A Vision-Based Approach for Real Time Fire and Smoke Detection Using FASDD

By

Qaiser Khan

(Registration No: 00000363689)

Supervisor: Dr. Kunwar Faraz Ahmed

Department of Mechatronics Engineering

College of Electrical and Mechanical Engineering

National University of Sciences & Technology (NUST)

Islamabad, Pakistan

Jan 2025

Annex A

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis entitled "**A Vision-Based Approach for Real-Time Fire and Smoke Detection Using FASDD**" written by **Qaiser Khan** Registration No. 00000363689, of College of E&ME has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS/M Phil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature : ____________________
Name of Supervisor: Dr. Kunwar Faraz Ahmed Khan
Date: 16 Jan 2025

Signature of HoD: ____________________
(Dr Hamid Jabbar)
Date: 16 Jan 2025

Signature of Dean: Rashid
(Brig Dr Nasir Rashid)
Date: 16 JAN 2025

ACKNOWLEDGEMENTS

In the name of Allah, the Compassionate and the Merciful. I express my gratitude to Almighty Allah for granting me the opportunity and abilities to complete my thesis.

I would like to express my overwhelming appreciation to my respected supervisor, **Dr. Kunwar Faraz Ahmed**, for guiding, supporting, and motivating me throughout my thesis. His knowledge and advice were incredibly helpful.

Furthermore, I would like to extend my gratitude to my Co-Principal Investigator, **Dr. Umer Asgher**, for his invaluable guidance and unwavering support throughout this research study.

I am deeply obliged to the esteemed members of my thesis committee, **Dr. Tahir Nawaz** and **Dr. Umer Shehbaz**, for their constructive feedback and insightful suggestions, which have greatly enhanced the quality and rigor of my thesis.

I am also thankful to all my friends and colleagues at the Department of Mechatronics Engineering for their guidance, cooperation, and support during my studies. My special thanks to **Dr. Rehan Zafar Paracha** and **Dr. Shahbaz khan** for their assistance in my research.

Lastly, I extend my sincere appreciation to my parents, siblings and my wife for their unwavering love, support, and encouragement during this journey. They have been my source of inspiration and motivation.

ABSTRACT

Fire and smoke detection is essential in safety-critical environments, yet traditional systems often struggle with maintaining accuracy and reducing false alarms in complex scenarios. Therefore, vision-based systems are used for preventing fire tragedies. There are different machines and deep learning techniques used to timely and effectively detect the fire/smoke and one of them is “You Only Look Once” (Yolo). Yolo is a type of neural network (CNN), which is good at detecting patterns in images. Yolov8 is the most widely used object detection model for vision-based systems. However, there still exist some challenges, such as high computational complexity and low detection performance. This study introduces a novel lightweight and optimal Yolov8 model to over these challenges. To enhance performance, Efficient Channel Attention (ECA) is integrated into the model’s head to focus on critical features, while the C3Ghost module in the backbone reduces computational overhead without sacrificing accuracy. The model is trained and evaluated on two datasets: FS and FASDD comprising diverse indoor, outdoor fire and smoke scenarios and has achieved a mAP@50 of 89%, precision:88%, recall: 84%, and an F1-score of 86.4% which shows an improvement of 4.56% in precision, 2% of recall and 8.10% in mAP@50 in comparison with the existing state of the art. Our findings have demonstrated significant improvements in detection accuracy and false-positive reduction compared to other computationally intensive models like Yolov5, Yolov7 and (vision) Transformers. Our model is lightweight architecture, more accurate in fire, and smoke detection, and makes it suitable for embedded device deployment.

Keywords: Deep learning, Efficient Channel Attention (ECA), C3Ghost, Indoor fire, Outdoor Fire, Object Detection, Yolov8

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS, ABBREVIATIONS AND ACRONYMS

3D-CNN	3D Convolution Neural Network
AP	Average precision
C3Ghost	Ghost convolution network
CBAM	Convolution block attention module
CBS	Convolution-Batch Normalizer-SiLU
CentOS	Linux operating system
CIoU	Complete Intersection over Union
CNN	Convolution neural network
COCO	Common Objects in Context
CSP	Cross Stage Partial Network
CV	Computer vision
DCGANs	Detection through Deep Convolutional Generative Adversarial Neural Networks
DCGC	Dual Channel Group Convolution
DETR	Detection transformer
DFL	Distribute Focal Loss

DIoU	Distance Intersection over Union
ECA	Efficient Channel Attention
ESE	Effective Squeeze Extraction
FASDD	Flame and Smoke Detection Dataset
FFT	Fast Fourier transform
FLOPs	Floating points operation per sec
FN	False Negatives
FP	False Positives
FPN	Feature Pyramid Network
FPS	Frames Per Seconds
FS	Fire and smoke
GAP	Global Average Pooling
GMMs	Gaussian Mixture Models
HSI	Hue-Saturation-Intensity
HSV	Hue-Saturation-Value
IR	Infrared
NFS	non-fire and non-smoke

PAN	Path aggregation network
R-CNN	Region based Convolution Neural Network
Res-CBAM	Residual Convolution Block Attention Module
RT-DETR	Real time detection transformer
SA	Shuffle Attention
SGD	Stochastic gradient descent
SPPF	Spatial pyramid pooling fusion
SSD	Single shot detection
TN	True Negatives
TP	True Positives
UV	Ultraviolet
VOC	Visual Objects Classes
VSD	Video Smoke Detection Dataset
WIoU	Wise-IoU
YOLO	You only look once

CHAPTER 1: INTRODUCTION

The ability to detect fires and smoke is critical for both safety and ecological reasons and holds significant significance. Fire detection systems which operate based on sensors have dire constraints. For instance, these varieties of devices are best suited for the indoor environment and are ineffective in larger or more active environments as they need to be stationed far too close to a flame. Moreover, they are incapable of locating the fire, the fire location's urgency, or the smoke movement in such detail with such accuracy. Recent advances in video-based fire detection systems which employ Convolutional Neural Networks (CNNs) or current object detection technology in image understanding such as YOLO (You Only Look Once) are a real game changer. These astonishing detectors function successfully for precise detection of flames within an extensive range of settings extending from confined indoor areas to expansive outdoor premises that require protection against wild flames.

CNNs have been especially useful in image classification and feature extraction, enabling powerful approaches to fire detection in diverse situations. The introduction of models like YOLO has marked a significant improvement in fast response fire detection systems with high precision and minimal delays, making them optimal for use in the most critical applications. These advancements help in the fight against fires, since their timely detection facilitates quicker response times and prevention of large-scale disasters. This thesis details the design and construction of a fire and smoke detection system based on YOLO, while maximally improving the detection and minimizing the false alarms with sophisticated neural networks.

Pakistan, like many other countries, faces significant challenges due to fires, particularly wildfires. These fires have a multifaceted impact, affecting public health, the environment, and the economy. Wildfires in Pakistan contribute to substantial carbon emissions, exacerbating climate change, while the smoke from these fires presents serious health risks, including respiratory and cardiovascular issues. Additionally, the destruction of habitats and ecosystems by wildfires leads to long-term environmental degradation.

Effective fire detection and management strategies are crucial for mitigating these adverse effects and protecting both human lives and the environment.

In order to mitigate these risks, this study puts forwards the state-of-the-art fire detection system built on YOLO model for monitoring and detecting flames in intricate settings. This study focuses on two datasets FASDD and FS which include both indoor and outdoor fires for developing fire detection system which offers timely solutions for fire detection and control not only in Pakistan but in other susceptible areas as well.

1.1 Motivation, Scope, and Background

An increasing rate of destructive wildfires and fires in urban areas points to the need for improved fire detection and suppression systems. Problem-oriented development usually works with algorithms and methods based on known sensors, but their implementation does work in real information diverse and mobile spaces. These issues need the evocative implementation of advanced fire technologies which risk civilians, structure, and nature. There is an opportunity for much improvement in these matters through CNNs and subsequently more advanced models like YOLO, SSD, and RT-DETR. The purpose of this research is to improve the ability to detect fires, reduce losses and damages, and combat fire threats in areas like Pakistan which are more susceptible to fire disasters comparatively.

The work addresses the goal of creating and deploying deep learning-based smoke and fire detectors using the Yolo state-of-the-art model augmented with Efficient Chanel Attention and C3Ghost module for speed and energy efficiency. This research includes these aspects for self-extinguishing scenarios both in houses and large scale outdoors. The objective of this project is to examine how these models can be utilized to enhance detection precision while lowering erroneous trigger rates, using real time computational video analysis which can seamlessly integrate into existing fire alarm systems. In journal articles, fire incidents are analyzed from different perspectives and this problem is tackled to achieve low false positive rates. The outcome of the investigation is a multi-faced fire alarm which provides adaptability to numerous conditions of frameworks and types of fires.

Over the years, fire detection technology has evolved significantly, transitioning from basic sensor-based systems to sophisticated video analysis techniques. Traditional methods, while effective in controlled environments, are often inadequate for early detection in complex and dynamic scenarios. The advent of CNNs has revolutionized fire detection, allowing for more precise and rapid identification of fire and smoke. Models such as YOLO represent the forefront of this technology, offering enhanced object detection capabilities that surpass the limitations of traditional systems.

The newer models of YOLO, particularly YOLOv8, YOLOv9, and YOLOv10, have proven to improve accuracy while also being faster, which is important for real-life applications. At the same time, some models like RT-DETR, SSD and Faster RCNN employ different methods of object detection and improve the ability to detect smoke and fire more effectively across different environments. These models can address the challenges created by the limitations in traditional detection technologies and present strong options for effective fire control management through early detection and intervention.

This aims to add value to the existing literature about fire detection and prevention of fire in a manner which is actionable and practical agitated places like Pakistan. This thesis intends to not only build practical and technical skill, but also to understand the extent to which neural network based enhanced detection systems can help mitigate damage caused by excessive fire and smoke, set a standard for the public health, environment, and safety.

1.2 Problem Complexity

Fire and smoke are intrinsically linked phenomena, often occurring concurrently. While smoke may, under specific conditions, ignite its source and escalate into flames, fire invariably produces smoke as a byproduct. Conversely, while smoke may not necessarily precede fire, fire is always accompanied by smoke. Despite their inseparable nature, fire and smoke possess distinct characteristics. A complete understanding of these elements is a must for developing effective detection technologies and systems.

1.2.1 Attributes of Fire

A flame is predominantly a combination of several gases which are produced when combustible matter, an oxidizer and heat are employed. The composition of combustion material together with several intermediate reactions determines the emitted frequency spectrum and light that is visible, infrared, and on some rare conditions, ultraviolet.

1.2.1.1 Shape

Normally, a flame's shape forms a triangle. It is always broad at the lower end and gets narrower at the top. Such is the case due to the angle and speed of the wind as well as the amount of combustion agents the fire has. A flame’s shape is altered depending of the shape and the surface area of the source, the combustion’s point of origin. The size, area, form and quantity of fire regions in an image vary from one frame to another.

1.2.1.2 Color

The multi-colored tendency of fire and the soot that can sometimes be produced does at times emit a light that seems to be dimming. The particles that flame emit are captured by the burning source and thus are turned into black bodies which releases a dark orange hue. Blue completely burned gas fire emits single wavelength radiation and gives off the infrared and recommend radiation. The color is found out by the chemical composition for emission spectra and the temperature for black body radiation. A flame’s dominant color is altered by temperature. The blue base of the fires is the hottest color achievable for organic material. Above that, other colors like yellow orange and red are achieved. The latter set of colors may or may not match the colors emittance source composition set. However, wildfires rarely consist of barium nitrate which is why the fire from barium nitrate is bright green.

1.2.2 Features of smoke

Smoke consists of aerosol airborne particulates and gases emitted during the combustion or pyrolysis of a material, combined with the amount of air mixed into the mass. The nature of the smoke formed is dependent upon the kind of fuel being burnt and the combustion conditions. The burning source tends to combust more completely in the presence of fire and oxygen, thus producing less smoke.

1.2.2.1 Shape

At the origin, the shape of smoke is denser, and it gets diluted as it ascends into the ambient surroundings. Smoke plumes usually have vertical or almost vertical axis that are overcome by the wind and buoyancy, so it expands in a circular region, which increases its size perpendicular to its axis. Smoke's movement pattern is not stable, which results in the changing sculpture.

1.2.2.2 Color

To assess the scenario involving a fire, smoke is also considered by firefighters. The color of the smoke is very crucial to the prediction of fire behavior. It reveals the type of fuels and their intensity which gives clues to the possible actions of the fire. White smoke indicates that the material is releasing moisture and water vapor and is therefore undergoing a fire start. White smoke can also suggest flashy fuels such as grass or twigs. Heavy unconsumed thick fuels burn on, and black smoke is a thick by-product. And sometimes, black smoke serves as a by-product of burning tires, cars, or buildings. The blacker smoke is emitted, the more aggressive the fire is. Silvery smoked suggests that the fire is in its final stages of existence.

Figure 1.1: Features of fire and smoke [1]

1.3 Fire and Smoke Categories

A clear understanding of the types of fires and smoke they emit is required for effective fire and smoke detection. These may be classified in a general way with regards to their environment and the type of fire. Each category poses specific issues which require specially designed detection systems.

1.3.1 Indoor Fire

Fires that are classified as indoor fires occur within buildings or enclosed spaces in which the fire was started deliberately or accidentally in a process termed as "ignition." The causes can be electrical failures, neglected cooking, heating devices failing or some other accidental igniting sources. Early warning of such fires is very significant because Indoor fires produce dense smoke which can spread at an alarming rate. These systems should always be on guard and accurate even in low visibility conditions because of the smoke and fire. The main aim here is to shift the focus towards the safekeeping of individuals inside and damage control to the building. Minimal harm to the occupants and structural damage to be avoided.

Figure 1.1: Sample images of Indoor fires [1]

1.3.2 Outdoor Fire

Outdoor fires appear in open areas such as fields, forests, and urban environments. They can be ignited by natural causes such as lightning strikes, or human activities, such as discarded cigarettes, campfires, or industrial accidents. Unlike indoor fires, outdoor fires are often fueled by wind and dry vegetation, which can cause them to spread rapidly and uncontrollably. Detection systems for outdoor fires must be resilient to environmental factors such as strong winds, rain, and varying light conditions. These systems must ensure reliable performance in dynamic, large-scale settings, where fire behavior can change rapidly.

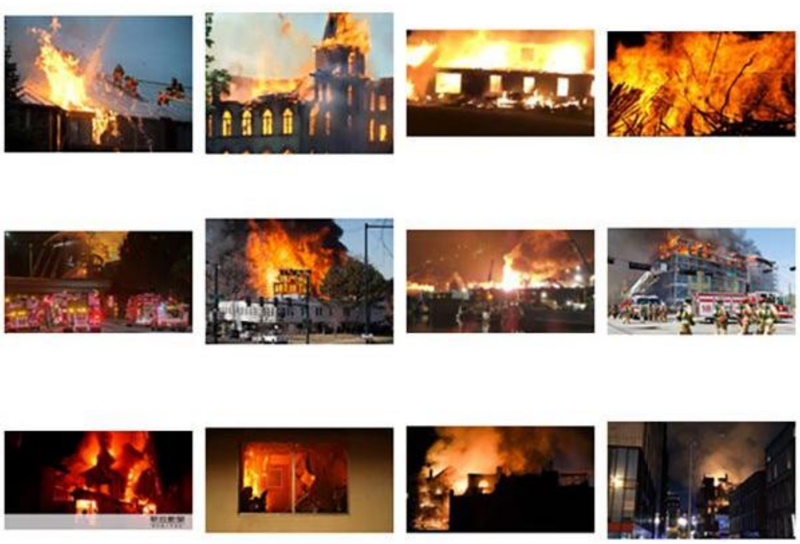


Figure 1.2: Samples of outdoor fires

1.3.3 Wild fire

Wildfires are a specific type of outdoor fire that can spread quickly across vast areas, particularly in forests and grasslands. Driven by dry vegetation, strong winds, and drought conditions, wildfires can produce vast amounts of smoke, which may travel great distances and significantly degrade air quality. Detecting wildfires at an early stage is crucial for containment efforts, especially in remote areas. Detection systems for wildfires must cover large geographical areas, often incorporating satellite or drone-based monitoring, and should be able to identify fire outbreaks at the earliest signs, enabling rapid response and resource deployment.

Figure 1.3: Sample images of wildfires

1.3.4 Smoke Emissions

The composition of smoke is highly influenced by the materials that are burning. As such, understanding the characteristics of different types of smoke is essential for optimizing detection systems. The three primary categories of smoke emissions are:

1.3.4.1 White Smoke

Generally created from the burning of organic matter like plants. White smoke is very thin and gets dispersed very fast which makes it harder to see from far away. While a fire is in its early stages, the detecting systems should be fine-tuned and be able to see the dim traces of smoke during the initial phases of the fire.

1.3.4.2 Black Smoke

Generated from the burning of synthetic materials, such as plastics or hydrocarbons, black smoke is dense, thick, and highly visible. It often indicates a high-intensity fire, making it easier to detect. However, dense black smoke can obscure other critical fire indicators, such as flame intensity or the spread of fire.

1.3.4.3 Gray Smoke

A mixture of different materials burning gray smoke is characterized by a blend of the features of both white and black smoke. Its varying density and color make it more challenging to detect accurately.

Each smoke type presents unique detection challenges. For instance, dense smoke from black fires can obscure visual cues, while the rapid dispersion of white smoke can make it difficult to track the fire's progress. Effective fire and smoke detection systems must be designed to adapt to these differences and respond accordingly.

1.4 Types of Various Detection Approaches

Through the various technological approaches, fire and smoke detection systems has achieved its significant improvement. These can be classified into the types of traditional, sensor-based, and sophisticated computer vision systems. Each exhibits remarkable strengths and weaknesses, and an appreciation of these approaches is required for the design of effective detection systems.

.

1.4.1 Traditional Approaches

1.4.1.1 Conventional Smoke Detectors

The most common fire detectors are traditional smoke detectors. These devices depend on two methods: photoelectricity and ionization. In a photoelectric detector, a light source and a photosensitive sensor are present; their combined functions cause an alarm to go off when a smoke particle enters the detection chamber and scatters the light. In contrast, Ionization detectors contain a small quantity of radioactive material for air ionization. When smoke interferes with it, the alarm is triggered. However, these types of detectors are highly effective for indoor use. There are, however, shortcomings for use in larger and more open spaces as they may not detect smoke or fire timely.

1.4.1.2 Heat Detectors

Heat detectors are designed to activate when a predetermined temperature threshold is exceeded. These detectors are particularly useful in environments prone to false alarms from smoke detectors, such as kitchens or garages. While they are reliable in detecting slow-burning fires, they may not respond quickly enough to rapidly spreading flames, which poses a limitation in situations where early detection is crucial.

1.4.2 Sensor-based Approaches

1.4.2.1 Infrared (IR) Sensors

Infrared sensors detect the infrared radiation produced by flames. These sensors are particularly effective in detecting fires that produce significant heat, making them ideal for industrial and outdoor applications where flames are more likely than smoke. However, they may struggle with detecting smoke, which can be a limitation in certain fire scenarios.

1.4.2.2 Ultraviolet (UV) Sensors

UV sensors detect ultraviolet light emitted by flames and are known for their fast response times. These sensors can identify fires within milliseconds, making them well-suited for high-risk environments where quick detection is essential. However, UV sensors are sensitive to false alarms caused by lightning, sunlight, or industrial activities like arc welding.

1.4.2.3 Gas Sensors

Gas sensors detect burning gases, such as carbon monoxide (CO) and carbon dioxide (CO2), which are released during the combustion process. These sensors are particularly useful for detecting smoldering fires that may not produce visible flames or substantial heat. While effective in certain fire scenarios, they may not provide early warnings for rapidly spreading or large fires.

1.4.3 Vision-based Approaches

1.4.3.1 Video-Based Detection System

Video-based detection systems use cameras and image processing algorithms to analyze video feeds for signs of fire and smoke. These systems offer real-time monitoring and can cover large areas, making them versatile for both indoor and outdoor applications. When integrated with Convolutional Neural Networks (CNNs), video-based systems have seen significant improvements in accuracy and speed, enabling more reliable detection in complex environments.

1.4.3.2 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) have revolutionized fire and smoke detection by providing advanced capabilities for feature extraction and object detection.

CNN-based models, such as YOLOv8, YOLOv9, YOLOv10, and RT-DETR, excel in detecting fire and smoke with high precision, even in challenging environments with varying lighting conditions and complex backgrounds. By analyzing complex visual patterns, these models offer robust solutions for fire detection in a variety of settings, from indoor spaces to large-scale wildfires.

1.4.3.3 Hybrid Systems

Hybrid systems combine multiple sensor types and computer vision techniques to create a comprehensive fire detection solution. For instance, integrating infrared sensors with video analysis can enhance the reliability of fire detection by cross-verifying signals from different sources. These hybrid systems aim to provide the benefits of both sensor-based and vision-based approaches, offering a more accurate and adaptive detection system for diverse fire scenarios.

1.5 Challenges in Fire Detection System

Fire and smoke detection systems face significant challenges in achieving high accuracy, adaptability, and efficiency, particularly in diverse and dynamic environments. Traditional sensor-based systems, though effective in controlled indoor settings, often exhibit limitations in large-scale or outdoor scenarios. These systems typically require closeness to the fire source and fail to offer detailed evidence on the fire's location, intensity, and progression, making them unsuitable for comprehensive monitoring. Furthermore, they are prone to false alarms, reducing their reliability and usability in critical applications. Environmental complexities further complicate fire detection. Indoor fires often involve dense smoke and low-visibility conditions, necessitating systems capable of functioning in constrained environments. Conversely, outdoor fires, including wildfires, must contend with rapidly changing environmental conditions such as strong winds, varying lighting, and expansive coverage areas. The detection of wildfires demands robust systems that can handle large geographical regions, often requiring satellite or drone-based monitoring for effective early warning. These scenarios underscore the need for advanced, adaptable fire detection technologies.

Another challenge lies in addressing the diverse nature of fire and smoke emissions. For instance, detecting faint white smoke from organic materials differs significantly from identifying dense black smoke produced by synthetic combustibles. Moreover, traditional models often lack generalization capabilities, struggling to perform consistently across different fire and smoke categories. Resource constraints, such as the need for computational efficiency in real-time systems, add an additional layer of complexity, especially in regions where advanced infrastructure is unavailable. Addressing these challenges is crucial for improving the performance of vision-based fire detection systems. For further insights into these challenges, we recommend referring to the literature review section which offers comprehensive discussions on the challenges in fire and smoke detection.

1.6 Research Objectives

To address the aforementioned problems, this thesis aims to employ modern deep learning technologies and computer vision techniques. The objectives of this study are formulated as follows:

- To recognize fire and smoke in diverse environments (Indoor and Outdoor scenes).
- To train the model on large scale datasets including FASDD and FS to solve the generalizability problem.
- To optimize the model for resource constrained real time embedded systems.
- To compare the suggested model with modern existing models.

In summary, this thesis bridges the gap between theoretical advancements in fire detection technologies and their practical application, delivering a versatile, efficient, and reliable solution for fire and smoke detection in diverse settings.

CHAPTER 2: LITERATURE REVIEW

Fire is one of the most hazardous and destructive forces, posing significant risks to human lives, property, and the environment. Effective handling of fire incidents necessitates robust detection systems to mitigate risks promptly. Fires exhibit distinct characteristics such as variations in flame color, heat intensity, spread rate, and burning patterns. For instance, flames may appear in different colors, including red, orange, and yellow, depending on the temperature range [2]. Fires are typically classified based on their environmental settings, with indoor and outdoor fires presenting unique challenges. Outdoor fires, particularly wildfires, have increased in severity due to climate change and human activity, resulting in catastrophic environmental damage and public health crises. Prominent wildfire events in the United States [3], Turkey [4], and Australia [5] have underscored the environmental toll, as wildfires account for almost 84% of CO_2 emissions in tropical and subtropical regions, significantly degrading air quality [6]. Meanwhile, indoor fires, often triggered by industrial accidents in urban areas, remain a critical yet overlooked hazard [7]. Fires in factories, chemical plants, and fuel storage sites place nearby schools, hospitals, and residential areas at substantial risk [8], [9]. Beyond physical damage, fire catastrophes evoke strong emotional responses and alter risk perceptions among those who experience them [10]. These realities emphasize the urgent need for effective, real-time fire detection systems to minimize fire-related losses and hazards.

In this chapter, a review of the major issues that deal with fire and smoke detection nowadays, as well as relevant research directed at formulating working detection solutions using the newest technologies, is provided.

2.1 Traditional Fire and Smoke Detectors

Traditional fire detection systems, such as smoke, heat, and flame detectors, are commonly used but suffer from significant limitations in detecting fires across diverse environmental conditions. For example, heat detectors are notably slow, requiring up to eight additional minutes to detect fire compared to smoke detectors [11]. While smoke detectors are faster and more reliable, their performance is heavily influenced by modern building ventilation systems, which can delay detection by 58 to 67 seconds depending on air velocity [12], [13]. Flame detectors, which operate by identifying ultraviolet or infrared flames, are prone to false positives caused by reflective surfaces or floating objects [14], [15]. Although flame detectors report minimal delays, the frequent occurrence of false alarms reduces their reliability and delays human responses during emergencies [9]. Multi-sensor systems that combine smoke, flame, and heat detection can address some of these issues but remain prohibitively expensive [16]. These limitations underscore the necessity for advanced, efficient, and reliable fire detection systems that can overcome the shortcomings of traditional approaches.

2.2 Computer Vision-Based Detection Techniques

Research and development work was carried out on the design of computer vision techniques for automatic fire or smoke detection in tunnels, aircraft hangars, ships and other small complex environments. Considerable efforts have also been made towards the development of robust video fire detection systems for big or open spaces. Numerous vision-based fire detection systems function by detecting color and motion which are two salient features.

2.2.1 Colour and Motion Detection

2.2.1.1 Color-based Detection

It is still common today that color detection was the first used method in automated video fire detection systems. Strategies for video fire detection that employ color uses the RGB color model, and sometimes the HSI or HVS color space systems are used as well [17], [18], [19], [20]. The RGB format is largely used in visible camera sensors, as it is fitted for video-based fire detection because it is suitable for the spectrum content of this color space. Thus, Phillips [21] introduced the Gaussian-smoothed color histogram of 'fire color' pixels in RGB color space with temporal changes and smoothed color histogram and then used the pixels' temporal changes to determine the fire rate. A flame detection rate of

93.5% has been achieved with this method. Subsequently, Liu et al. [22] applied HSV Color space with Gaussian distributions to represent fire colors and used Fast Fourier Transformations fire contours for detection. The rates achieved were impeccable and set at 0.999. Toreyin et al. [23]used hidden Markov model with wavelets to study periodic movements of the smoke edge. They regarded the changes in chrominance components U and V in YUV color space and noted that these components tend to lower in grayish scenes which has smoke. Towards the smoke flicker modeling, these authors focused on active edges of the smoke and applied 97 frames delay to the 25m flame detection algorithm. Çelik et al. [24] developed fuzzy models of statistical analysis of video sequences still images and developed fuzzy color models in the YCbCr and RGB spaces. In an effort to detect destruction via flames and smoke within a scene, they developed a technique that fused spectral analysis with the features of a fire flicker. Their model performs with an accuracy of 99% when identifying a fire from lookalike objects. Qi et al. n.d. [25], invented a cumulative fire matrix that joins RGB color with HSV saturation. The developer logic suggests that the green portion of the fire pixel experiences greater fluctuation compared to the red and blue parts. This method separates uncontrolled fires from other moving objects by analyzing the spatial color transformation of pixel values. The system succeeds in the detection of 60 different classes of fire videos. Table 2.1 provides a brief comparison of the methods.

Table 2. 1: Color-based detection models (literature survey)

References	Flame Detection	Smoke Detection	Color Detection	Remarks	Results
Phillips, Shah, and Da Vitoria Lobo (2002) [21]	✓		RGB	Used temporal variation	Flame detection rate is 93.5%
Liu and Ahuja (2004) [22]	✓		HSV	Used FFT	Flame detection rate is 99.9%
Toreyin, Dedeoglu, and Cetin (2006) [23]	✓		YUV	Used a hidden Markov model and wavelets	Delay in detection: 97 frames with a distance of 25m

Çelik, Özkaramanl, and Demirel (2007) [24]	✓	✓	YCbCr, RGB	Fuzzy color models	Detection rate is 99%
Qi and Ebert (2009) [25]	✓		RGB/HSV	Cumulative fire color matrix	The system correctly detects 60 types of fire videos

2.2.1.2 Motion-based Detection

Since there is smoke and fire in the image, using video content for detecting the moving objects is suitable. However, to prevent misanalysis of the fire movement as an ordinary moving object, the video footage containing the fire region always requires further examination. These algorithms determine the motion of the region by means of Background Subtraction (Chen et al. 2010) [17], analysis of motion employing optical flow (Kolesov et al. 2010) [26], by temporal differencing Lee et al. 2007 [27]. These algorithms can be easily adapted to the automated detection of fires in videos systems.

Foggia et al. 2015 [28], proposed a blended approach for candidate region selection which incorporates a balanced voting algorithm with YUV and morphological changes to enhance motion change detection. The approach enabled greater accuracy of 93,55%, and further adding features such as optical flow reduced the number of false detections. However, they had insufficient focus on shape and texture considerations. A year later, Han et al. 2017 [29], introduced a hybrid model which incorporated YUV and RGB spaces together with multicolor Gaussian mixture models to enhance motion detection. The model was effective with a remarkable detection rate of 96% within the confines of a laboratory, but it was unable to transduce to practical use in the field. Similarly, Gong et al. 2019 [30], introduced a new novel approach to fire detection analysis based on the analysis of fire characteristics. Detecting changes of color in sequential frames enabled the suspected fire regions to be identified. Tracking the center of mass of fire across frames and observing variations in shape, distribution in space, and area greatly increased identification accuracy. This method was shown to have a lower false positive rate in experiments but is not widely used in practice. Gagliardi et al. 2020 [31], a multi-step smoke detection framework was designed which comprised a Kalman filter used for motion tracking, color segmentation, blob analysis and labeling, and an overlapping bounding box alert system. An alert was raised when seven or more overlapping bounding boxes were identified. This method performed very well for all the datasets and was found to outperform all other methods in terms of effectiveness. The method also achieved an unprecedented recall rate of 100%. In another study Gagliardi et al. 2021 [32], a proprietary method was introduced which combined image processing and deep learning. His method had more advanced smoke detection capabilities as a Kalman filter was used for motion tracking, color was segmented, and bounding boxes were placed around the moving gray objects. Moving gray objects were fed into a Convolutional Neural Network (CNN) for predictions which were accurate 90.49% of the time. Khalil et al. 2021 [33], a multi-space color model was utilized together with motion detection for the classification of fire objects. This was aimed at reducing parameters. They worked with RGB and LAB color space to differentiate fire from fire-like objects. Motioning objects were discovered using Gaussian Mixture Models or GMMs and the fire-like regions were masked to lower interference. Although they achieved high detection accuracy, the model suffered an alarming 88.81% case of false alarms. Wahyono et al. 2022 [34], included the fire color characteristics by employing probabilistic Gaussian mixture models and utilized motion-based moment invariant analysis for dynamic fire movements modeling. Through this technique, a True Positive Rate of 89.92% was realized. Nevertheless, significant barriers to implementation, such as the positioning of the camera, still pose substantial practical difficulties. Table 2.2 presents a brief comparison of the methods.

Table 2. 2: Motion-based detection models (literature survey)

References	Flame Detection	Color Detection	Motion Detection	Remarks	Results
Foggia, Saggese, and Vento (2015) [28]	✓	YUV	Optical Flow (Real-Time)	Enables real-time detection	Achieved an accuracy of 93.55%

X. F. Han et al. (2017) [29]	✓	RGB/HSI/YUV	Gaussian Mixture Models	Effective in motion detection	Average detection rate of 96%
Gong et al. (2019) [30]	✓	RGB/HIS	Frame Differences (Real-Time)	Reduces false positives	High accuracy with minimized false positives
Gagliardi and Saponara (2020) [31]	✓	HSV	Kalman Estimator + Geometric Analysis	Analyzes geometric features	Achieved 100% recall compared to other methods
Gagliardi, de Gioia, and Saponara (2021) [32]	✓	HSV	Kalman Filter + CNN Classifier	Combines motion tracking and classification	Achieved a hit rate of 90.49%
(Khalil et al. 2021) [33]	✓	RGB/LAB	Gaussian Mixture Models	Tracks fire growth and static objects	High performance relative to other models
(Wahyono et al. 2022) [34]	✓	RGB/HSV/ YCbCr	Moment Invariants (Real-Time)	Analyzes dynamic characteristics	True positive rate of 89.92%

2.3 Deep Learning Based Detection Techniques

The ability of deep learning has grown tremendously in the various fields of machine learning. Deep learning algorithms have been successfully applied to activities like image object/caption detection and classification, speech to text and text to speech, and language translation. Studies on deep learning techniques for detecting fire and smoke have focused on improving detection performance and operational efficiency. An overview of some of the work done in this area is given below.

With the aid of a Deep Neural Network, Zhang et al. 2016 [35], conducted fire detection using deep learning methods on forests in 2016 [{34}]. Two classifiers and the Global Convolutional Neural Network Deep CNN designed for image processing were used. Initially, the global image-level classifier processes the image. If any fire is detected, the fine-grained patch classifier is used next in a cascaded form. In this research, two methods of fire patch detection were presented. The first method was to classify the image containing a fire patch using a linear SVM classifier as a binary classifier, and taking the patches tagged with fire into the non-linear Convolutional Neural Network trained on CIFAR 10. The second method proposes the use of a cascade of CNN fire classifiers. The base CNN, a deep CNN, was trained on the ImageNet dataset using the AlexNet model

[36] and the upper layers were connected to the fined-grained patch classifier which was trained on the up sampled Pool features. This method was only deployed in a few videos, but in roughly 90% of the tested 59 images, fire patches were detected.

S. Frizzi et al. (2016) [37], proposed a small convolutional neural network for detecting fire and smoke in the videos. The classifier is a nine-layer CNN model with two fully connected layers. In this work, Leaky ReLu activation was used with a 0.5 drop out rate on fully connected layers. The achieved classification accuracy was 97.9 % on previously unseen data using 5584 images as test set. Fire and smoke features from videos were detected for fast classification by sliding windows of size 12 x 12. These windows are the inputs to be classified by the convolutional neural network and fully connected neural networks. The window position must change to analyze an entire video frame so the window can go through the convolutional neural network again. Here, only the last feature map of the convolutional neural networks is used with sliding windows and GPU.

Luo et al. 2018 [38], implemented a surveillance system that detects smoke using the motion characteristics of smoke visualized using CNN techniques. Initially, a smoke region examiner is proposed where a moving object observation algorithm is assisted by a background dynamic update and dark channel prior. Afterward, the required features from the suspected region of the video are fetched using custom CNNs, followed by smoke needlepoint identification. Applying the techniques improved the accuracy of detection by 99% in videos post-testing. The challenge of monitoring small-sized regions was addressed adequately in the proposed techniques, which altered the regions suspected to be smoke. Implicit enlarging of the areas also improved the speed of detection. Moreover, an algorithm that tuned the network's training parameters using a limited dataset achieved remarkable results. The approaches garnered showed exemplary versatility when tested within different video scenarios.

J. Sharma et al. 2017 [39] proposed deeper Convolutional Neural Networks for better fire detection in images, which was further improved by fine tuning with a fully connected layer Within this strategy, two pretrained Deep CNNs VGG16 and Resnet50 which are at the topmost stage of fire detection model development were exploited. The Deep CNNs are tested using an unbalanced dataset designed to replicate real world conditions. Following this methodology, Resnet50 performs better than VGG16 on the unbalanced dataset. The performance of the deep models is commendable; more than 90% accuracy has been achieved in testing the performance of the Deep CNNs. However, the modified VGG16 and Resnet50 deep models with other additional fully connected layers tend to do marginally better than the base models, but at the price of having to endure longer training periods.

Muhammad et al. 2018 [40], developed a CNN model for fire detection in video surveillance which is cost-effective by which they refer to ‘low cost.’ The model is based on GoogleNet [41] architecture and altered to suit the problem of classification. The GoogleNet model uses transfer learning techniques. To increase accuracy without afflicting the other, the model has been optimized to address target issue and fire data. The final model achieved accuracy of 94.43 % on the test set of the frames from videos.

Aslan et al. 2019 [42], presented a video-based technique for flame detection through Deep Convolutional Generative Adversarial Neural Networks (DCGANs). This method relies on video frame clustering to achieve image temporal slices and the latter being processed through a DCGANs structure. They proposed an initial out-of-the-box training model for a DCGAN which greatly increased the ability of the deep learning model's discriminator. The first stage was trained identifying the differences between normal sequences and flame sequences. After the deep learning model obtained a powerful discriminator, the discriminator was finally trained without the generator. The non-flame images that were utilized constituted the “generated” training data while all other images remained unchanged. This technique reached an impressively increased false-positive rate of 3.91% corresponding to a hit-rate of 92.19%.

Kim et al. 2019 [43], proposed an approach for detecting fires in video sequences based on deep learning techniques. The approach proposed employs a Faster R-CNN spatial classification model to identify supplementary regions of interest (SRoFs) along with ordinary background images. Summary classifying features of the bounding boxes created in consecutive frames are extracted by LSTM networks and used to detect the presence of fire for a limited duration of time. The short-term decisions made consecutively are also passed through the majority voting mechanism for final decision making and detection during the longer period. The final fire decision is made after the estimation of the flame and smoke area is done, and the fire's behavior is analyzed with its spatial and temporal changes. This method was successful in achieving the highest reported accuracy of 97.92 percent in fire detection.

Li et al. (2019) [44], conducted a comparative study of different deep learning networks for smoke detection from video footage in real time. Their work consists of the smoke identification using MobileNet networks models, followed by Transfer Learning methods to create contrasts with other networks such as AlexNet, VGG16, GoogLeNet, and ResNet50. The main contribution of this research is to define how the training time and model update expense can be improved with the use of lightweight *MobileNet* [45] This basic proof allows the model to be applied to devices with limited computational resources. As stated, the pre-trained MobileNet achieved an accuracy of 98.78% over training done for 200 epochs.

2.4 State of the Art Methods

Recent research has been focusing on computer vision (CV) and deep learning (DL) methods for fire and smoke detection, utilizing advanced image and video analysis techniques to address the limitations of conventional systems. Deep learning-based systems typically focus on three key tasks: classification, which determines whether an image contains fire or smoke [46]; detection, which identifies the location of fire and smoke within an image; and segmentation, which annotates the shape and extent of fire and smoke regions [47]. Lin et al. [48] developed a joint detection framework that combines Faster R-CNN and 3D-CNN, enabling smoke localization by leveraging both spatial and temporal information. Li et al. [49] explored multiple architectures, including Faster R-CNN [50], R-FCN [51], SSD [52], and YOLOv3 [53], concluding that convolutional neural networks

(CNN) provide a favorable contrast between detection speed and accuracy. Saponara et al.[54] achieved real-time fire and smoke detection on embedded devices by deploying a lightweight YOLOv2 model, reducing computational costs while maintaining performance.

Several studies have explored the advancements in YOLO-based architectures to enhance fire detection. In Avazov et al. [55], YOLOv4 was implemented on a three-layer Banana Pi M3 board, effectively triggering alarms within eight seconds of fire outbreaks. Xue et al. [56] modified the YOLOv5 backbone by incorporating an SPPFP layer, which improved global feature extraction, particularly for small fire targets. Hu et al. [57] introduced a value-transformed attention mechanism that used color and texture features of smoke images to enhance feature weight distribution. They employed a Mixed-NMS technique to improve localization accuracy. Khudayberdiev et al. [58] developed Light-FileNet, a lightweight fire detection model inspired by H-swish convolution mechanisms, which demonstrated efficiency and reliability in detecting fire. More recent efforts have focused on real-time deployment in complex environments. For instance, Saydirasulovich et al. [59] tailored YOLOv8 for wildfire detection using UAV imagery, integrating Wise-IoU (WIoU) v3 to enhance bounding box regression. Ma et al. [60] modified YOLOv5s by integrating ODConvBS blocks for improved feature extraction, while He et al. [61] incorporated Dual Channel Group Convolution (DCGC) and Effective Squeeze Extraction (eSE) mechanisms into YOLOv5, enhancing the model’s receptive field and its ability to focus on relevant features.

2.5 Available Datasets for Fire and Smoke Detection

The datasets play an essential role in the success of deep learning applications, particularly in fire detection research. To significantly improve detection accuracy, it's crucial to have a large, well-curated dataset that includes high-dimensional images and video sequences, allowing the model to learn a wide range of features. Fire images often display a variety of flame colors, ranging from blue to red, influenced by factors like the burning material and flame temperature. Smoke, a key element in fire detection, usually appears in shades of gray, white, or black in videos. To ensure comprehensive coverage, datasets should include samples from diverse sectors, such as industry, agriculture, infrastructure, households, and forests. This variety is necessary to effectively distribute image features across various application domains.

Some commonly used and publicly available fire and smoke detection datasets are mentioned in Table 2.3.

Table 2.3: Fire and smoke datasets

Dataset Name	Description	Use Case	Environment	Availability
VisiFire Dataset	Public video format with four categories: flame, smoke, other, and forest smoke. Includes 57 videos, with a subset annotated for frame-by-frame segmentation.	Fire and smoke detection in videos.	Both	Publicly available
BoWFire Dataset	Contains 226 images depicting fires and non-fire scenes, encompassing building fires, industrial fires, and fire-like objects.	Training and validating fire detection models.	Both	Publicly available
Corsican Fire Database	A comprehensive collection of multi-modal wildfire images and videos, annotated for fire-related features such as flame color and smoke obscuration.	Fire segmentation and detection in wildfire scenarios.	Outdoor	Available upon request
FESB MLID Dataset	Includes 400 Mediterranean landscape images categorized into 12 categories, with challenging samples of small-scale or distant smoke-like features.	Smoke detection in natural landscapes.	Outdoor	Available upon request
Smoke100k	Large-scale artificial smoke image dataset with three subsets, simulating various smoke densities and backgrounds for training smoke-detection models.	Training smoke-detection models with synthetic data.	Both	Publicly available
Video Smoke Detection Dataset (VSD)	Comprises smoke and non-smoke videos and image datasets, annotated with similarities in color, shape, and texture to non-smoke objects.	Reducing false positives in smoke detection.	Both	Publicly available
FLAME Dataset	Aerial images and videos from Northern Arizona forests captured by UAVs, featuring pixel-level annotations for wildfire recognition and segmentation tasks.	Wildfire recognition and fire segmentation.	Outdoor	Publicly available
D-Fire Dataset	Diverse fire and smoke images, including synthetic samples, designed for developing object-detection methods.	Object detection in fire and smoke scenarios.	Both	Publicly available

DSDF (Dataset for Smoke Detection in Foggy Environments)	Real-world images for smoke detection in foggy environments, with annotations for smoke variations and background information.	Enhancing model generalization in foggy conditions.	Outdoor	Publicly available
DFS (Dataset for Fire and Smoke Detection)	9462 fire images categorized by flame size, including a category for non-fire objects such as vehicle lights and streetlights to reduce false positives.	Fire and smoke detection with false positive reduction.	Both	Publicly available
Flame and Smoke Detection Dataset (FASDD)	Large-scale dataset of 100,000-level flame and smoke images from various sources, including challenges for small object detection, is a large-scale dataset.	Training and validating small object detection models.	Both	Publicly available

As part of our investigation this study is to detect fire and smoke in both (Indoor and Outdoor scenes), so we are considering FASDD and FS for our research. A brief discussion of the datasets is given in the Methods and Materials section (Chapter 3).

2.6 Research Gap Analysis and Contributions of the Proposed Methodology

From the literature review, it is evident that current state of the art models has made significant progress. However, despite these advancements, existing methods for fire detection continue to face significant challenges. A major limitation is the lack of generalizability, as most models are optimized for either indoor or outdoor fire detection, with limited capability to handle both environments effectively. Additionally, most of the models are typically costly, making them unsuitable for real-time deployment on embedded devices. These challenges highlight the need for a lightweight, computationally efficient model that maintains high detection accuracy and generalizability across diverse environments.

To address these challenges, this study proposes a new fire and smoke detection framework based on the YOLOv8 architecture with the following main contributions.

- The Efficient channel attention (ECA) mechanism is proposed in the neck and head network of Yolov8. Therefore, the model's accuracy is improved, and the cost of computation is reduced.

- The lightweight C3Ghost module is incorporated into the backbone network to compress the model size further while maintaining accuracy and speed of detection.
- The model is trained and evaluated on a large variety of indoor and outdoor images to validate its generalizability in different environment settings.
- The proposed model experimental results are compared with related studies to validate our findings.

This work tries to overcome the gap between computational efficiency and real-time fire detection capabilities, resulting in advancements in fire safety technologies. The other key contents of this research are as follows: The third section containing Methods and Materials introduces algorithm, its improved framework and experiments with the algorithm model before and after improvements. The fourth chapter containing Results and Discussion concludes the experimental results and compares the proposed model results with existing state of the models for validations and analysis. The fifth section summarizes the work.

CHAPTER 3: METHODS AND MATERIALS

This chapter highlights proposed model framework before and after improvements, key changes made to architecture followed by brief discussion on Efficient Chanel Attention (ECA) and C3Ghost convolution module, description of datasets (FASDD and FS) and preprocessing steps performed, performance metrics to validate study experimentation results. The workflow diagram is illustrated in Figure 3.1.

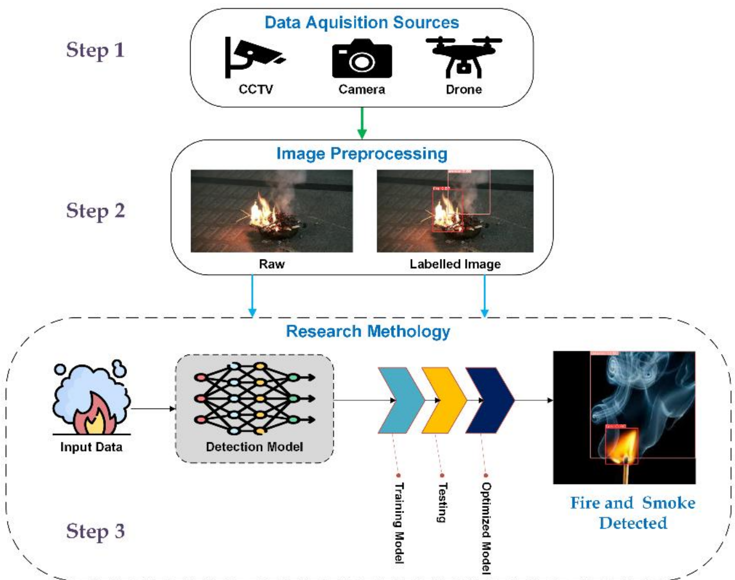


Figure 3.1: Proposed methodology flowchart showing steps from input to output stages

3.1 YOLOv8 Architecture

YOLOv8 is divided into four sections Backbone, Neck, Head and Loss Function. The primary division is augmented by CSPNet whose role it is to ease computational burden while enhancing the learning capability of a CNN [62]. When comparing both,It can be seen from Figure 3.2 that the developers of YOLOv8 turned to the implementation of the C2f block. This block can be seen as a combination of the C3 block with the E-ELAN [63] framework pioneered in YOLOv7 [64]. More specifically, the C3 block is comprised of 3 convolutional blocks in parallel with a multitude of bottleneck connections while the C2f block is constructed using 2 convolutional blocks in series with 3 or more bottleneck connections in between. The structure of the convolution module is as follows; Convolution-Batch Normalizer-SiLU (CBS).

Furthermore, Lin et al. [65], YOLOv5 utilizes the Feature Pyramid Network (FPN) topology for performing top-down sampling, which enables information richer in features to be integrated into the lower feature map. In the same way, Liu et al. [66], a path aggregation network (PAN) was utilized for bottom-up sampling, which enables to improve the top feature map by utilizing the location information of the features more accurately. Their combination generates the accurate estimation of the image possible over various dimensions. The authors retained the FPN and PAN concepts, however, due to the non-reinforced convolution operation, the up-sampling aspect becomes altered as seen in Figure 3.2.

Unlike YOLOv5 that relies on a single coupled head, YOLOv8 has a decoupled head which separates the classification and detection processes. Interestingly, YOLOv8 omits the objectness branch and focuses only on the classification and regression branches. In this manner, it also favors the target center predictor strategy over the anchor-based method which sets an anchor and uses the distance from the anchor to the target center edge as the predictor.

For classification in Yolov8,the loss function using the (BCE) loss, as given by the Equation (3.1) follows:

$$\text{Loss}_{BCE} = -w[y_n \log x_n + (1 - y_n) \log(1 - x_n)] \qquad \textbf{3.1}$$

Where *w* is the weight; *yn* is the labeled value; and *xn* is the predicted value generated by the model.

In the regression branch, YOLOv8 has integrated Distribute Focal Loss (DFL) and Complete Intersection over Union (CIoU) Loss. DFL's objective is to emphasize the growth of probability margins surrounding object 'y'. Its Equation (3.2) is displayed as follows:

$$\text{Loss}_{DF} = -\left[(y_{n+1} - y)\log\frac{y_{n+1}-y_n}{y_{n+1}-y_n} + (y - y_n)\log\frac{y-y_n}{y_{n+1}-y_n}\right] \qquad \textbf{3.2}$$

The CIoU Loss is the first to apply an influence factor to the DIoU Loss with the aim of incorporating an object's aspect ratio into distance bounding box evaluation. The corresponding Equation (3.3) is as follows:

$$Loss_{CIoU} = 1 - IoU + \frac{d^2}{c^2} + \frac{v^2}{(1-IoU)+v} \qquad \textbf{3.3}$$

whereas the IoU is a measure of the overlap between the bounding box predicted and the ground truth bounding box; d is the Euclidean distance between the center points of the predictive and the ground truth bounding boxes and c is the diagonal distance of the minimum bounding box which fully encapsules the predicted and ground truth bounding boxes. In addition, v is the parameter measuring the constancy of the aspect ratio of the object which is formulated with the next Equation (3.4):

Furthermore, v stands for the parameter describing the consistency of the aspect ratio of the object which is described through the subsequent Equation (3.4):

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w_{gt}}{h_{gt}} - \arctan\frac{w_p}{h_p}\right)^2 \qquad \textbf{3.4}$$

Where *w* indicates the weight of the bounding box; *h* indicates the height of the bounding box; *gt* indicates the ground truth; and *p* indicates the prediction.

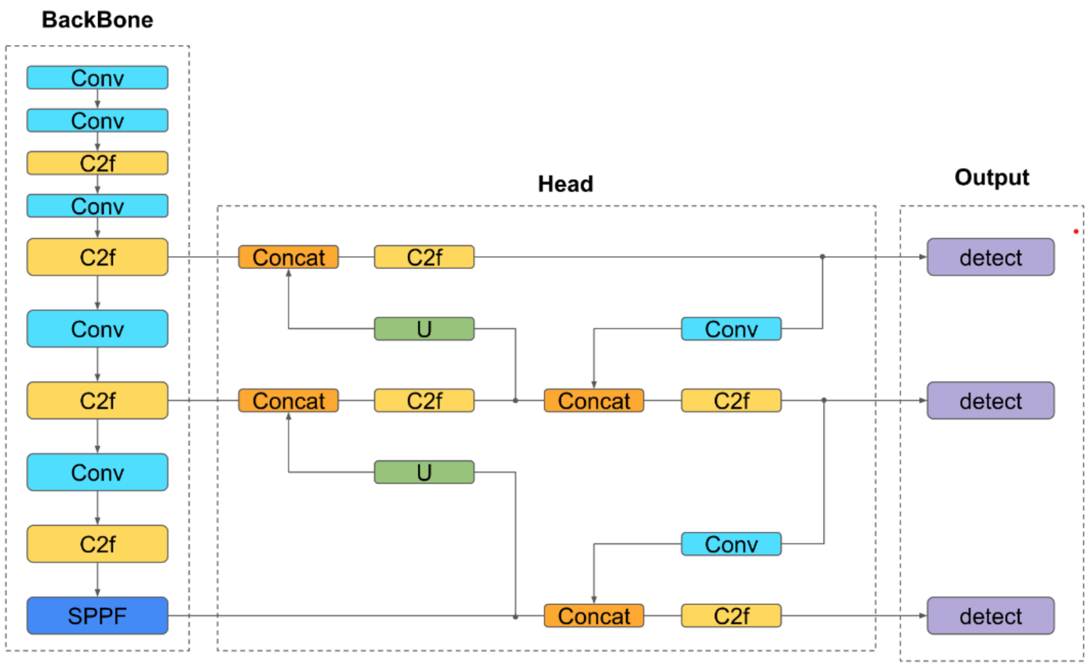


Figure 3.2: YOLOv8 architecture visualization

3.2 Improved YOLOv8 Architecture

This study proposes an efficient and lightweight deep learning model based on YOLOv8 architecture to make the model run fast on resource constrained devices (embedded devices) and improve speed and efficiency for detecting fire and smoke. Figure 3.3 is the illustration of an improved yolov8 structure. This paper uses the ECA attention module in the Neck component of the YOLOv8 model to improve computational power while maintaining its efficiency and accuracy. Attention modules, namely, CBAM [67], ECA [68], and SA [69] are distinctly used after each of the four C2f modules. An in depth introduction of the ECA attention module and Ghost convolution is presented in Sec (C, D).

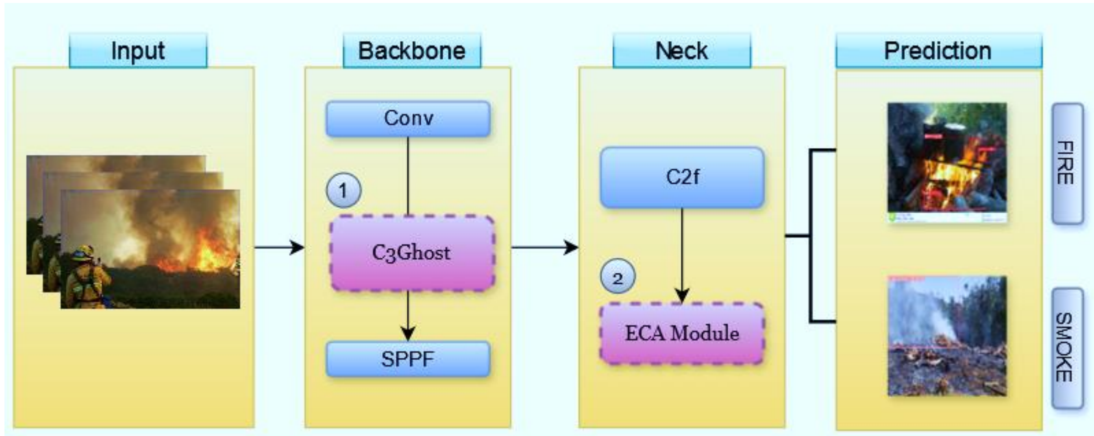


Figure 3. 3: The structure of improved yolov8 model proposed in this study (Added C3Ghost (1), and ECA (2) module to the original model)

3.3 Integrating Attention Mechanisms

In recent years, attention mechanisms have improved the performance of object detection models by enhancing the model's ability to focus on relevant features while reducing unnecessary information. In this study, three attention mechanisms Shuffle Attention (SA), Residual Convolution Block Attention Module (Res-CBAM), and Efficient Channel Attention (ECA) are integrated one by one into the YOLOv8 framework to assess their impact on fire and smoke detection. A detailed comparison of these attention mechanisms is presented in Sec (III).

3.3.1 The Efficient Channel Attention mechanism (ECA)

The Efficient Channel Attention mechanism in Figure 3.4 enhances feature extraction by enabling local cross-channel interaction. The input feature map $F_{input} \in R^{C\times H\times W}$ is processed using Global Average Pooling (GAP) and cross-channel interaction to produce the aggregated feature map F_a given by Equation (3.5). In this way, the interaction between the features of each channel and their neighboring channels is observed, avoiding 1D convolution for dimensionality reduction.

$$F_a = C\left(GAP(F_{input})\right) \quad \mathbf{3.5}$$

The weight w of each feature F is computed using the sigmoid function σ, as described in Equation (3.6):

$$\omega_i = \sigma\left(\sum_{j=1}^{k} W^j F_{ai}^j\right), F_{ai}^j \in \Omega_i^k, \quad \mathbf{3.6}$$

Where $\boldsymbol{\Omega_i^k}$ indicates the set of $\boldsymbol{k}$ neighboring channels of F_{ai}, and $\boldsymbol{W^j}$ signifies the learned weights for each channel. The adaptive convolution kernel size $\boldsymbol{k}$ is related to the channel dimension $\boldsymbol{C}$ using the nonlinear representation shown in Equation (3.7):

$$C = \emptyset(k) = 2^{\gamma * k - b} \quad \mathbf{3.7}$$

Lastly, the kernel size k can be adaptively determined based on the channel dimension, as explained by Equation (3.8):

$$k = \psi(C) = \left|\frac{log_2 C}{\gamma} + \frac{b}{\gamma}\right|_{\mathrm{odd}} \quad \mathbf{3.8}$$

Where t is closest to $|t|_{odd}$, based on tentative results [68], where γ and b are set to 2 and 1, respectively.

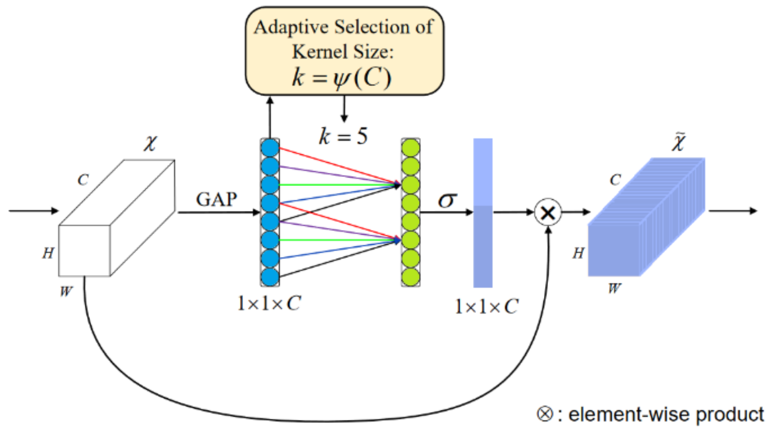


Figure 3.4: Workflow diagram of Efficient channel attention (ECA) mechanism

3.4 Integrating C3Ghost Convolution in the Backbone of YOLOv8

To lower model size and FLOPs, the standard convolutions are restored with Ghost convolutions [70]. It is inserted in the C3 module to form the C3Ghost module. This alteration enables the backbone to extract features efficiently while preserving a high level of feature expression, even with limited computations. In a typical convolutional layer, each filter employs a full convolution to create a feature map. C3Ghost convolution, however, introduces a two-step process where the feature maps are split into primary features, produced via standard convolution, and ghost features, which are computed through lightweight linear operations, as described in Equation (3.9):

$$F_{ghost} = LinearOp(F_{primary}), \tag{3.9}$$

Here, $F_{primary}$ represents the primary feature map obtained through regular convolution, and F_{ghost} denotes the additional features created using linear transformations, such as depth wise convolution or linear projections. This methodology notably reduces the number of required calculations while retaining the expressive power of the convolutional layer.

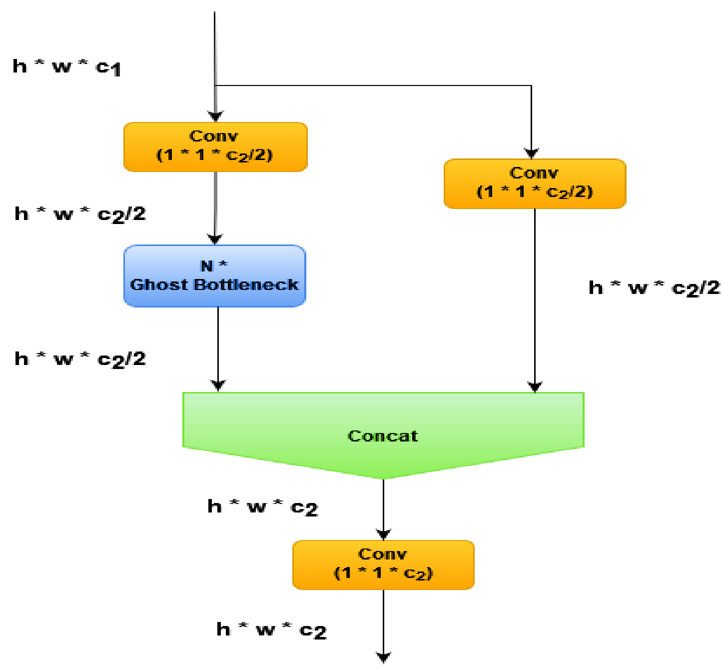
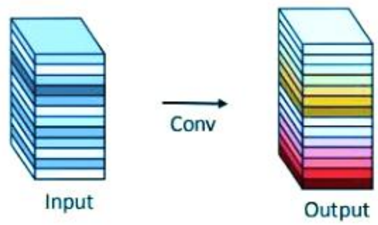
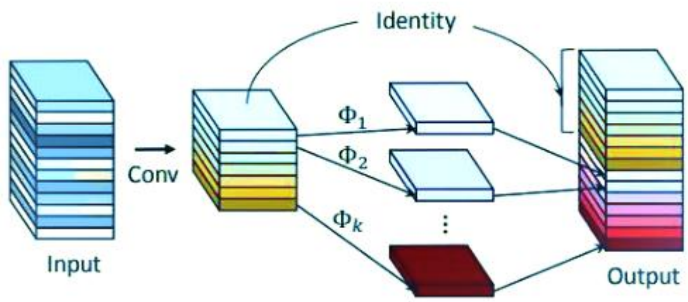


Figure 3.5: Workflow of C3Ghost Convolution



(a) The convolutional layer.



(b) The Ghost module.

Figure 3.6: Traditional Convolution and C3Ghost Convolution modules

The C3Ghost convolution is particularly beneficial when handling large datasets and complex scenes, as it reduces computational demand without sacrificing detection accuracy. The operation of C3Ghost convolution in YOLOv8 is defined as follows in Equation (3.10):

$$Y = GhostConv(X) = Conv(X) + GhostOp(X) \quad \mathbf{3.10}$$

Where $\boldsymbol{X}$ represents the input feature map, $\boldsymbol{Y}$ is the output feature map, and $\boldsymbol{GhostOp}(\boldsymbol{X})$ refers to the ghost feature generation operation. In Figure 3.6, the structure depicted in Figure 3.6 displays the intact C3 module structure, and various Bottleneck can form different C3 configurations. When the Bottleneck is replaced with Ghost Bottleneck, the structure is referred to as C3Ghost; while other ordinary convolutions are replaced by the Ghost module in the network, then we can achieve not only compressing the size of the model but also diminishing the amount of computation.

3.5 Data Acquisition

This section describes the characteristics and characteristics of image datasets utilized for training and testing. In recent fire detection studies, the models were trained on both indoor and outdoor fires. However, certain studies only determined fire detection, not smoke. In this study, a model based on fire and smoke was developed to detect them both. This study utilizes two datasets, Fire and Smoke Detection Dataset (FASDD) [71] and the FS dataset [72].

This study utilizes two datasets for training and evaluation of the proposed model to address the generalizability issue. The description of both the datasets is given below.

3.5.1 The Fire and Smoke Detection dataset (FASDD)

The FASDD (Flame and Smoke Detection Dataset) is a comprehensive dataset designed to enhance fire and smoke detection algorithms, particularly in remote sensing applications. It includes over 122,624 flame and smoke images from a variety of sources, including surveillance cameras, drones, multi-source remote sensing satellites, and computer-generated graphics. The dataset involves diverse scenes such as urban areas, forests, industrial sites, and remote wilderness, ensuring comprehensive coverage for training deep learning models according to the authors of [71]. It is noteworthy that it includes small-scale flame and smoke objects, posing challenges for small object detection in deep learning models. Additionally, the dataset provides annotation files in three different formats like JSON format designed in the Microsoft Common Objects in Context (COCO) dataset, XML format used in Pascal Visual Objects Classes (VOC) dataset, and TXT format compatible with YOLO series models. This public-available dataset is a valuable tool for developing robust fire and smoke detection models that can be utilized in a wide range of applications. The dataset has three variants FASDD_CV, FASDD_UAV, and FASDD_RS. We are using **FASDD_CV** in this study. Figure 3.7 shows sample images from the dataset.

Figure 3.7: Fasdd dataset samples showing (a) Indoor, (b) Outdoor, (c) Day fire, (d) Night fire and smoke images

3.5.2 The FS (Fire and Smoke) Dataset

The FS dataset is composed of 11,667 images obtained from different sources, as described in the article [72]. The dataset includes various lighting conditions daytime, dusk, and night-time providing a rich source of data for training models under various environmental conditions. Both datasets used the (.txt) annotation format for consistency in training. Figure 3.8 presents sample images from the dataset.

Figure 3.8: Sample images from FS datasets of various scenes like only fire, smoke, and both

3.6 Preprocessing

Data pre-processing is a method of converting raw data into clean data. As the data gathered is from different sources, it needs to be standardized and cleaned up before feeding it to machine learning algorithms. Pre-processing is a necessary step to reduce complexity and improve the accuracy of the algorithm. None of the color-changing or intensity changing operations are applied to this data as these properties impact the learning model in real-time. Changing color and luminous intensity will alter the color of _re and smoke, which must be avoided since the algorithm must learn them as primary features. Data cleaning was performed to remove the unnecessary images that contain either too much detail or too little detail. After cleaning, the data is classified into three categories: Fire, Smoke, Non-fire and Non-smoke (NFS).

This research is limited to fire and smoke picture only. The pictures are resized to 600 * 600 \pixel and I annotate the two datasets using txt format to ensure that both datasets are trained effectively by the proposed model. The datasets are divided into training, validation, and testing (Subsets of original dataset used To assess how well machine learning models are able to identify unseen data. He or she is used to unbiased data which In this instance consists of data not used in model training, regardless of its division) segments. The validation (A validation dataset is one which is designed and constructed to identify how well the models work on the specified data. It is utilized to set and calibrate the parameters of the model (Neural network) The model does not get trained on the validation dataset but instead employed to set parameters for improved generalization) set is used to check how effective the model captures the given data and employing it will modify the set parameters so that the model's generalization is enhanced. Usually, this data is never seen by the network) Table 3.1 illustrates the information about data splitting.

Table 3.1: Statistics of Preprocessed data

SR. NO.	DATASET	CLASSES	TOTAL IMAGES	TOTAL SMOKE INSTANCES	TOTAL FIRE INSTANCES	TRAINING SAMPLES	VALIDATION SAMPLES	TESTING SAMPLES
1	FASDD_CV	Smoke, Fire	69,226	53080	73297	48,458	10,384	10,384
2	FS	Smoke, Fire	11,667	8693	16232	8494	2114	1059

3.7 Transfer Learning Technique

An effective technique that is extensively used in today's world is transfer learning. This is where pre-trained models are utilized to improve the performance of learning algorithms on different yet related problems, thereby enhancing the already learned knowledge. In case of transfer learning, it is way more effortless and quicker to adjust an already existing framework rather than going for fresh training. The primary layers within the structures of CNNs tend to hold broad features which can be utilized for various tasks. However, the final layers focus more on specific details for individual applications. Using this property, the initial layers are kept intact, and the final ones are modified in order to train on the new data set [73].

The FASDD and FS datasets are critical to this study in developing a highly effective and versatile fire detection model suitable for safety-critical applications. The design begins with a primary training phase using the FASDD dataset to acquire basic skills in identifying fire and smoke. In the second phase, generalization is enhanced by performing transfer learning on the FS dataset. This is accomplished by adjusting the FASDD-trained model's parameters prior to training on the FS dataset. The model becomes proficient in performing a wide gamut of tasks, leading to enhanced adaptability. Transfer learning also goes a step further by improving the model's accuracy while maintaining tolerance to changes in anthropometric factors including lighting, camera position, or fire and smoke volume. Transfer learning's practicality, scalability, and efficiency in solving these issues demonstrates that it is a superior and dependable method in developing fire and smoke detection systems.

3.8 Performance Evaluation Metrices

The activity of describing how a trained model performs in new situations is called performance evaluation. Defining the recognition problem and selecting the model leads to a variety of techniques and measures that enable us to quantify the effectiveness and the generalization capability of the model. When defining classification performance metrics, it's vital to know what a Confusion Matrix is. A confusion matrix or Error Matrix is a table that is used to describe the behavior of a classification model. A general binary classification confusion matrix is shown in Figure 3.9 which is a 2x2 matrix to understand the terminologies which is useful when finding the performance metrics. A confusion matrix is a nxn matrix where n represents the number of labels in the data. It is an organized and complete representation of the prediction results of a model and its comparison to the real values for the performed analyses and a table version of it. The general 2x2 matrix is composed of four different combinations of predicted and actual classes.

3.8.1 True Positives (TP)

Instances where the model correctly predicts the positive class. It means that the true class label case as in ground truth has correctly predicted by the model as true.

3.8.2 True Negatives (TN)

Instances where the model correctly predicts the negative class. It means that the negative class label case as in ground truth has correctly predicted by the model as negative.

3.8.3 False Positives (FP)

Instances where the model predicts the positive class incorrectly. It's a misclassification case where a true class label as in ground truth is predicted as negative class label by the model.

3.8.4 False Negatives (FN)

Instances where the model predicts the negative class incorrectly. It's a misclassification case where a negative class label as in ground truth is predicted as positive class label by the model.

A confusion matrix is a useful tool to locate classification performance metrics that include overall accuracy, precision, recall/sensitivity, and F1-Score.The details of each performance metric are explained in Table 3.2.

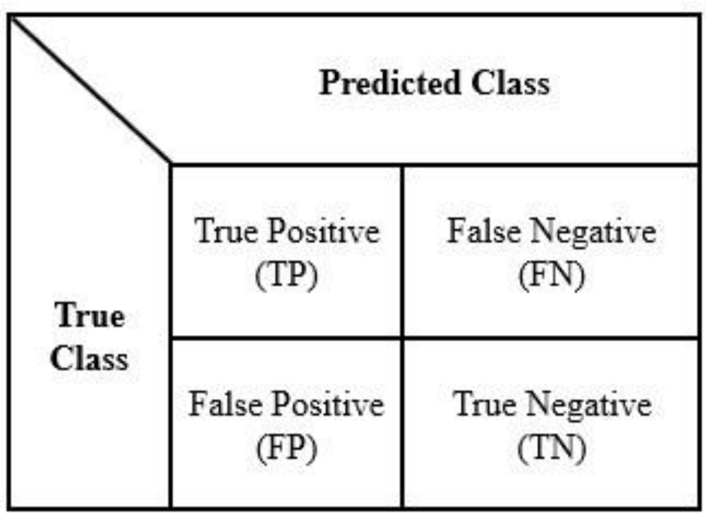


Figure 3.9: A general 2x2 confusion matrix.

Table 3.2: Classification performance metrics

Performance Metric	Definition	Purpose	Formula
Overall Accuracy	The proportion of correct predictions to the total number of predictions.	Offers a comprehensive assessment of the model's accuracy across all predictions.	$\frac{TP + TN}{TP + TN + FP + FN}$
Precision	The proportion of true positive predictions out of the total predicted negative instances.	Evaluate the model's accuracy in predicting positive events, minimizing false positives.	$\frac{TP}{TP + FP}$
Recall/Sensitivity	The ratio of true positives to false negatives and false negatives is the same.	Evaluate the model's ability to identify actual positive events, minimizing false positives.	$\frac{TP}{TP + FN}$
F1-Score	The harmonic means of precision and recall, providing a balanced measure between the two metrics.	Balances precision and recall, resulting in a single metric that identifies both false positives and false negative.	Precision x Recall 2 x $\overline{\text{Precision + Recall}}$

Generalization refers to the skill of a model to cope with new, previously unseen data without any difficulty. The two principal kinds of generalization challenges such as underfitting and overfitting are unwanted when it comes to evaluating a model's effectiveness on unexplored data.

An underfitted model is unable to learn because the patterns provided to it are very simplistic. It is not able to grasp the underlying intricacies and hence performs in a subpar manner during both training and even within the unseen dataset. This could result from a variety of things such as not using suitable algorithms, failure to comprehend the concealed information within the dataset, as well as insufficient training. Underfitting models show high errors rates and low accuracy in comparison to both the training and test datasets. Any combination of attempting to use intelligent algorithms, adding more factors, increasing model strength, or extending training can help deal with underfitting.

Overfitting is usually reflected in models which are too sophisticated and complex to grasp the underlying data patterns. This tends to capture noise or other random fluctuations present within the training dataset. An overfit model would perform exceedingly well on the training whereas totally deficient data.

Encountering noise instead of genuine trends is a possibility with overfitting that occurs during extended training or using overly sophisticated models. Overfitting can also occur at the same time as working with small datasets or training a model that is too specific. A gap in performance between high results of the training phase and remarkably low results while testing or validating the model typically suggests that overfitting has taken place. The smart algorithm should learn the patterns but the model simply retains the details of datasets used for training. In case of overfitting, the model needs to be reworked or rather, there are a range of options such as imposing limits on the model's complexity or using dropout, increasing the records that are present in the dataset, or using cross validation to lessen the chances of overfitting.

Furthermore, the number of computational resources is crucial when executing and deploying the model. To quantify the model's complexity, parameters, flops, fps, inference time, or other relevant parameters are used, as these data aid in preparing the suitable resources for model training and implementing it in real time scenarios. These performance metrics, when combined with computational complexity aid in model efficiency. These metrics are explained as follows:

3.9 Model Parameters

All the fixed parameters that are also known as non-trainable parameters and the updatable parameters (weights and biases) that the model learns during training are divided into a single scale called the model parameters. A model structure where the number of parameters increases creates complexity to the model. Such complex model helps in capturing all complex patterns of the data and increases the accuracy however it increases the computational complexity that results in high utilizing of the hardware resources and that model might not be suitable to deploy for real-time applications. On the other hand, the smaller number of parameters makes the model less computationally expensive, it utilizes less hardware resources. However, in such a model, the accuracy of the model may be limited, but it is suitable for deployment in real-time applications.

3.9.1 Floating-point Operations (Flops)

When a model is trained and tested, the arithmetic operations he performs are calculated in floating point operations per second "flops". As the number of flops increases, the complexity of the model also increases, and this is also true in reverse. Higher flop count models tend to consume more hardware resources and time during execution compared to lower flop count models

3.9.2 Frames Per Seconds (FPS)

Frames per second refers to the rates at which an image is sent to a model within a given template. This parameter is extensively utilized in real-time systems to evaluate a model's feasibility for implementation. A higher metric value implies a greater speed, which is essential for models in this domain that require quick processing. However, a sophisticated model usually requires greater computational power, which is required to assist with greater fps per second and the opposite is true.

3.9.3 Inference Time

Inference time while making new predictions using a trained model is typically the time taken for the predictions. An effective model should have a low inference time to permit usage in real-time applications.

CHAPTER 4: RESULTS AND DISCUSSION

4.1 Case Description

In this study, the models are trained using two different datasets namely FASDD and FS. The case study is divided into two categories given below:

In case 1, the improved yolov8 model is trained on FASDD data using the hyperparameters listed in Table 4.3. The results of model are compared with related study [71].

In case 2, the proposed model is fine-tuned and trained on FS data using the pre trained weight of case (1) through transfer learning to enhance the performance of the model and make the model more generalizable. The hyperparameter settings used for this case study are included in Table 4.5. The experimentational results are then compared to other studies to validate the performance of the model.

4.2 Experimental Setup

The proposed models are equipped with a Tesla T4 GPU (16 GB memory) and an AMD EPYC 7452 32-Core CPU. The operating system used was Linux CentOS. The study utilized PyTorch version 2.0.1, CUDA version 11.7, and Python version 3.9.7 to implement and train the deep learning models. This hardware-software combination ensured efficient training and testing of the improved model.

4.3 Assessing Yolo Models

We examined different YOLO models, including YOLOv8, YOLOv9, and YOLOv10, in their nano, small, and medium configurations. The model's performance was assessed using precision, recall, mAP@50, mAP@50-95, inference speed and process speed. For selecting an optimal base model, the detailed model's comparisons using the above performance metrics, is given in Table 4.1.

Table 4.1: Model comparison for optimal model selection

Sr. No.	Model	Precision	Recall	mAP50	mAP50-95	Inference speed	Process speed
1	YOLOv8n	0.633	0.486	0.525	0.282	2.1ms	4.8ms
2	YOLOv8s	0.816	0.754	0.828	0.587	3.4ms	1.2ms
3	YOLOv9 Custom	0.810	0.728	0.825	0.538	49.0ms	0.9ms
4	YOLOv9 Gelan	0.828	0.732	0.829	0.580	--	--
5	YOLOv10 Medium	0.796	0.711	0.805	0.552	--	--
6	YOLOv8m	0.828	0.740	0.83	0.578	7.0ms	0.7ms

As illustrated in Table 4.1, YOLOv8 Medium (YOLOv8m) achieves the best balance between detection accuracy and processing speed. With a precision of 0.828, recall of 0.740, and mAP@50 of 0.830, YOLOv8m surpasses the other models in terms of overall performance. Moreover, its inference speed of 7.0ms ensures suitability for real-time applications. YOLOv8 Small (YOLOv8s) closely follows, delivering faster processing at 1.2ms, with a minor compromise in accuracy, making it ideal for time-sensitive applications. YOLOv8 Nano, while the fastest at 2.1ms, exhibits a significant reduction in mAP@50-95 (0.282), indicating a notable trade-off in detection accuracy. On the other hand, YOLOv9 Custom and YOLOv10 Medium, although reasonably accurate, lag in both inference and processing speed—especially YOLOv9 Custom with an inference time of 49.0ms—rendering them less suitable for real-time scenarios. The YOLOv8 models, particularly YOLOv8 Medium, emerge as the most efficient choice for fire and smoke detection tasks due to their balance of high precision and speed.

YOLOv8’s streamlined architecture and reduced computational overhead make it highly efficient for real-time fire and smoke detection tasks. With faster inference times and comparable accuracy to more advanced models such as YOLOv9 and YOLOv10, YOLOv8 is suitable for scenarios where rapid detection is critical. Additionally, its lightweight design allows for deployment on embedded systems, which are commonly used in fire detection setups. Thus, YOLOv8 is the most advantageous balance between accuracy, speed, and resource efficiency, making it the most suitable choice for this study.

4.4 Performance Comparison of Attention Modules

Attention mechanisms are essential in enhancing deep learning architectures by enabling models to prioritize essential features while minimizing distractions from irrelevant information. This study focuses on three distinct mechanisms, Efficient Channel Attention (ECA), Shuffle Attention (SA), and Residual Convolutional Block Attention Module (Res-CBAM) to optimize the YOLOv8 model for fire and smoke detection.

Each mechanism was integrated into the model's design and evaluated under identical training conditions. ECA focuses on local cross-channel interactions, improving feature extraction with minimal computational overhead. In contrast, SA combines spatial and channel attention to enhance the model’s adaptability to complex visual patterns. Res-CBAM incorporates both residual learning and attention, balancing feature refinement with computational cost.

These mechanisms were assessed using identical hardware and software configurations and trained for two hundred epochs with a batch size of 16. The optimizer and learning rate schedules were fine-tuned to ensure consistent evaluation. This systematic approach enables a clear comparison of their performance on fire and smoke detection tasks.

Table 4.2: Comparison of different attention mechanisms

Model	Images	Model GFLOPS	Precision	Recall	mAP50	mAP50-95
YOLOv8m-SA	2114	78.7	0.77	0.72	0.75	0.45
YOLOv8m-RES_CBAM	2114	97.8	0.88	0.82	0.88	0.62
YOLOv8m-ECA	2114	78.7	0.88	0.83	0.89	0.61

Table 4.2 highlights the performance comparison of three attention mechanisms integrated into the YOLOv8m model. The results demonstrate that the Efficient Channel Attention (ECA) mechanism achieves the most balanced trade-off between accuracy and computational efficiency. With a precision of 0.88, recall of 0.83, and mAP@50 of 0.89, ECA outperforms the Shuffle Attention (SA) mechanism while maintaining a significantly lower computational cost than Res-CBAM.

Res-CBAM delivers the highest mAP@50 (0.88) and mAP@50-95 (0.62), but its higher computational demand (97.8 GFLOPS) limits its applicability in real-time or resource-constrained settings. In contrast, SA, though computationally lightweight, records lower precision (0.77) and recall (0.72), making it less suitable for complex detection tasks.

Overall, YOLOv8m-ECA emerges as the optimal choice for fire and smoke detection applications, striking a critical balance between detection accuracy and efficiency. This advantage makes it particularly effective for deployment in embedded systems where resource constraints are a significant consideration.

To evaluate the effectiveness of the modifications discussed, including the integration of attention mechanisms and C3Ghost convolution, we conducted extensive experiments using two datasets (FASDD and FS). The results presented in this section compare the performance of the YOLOv8 variants in terms of detection accuracy, computational efficiency, and inference speed. The following subsections detail the quantitative evaluation and comparison with existing state-of-the-art models.

4.5 Experimentation of Proposed Model in FASDD (Case-1)

To evaluate our proposed YOLOv8m-ECA model, we compared its performance on the FASDD dataset against several state-of-the-art architectures. The FASDD dataset, introduced in Section 3.1, includes a variety of fire and smoke images from both indoor and outdoor environments, split into training, validation, and testing subsets as detailed in Table 3.1. For consistency with prior work [71], we maintained the same hyperparameter settings, which are summarized in Table 4.3.

Table 4.3: Hyperparameters settings for (Case-1)

Hyperparameters	Values
Epochs	36
Batch size	8
Optimizer	SGD with Momentum: 0.85
Learning rate	Initial: 0.005, final: 0.01
Image- Size	640*640

Following the completion of training, the YOLOv8m-ECA model was rigorously evaluated to determine its performance across key metrics, including mean Average Precision (mAP@50), precision, recall, and F1-score. These metrics provide a comprehensive assessment of the model's ability to accurately detect and classify fire and smoke in various scenarios, particularly in complex indoor and outdoor environments.

On the FASDD dataset, the model demonstrated robust performance, achieving an overall mAP@50 score of 0.799, with a fire detection mAP of 0.75 and a smoke detection mAP of 0.84. Furthermore, it recorded precision, recall, and F1-scores of 0.816, 0.715, and 0.76, respectively, reflecting a robust balance between detection accuracy and consistency. The precision, recall, and F1-score curves, illustrated in Figure 4.1, visually capture the model's progression and performance trends throughout the training process.

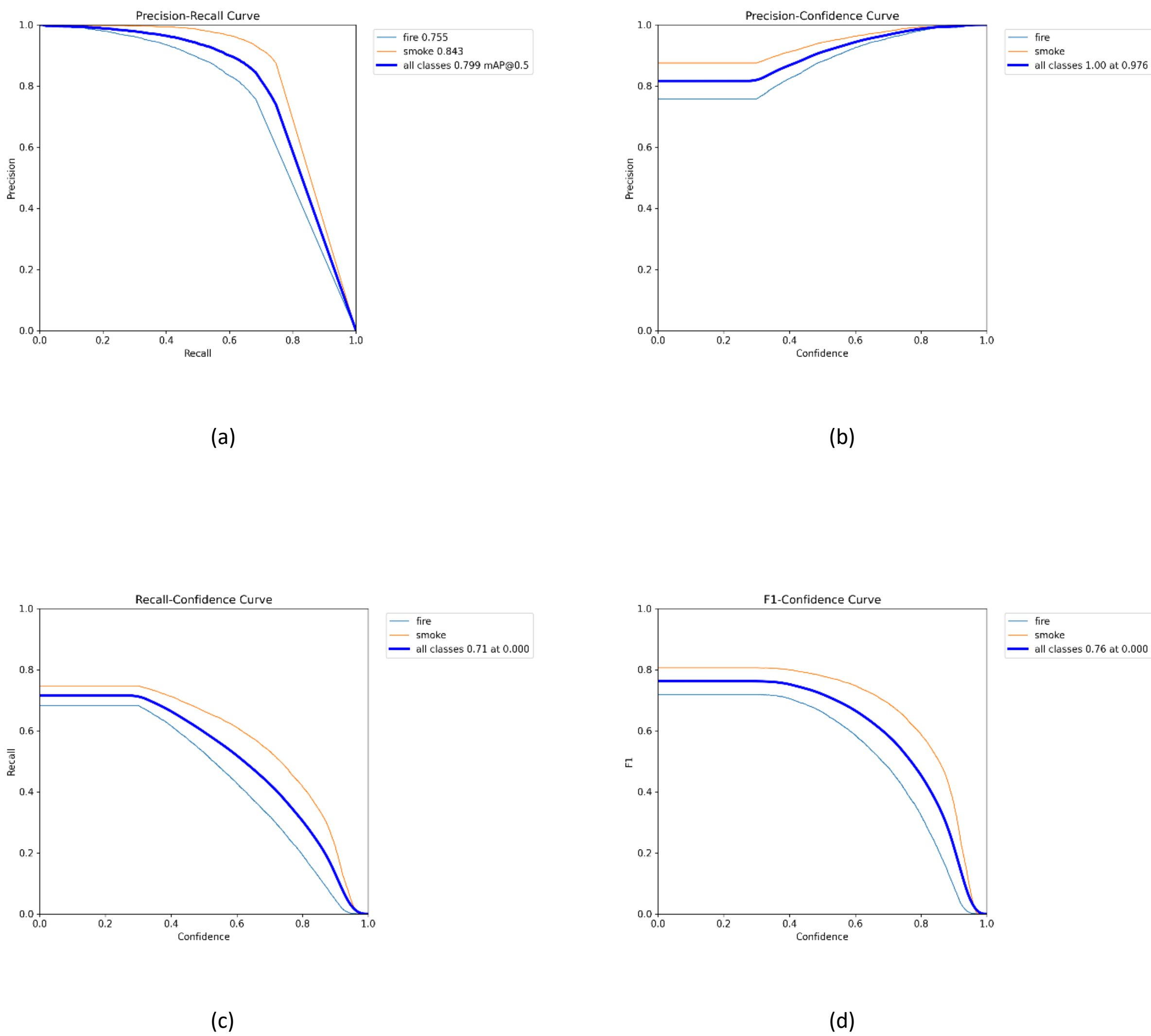
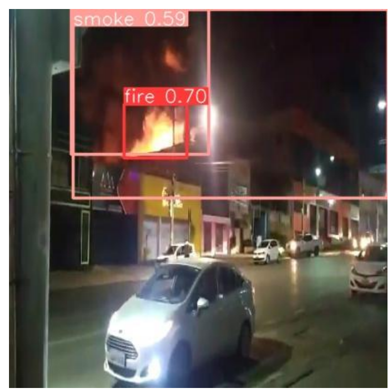


Figure 4.1: Confidence curves (a) mAP:50 score, (b) Precision curve, (c) Recall, (d) F1-score

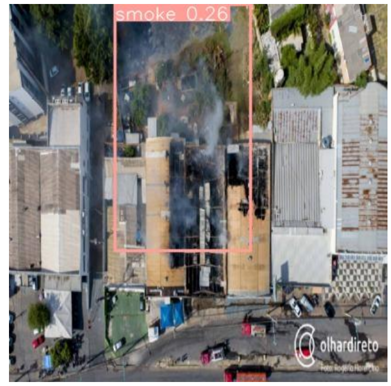
To further validate the model, we tested it on unseen data for fire and smoke detection across different challenging environments. Figure 4.2 illustrates sample detection results, highlighting the robustness of YOLOv8m-ECA in various scenes, including day and night views, urban settings, and aerial perspectives. For instance, in a daytime city scene, the model accurately detected fire and smoke with confidence scores of 0.76 and 0.87, respectively Figure 4.2(a). In a more complex night scene, the model achieved fire and smoke detection with confidence scores of 0.70 and 0.59, respectively Figure 4.2(b), demonstrating strong adaptability in low-light conditions. Additional examples in Figure 4.2(c, d) demonstrate smoke detection in challenging night and aerial views, with confidence scores of 0.58 and 0.26, further enhancing the model's generalizability across different scenarios.

(a) Fire and smoke detected in Daytime



(b) Fire and smoke in Night-time

(c) Smoke detected in Night-time



(d) Smoke detected in Aerial view

Figure 4.2: Fire and smoke detected by proposed YOLOv8m-ECA model

4.5.1 Comparative Analysis of Proposed Model with Related Studies

To analyze the results of our proposed model, a detailed comparison is made with related studies illustrated in Table. 4.4 against other prominent models used for fire and smoke detection on the FASDD dataset, specifically comparing average precision (AP) for fire and smoke, recall, and mAP@50. Our proposed YOLOv8m-ECA model achieved a mAP@50 score of 79.9, surpassing YOLOv5x (75.5), Intern-Image (78.1), DETR (74.5), and Swin Transformer (78.1), representing a 5.83% improvement over YOLOv5x, 2.3% over Intern-Image and Swin Transformer, and 7.25% over DETR. Notably, YOLOv8m-ECA achieved the highest AP for smoke detection (87.5) and competitive AP for fire detection (75.8), outperforming other models in smoke detection accuracy.

While the YOLOv8m-ECA model shows slightly lower recall scores compared to models like InternImage and Swin Transformer, it achieves a favorable trade-off between precision and recall, optimized for real-time applications. This performance balance, coupled with its lightweight architecture, makes YOLOv8m-ECA particularly suitable for embedded systems requiring both high accuracy and computational efficiency, such as fire and smoke detection in smart building environments.

Table 4.4: Comparison of proposed model with state of the art

Dataset	**Batch-size**	**Epochs**	**Models**	**AP (fire)**	**AP (smoke)**	**Recall (fire)**	**Recall (smoke)**	**mAP@50**
FASDD	8	36	Yolov5x	73.8	77.2	75.0	61.0	75.5
			InternImage	74.0	82.2	90.7	90.8	78.1
			DETR	68.8	80.3	89.9	96.2	74.5
			Swin Transformer	72.8	83.4	91.5	92.8	78.1
			Proposed Model (Yolov8m-ECA)	75.8	87.5	68.3	74.7	79.9

4.6 Experimentation of Proposed Model on FS Dataset (Case-2)

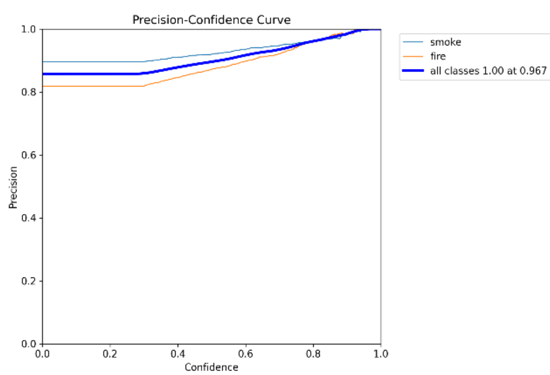
To further fine tune and optimize the proposed models, YOLOv8m-ECA-C3Ghost: model with medium configuration and YOLOv8n-ECA-C3Ghost: model with nano configuration is further trained on FS dataset using transfer learning. For transfer learning, the weights of the pre trained model on FASDD data are used with the hyperparameters settings as shown in Table 4.5 The dataset is split into training, validation, and testing subsets, as shown in Table 3.1.

Table 4.5: Hyperparameter Settings for (Case-2)

Hyperparameters	Values
Epochs	300
Batch size	32
Optimizer	SGD with Momentum: 0.85
Learning rate	Initial: 0.005, final: 0.01
Image size	640*640

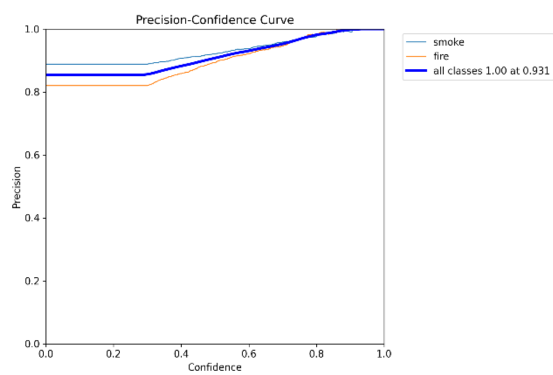
After training, YOLOv8m-ECA-C3Ghost achieved excellent performance, with an overall mAP@50 of 0.889 (0.866 for fire and 0.912 for smoke), precision of 0.892, recall of 0.831, and F1-score of 0.86 on the FS dataset. The YOLOv8n-ECA-C3Ghost model achieved a slightly lower overall mAP@50 of 0.853, with precision at 0.854, recall at 0.805, and F1-score of 0.83. The precision, recall, mAP@50, and F1-score curves are shown in Figure 4.4, illustrating the models' performance over the training epochs.

Yolov8m – ECA – C3Ghost

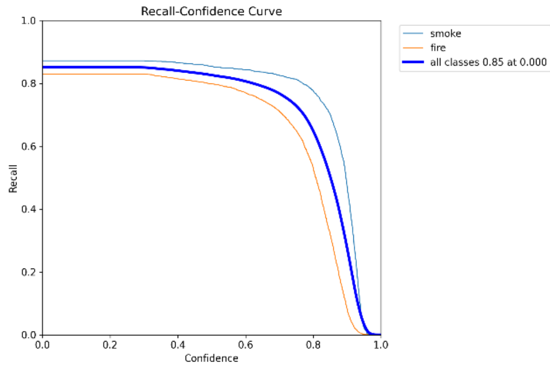


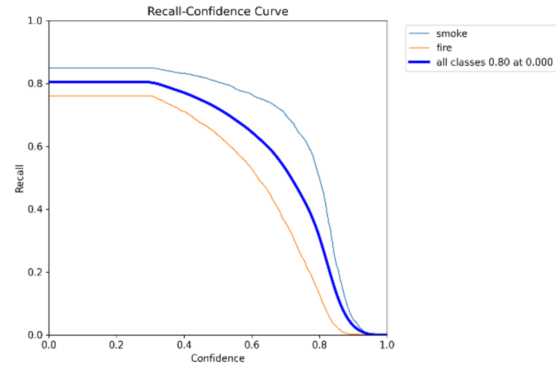
(a) Precision

Yolov8n – ECA – C3Ghost



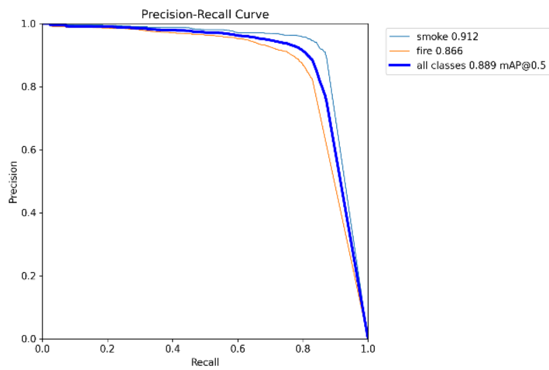
(b) Precision

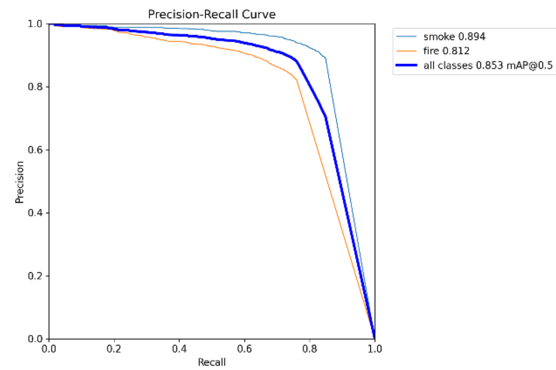




(c) Recall

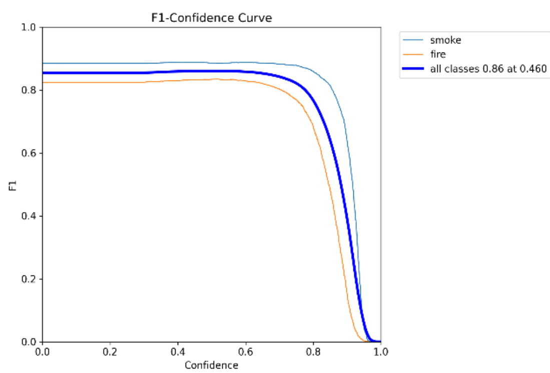
(d) Recall

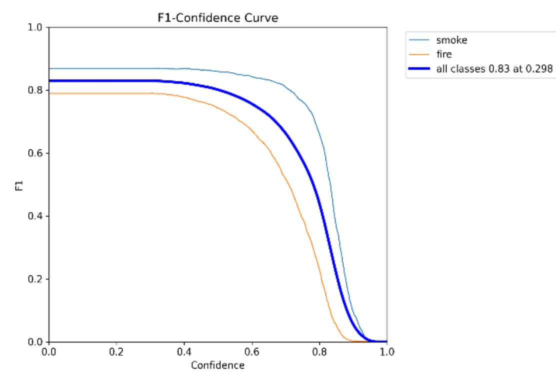




(e) Precision-Recall

(f) Precision-Recall





(g) F1- score

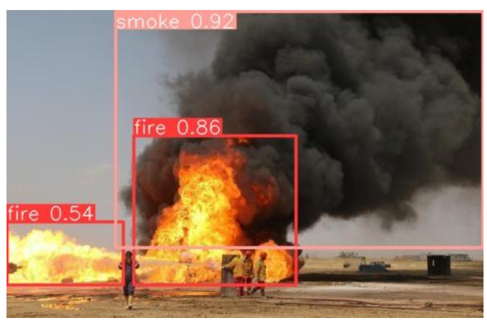
(h) F1- score

Figure 4.4: Performance metrics of proposed YOLOv8-ECA-C3Ghost models.

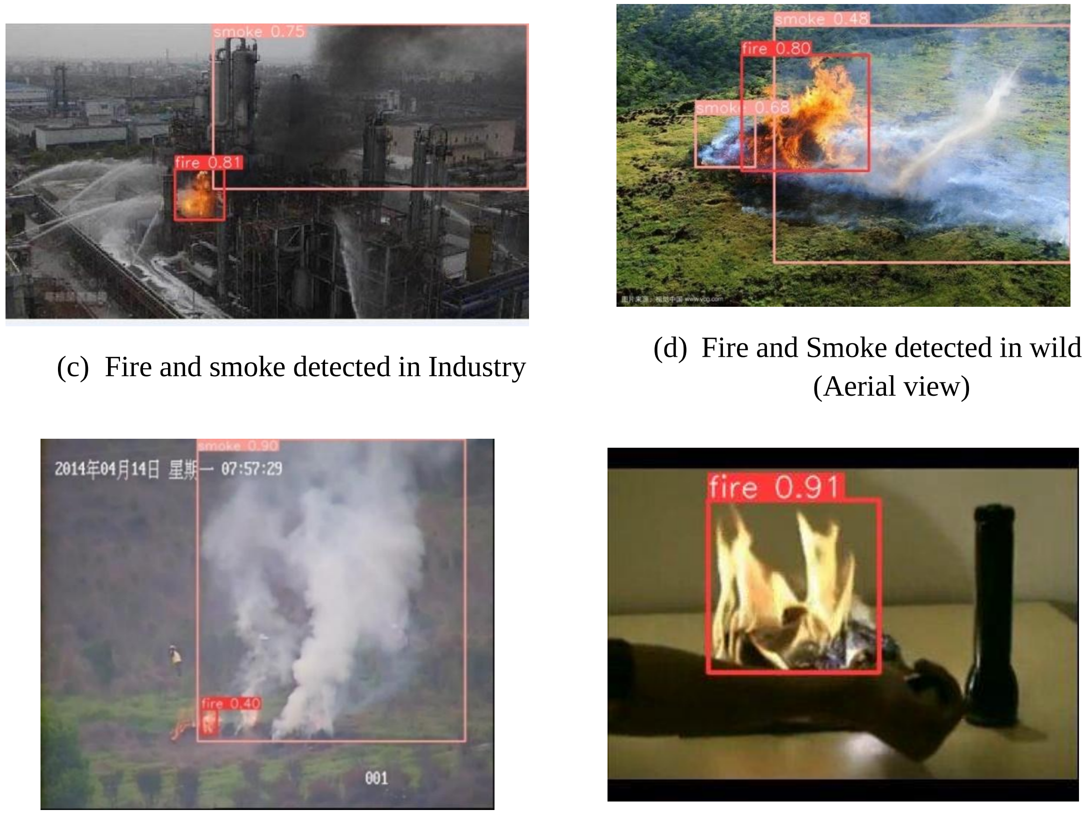
4.6.1 Inference on Diverse Test Scenarios

To evaluate the real-world applicability of our models, we tested YOLOv8m-ECA-C3Ghost and YOLOv8n-ECA-C3Ghost on previously unseen data across a range of challenging scenarios. These scenarios included diverse settings such as indoor and outdoor daytime environments, industrial facilities, and aerial views of wildfires. Figure 4.5 presents representative detections: in an indoor daytime setting, the models achieved confidence scores of 91% for fire and 92% for smoke, highlighting their accuracy under well-lit conditions. In outdoor scenes, the models maintained high detection accuracy, and in industrial settings, they effectively identified fire and smoke, adapting to complex backgrounds and potential visual noise. Additionally, aerial views of wildfires were detected accurately, as were long-distance and close-range scenes, demonstrating the models' adaptability across varying spatial and lighting conditions. These examples underscore the robustness and generalizability of our models, as they consistently performed well across diverse environments. The results confirm that YOLOv8m-ECA-C3Ghost and YOLOv8n-ECA-C3Ghost can maintain high detection accuracy and reliability in real-world applications, where environmental factors may vary significantly.

(a) Fire and smoke detected in Daytime (Indoor)



(b) Fire and smoke in Daytime (Outdoor)



(c) Fire and smoke detected in Industry

(d) Fire and Smoke detected in wild (Aerial view)

(e) Fire and smoke in Far away scene

(f) Fire detection in near scene

Figure 4.5: Inference of fire and smoke detection in mixed scenes.

4.6.2 Comparative Analysis with Related Studies (Case-2)

Table 4.6. provides a comparative analysis of advanced fire and smoke detection models, highlighting that our **YOLOv8m-ECA-C3Ghost** and **YOLOv8n-ECA-C3Ghost** models achieve an exceptional balance of accuracy, speed, and computational efficiency, making them ideal for real-time detection in resource-constrained environments. While models like **Zhao et al. 2022** and **Chetoui et al. 2024** show strong precision and recall, but due to a lack of data on FPS and Flops, which are critical for real-time applications. For instance, **Chetoui et al. 2024** achieves the highest recall (0.952) and F1-Score (0.89), but such high recall can sometimes result in false positives, reducing reliability in real-world scenarios where balanced precision and recall are needed. Additionally, **Ma et al. 2023**

and **Yang et al. 2023** demonstrate improvements in both precision and recall but still fall short in real-time efficiency and computational cost.

In contrast, our **YOLOv8m-ECA-C3Ghost** offers superior mAP50 (0.891), the highest in Table 4.6, indicating better detection accuracy across varying IoU thresholds, while maintaining an excellent FPS of 113 and Flops of 78.7G making it a versatile, high-performing model for complex detection tasks. Similarly, **YOLOv8n-ECA-C3Ghost** balances performance with computational efficiency, achieving 277 FPS and only 6.6G Flops, outperforming other models in real-world scenarios where speed and resource efficiency are critical. Although some models might slightly outperform ours in isolated metrics like precision or recall, the overall superiority of **YOLOv8m-ECA-C3Ghost** and **YOLOv8n-ECA-C3Ghost** lies in their ability to provide high accuracy, exceptional speed, and low computational overhead, ensuring generalizability and robustness for real-ire and smoke detection applications.

Table 4.6: Performance comparison of our models with recent state of the art models for fire and smoke detection.

Model	No. of Images	F1-Score	Precision	Recall	mAP50	mAP50-95	FPS	FLOPs(G)
Zhao et al. 2022	19819	0.73	0.915	0.596	0.802	---	-----	-----
Ma et al. 2023	4998	0.83	0.83	0.83	0.87	0.57	33	14.8
Yang et al. 2023	11667	0.85	0.892	0.827	0.873	0.566	----	----
Xu et al. 2024	2058	0.83	0.861	0.818	0.883	------	----	---
YOLOv8m-ECA-C3Ghost	11667	0.86	0.892	0.83	0.889	0.67	97	53

YOLOv8n-ECA-C3Ghost	11667	0.83	0.854	0.805	0.853	0.597	277	6.6

4.7 Discussion

Traditional image processing techniques and early deep learning models for fire and smoke detection often struggle with generalization, especially when applied to diverse, unseen scenarios. They also present challenges in real-time applications due to high computational demands, making them impractical for deployment on resource-constrained devices. This study addresses these issues by introducing a lightweight YOLOv8-based model incorporating Efficient Channel Attention (ECA) and C3Ghost convolution. The proposed model demonstrates high detection accuracy across both indoor and outdoor environments, achieving a competitive mAP@50 of 89% on the FS dataset while maintaining a significantly lower computational cost compared to state-of-the-art models. This balance of accuracy and efficiency underscores the model's adaptability to real-world scenarios, including resource-constrained platforms like Raspberry Pi and Jetson Nano. The ability to achieve high frames per second (FPS) rates further validates its potential for real-time deployment. Notably, the inclusion of ECA enables the model to focus on critical features without adding significant computational overhead, while the C3Ghost module ensures efficient feature extraction. Compared to similar studies, such as the integration of Res-CBAM, our model provides a more resource-efficient alternative with comparable accuracy, bridging the gap between performance and practicality.

However, some limitations persist. The model's lightweight nature, while advantageous for embedded devices, has limited its ability to capture intricate details in overly complex fire and smoke patterns. Future research should explore more advanced attention mechanisms or hybrid architectures to further enhance detection capabilities without sacrificing computational efficiency. Additionally, testing under dynamic real-world conditions, such as varying weather or lighting, is necessary to validate its robustness comprehensively

CHAPTER 5: CONCLUSIONS

This study introduces a lightweight YOLOv8-based deep learning model, enhanced with Efficient Channel Attention (ECA) and C3Ghost convolution, tailored for real-time fire and smoke detection. The model addresses critical challenges associated with related methods, including high computational demands and limited generalizability. By leveraging two diverse datasets, FASDD and FS, the proposed model achieves a competitive mAP@50 of 89% and demonstrates robust performance across varied indoor and outdoor scenarios. Its computational efficiency and adaptability make it particularly suited for deployment on embedded systems, enabling practical applications in resource-constrained and safety-critical environments. The integration of ECA and C3Ghost modules significantly improves the model's ability to focus on relevant features while maintaining low computational overhead, ensuring high detection accuracy without compromising speed. This advancement bridges the gap between accuracy and efficiency, paving the way for more effective real-time fire and smoke detection systems in smart buildings, industrial facilities, and remote wildfire monitoring setups.

Despite these advancements, the lightweight nature of the model presents limitations in detecting highly intricate fire and smoke patterns in complex scenarios. Future research will focus on incorporating advanced attention mechanisms and hybrid architectures to enhance feature extraction capabilities. Additionally, extending the scope of testing to include real-world conditions, such as varying weather, dynamic lighting, and diverse environmental challenges, will further validate the model's robustness and reliability. Expanding the datasets with more diverse fire and smoke scenarios will also improve generalizability, ensuring the model's adaptability to unforeseen situations. Overall, this study underscores the importance of balancing computational efficiency and detection accuracy, offering a practical, scalable solution for real-time fire and smoke detection in embedded environments. Its potential to enhance safety systems in a variety of settings marks a significant step forward in fire hazard mitigation technologies.

REFERENCES

[1] R. Simone Scardapane Relatore Esterno Giovanni Trovini, "Fire and Smoke Detection for Smart Cities Using Deep Neural Networks and Edge Computing on Embedded Sensors," 2019.

[2] *2017 2nd International Conference for Convergence in Technology (I2CT)*. IEEE, 2017.

[3] Y. Xie *et al.*, "Tripling of western US particulate pollution from wildfires in a warming climate," 2022, doi: 10.1073/pnas.

[4] S. Tariq, A. Mariam, Z. ul-Haq, and U. Mehmood, "Spatial and temporal variations in PM2.5 and associated health risk assessment in Saudi Arabia using remote sensing," *Chemosphere,* vol. 308, Dec. 2022, doi: 10.1016/j.chemosphere.2022.136296.

[5] G. Di Virgilio, M. A. Hart, A. M. Maharaj, and N. Jiang, "Air quality impacts of the 2019–2020 Black Summer wildfires on Australian schools," *Atmos Environ*, vol. 261, Sep. 2021, doi: 10.1016/j.atmosenv.2021.118450.

[6] G. R. Van Der Werf *et al.*, "Global fire emissions estimates during 1997-2016," Sep. 12, 2017, *Copernicus GmbH*. doi: 10.5194/essd-9-697-2017.

[7] S. D. Griffiths, J. A. Entwistle, F. J. Kelly, and M. E. Deary, "Characterising the ground level concentrations of harmful organic and inorganic substances released

during major industrial fires, and implications for human health," *Environ Int*, vol. 162, Apr. 2022, doi: 10.1016/j.envint.2022.107152.

[8] S. Alkış, E. Aksoy, and K. Akpınar, "Risk assessment of industrial fires for surrounding vulnerable facilities using a multi-criteria decision support approach and gis," *Fire*, vol. 4, no. 3, Sep. 2021, doi: 10.3390/fire4030053.

[9] M. A. Ibrahim, "Risk of spontaneous and anthropogenic fires in waste management chain and hazards of secondary fires," *Resour Conserv Recycl*, vol. 159, p. 104852, Aug. 2020, doi: 10.1016/J.RESCONREC.2020.104852.

[10] T. Shavit, S. Shahrabani, U. Benzion, and M. Rosenboim, "The effect of a forest fire disaster on emotions and perceptions of risk: A field study after the Carmel fire," *J Environ Psychol*, vol. 36, pp. 129–135, Dec. 2013, doi: 10.1016/j.jenvp.2013.07.018.

[11] "A Study on the Response Characteristics of Fire Detector by Full-scale Experiment of Fire Phenomena in the Row House".

[12] J. H. Klote and P. Fellow ASHRAE, "CFO Simulations of the Effects of HVAC-lnduced Flows on Smoke Detector Response."

[13] T. T. Aralt and A. R. Nilsen, "Automatic fire detection in road traffic tunnels," *Tunnelling and Underground Space Technology*, vol. 24, no. 1, pp. 75–83, Jan. 2009, doi: 10.1016/j.tust.2008.04.001.

[14] S. Festag, “False alarm ratio of fire detection and fire alarm systems in Germany - A meta analysis,” Jan. 01, 2016, *Elsevier Ltd*. doi: 10.1016/j.firesaf.2015.11.010.

[15] J. Dinaburg and D. Gottuk, “Smoke Alarm Nuisance Source Characterization: Review and Recommendations,” *Fire Technol*, vol. 52, no. 5, pp. 1197–1233, Sep. 2016, doi: 10.1007/s10694-015-0502-1.

[16] T. Toulouse, L. Rossi, A. Campana, T. Celik, and M. A. Akhloufi, “Computer vision for wildfire research: An evolving image dataset for processing and analysis,” *Fire Saf J*, vol. 92, pp. 188–194, Sep. 2017, doi: 10.1016/j.firesaf.2017.06.012.

[17] J. Chen, Y. He, and J. Wang, “Multi-feature fusion based fast video flame detection,” *Build Environ*, vol. 45, no. 5, pp. 1113–1122, May 2010, doi: 10.1016/j.buildenv.2009.10.017.

[18] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, “An Early Fire-Detection Method Based on Image Processing.”

[19] O. Günay, K. Taşdemir, B. U. Töreyin, and A. E. Çetin, “Fire detection in video using LMS based active learning,” *Fire Technol*, vol. 46, no. 3, pp. 551–577, Jul. 2010, doi: 10.1007/s10694-009-0106-8.

[20] X. Yang, Y. Shi, L. Chen, and Z. Quan, “The lifting scheme for wavelet Bi-frames: Theory, structure, and algorithm,” *IEEE Transactions on Image Processing*, vol. 19, no. 3, pp. 612–624, Mar. 2010, doi: 10.1109/TIP.2009.2038762.

[21] W. Phillips, M. Shah, N. Da, and V. Lobo, “Flame recognition in video q.” [Online]. Available: www.elsevier.com/locate/patrec

[22] C.-B. Liu, N. Ahuja, and B. Institute, “Vision Based Fire Detection.”

[23] B. U. Toreyin, Y. Dedeoglu, and A. E. Cetin, “CONTOUR BASED SMOKE DETECTION IN VIDEO USING WAVELETS.”

[24] T. Celik, H. Demirel, H. Ozkaramanli, and M. Uyguroglu, “Fire detection using statistical color model in video sequences,” *J Vis Commun Image Represent*, vol. 18, no. 2, pp. 176–185, Apr. 2007, doi: 10.1016/j.jvcir.2006.12.003.

[25] X. Qi and J. Ebert, “A Computer Vision-Based Method for Fire Detection in Color Videos.”

[26] I. Kolesov, P. Karasev, A. Tannenbaum, and E. Haber, “FIRE AND SMOKE DETECTION IN VIDEO WITH OPTIMAL MASS TRANSPORT BASED OPTICAL FLOW AND NEURAL NETWORKS.”

[27] B. Lee and D. Han, “Real-Time Fire Detection Using Camera Sequence Image in Tunnel Environment,” 2007.

[28] P. Foggia, A. Saggese, and M. Vento, “Real-Time Fire Detection for Video-Surveillance Applications Using a Combination of Experts Based on Color, Shape, and Motion,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 9, pp. 1545–1556, Sep. 2015, doi: 10.1109/TCSVT.2015.2392531.

[29] X. F. Han, J. S. Jin, M. J. Wang, W. Jiang, L. Gao, and L. P. Xiao, "Video fire detection based on Gaussian Mixture Model and multi-color features," *Signal Image Video Process*, vol. 11, no. 8, pp. 1419–1425, Nov. 2017, doi: 10.1007/s11760-017-1102-y.

[30] F. Gong *et al.*, "A real-time fire detection method from video with multifeature fusion," *Comput Intell Neurosci*, vol. 2019, 2019, doi: 10.1155/2019/1939171.

[31] A. Gagliardi and S. Saponara, "Advised: Advanced video smoke detection for real-time measurements in antifire indoor and outdoor systems," *Energies (Basel),* vol. 13, no. 8, Apr. 2020, doi: 10.3390/en13082098.

[32] A. Gagliardi, F. de Gioia, and S. Saponara, "A real-time video smoke detection algorithm based on Kalman filter and CNN," *J Real Time Image Process*, vol. 18, no. 6, pp. 2085–2095, Dec. 2021, doi: 10.1007/s11554-021-01094-y.

[33] A. Khalil, S. U. Rahman, F. Alam, I. Ahmad, and I. Khalil, "Fire Detection Using Multi Color Space and Background Modeling," *Fire Technol*, vol. 57, no. 3, pp. 1221–1239, May 2021, doi: 10.1007/s10694-020-01030-9.

[34] Wahyono, A. Harjoko, A. Dharmawan, F. D. Adhinata, G. Kosala, and K. H. Jo, "Real-Time Forest Fire Detection Framework Based on Artificial Intelligence Using Color Probability Model and Motion Feature Analysis," *Fire*, vol. 5, no. 1, Feb. 2022, doi: 10.3390/fire5010023.

[35] Q. Zhang, J. Xu, L. Xu, and H. Guo, "Deep Convolutional Neural Networks for Forest Fire Detection," 2016.

[36] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks.” [Online]. Available: http://code.google.com/p/cuda-convnet/

[37] *Proceedings of the IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society : Florence (Italy), October 24-27, 2016*. IEEE, 2016.

[38] Y. Luo, L. Zhao, P. Liu, and D. Huang, “Fire smoke detection algorithm based on motion characteristic and convolutional neural networks,” *Multimed Tools Appl*, vol. 77, no. 12, pp. 15075–15092, Jun. 2018, doi: 10.1007/s11042-017-5090-2.

[39] G. Boracchi, L. Iliadis, C. Jayne, and A. Likas, Eds., *Engineering Applications of Neural Networks*, vol. 744. in Communications in Computer and Information Science, vol. 744. Cham: Springer International Publishing, 2017. doi: 10.1007/978-3-319-65172-9.

[40] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, “Convolutional Neural Networks Based Fire Detection in Surveillance Videos,” *IEEE Access*, vol. 6, pp. 18174–18183, Mar. 2018, doi: 10.1109/ACCESS.2018.2812835.

[41] C. Szegedy *et al.*, “Going Deeper with Convolutions,” Sep. 2014, [Online]. Available: http://arxiv.org/abs/1409.4842

[42] S. Aslan, U. Güdükbay, B. U. Töreyin, and A. E. Çetin, “Deep Convolutional Generative Adversarial Networks Based Flame Detection in Video,” Feb. 2019, [Online]. Available: http://arxiv.org/abs/1902.01824

[43] B. Kim and J. Lee, "A video-based fire detection using deep learning models," *Applied Sciences (Switzerland)*, vol. 9, no. 14, Jul. 2019, doi: 10.3390/app9142862.

[44] Y. Li, A. Wu, N. Dong, J. Han, and Z. Lu, "Smoke Recognition Based on Deep Transfer Learning and Lightweight Network."

[45] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," Apr. 2017, [Online]. Available: http://arxiv.org/abs/1704.04861

[46] *Proceedings of the 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA 2020) : 5-7, November 2020*. IEEE, 2020.

[47] S. Khan *et al.*, "DeepSmoke: Deep learning model for smoke detection and segmentation in outdoor environments," *Expert Syst Appl*, vol. 182, Nov. 2021, doi: 10.1016/j.eswa.2021.115125.

[48] G. Lin, Y. Zhang, G. Xu, and Q. Zhang, "Smoke Detection on Video Sequences Using 3D Convolutional Neural Networks," *Fire Technol*, vol. 55, no. 5, pp. 1827–1847, Sep. 2019, doi: 10.1007/s10694-019-00832-w.

[49] P. Li and W. Zhao, "Image fire detection algorithms based on convolutional neural networks," *Case Studies in Thermal Engineering*, vol. 19, Jun. 2020, doi: 10.1016/j.csite.2020.100625.

[50] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans Pattern Anal Mach Intell*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[51] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object Detection via Region-based Fully Convolutional Networks." [Online]. Available: https://github.com/daijifeng001/r-fcn.

[52] W. Liu *et al.*, "SSD: Single Shot MultiBox Detector," Dec. 2015, doi: 10.1007/978-3-319-46448-0_2.

[53] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement." [Online]. Available: https://pjreddie.com/yolo/.

[54] S. Saponara, A. Elhanashi, and A. Gagliardi, "Real-time video fire/smoke detection based on CNN in antifire surveillance systems," *J Real Time Image Process*, vol. 18, no. 3, pp. 889–900, Jun. 2021, doi: 10.1007/s11554-020-01044-0.

[55] K. Avazov, M. Mukhiddinov, F. Makhmudov, and Y. I. Cho, "Fire detection method in smart city environments using a deep-learning-based approach," *Electronics (Switzerland)*, vol. 11, no. 1, Jan. 2022, doi: 10.3390/electronics11010073.

[56] Z. Xue, H. Lin, and F. Wang, "A Small Target Forest Fire Detection Model Based on YOLOv5 Improvement," *Forests*, vol. 13, no. 8, Aug. 2022, doi: 10.3390/f13081332.

[57] Y. Hu *et al.*, “Fast forest fire smoke detection using MVMNet,” *Knowl Based Syst*, vol. 241, Apr. 2022, doi: 10.1016/j.knosys.2022.108219.

[58] O. Khudayberdiev, J. Zhang, S. M. Abdullahi, and S. Zhang, “Light-FireNet: an efficient lightweight network for fire detection in diverse environments,” *Multimed Tools Appl*, vol. 81, no. 17, pp. 24553–24572, Jul. 2022, doi: 10.1007/s11042-022-12552-5.

[59] S. N. Saydirasulovich, M. Mukhiddinov, O. Djuraev, A. Abdusalomov, and Y. I. Cho, “An Improved Wildfire Smoke Detection Based on YOLOv8 and UAV Images,” *Sensors (Basel)*, vol. 23, no. 20, Oct. 2023, doi: 10.3390/s23208374.

[60] J. Ma, Z. Zhang, W. Xiao, X. Zhang, and S. Xiao, “Flame and Smoke Detection Algorithm Based on ODConvBS-YOLOv5s,” *IEEE Access*, vol. 11, pp. 34005–34014, 2023, doi: 10.1109/ACCESS.2023.3263479.

[61] Y. He, J. Hu, M. Zeng, Y. Qian, and R. Zhang, “DCGC-YOLO: The Efficient Dual-Channel Bottleneck Structure YOLO Detection Algorithm for Fire Detection,” *IEEE Access*, vol. 12, pp. 65254–65265, 2024, doi: 10.1109/ACCESS.2024.3385856.

[62] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, “CSPNet: A New Backbone that can Enhance Learning Capability of CNN.”

[63] C.-Y. Wang, H.-Y. M. Liao, and I.-H. Yeh, “Designing Network Design Strategies Through Gradient Path Analysis,” Nov. 2022, [Online]. Available: http://arxiv.org/abs/2211.04800

[64] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors." [Online]. Available: https://github.com/

[65] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection."

[66] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path Aggregation Network for Instance Segmentation."

[67] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module."

[68] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks."

[69] Q. L. Zhang and Y. Bin Yang, "SA-Net: Shuffle attention for deep convolutional neural networks," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 2235–2239. doi: 10.1109/ICASSP39728.2021.9414568.

[70] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More Features from Cheap Operations."

[71] M. Wang, P. Yue, L. Jiang, D. Yu, T. Tuo, and J. Li, "An open flame and smoke detection dataset for deep learning in remote sensing based fire detection," *Geo-Spatial Information Science*, 2024, doi: 10.1080/10095020.2024.2347922.

[72] J. Yang, W. Zhu, T. Sun, X. Ren, and F. Liu, "Lightweight forest smoke and fire detection algorithm based on improved YOLOv5," *PLoS One*, vol. 18, no. 9 SEPTEMBER, Sep. 2023, doi: 10.1371/journal.pone.0291359.

[73] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," Nov. 2014, [Online]. Available: http://arxiv.org/abs/1411.1792