Deep Learning System For Fire Detection And Alerts Using Yolo

Chaya P[1], Anu Priya[2], ChinmayiD[3], Lavanya S[4], Waiza Fathima[5]
[1]Assistant Professor,Dept of ISE
[2, 3, 4, 5]Dept of ISE
[1, 2, 3, 4, 5] GSSSIETW,Mysuru,India

Abstract- *This project introduces a real-time fire detection and alert system based on the YOLO algorithm, aiming to enhance safety in various settings. Designed to mitigate fire hazards across diverse environments, including industrial, residential, and public spaces. Leveraging the YOLO deep learning algorithm, the system accurately identifies fire from live camera feeds and triggers immediate responses through integrated hardware. On detecting fire, it activates an alarm, captures evidence, emails alerts, and initiates a water pump via an ESP32 microcontroller. This AI-powered system eliminates the need for traditional, maintenance-heavy sensors and instead offers a scalable, cost-effective alternative that integrates seamlessly with existing surveillance setups. By ensuring rapid detection and response, it enhances environmental safety and operational efficiency.*

Keywords- Fire Detection, Object Detection, YOLO

I. INTRODUCTION

Fires endanger people, property, and ecosystems, particularly in crowded or industrial zones, particularly in industrial, residential, and densely populated areas. Conventional fire detection tools like heat or smoke detectors often respond slowly and can be impacted by environmental conditions, high maintenance requirements, and environmental constraints that hinder their effectiveness in real-time scenarios. Recent advances in computer vision and deep learning provide promising avenues to enhance fire safety through intelligent, automated systems capable of rapid and accurate hazard detection.

This research presents a real-time Fire Detection and Alert System that leverages the YOLO (You Only Look Once) object detection model in conjunction with embedded hardware for responsive fire suppression and remote notification. By integrating pre-trained deep learning models with video surveillance input, the system identifies fire instances with high precision and triggers immediate actions, including activating a water pump, sounding alarms, and emailing visual evidence to designated recipients. The system's low-cost, scalable design and compatibility with existing infrastructure make it an efficient and accessible solution for improving fire safety across diverse environments.

II. LITERATURE SURVEY

Literature Survey is most important step in the software development process. Before developing of development of intelligent safety systems, it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, the next step is to determine which operating system and language can be used in developing the system.

[1]IEEE Xplore, 2019 – Deep Learning for Fire Detection in Surveillance Videos In this work by M. Johnson and K. Lee, a VGG16-based deep learning model was used to detect fire in surveillance footage. With an accuracy of 91.3%, the system highlighted the strength of convolutional neural networks in real-time fire surveillance.However, it struggled with high computational overhead, leading to processing delays in high-resolution video feeds. Furthermore, the model lacked integration with any real-time fire response mechanisms.Despite its strong performance in static conditions, the system exhibited certain drawbacks. The computational complexity of the VGG16 model caused delays in processing high-resolution video feeds, rendering it less ideal for real-time scenarios.

[2] PLOS ONE, 2023 – Fire Detection Using YOLOv5 with Attention Mechanism In their work, Wang and colleagues improved YOLOv5 by incorporating attention layers, which helped in better focus during fire detection tasks, and dynamic anchor boxes. The system was trained on a mixed dataset of synthetic and natural fire images, leading to improved detection accuracy. However, the researchers noted that the model needed additional robustness for deployment in complex real-world environments, where background clutter and varying lighting conditions posed challenges.The model was trained on a combination of real-world and synthetically generated fire images, making it robust against diverse fire patterns and backgrounds. As a result, the system demonstrated a significant increase in precision and recall metrics compared to conventional YOLOv5.

[3] Springer Link, 2022 – Real-Time Fire Detection with YOLOv5s in Smart Cities Zhang et al. implemented a fire detection solution for smart city applications using a lightweight YOLOv5s model. They incorporated SE (Squeeze-and-Excitation) and SPPF (Spatial Pyramid Pooling Fast) modules for improved performance and reduced inference time. The system was successful in detecting urban fire hazards but faced limitations in resource-constrained environments due to its computational demands.

[4] IJCRT, 2022 – YOLOv4-Based Fire Detection for IoT-Integrated Systems Ahmad developed a fire detection system combining YOLOv4 with IoT technologies. The system provided real-time fire alerts and included basic suppression mechanisms.Despite performing well generally, the system occasionally misidentified reflections as fire, leading to inaccuracies. The authors emphasized the need for improved differentiation between fire and other light sources for practical reliability.

[5] Journal of Intelligent Systems, 2020 – Real-Time Fire Detection Using YOLOv3 for Early Fire SuppressionR. Kumar and M. Patel utilized YOLOv3 with a custom dataset tailored for industrial environments. The system was integrated with a fire suppression unit, allowing automated response upon detection. While the model performed well in real-time conditions, it showed difficulty identifying small fires or those obscured by smoke and fog, which reduced overall detection consistency.

III. METHODOLOGY

The methodology of the Fire Detection and Alert System Using YOLO involves a multi-stage process that integrates deep learning, real-time video processing, embedded control, and alert mechanisms to detect and respond to fire hazards efficiently. The system begins by capturing a live video feed through a connected webcam, which is broken into individual frames for analysis. These frames are then processed using OpenCV libraries and passed through a YOLOv3 deep learning model to identify the presence of fire. If fire is detected, the system activates multiple alert functions—including visual bounding boxes, voice alerts, buzzer signals, and an automatic water pump via ESP32 microcontroller. Additionally, the system captures an image of the fire incident and sends it via email for remote monitoring.

Deep Learning System for Fire Detection and Alerts system: real time fire detection with distance and voice alertsthe diagram illustrates the operational workflow

1. Web camera: The system initiates by capturing a continuous live video feed from the surroundings using a webcam. The cap.read() function in OpenCV reads each frame in real time. These frames are then resized and preprocessed for input into the YOLO model. This step ensures real-time responsiveness, allowing early hazard detection.

2. YOLO-Based Object Detection: The YOLOv3 model, trained on fire and non-fire datasets, analyzes each frame using net.forward() to detect potential fire. The model applies bounding boxes and confidence thresholds to isolate regions likely to contain flames. Detected outputs include object class (i.e., "fire"), position, and confidence score, all handled through the blobFromImage() and NMSBoxes() functions

3.Fire Classification and Confidence Thresholding: Detected objects are classified based on the YOLO model's class IDs. Only those with the label "fire" and a confidence level above a set threshold (e.g., 0.3) are considered valid. The argmax() function helps determine the object class with the highest likelihood. This ensures only true fire instances proceed to the next steps, reducing false alarms.

4. Bounding Box and Labeling: Detected fire instances are visually marked on the video feed using bounding boxes. The system uses class IDs to determine object type and overlays text labels (e.g., "Fire 0.87") along with confidence values. This step helps confirm detection and visualize the threat on screen.

5.ESP32 Microcontroller Activation: Upon detection of fire, a command is sent to the ESP32 microcontroller via serial or Bluetooth communication using the serial.write() function. The ESP32 then activates a 9–12V water pump connected through a relay module, initiating automatic fire suppression. Simultaneously, a buzzer is triggered to alert nearby personnel.

6.Image Capture and Email Notification: The system captures a snapshot of the detected fire frame using cv2.imwrite() and sends it via an SMTP-based email function. This email includes the image and relevant alert message to a preconfigured recipient, enabling remote monitoring and quick decision-making.

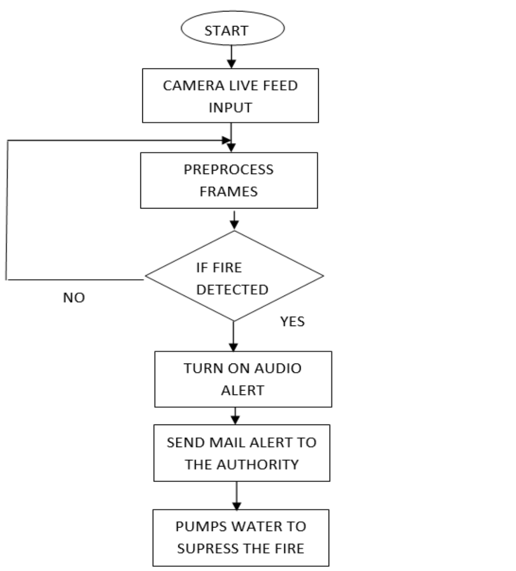
7.Voice Alert System:A voice alert is generated using text-to-speech functionality (pyttsx3) when fire is detected. The system speaks a pre-defined message such as "Fire detected!"

to audibly inform users in the vicinity. This is particularly helpful in noisy environments or when the video feed is not actively monitored.

8.False Positive Handling and Deactivation: If no fire is detected for a specific interval (e.g., 3 seconds), a command is sent to deactivate the pump and reset the alert system. This prevents unnecessary water usage and system fatigue. The logic for this is handled through a cooldown timer and conditional checks on detection frequency.

9. Water Pump Activation and Fire Suppression: Upon receiving the command, the ESP32 activates the water pump connected via a relay. The motorized system begins spraying water or fire suppressant to extinguish the fire automatically. This automation allows for early mitigation, especially in unmanned or remote environments.

10. Post-Detection Monitoring and Reset: After the fire is no longer detected in the frame for a fixed duration (e.g., 3 seconds), the system resets the flags and sends a new command (e.g., "2") to stop the pump and resume monitoring. This post-detection logic ensures the system doesn't trigger repetitive actions unnecessarily and is always ready for new events.



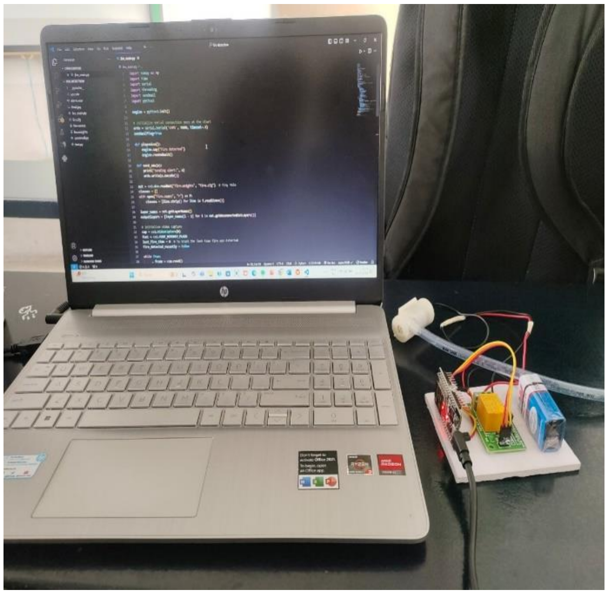
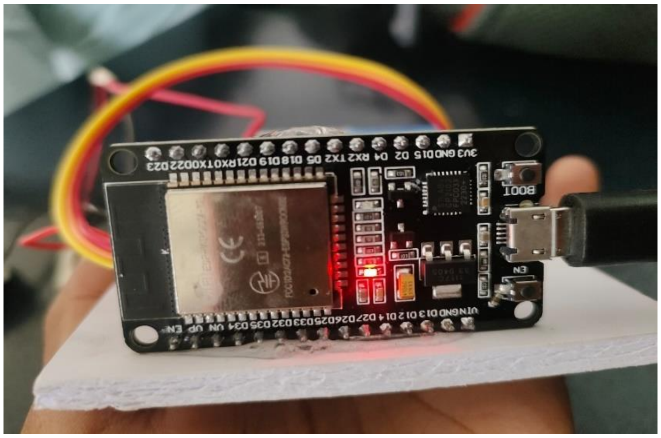
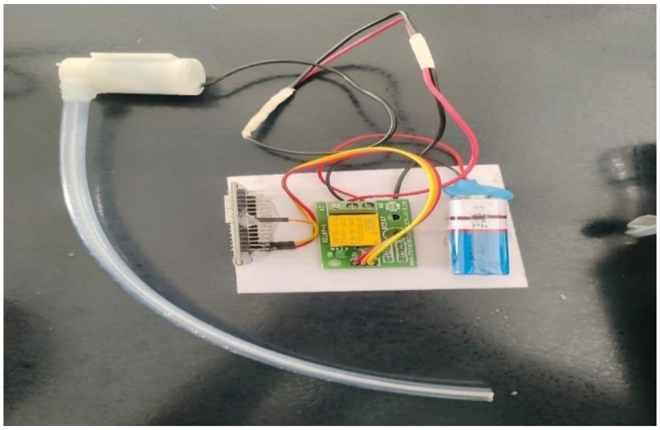


Figure1 :Methodology of Fire Detection with Voice Alerts

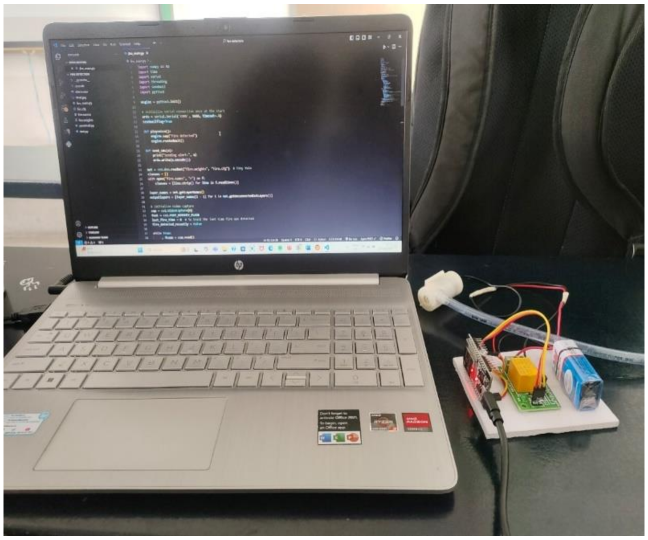
IV. SNAPSHOTS



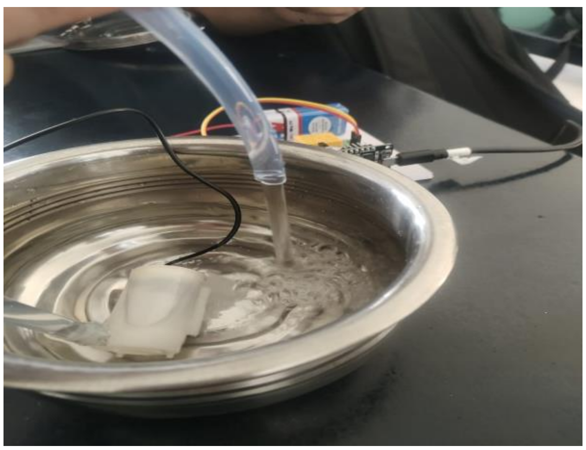
Snapshot1:ESP32 Microcontroller



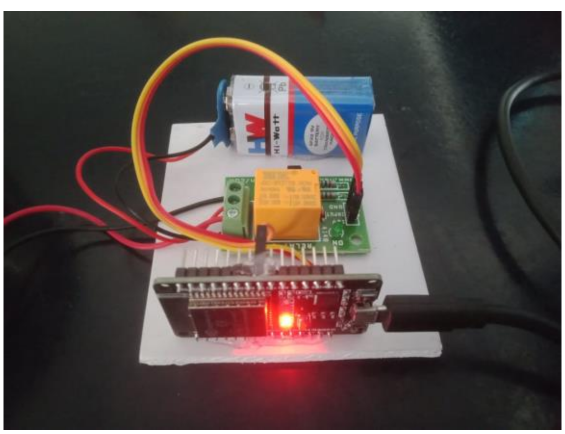
Snapshot2:Pump Control System



Snapshot3: Fire Detection System

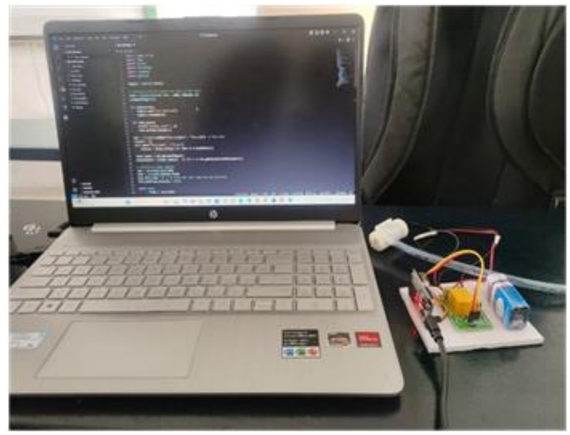


Snapshot4:Automatic water spray for fire extinction



Snapshot5:Relay and Motor integration

Snapshot6: Implementation of bounding box



Snapshot7:YOLO based Fire detection and alerts system

V.CONCLUSION

This project successfully addresses the critical need for rapid, automated fire detection and response in various environments. By integrating real-time object detection using the YOLO deep learning model with hardware-based suppression mechanisms, the system offers an efficient and scalable solution for enhancing fire safety. Through accurate detection, immediate alerts, and automatic activation of fire suppression devices, it ensures timely intervention, potentially saving lives and minimizing property damage. The project demonstrates the practical integration of artificial intelligence, computer vision, embedded systems, and IoT technologies to solve real-world safety challenges.

By focusing on accuracy, responsiveness, and low-cost implementation, the system is suitable for both industrial and residential applications. Enhancements could involve recognizing various types of fires and optimizing performance under different visibility conditions, optimizing the system for low-light or occluded conditions, and integrating cloud-based monitoring for multi-location deployments. This work not only showcases the transformative impact of AI in emergency response systems but also sets a strong foundation for continued research and innovation in intelligent safety technologies. The project underlines how proactive automation can play a key role in emergency preparedness and disaster mitigationand highlights the role of smart systems in building safer, more resilient communities.

REFERENCES

[1] National Risk Survey Report - Pinkerton, FICCI (2018).
[2] Janku P., Kominkova Oplatkova Z., Dulik T., Snopek P. and Liba J. 2018. "Fire Detection in Video Stream by Using Simple Artificial Neural". Network. MENDEL. 24, 2 (Dec. 2018), 55–60.
[3] Shen, D., Chen, X., Nguyen, M., & Yan, W. Q. (2018). "Flame detection using deep learning". 2018 4th International Conference on Control, Automation and Robotics (ICCAR).
[4] Li, C., & Bai, Y. (2018). "Fire Flame Image Detection Based on Transfer Learning". 2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS).
[5] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho and S. W. Baik, "Convolutional Neural Networks Based Fire Detection in Surveillance Videos," in IEEE Access, vol. 6, pp. 18174-18183, 2018.
[6] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," in IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345-1359, Oct. 2010.
[7] Zhang, Qingjie & Xu, Jiaolong & Xu, Liang & Guo, Haifeng. (2016). "Deep Convolutional Neural Networks for Forest Fire Detection"
[8] K. Muhammad, J. Ahmad, Z. Lv, P. Bellavista, P. Yang and S. W. Baik, "Efficient Deep CNN-Based Fire Detection and Localization in Video Surveillance Applications," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 49, no. 7, pp. 1419-1434, July 2019.
[9] Z. Jiao et al., "A Deep Learning Based Forest Fire Detection Approach Using UAV and YOLOv3," 2019 1st International Conference on Industrial Artificial Intelligence (IAI), Shenyang, China, 2019.
[10] Faming Gong ,1 Chuantao Li,1 Wenjuan Gong ,1 Xin Li,1 Xiangbing Yuan,2 Yuhui Ma and Tao Song. "A Real-Time Fire Detection Method from Video with Multi-feature Fusion". Hindawi, Computational Intelligence and Neuroscience, Volume 2019.
[11] "YOLOv4: Optimal Speed and Accuracy of Object Detection" , arXiv (2020) , Alexey Bochkovskiy, Chien Yao Wang, Hong Yuan Mark Liao .
[12] Fire Detection Based on Computer Vision: A Review", MDPI (2019) , S. Saponara, G. Elhanashi, and M. Rundo.
[13] "A Survey on Fire Detection Techniques Using Sensors, Drones, and Image Processing" , IEEE Xplore , Yadav S., Chauhan D.
[14] Fire Detection using Deep Learning and IoT by M. Sharma, P. Gupta. A YOLO-based fire detection system integrated with IoT for real-time alerts and response in smart homes.
[15] IoT-Based Fire Alarm and Monitoring System by S. Patel, A. Verma. A smart fire detection system using IoT sensors for early alerts and real-time monitoring in buildings. Published in Sensors Journal, 2023.
[16] Embedded System for Fire Detection and Suppression Using ESP32 by D. Kumar, V. Rao. An embedded fire suppression system using an ESP32 controller for activating alarms and water pumps. Published in IEEE Sensors Conference, 2022.
[17] A Hybrid Approach to Fire Detection Using YOLOv4 and IoT by L. Wang, H. Zhang. A hybrid fire detection system combining YOLOv4 with IoT for improved detection accuracy and response time. Published in MDPI Sensors, 2023.