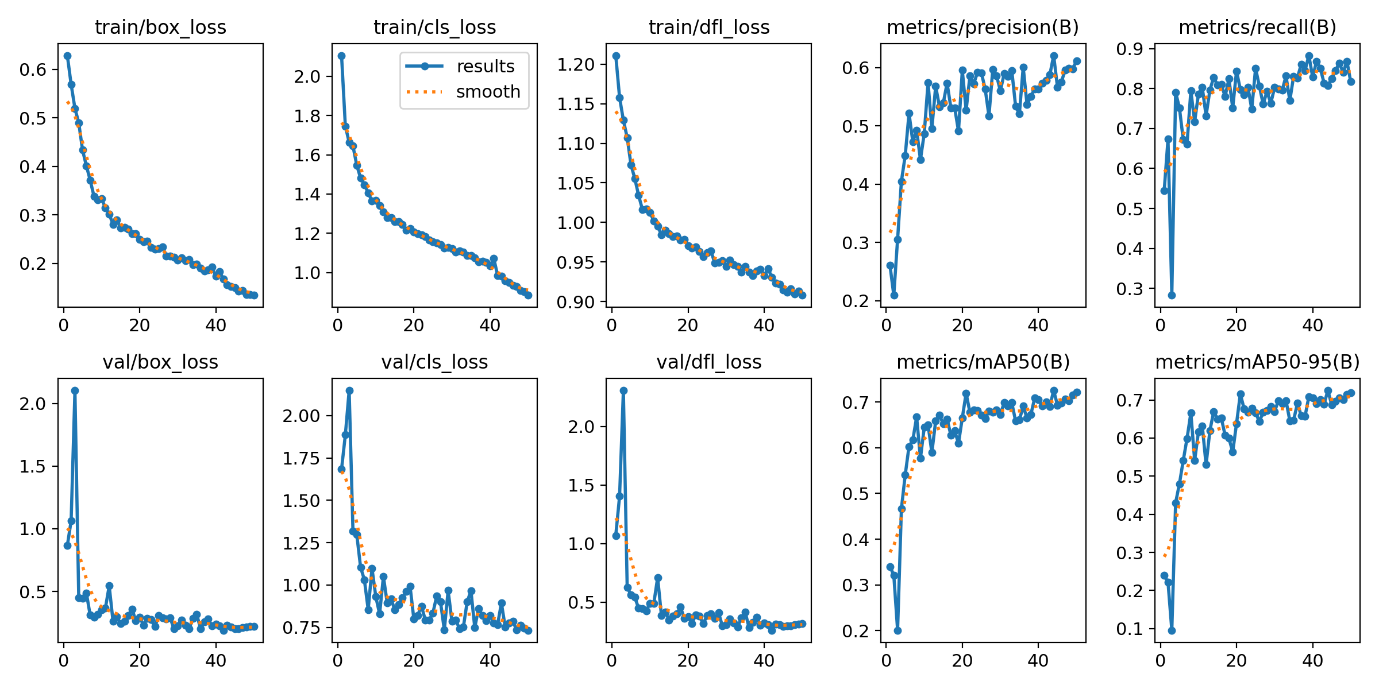
1. **Evaluation:**

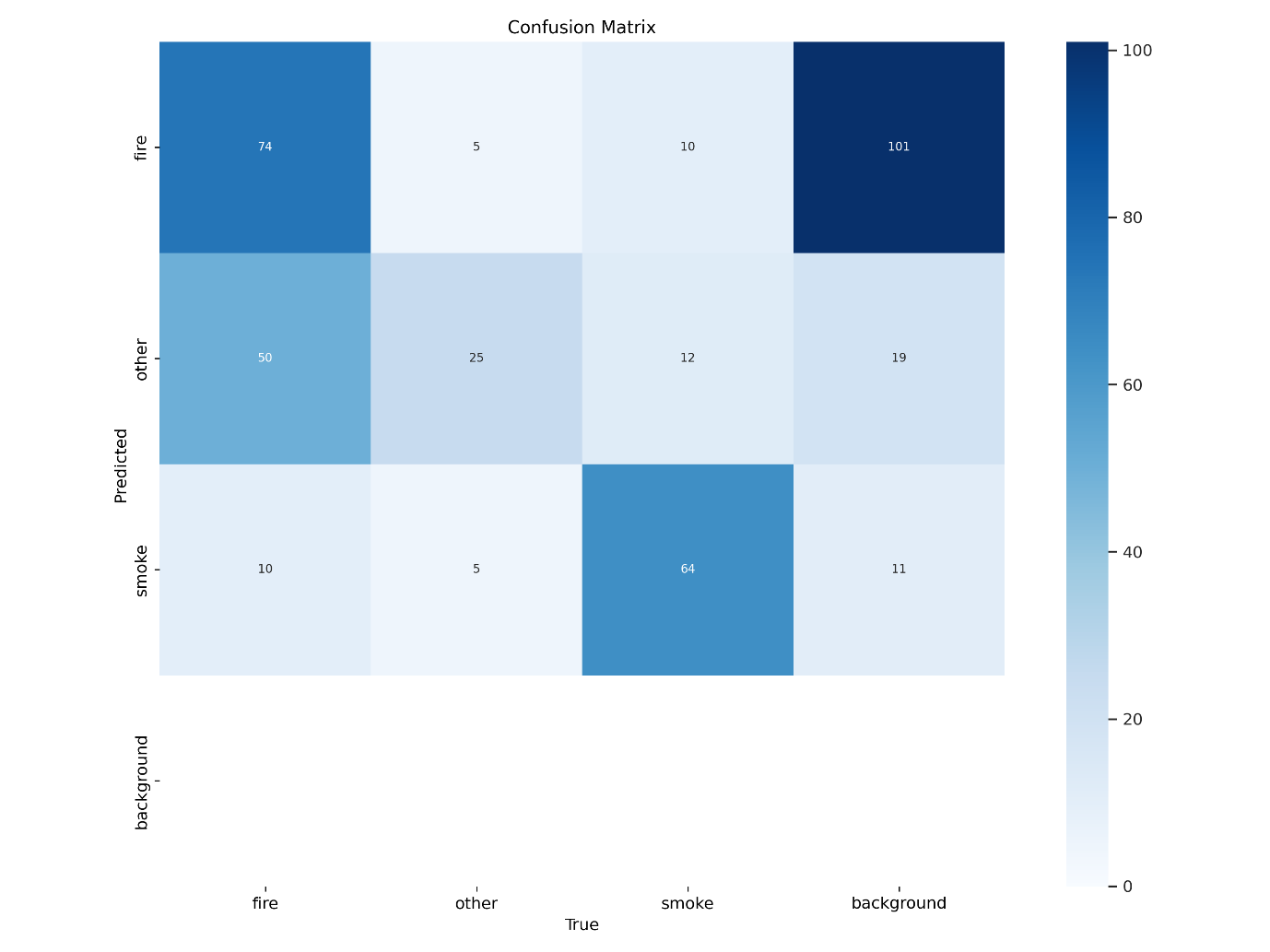
**YOLO 5**

****

****

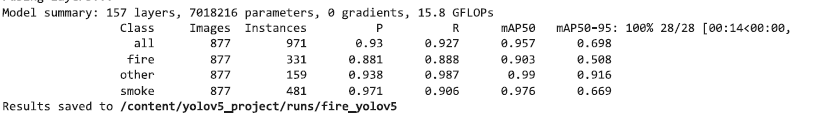
**YOLO 8**

****

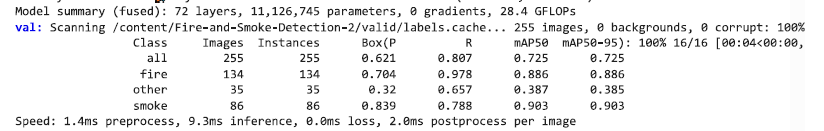
  
The models' performance was evaluated using key metrics such as mean Average Precision (mAP), precision, and recall, which helped assess their effectiveness in detecting fire and smoke. Visual validation was performed on the test images and videos to confirm that the models were detecting the fire and smoke regions accurately.

1. **Inference:**

**YOLO 5**

****

**YOLO 8**

****

Once trained, the models were deployed for inference on test videos and real-world surveillance footage. The system generated bounding boxes around detected fire and smoke regions in the videos, providing real-time visual feedback.

**6.Final Deployment:**

The final system is capable of processing live video feeds to

generate real-time alerts and visual overlays, significantly improving the speed and accuracy of fire detection in real-world scenarios. This makes it suitable for deployment in various environments such as urban areas, industrial zones, and forest regions.

** **

**7. HARDWARE AND SOFTWARE REQUIREMENTS**

**Hardware Requirements:**

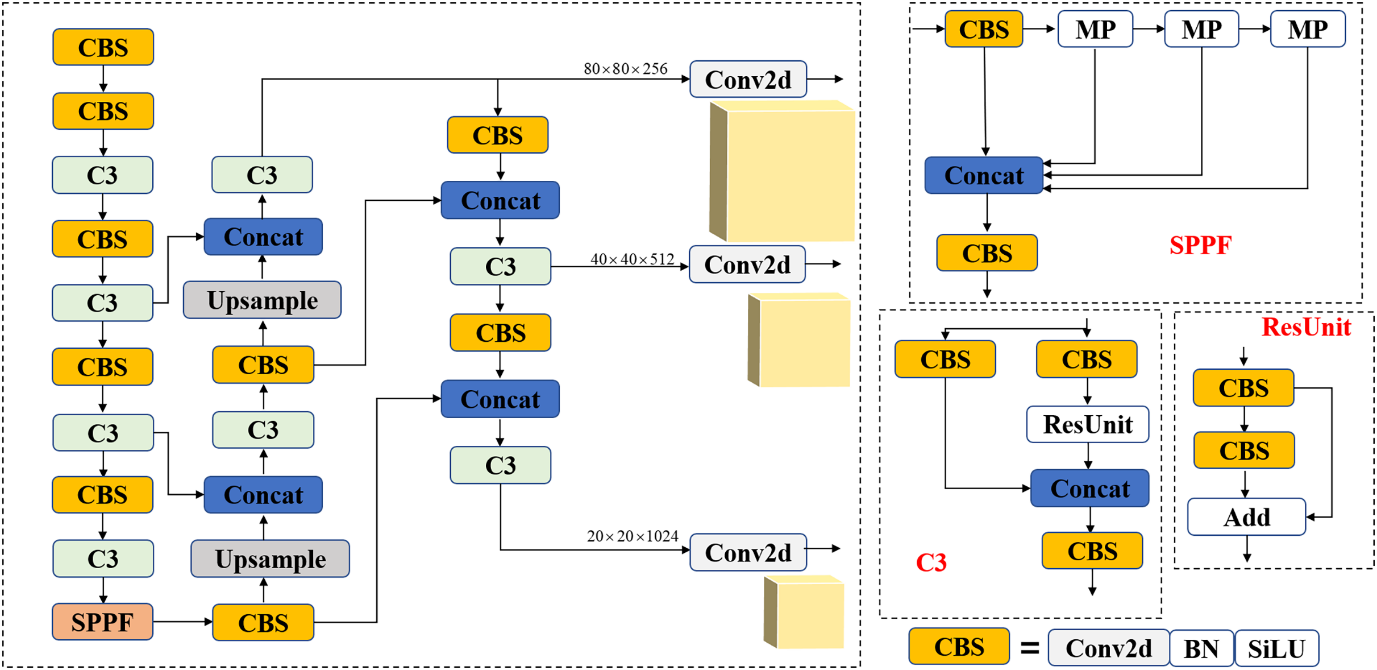
* Processor: A multi-core processor (e.g., Intel i7 or AMD Ryzen 7) is essential to handle the intensive computations required during training and inference phases of the deep learning models. A GPU is highly recommended for faster processing.
* RAM: A minimum of 16 GB of RAM is suggested to ensure smooth training, particularly when handling large datasets. More RAM can enhance the performance when processing real-time video streams.
* Storage: Sufficient storage space (at least 50 GB) is needed to store the dataset, model weights, and logs from training. An SSD is preferred to speed up data access and model loading times.
* Graphics Processing Unit (GPU): A high-performance GPU such as NVIDIA GeForce GTX 1660 or better (e.g., RTX 2060, 3060, or Tesla GPUs) is crucial for training deep learning models efficiently.

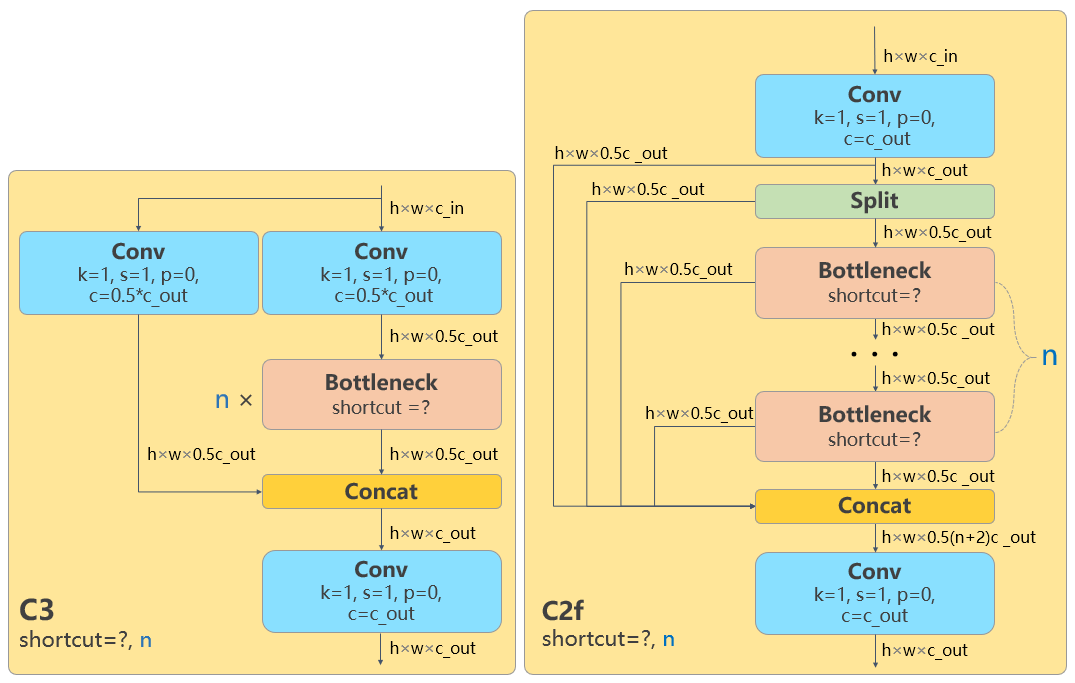
**Software** **Requirements**:

* Operating System: **Windows 10** (Development environment)  
  *Note: The project is platform-independent and can be run on other operating systems such as Linux, macOS, or cloud environments with compatible hardware and software setups.*
* Programming Languages: Python 3.11.12 is used for model development and implementation due to its rich support for machine learning libraries and frameworks.
* Development Environment: Google Colab is utilized for model development and training. It provides a cloud-based environment with access to free GPUs, facilitating the rapid training of deep learning models.
* Libraries & Frameworks:
  + PyTorch for model building and training (specifically for YOLOv5 and YOLOv8).
  + OpenCV for handling image/video processing.
  + Roboflow for dataset generation and augmentation.
  + NumPy and Pandas for data manipulation and handling.
  + Matplotlib and Seaborn for visualizations.
* Version Control: Git for version control to track code changes and collaboration.

1. **SYSTEM ARCHITECTURE**

**YOLO V5**



**YOLO V8**

The system architecture for this fire and smoke detection solution can be broken down into four main layers:

1. Input Layer (Image/Video Feed):
   * The system receives images or video streams, either from cameras in the environment or uploaded data. The input could be live surveillance footage or pre-recorded video files.
2. Processing Layer (Deep Learning Model):
   * The core of the system, where the trained YOLOv5 and YOLOv8 models process the images or video frames. This layer utilizes object detection to identify and localize fire and smoke in the data stream.
   * The models output bounding boxes around the detected fire and smoke regions.
3. Alerting and Visualization Layer:
   * Once fire or smoke is detected, this layer generates real-time alerts and visual overlays (bounding boxes) on the video. The overlay helps identify the affected areas and trigger necessary actions.
4. Output Layer (Emergency Action/Logs):
   * If fire or smoke is detected, real-time alerts (either through visual cues or notifications) are sent to the relevant authorities. Logs of detection and alert activities are also stored for future reference or analysis.

**9. MODULES OF THE PROJECT**

1. Data Collection Module:
   * This module is responsible for collecting and annotating fire and smoke images. The dataset is sourced from Roboflow, where the data is categorized into fire, smoke, and other classes and converted into the YOLO format.
2. Model Training Module:
   * This module handles the training of YOLOv5 and YOLOv8 models. It performs preprocessing, data augmentation, and training using PyTorch. The module also monitors model performance through metrics such as accuracy, precision, recall, and mAP.
3. Model Evaluation Module:
   * After training, the models are evaluated on the validation and test sets. This module compares the performance of YOLOv5 and YOLOv8 based on various criteria such as mAP, FPS, precision, and recall.
4. Inference and Detection Module:
   * This module is responsible for real-time inference. It processes live video or test images, applies the trained models, and detects fire and smoke in the footage. Bounding boxes are generated and visualized on the detected areas of interest.
5. Alerting and Notification Module:
   * The system generates real-time alerts when fire or smoke is detected. Notifications are sent to the relevant emergency teams or system operators, and visual alerts are displayed on the monitoring interface.

**10. TECHNOLOGY STACK**

The technology stack for this project includes the following:

* Programming Languages: Python 3.x for model training and processing.
* Deep Learning Frameworks: PyTorch for YOLOv5 and YOLOv8 model implementation.
* Cloud Platform: Google Colab for cloud-based computation with access to GPU acceleration.
* Image Processing Library: OpenCV for reading and processing video frames.
* Dataset Management: Roboflow for collecting and annotating images.
* Version Control: Git for source code management and version control.
* Visualization: Matplotlib and Seaborn for plotting training results and visualizing model performance.

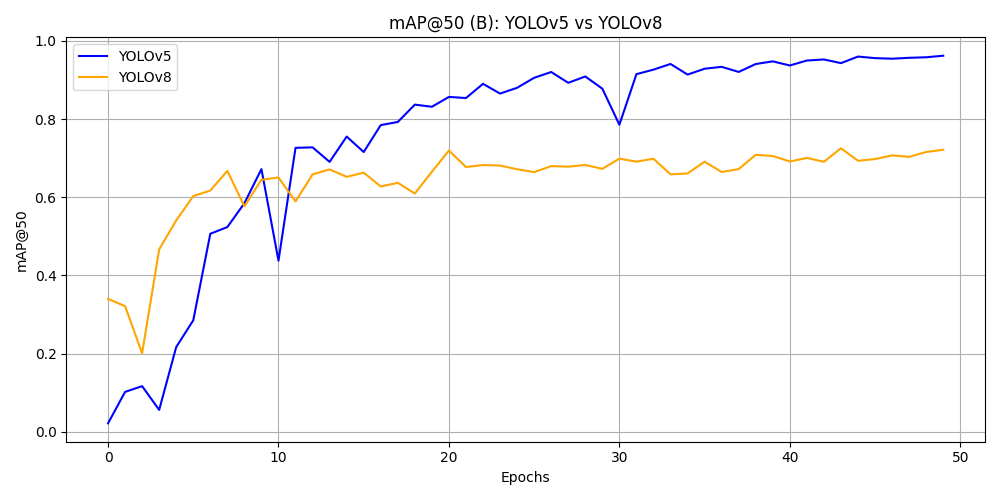
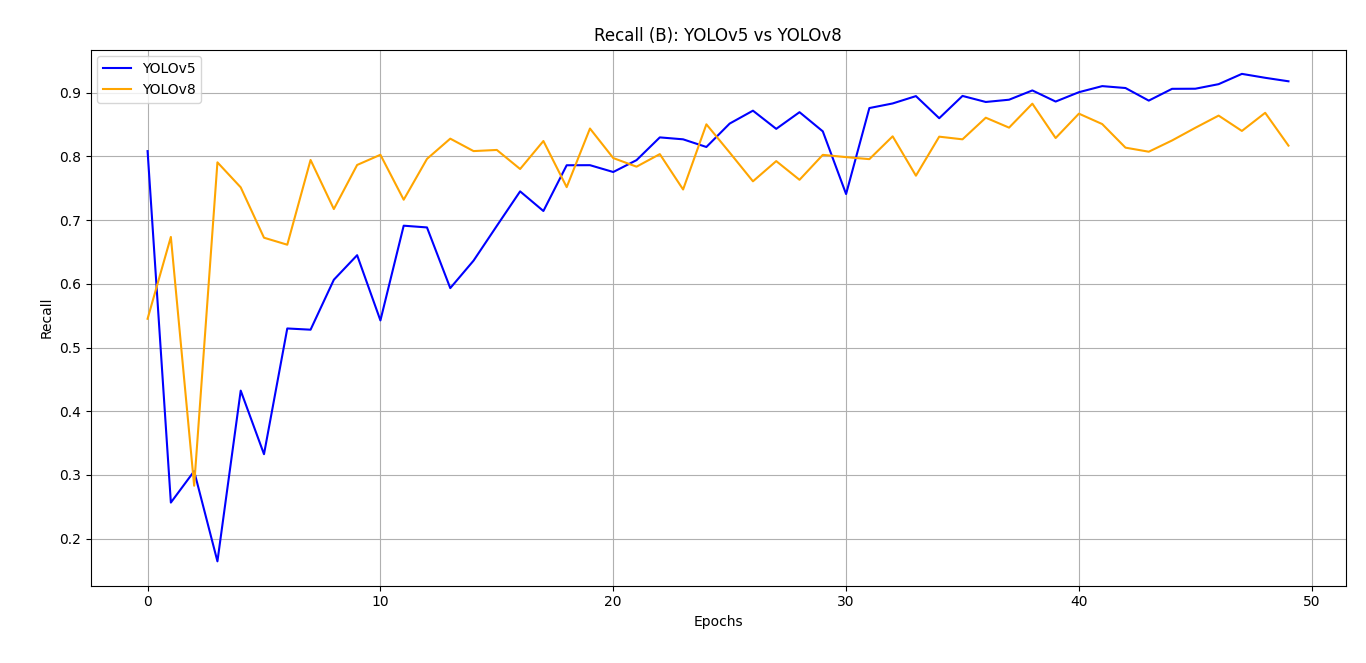
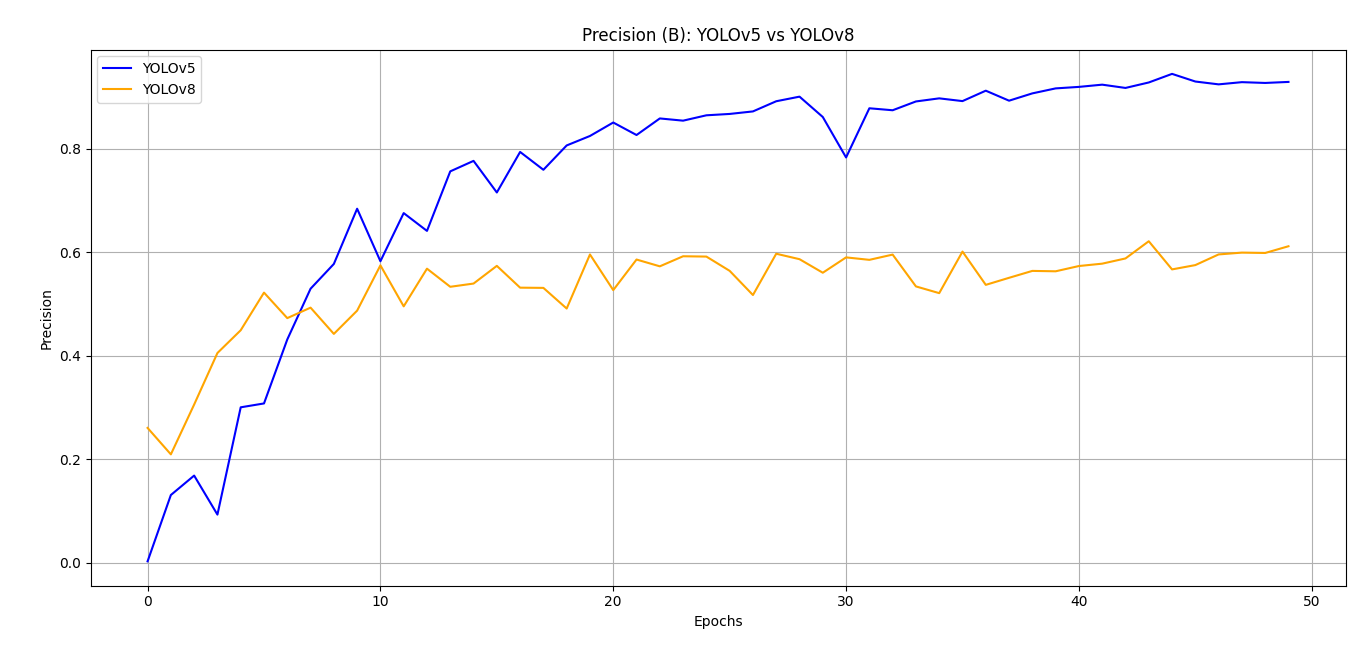
**11. EXPECTED OUTCOMES**

The expected outcomes of this project are as follows:

1. Improved Fire Detection Accuracy:
   * The system is expected to significantly enhance fire detection capabilities with minimal false alarms. YOLOv5 and YOLOv8 models should offer high detection accuracy, with precise identification of fire and smoke in various environments.
2. Real-Time Alert System:
   * The system will generate real-time alerts and visual overlays, providing immediate feedback for emergency responders. This will help improve response times and potentially reduce the impact of fires on lives and property.
3. Scalability and Flexibility:
   * The fire detection system should be scalable, capable of being deployed in various environments, including homes, industrial zones, and large outdoor areas (such as forests).
4. Comparison of YOLOv5 and YOLOv8 Models:
   * A detailed performance comparison of YOLOv5 and YOLOv8 based on accuracy, speed, and computational efficiency will be made, providing valuable insights into the strengths and limitations of each model for fire detection tasks.

**12. CONCLUSION**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| METRIC | MODEL | PRECISION | RECALL | MAP-50 | MAP-50-95 |
| FIRE | **YOLO V5** | **0.881** | **0.888** | **0.903** | **0.508** |
|  | **YOLO V8** | **0.704** | **0.978** | **0.886** | **0.886** |
| SMOKE | **YOLO V5** | **0.971** | **0.906** | **0.976** | **0.669** |
|  | **YOLO V8** | **0.839** | **0.788** | **0.903** | **0.903** |
| OTHER | **YOLO V5** | **0.938** | **0.987** | **0.99** | **0.916** |
|  | **YOLO V8** | **0.32** | **0.657** | **0.387** | **0.385** |
| OVERALL | **YOLO V5** | **0.93** | **0.927** | **0.957** | **0.698** |
|  | **YOLO V8** | **0.621** | **0.807** | **0.725** | **0.725** |

****

This fire and smoke detection system, built using state-of-the-art YOLOv5 and YOLOv8 models, addresses the critical limitations of traditional fire detection systems. By leveraging deep learning, the system provides an automated, real-time solution that can detect fire and smoke with high precision. It is expected to improve response times in emergency situations, reduce the impact of fires, and be applicable across a variety of environments, from homes and industrial zones to large outdoor areas. This innovative system has the potential to be a game-changer in fire safety and emergency management.

**13. FUTURE SCOPE**

The fire and smoke detection system developed in this project holds promising potential for future enhancements. These improvements aim to expand the system's capabilities and efficiency in various scenarios:

1. Model Improvement:

Future work can focus on extending the model training with more epochs and performing hyperparameter tuning to improve the accuracy of fire and smoke detection. This can also involve experimenting with different backbones or architectures to further optimize the model's performance.

1. Real-Time Deployment:

The system can be integrated with live surveillance systems to provide real-time fire and smoke detection. This would involve establishing a seamless communication channel with camera systems or sensor networks in different environments to ensure that fires are detected immediately.

1. Edge Optimization:

To enable deployment in resource-constrained environments, such as low-power devices (e.g., drones, IoT sensors), future improvements can focus on model compression. This would reduce the model size and improve inference speed without sacrificing detection accuracy.

1. Dataset Expansion:

To further enhance the robustness of the system, the dataset can be expanded to include more varied real-world data, such as images from different geographical regions, weather conditions, and lighting. A more comprehensive dataset would improve the model's ability to generalize to new and unseen environments.

1. Multi-Hazard Detection:

The system's detection capabilities can be extended to other safety hazards, such as gas leaks, structural failures, and even flooding. This would increase the versatility of the system and make it an integral part of a smart city’s safety infrastructure.

**14. REFERENCES**

1. Talaat, Fatma M., and Hanaa ZainEldin. "An Improved Fire Detection Approach Based on YOLO-v8 for Smart Cities."
2. Roboflow, Fire & Smoke Detection Dataset.
3. Edmundo Casas , Leo Ramos , Eduardo Bendek and Francklin Rivas-Echeverria 1. “YOLOv5 vs. YOLOv8: Performance Benchmarking in Wildfire and Smoke Detection Scenarios”
4. Roboflow Dataset (https://universe.roboflow.com/data2/fire-smoke-detection-ua3dm)
5. PyTorch Documentation. (2021). PyTorch.
6. <https://github.com/Abonia1/YOLOv8-Fire-and-Smoke-Detection>
7. <https://github.com/spacewalk01/yolov5-fire-detection>
8. *Ultralytics: The YOLO AI company*. <https://ultralytics.com>
9. Glenn Jocher, & Ultralytics. (2020). *YOLOv5: The next-generation YOLO* [Software]. <https://github.com/ultralytics/yolov5>
10. Ultralytics. (2023). *YOLOv8: The latest evolution of YOLO* [Software]. <https://github.com/ultralytics/yolov8>
11. <https://peerj.com/articles/cs-1830/>
12. https://mmyolo.readthedocs.io/en/latest/recommended\_topics/algorithm\_descriptions/yolov8\_description.html
13. https://www.researchgate.net/figure/The-architecture-of-YOLOv8-algorithm-model-The-overall-architecture-is-divided-into-four\_fig1\_377492266
14. OpenAI. (2023). *ChatGPT* (April 2023). <https://chat.openai.com>