frontiers | Frontiers in Computer Science

TYPE Original Research
PUBLISHED 15 August 2025
DOI 10.3389/fcomp.2025.1636758

OPEN ACCESS

EDITED BY
Francisco Gomez-Donoso,
University of Alicante, Spain

REVIEWED BY
Edmanuel Cruz,
Technological University of Panama, Panama
Buhari Umar,
Federal University of Technology Minna, Nigeria

*CORRESPONDENCE
Uttam U. Deshpande
✉ uttamudeshpande@gmail.com
Goh Kah Ong Michael
✉ michael.goh@mmu.edu.my

RECEIVED 28 May 2025
ACCEPTED 28 July 2025
PUBLISHED 15 August 2025

CITATION
Deshpande UU, Michael GKO, Araujo SDCS, Srinivasaiah SH, Malawade H, Kulkarni Y and Desai Y (2025) Real-time fire and smoke detection system for diverse indoor and outdoor industrial environmental conditions using a vision-based transfer learning approach.
Front. Comput. Sci. 7:1636758.
doi: 10.3389/fcomp.2025.1636758

Real-time fire and smoke detection system for diverse indoor and outdoor industrial environmental conditions using a vision-based transfer learning approach

Uttam U. Deshpande[1]*, Goh Kah Ong Michael[2]*, Sufola Das Chagas Silva Araujo[3], Sowmyashree H. Srinivasaiah[4], Harshel Malawade[1], Yash Kulkarni[1] and Yash Desai[1]

[1]Department of Electronics and Communication Engineering, KLS Gogte Institute of Technology, Belgaum, India, [2]Center for Image and Vision Computing, COE for Artificial Intelligence, Faculty of Information Science and Technology, Multimedia University, Melaka, Malaysia, [3]Department of Computer Science and Engineering, Padre Conceição College of Engineering, Goa, India, [4]Department of Computer Applications, Bangalore Institute of Technology, Bengaluru, India

The risk of fires in both indoor and outdoor scenarios is constantly rising around the world. The primary goal of a fire detection system is to minimize financial losses and human casualties by rapidly identifying flames in diverse settings, such as buildings, industrial sites, forests, and rural areas. Traditional fire detection systems that use point sensors have limitations in identifying early ignition and fire spread. Numerous existing computer vision and artificial intelligence-based fire detection techniques have produced good detection rates, but at the expense of excessive false alarms. In this paper, we propose an advanced fire and smoke detection system on the DetectNet_v2 architecture with ResNet-18 as its backbone. The framework uses NVIDIA's Train-Adapt-Optimize (TAO) transfer learning methods to perform model optimization. We began by curating a custom data set comprising 3,000 real-world and synthetically augmented fire and smoke images to enhance models' generalization across diverse industrial scenarios. To enable deployment on edge devices, the baseline FP32 model is fine-tuned, pruned, and subsequently optimized using Quantization-Aware Training (QAT) to generate an INT8 precision inference model with its size reduced by 12.7%. The proposed system achieved a detection accuracy of 95.6% for fire and 92% for smoke detections, maintaining a mean inference time of 42 ms on RTX GPUs. The comparative analysis revealed that our proposed model outperformed the baseline YOLOv8, SSD MobileNet_v2, and Faster R-CNN models in terms of precision and F1-scores. Performance benchmarks on fire instances such as mAP@0.5 (94.9%), mAP@0.5:0.95 (87.4%), and a low false rate of 3.5% highlight the DetectNet_v2 framework's robustness and superior detection performance. Further validation experiments on NVIDIA Jetson Orin Nano and Xavier NX platforms confirmed their effective real-time inference capabilities, making them suitable for deployment in safety-critical scenarios and enabling human-in-the-loop verification for efficient alert handling.

KEYWORDS

artificial intelligence, fire and smoke detection, DetectNet_v2, transfer learning, model pruning, train adapt optimize (TAO), quantization-aware training (QAT), human-in-the-loop (HITL)

1 Introduction

The threat of fire accidents has always been a greater cause of concern; it has a high potential to cause severe damage to the economy, society, and mankind. National Disaster Management Authority (NDMA) of India (Fires in India: Learning Lessons for Urban Safety, 2020) has listed numerous causes of fire incidents, including gas or electric device malfunctioning, electrical faults, and explosions caused by flammable objects. As per the statistics, most of the fire-related fatalities are due to inhalation of toxic gases, which are more dangerous than external burn injuries. Oxygen deprivation by the release of poisonous gases from burning objects is the other serious cause of fire-related deaths. Fire is categorized as a human-induced disaster by the NDMA, highlighting the necessity of effective prevention measures. According to UK government data, the fire service responded to 5,55,759 events in 2019, out of which fires accounted for 28% (1,57,156) of these incidents (Fires in India: Learning Lessons for Urban Safety, 2020). In India, as per the National Crime Records Bureau's (NCRB) Accidental Deaths and Suicides in India (ADSI) report [Accidental Deaths and Suicides in India (ADSI), 2022], as many as 7,435 persons lost their lives in over 7,500 fire incidents in 2022. Currently, smoke and flame detector technologies are widely used for indoor environments (Gov, Fire Alarms-Property Management, 2020). However, these detectors have several restrictions, especially in detecting the rate of fire spread, size, and spatial extent, which are crucial inputs for fire personnel required to act in time. Conventional methods that rely on thermal cameras or smoke detectors often fail to cover bigger industrial spaces, and large-scale deployments are practically not viable considering the deployment and maintenance costs. These are non-invasive approaches and lack a thorough understanding of the surrounding environment. In contrast, computer vision (CV) based systems are more intrusive because they rely on continuous video streams to provide greater spatial coverage and real-time analysis. Video surveillance, a critical component of early warning systems, plays an important role in monitoring and alerting authorities about fire events occurring in various environments, such as buildings, factories, and public spaces. The AI-based video surveillance solutions are one of the fastest-growing fields of computer vision (CV) research, which provides automated detection, tracking, and monitoring features to various end-users, including home security, traffic control, and airport surveillance (Wang, 2013) departments. With the help of video surveillance systems, the early detection of accidental and catastrophic fire events can significantly reduce the likelihood of casualties and significant damage to properties. The practical limitations of the conventional smoke detectors create better opportunities for computer vision researchers to develop intriguing alternative solutions (Xiong et al., 2007). In recent years, several video-based fire and smoke detection algorithms have been proposed to overcome the traditional detectors' limitations, providing more efficient and scalable solutions. Fire behavior modeling by combining various signals and image processing techniques (Çetin et al., 2013) led to early breakthroughs in the video-based fire detection research. Vision-based systems are a part of the broader artificial intelligence (AI) research and are becoming increasingly important in the development of fire safety applications (Healey et al., 1993).

Earlier systems employed handcrafted feature extraction methods to carry out fire or smoke detection tasks under diverse environmental conditions (Jadon et al., 2020). However, these methods lacked precision and robustness. To accomplish these tasks, several video-based detection systems proposed the use of textural features like color, shape, and motion cues. Yet, accurately capturing and representing fire and smoke features involving irregular motion, fluctuating shape, varying color, texture, and density remains an open challenge, leading to high false alarm rates. As discussed at the beginning of this section, conventional thermal imaging cameras and smoke detectors provide very limited coverage, particularly in large-scale workplaces. The high initial setup and maintenance costs make them less accessible to many organizations. Hence, the demand for affordable alternatives that improve fire and/or smoke detection capabilities by utilizing emerging CV and AI-based technologies is consistently rising. By utilizing existing CCTV camera infrastructure, organizations can easily integrate these intelligent systems into their setups. Thus, offering budget-friendly real-time AI surveillance solutions (Pincott et al., 2022) that are not only efficient but can easily adapt to any given operational environment. These not only enable prompt identification of fire hazards but also quickly alert the concerned to control the damage, as illustrated in Figure 1. Given the critical need for early warning systems in industrial and manufacturing sectors, innovative solutions are essential for protecting both human lives and factory assets.

To address these issues, we propose an advanced as well as a cost-effective computer vision and artificial intelligence-based fire and smoke detection system. The workflow of the proposed framework involves the generation of the synthetic dataset, structuring the data, and training on the DetectNet_v2 model to evaluate its performance in terms of detection accuracy, inference speed, and deployable parameters. The dataset was constructed by utilizing publicly available fire and smoke repositories and extending it by extracting high-quality frames from online video sources. The next step involves accurate bounding box annotations around fire and smoke regions, before training. The training operation is carried out on a pre-trained DetectNet_v2 architecture that is later optimized via the NVIDIA TAO Toolkit. Model reliability is validated by experimenting on test datasets containing complex industrial operational scenarios to analyze false-positive and missed-detection cases.

2 Related work

This section provides a detailed literature review to highlight the gaps in the research as well as the rationale for the goals of the study and the methods that were selected.

2.1 CNN-based forest fire and smoke detection systems

Climate change and human activities in forest areas have caused the frequency of wildfire incidents to increase drastically in recent years. High deployment and maintenance costs of thermal sensors and smoke detectors, coupled with limited coverage area and slow response times, especially in dense vegetated forests, make them unpopular choices. On the other hand, computer vision-based detection systems offer a cost-effective alternative that enables automatic and real-time forest surveillance solutions even under low

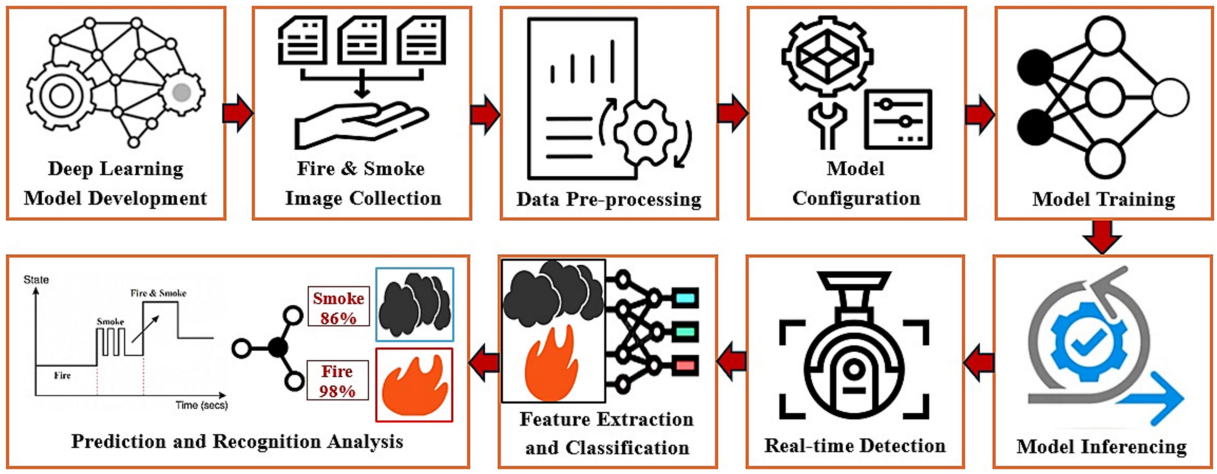


FIGURE 1
Workflow of the computer-vision-based indoor fire and smoke detector system.

visibility and challenging weather conditions. These computer vision-based systems are often developed using Convolutional Neural Networks (CNN) and deep learning-based frameworks to achieve high detection performances on input frames supplied from live video feeds (Zhou et al., 2016). Since CNN architectures can perform feature extraction and classification operations in a single pass, they offer highly accurate detections at low latency even in applications involving complex or low-light scenes. Thus, eliminating the need to develop handcrafted ground truth features. Their deployment in wildfire monitoring can significantly contribute to timely alerting and mitigation efforts. Several CNN-based studies have demonstrated their applicability in forest fire detection scenarios through acceptable precision and recall performances. The use of CNNs in vision-based coal mine area surveillance has reported good detection accuracies with negligible false detection rates, thereby creating an impact in critical applications (Toptaş and Hanbay, 2020). These research findings influence their deployment in similar forest areas, where early smoke detection can be particularly challenging due to occlusion, poor lighting, and cluttered backgrounds. Further, CNN-based models need to constantly adapt to changing weather conditions and diverse vegetation types in forest landscapes. This emphasizes the need for further investigation into the development of more resilient and generalizable architectures that utilize contextual and temporal information for improved reliability.

Numerous studies have concentrated on CNNs' adoption in CV-based fire and smoke detections, demonstrating their superior performance over traditional vision-based fire detection methods (Avazov et al., 2022; Zhang et al., 2016). To classify fire and smoke patterns from real-time video streams, wildfire detection systems (Wu and Zhang, 2018) employed DL-based frameworks, including SSD (Single Shot MultiBox Detector), YOLO (You Only Look Once), and Faster R-CNN. Even though methods reported good detection accuracies at reasonable computational costs, the detection latency and validation under noisy environments are not reported in this literature. Generalization capabilities in unseen environments when these models are trained on small, customized datasets are also not well explained. Without clear evidence on trade-offs, selecting the optimal detection framework for specific deployment scenarios will become more challenging.

In another study, the forest fire and smoke detection using a DL-based method (Sathishkumar et al., 2023) utilizes transfer learning techniques on pre-trained models like VGG16, InceptionV3, and Xception, combined with a "learning without forgetting" approach. While learning without forgetting guarantees that the model can continuously learn new classes without forgetting previously learned ones, transfer learning enables the model to leverage the information from large-scale datasets. While this approach improves detection adaptability and classification accuracy, it lacks a detailed examination of hyperparameters, model configurations, and performance scalability required for fine-tuning the models before deployment in diverse conditions. So far, the researchers have predominantly focused on outdoor environments, particularly in forest fire scenarios. Still, a significant research gap exists in the development and evaluation of CNN-based fire detection models customized for commercial or industrial workplace scenarios, where fire characteristics, visual obstructions, and environmental variations pose key challenges. Urgent need to address these gaps is essential for developing comprehensive, real-world fire detection solutions that extend beyond forest environments.

2.2 CNN-based fire and smoke detection systems in industrial environments

An important method of real-time fire and smoke detection from video surveillance involves the use of a parallel computing CUDA (Compute Unified Device Architecture) (Filonenko et al., 2018) framework, designed to accelerate NVIDIA GPUs' performance. In applications involving moving object detection, the CUDA framework adopts a background subtraction technique and performs color probability analysis to segregate smoke from non-smoke objects. Apart from this, it performs boundary roughness predictions and edge density analysis to detect unseen smoke patterns. Since the CUDA frameworks heavily rely on the static visual fire patterns and attempt to speed up these computations to match the best performances in

controlled environments (stationary cameras). Hence, making it inappropriate for dynamic industrial surveillance applications.

An alternative approach, a deep CNN-based fire detection system called "DeepCNN" (Muhammad et al., 2019), utilizes AlexNet (Krizhevsky et al., 2017) and a transfer learning technique to develop an intelligent feature map selection strategy for fire detection. Despite its smaller model size (3 MB compared to AlexNet's 238 MB), this method achieves 94.50% accuracy, while minimizing false alarms, making it suitable for resource-constrained environments. However, the study lacks emphasis on contextual scene understanding or object-level reasoning, which act as important deciding parameters for robust decision-making in complex industrial scenarios. The absence of a specific and tailored dataset containing varied industrial scenes further limits the model's differentiating capacity.

Another complex video-based smoke detection system called "AdVISED" (Gagliardi and Saponara, 2020) integrates multiple analytical techniques to perform image segmentation, color space analysis, Kalman filtering for object tracking, and geometric feature extraction tasks. The objective of this multifaceted method is to enhance the resilience and accuracy of smoke detections in systems where smoke appearance is influenced by textured or colored backgrounds. Thus, limiting the robustness due to varying lighting conditions or when smoke characteristics deviate from their required patterns. Additionally, the cumulative computational load imposed by sequential processing stages may hinder real-time performance, especially in embedded or resource-constrained devices.

CNN-based YOLOv2 architecture offers fast processing capabilities on a small training dataset, making it the best choice in the development of efficient fire-smoke surveillance applications (Saponara et al., 2021). Several studies utilized Faster R-CNN (Tien et al., 2020), Inception_v2 (Wei et al., 2020; Feng et al., 2016), and SSD MobileNet_v2 (Sandler et al., 2018; Tsang, 2020) models to evaluate on indoor-specific datasets. Training with a limited and diverse dataset resulted in more missed detection rates, achieving average accuracy.

To boost the real-time performance on resource-constrained applications such as coal mines, an improved YOLOv8s-based model with faster convolution layers and attention mechanisms was developed. This approach achieved a mean Average Precision (mAP) of 91.0%, while consuming fewer computational resources, making it appropriate for mine surveillance environments (Kong et al., 2024; Redmon et al., 2016). Further research is required to fully assess its scalability and versatility across different coal mining conditions before deploying it in larger mine areas. A method to detect small objects such as cell phones utilised a combination of YOLOv8 and ResNet-18 feature classification layers (Deshpande et al., 2025b), to produce a decent mean Average Precision (mAP@0.5) of 49.5% at an Intersection-over-Union (IoU) threshold of 0.5. Similarly, the ResNet-18-based traffic monitoring systems (Deshpande et al., 2025c; Deshpande et al., 2025a) under complex backgrounds achieved a high 91.42% accuracy for triple riding violations, and 98.5% for helmet violations at considerably slower detection rates.

The incidents involving fire hazards, early detection, and swift actions are essential to prevent production disruptions, property losses, and fatalities in manufacturing industries. Despite advances in CV and DL technologies, many existing fire and smoke detection systems still face significant limitations such as high false detections, inadequate inference speeds, and poor generalization issues. Unpredictable fire and smoke patterns, inappropriate lighting conditions, occlusions, and limited industry-specific fire and smoke datasets are the additional factors that hinder the model's performance. Very few researchers have tried to solve the indoor residential fire and smoke detection problem. To bridge these gaps, we propose a computer vision and AI-based fire/smoke detection system specifically designed to serve industrial indoor and outdoor use cases using the DetectNet_v2 architecture. To quickly optimize and develop a reliable fire and smoke hazard detection system, we employ a transfer learning approach that enables us to seamlessly integrate the proposed solution with the existing CCTV infrastructure to serve a variety of industrial use cases. The rest of this paper is organized as follows: Section 3 describes our proposed DetectNet_v2 framework, Section 4 presents a comprehensive experimental setup and evaluation procedures. Section 5 concludes the study by highlighting contributions and potential future research directions.

3 Proposed computer vision-based fire and smoke detection system

To address the challenges discussed in the previous section, we present a transfer learning based real-time fire and smoke detection system using the DetectNet_v2 framework that utilizes ResNet-18 as its backbone network. The overall system workflow, comprising dataset engineering, model training and validation, and performance evaluation, is depicted in Figure 2.

3.1 Data engineering and preprocessing

The first phase in the implementation process involves the construction of a comprehensive and diverse custom dataset to perform the training and evaluation of the proposed fire and smoke detection model. The dataset comprises synthetically generated (augmented) images from video frames featuring a wide range of fire and smoke scenarios captured in industrial and indoor environments. Bounding box annotations across accurate fire and smoke regions of interest (ROIs) are marked to ensure that the model learns to localize fire and smoke instances effectively.

Image acquisition, pre-processing, and data augmentation: we started the process by collecting a total of 1,360 images containing fire and smoke incidents from publicly available internet sources and a publicly available dataset (Senthil, 2025). These samples feature various industrial parameters across different lighting conditions and environmental variations, as indicated in Figures 3a,b.

To extend the dataset size and simulate on realistic scenarios, synthetic augmentation using Adobe Premiere Pro is performed as shown in Figure 3c. The fire and smoke assets extracted from stock video repositories are blended into base images to generate synthetic cases, resulting in 1,580 augmented samples, as summarized in Table 1. This process involves strategic placement and blending to achieve a natural integration of the effects into the existing environment. Techniques such as color matching, scaling, and motion tracking are employed to ensure that the fire and smoke appear seamless within the footage. In addition, 1420 real-world images from various online sources and public repositories (Senthil, 2025) are curated, representing authentic fire and smoke incidents across diverse environments as listed in Table 1. The pre-processing techniques, like

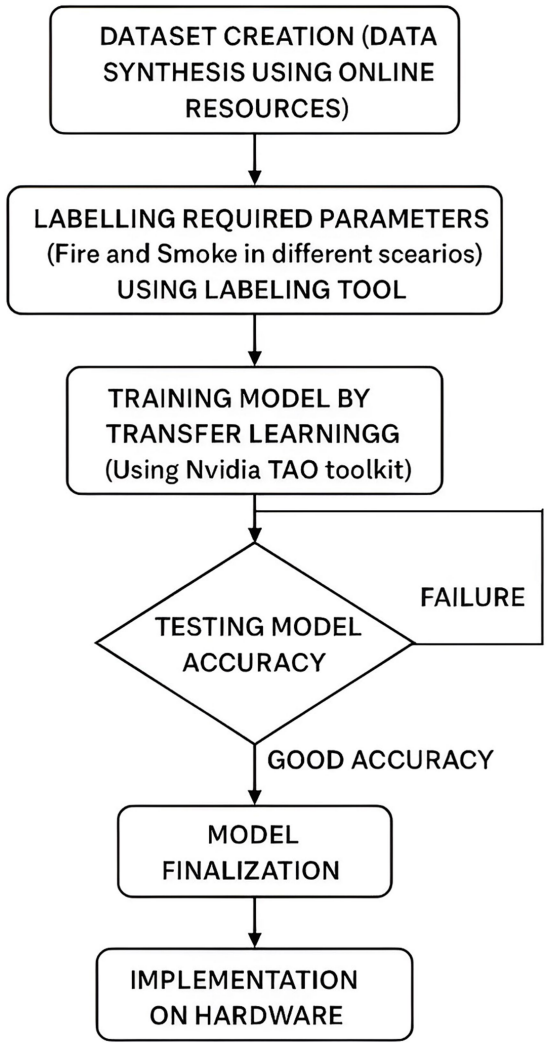


FIGURE 2
Workflow of our proposed computer vision-based fire and smoke detection system.

resizing and denoising, are applied to enhance the image quality and make it best suitable for feature extraction. Denoising is achieved primarily through built-in effects and plugins designed to reduce video noise, especially in low-light footage. In the later step, the gathered images are carefully processed and annotated with the help of a flexible labelling application called "LabelImg". To ensure correct ground truth for supervised learning, critical fire and smoke characteristics are carefully labeled and annotated, as indicated in Figure 4a. To enable edge devices such as Jetson Orin to efficiently perform real-time processing at a minimum average benchmark inference speed of 22 frames by keeping GPU memory usage as low as possible, we limit image size to 1280x720 pixel resolution. This resolution helps to balance model accuracy and computational load on resource-constrained edge platforms. After the data cleaning process, the combined effort resulted in a dataset of 3000 (1640 fire and 1360 smoke) images containing 1580 augmented and 1420 real-world images, as detailed in Table 1.

KITTI format: the training process on the DetectNet_v2 architecture requires the utilization of NVIDIA's TAO Toolkit. To comply with the TAO's object detection pipeline, the annotated images are carefully converted to the KITTI format. Each KITTI label file is a plain text file where each line corresponds to one object instance. A total of 15 elements per object are included as indicated by a sample KITTI file (top) and its description (bottom) depicted in Figure 4b. The description typically indicates the following:

- class name: e.g., "fire"
- xmin, ymin, xmax, ymax: e.g., 27, 18, 127, 203

The remaining fields can be set to default values (e.g., 0) as they are not used during training. After all images are labelled and converted to the appropriate format, the final dataset is ready to be used for training the DetectNet_v2 model.

3.2 Model selection and training

Several state-of-the-art deep learning-based object detection algorithms have been explored recently to address the challenges discussed in Section 2.2. Here is a list of the most well-established and widely used object detection architectures:

- Single-shot MultiBox Detector (SSD)
- Region-Based Convolutional Neural Network (R-CNN)
- You Only Look Once (YOLO)
- RetinaNet
- DetectNet
- CenterNet

Among these, the YOLOv8 framework has gained popularity for its speed and accuracy since it treats the object detection task as a regression issue rather than a classification task and hence predicts bounding boxes with its class probabilities directly from complete images in a single pass. YOLOv8 demonstrated suboptimal performance in generalizing fire and smoke detection under varying industrial conditions, including changing lighting, texturing, and scene complexities. In contrast, DetectNet_v2 works effectively in diverse environmental situations and is tailored for real-world deployment, making it a more suitable choice for fire and smoke detection tasks. The model offers robustness to cluttered backgrounds, and it works effectively in places that produce varying lighting conditions. Transfer learning, DetectNet_v2's grid-based prediction, and a reliable feature extraction pipeline help to accurately localize fire and smoke objects in complex manufacturing process scenarios that contain amorphous fire or smoke-like objects.

3.2.1 NVIDIA TAO toolkit and DetectNet (ResNet-18) model

To perform model customization and training, we employed a low-code, Python-based AI toolkit called the "NVIDIA TAO Toolkit," specifically designed for accelerating the development of computer vision (CV) applications. The transfer learning feature of TAO enables users to adapt pre-trained models on user-defined custom-labeled datasets, facilitating effective pattern identification in complex setups without requiring extensive model knowledge. The toolkit is based on

10.3389/fcomp.2025.1636758

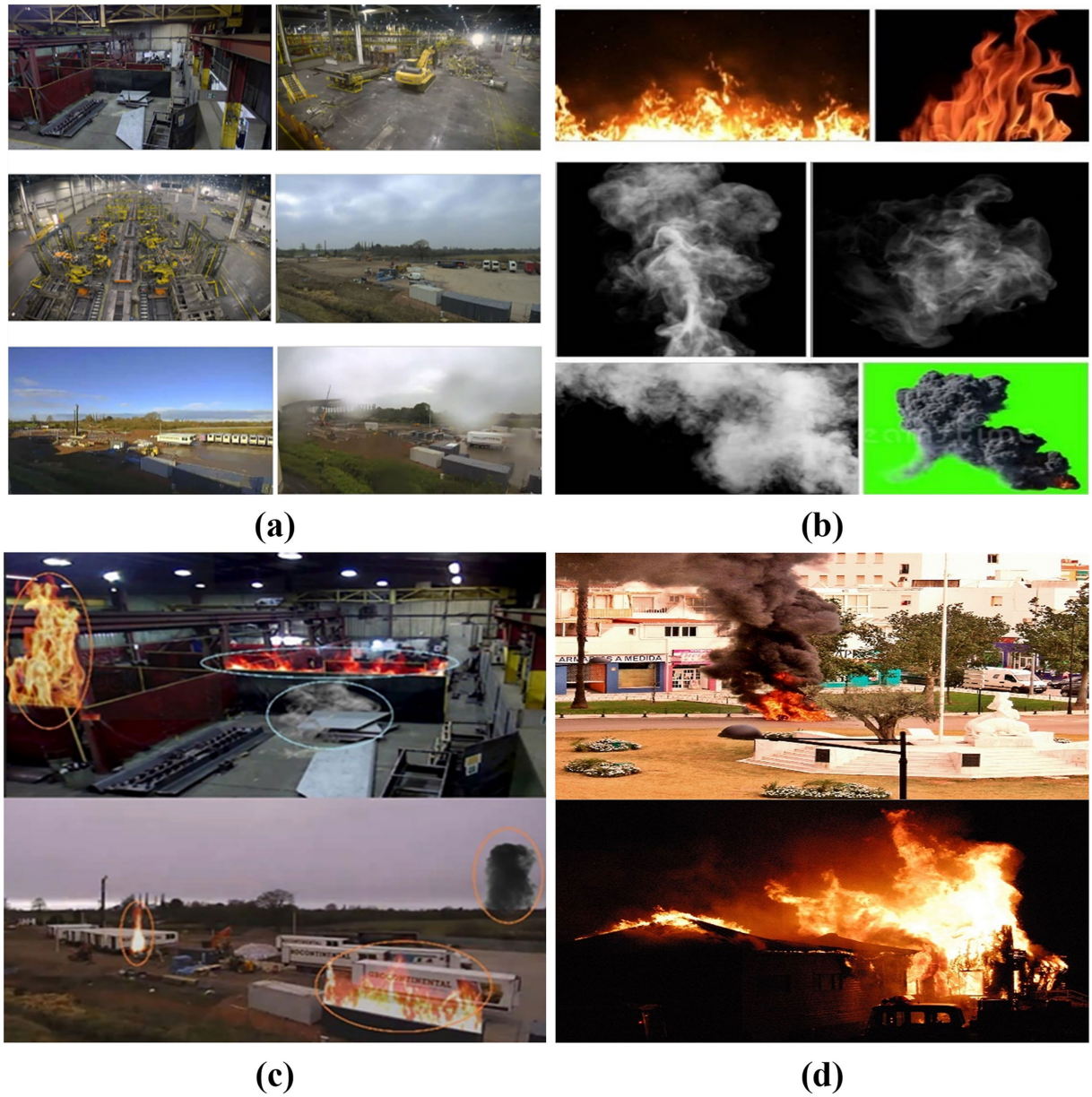


FIGURE 3
Sample fire and smoke datasets used in our work representing **(a)** indoor and outdoor industrial environment, **(b)** fire and smoke samples, **(c)** augmented fire (orange) and smoke (blue) images, **(d)** real-world fire and smoke instances (Senthil, 2025).

TABLE 1 **Summary of augmented and real fire and smoke images used in our study.**

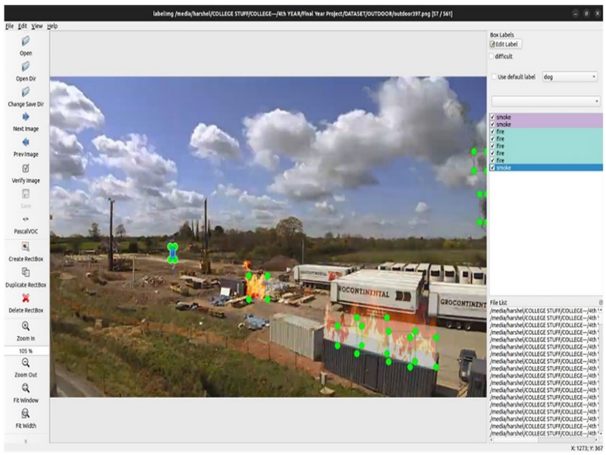
Category	Class		Total images in dataset
	Augment	Real	
Fire	861	779	1,640
Smoke	719	641	1,360
Total	1,580	1,420	3,000

TensorFlow and PyTorch and provides access to more than 100 pre-trained models. The models can be exported in ONNX format, ensuring compatibility across multiple inference engines and platforms, as indicated in Figure 5.

As seen in Figure 5, the TAO workflow supports the following crucial operations:

- Dataset preparation and augmentation
- Model training and evaluation
- Inference and performance tuning
- Pruning, quantization, and export for deployment

Detectors such as Faster R-CNN follow a two-stage mechanism where the candidate object regions are generated before accomplishing classification operations. On the contrary, NVIDIA's



(a)

Text Editor

Open

```
1 fire 0 0 0 27 18 127 203 0 0 0 0 0 0 0
2 fire 0 0 0 9 177 168 451 0 0 0 0 0 0 0
3 fire 0 0 0 393 277 808 333 0 0 0 0 0 0 0
4 smoke 0 0 0 451 339 587 417 0 0 0 0 0 0 0
5 smoke 0 0 0 541 348 755 426 0 0 0 0 0 0 0
```

#Value	Name	Description
1	Type	Describes the type of object: 'Car', 'Van', 'Truck', 'Pedestrian', 'Person_sitting', 'Cyclist', 'Tram', 'Misc' or 'DontCare'
1	Truncated	Float from 0 (non-truncated) to 1 (truncated), where truncated refers to the object leaving image boundaries
1	Occluded	Integer (0,1,2,3) indicating occlusion state: 0 = fully visible, 1 = partly occluded, 2 = largely occluded, 3 = unknown
1	Alpha	Observation angle of object, ranging [-pi..pi]
4	Bbox	2D bounding box of object in the image (0-based index): contains left, top, right, bottom pixel coordinates
3	Dimensions	3D object dimensions: height, width, length (in meters)
3	Location	3D object location x, y, z in camera coordinates (in meters)
1	Rotation_y	Rotation ry around Y-axis in camera coordinates [-pi..pi]

(b)

FIGURE 4
Sample LabelImg and KITTI file **(a)** fire and smoke labelling indicated by green boxes, **(b)** a sample text file of an image containing 15 elements per object in an image.

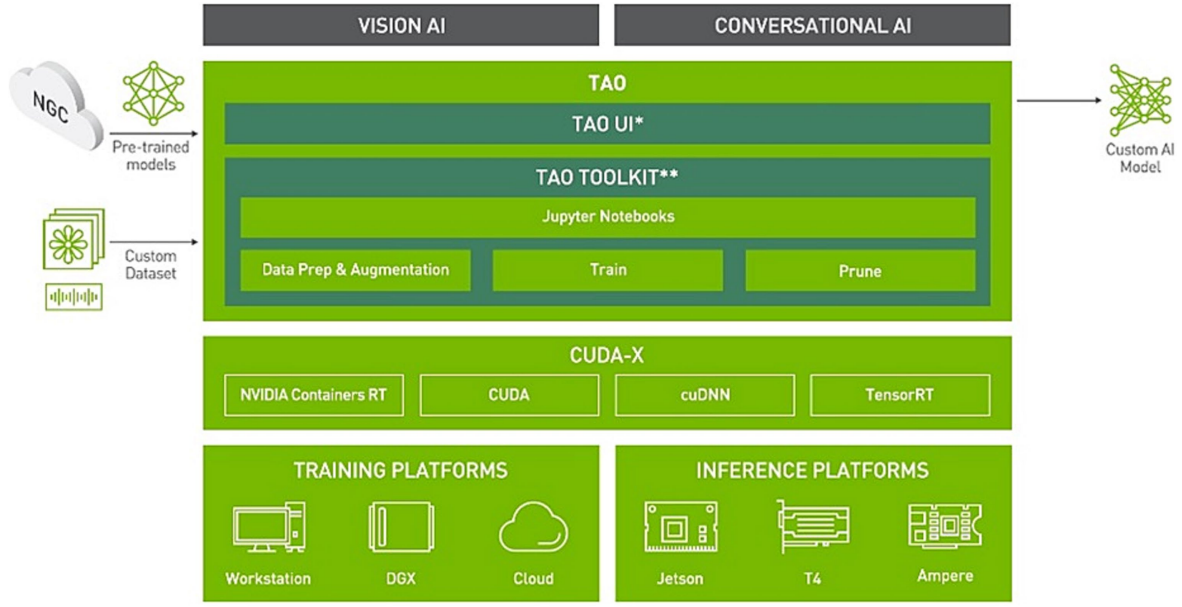


FIGURE 5
NVIDIA TAO overview (NVIDIA, 2025).

DetectNet_v2 is a single-stage deep learning-based framework optimized for deployment for real-time inference and robust object localization operations. The model's unified pipeline helps to perform object classification and bounding box regression tasks together and thereby maintains high detection accuracy despite reduced computational overheads. To meet the object detection task requirements, the framework accepts the datasets to be converted from other common formats (e.g., COCO) to KITTI. During the preprocessing step, the raw labeled KITTI data is converted into a binary format compatible with TensorFlow called "TFRecords". Further, DetectNet_v2 performs dataset translation, model training, evaluation, inference, pruning, calibration, tensor generation, and model export tasks as depicted in Figure 6.

We employ a pre-trained ResNet-18 model as the backbone feature extractor within the DetectNet_v2 architecture and retrain it on the KITTI dataset. ResNet (Residual Network) is available in several configurations (e.g., ResNet-18, ResNet-50). Derived from the concept of residual connections, the ResNet allows gradients to skip one or more layers to mitigate the vanishing gradient problem and effectively train deeper neural networks. Figure 7 illustrates the ResNet architecture that comprises 18 convolutional layers (ResNet-18) grouped into residual blocks.

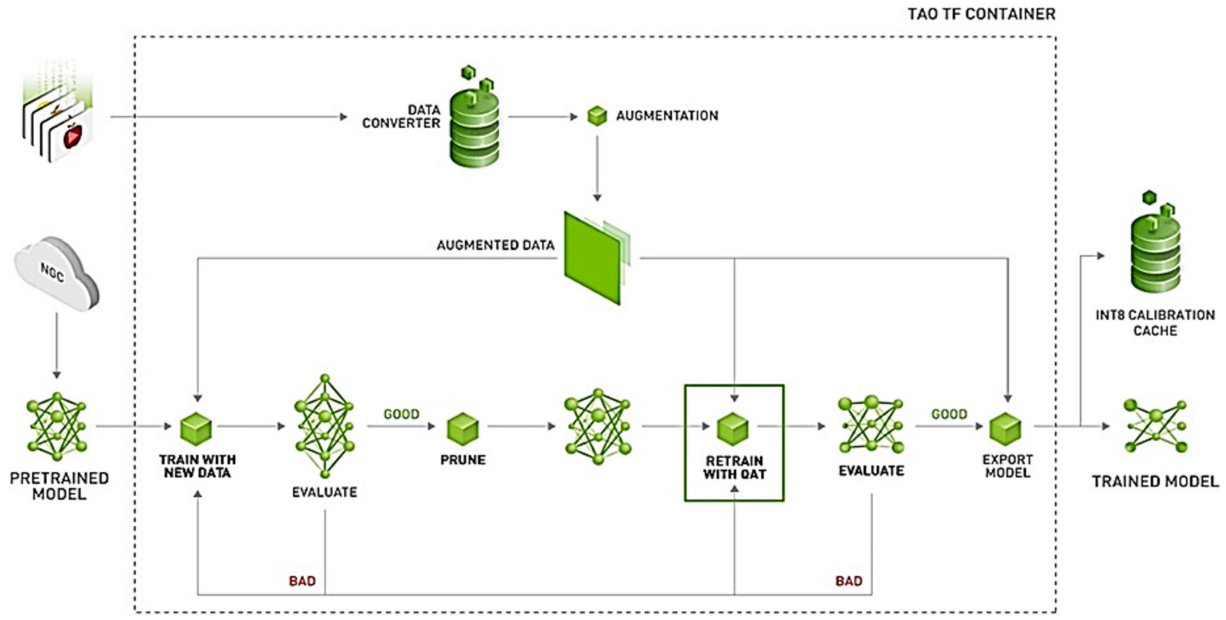


FIGURE 6
Workflow for creating an optimized and trained DetectNet_v2 (NVIDIA, 2025).

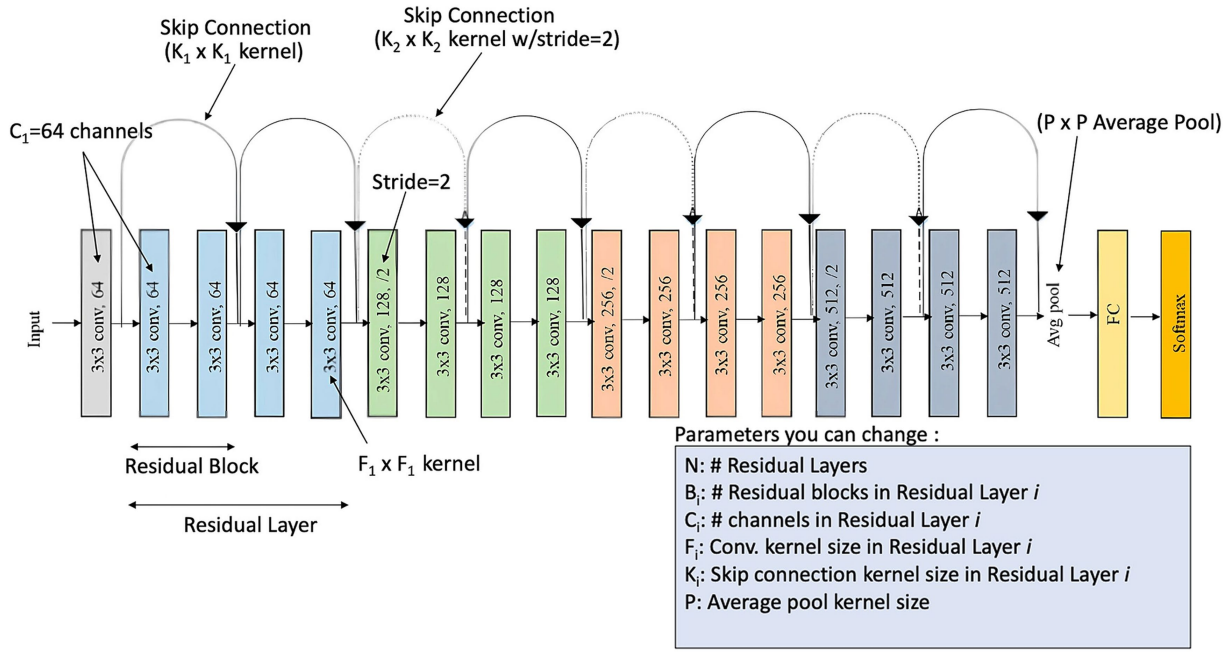


FIGURE 7
DetectNet_v2 with ResNet-18 backbone: layer overview (Beji, 2025).

Each pair of identically coloured layers denotes a residual block, and the black arrows indicate shortcut (skip) connections. Residual connections, which omit some layers to allow for deeper networks, are scattered throughout the network. Feature maps' spatial dimensions are decreased via pooling layers, and classification is carried out by the last fully connected layer.

Convolutional layers: convolutional layers process the input image by applying a series of learnable filters. To create an activation map, each filter moves across the input image (or feature map). The convolution operation for a particular filter (W) and input (X) can be expressed mathematically using Equation (1):

$$(W * X)(i,j) = \sum_{m=1}^{M} \sum_{n=1}^{N} X(i+m-1, j+n-1) \cdot W(m,n) \quad (1)$$

Here, M and N are the filter's dimensions, while (i, j) are the spatial coordinates in the output activation map.

Activation functions: the convolutional layer's output is subjected to a non-linear application of the ReLU (Rectified Linear Unit) activation function. The ReLU function is represented with the help of Equation (2):

$$\mathrm{ReLU}(x) = \max(0, x) \tag{2}$$

By introducing non-linearity, this enables the network to learn intricate patterns.

Batch normalization: to stabilize the learning process and minimize the number of training epochs needed, this layer normalizes the input to each layer. Given an input x with mean μ and variance σ^2, batch normalization is defined using Equation (3):

$$\hat{x} = \frac{x - \mu}{\sqrt{\sigma^2 + \hat{o}}} \tag{3}$$

$$y = \gamma \hat{x} + \beta$$

Here, γ and β are learnable parameters, and ϵ is a small constant to prevent division by zero.

Residual connections: the main principle of ResNet is the insertion of residual (skip) connections that bypass one or more layers. A residual block for an input x is defined in Equation (4):

$$y = F(x, \{W_i\}) + x \tag{4}$$

Here, the input sent through the shortcut connection is denoted by x, and $F(x, \{Wi\})$ denotes the residual mapping to be learned (a function of the weights Wi).

Pooling layers (max pooling): by choosing the largest value in each window, this stage shrinks the spatial dimensions of the input feature map (or patch). For a feature map X and a pooling window of size $k \times k$, max pooling is obtained using Equation (5):

$$Y(i,j) = \max 0 \le m < k, 0 \le n < k \; X(i\,k + m, j\,k + n) \tag{5}$$

3.3 Performance evaluation of the trained detection models

An overview of a machine learning model's performance on a set of test data is provided using a confusion matrix. Based on the model's predictions, the matrix showcases the proportion of accurate and inaccurate instances. This method is commonly utilized to test the classification model's performance to predict a categorical label for each input instance. We construct a confusion matrix including true positives, false positives, and negative results during model testing.

Precision: precision measures the accuracy of the positive predictions made by the model. One way to describe it is as the ratio of all positive forecasts to precisely predicted positive observations as expressed in Equation (6).

$$\mathrm{Precision} = \mathrm{TP} / (TP + \mathrm{FP}) \tag{6}$$

- The number of correctly detected objects is known as True Positives (TP).
- The amount of erroneously detected objects (false alarms) is defined as False Positives (FP).

Precision gauges how well the model predicts good outcomes. It may be defined as the ratio of all positive forecasts to precisely predicted positive observations.

Recall/sensitivity: recall measures the ability of the model to detect all relevant objects in the dataset. It is the proportion of all truly positive observations to all accurately projected positive observations as defined in Equation (7).

$$\mathrm{Recall} = \mathrm{TP} / (\mathrm{TP} + \mathrm{FN}) \tag{7}$$

False Negatives (FN) are the number of true positive cases that were incorrectly predicted as negative. It additionally reveals missing objects that were not detected. When a model accurately predicts a negative outcome when the actual result is negative, this is known as a True Negative (TN) measurement. Additionally, it displays accurately recognized non-objects.

4 Results and discussion

To evaluate the effectiveness of the proposed fire and smoke detection system, we perform transfer learning utilizing the NVIDIA TAO (Train, Adapt, Optimize) Toolkit, and carry out all experiments on an NVIDIA RTX 2000 Ada GPUs. Later, to achieve edge deployment compatibility, we optimize the final pruned DetectNet_v2 engine through the INT8 quantization technique. The resulting TensorRT engine is then deployed to produce good inference runtime efficiency. After post-processing operations, the DBSCAN algorithm is used to refine and merge bounding box predictions, which is especially helpful in situations where fire and smoke instances appear fragmented or spatially dispersed. To ensure consistency during training and inference tasks, all input images are resized to a uniform resolution of 1280 × 720 pixels. We use the Adam optimizer and train the DetectNet model with a weight decay set to $3e^{-9}$, epochs set as 120 with a batch size of 4 per GPU, using a learning rate that ramps up gradually from 5e-06 to 5e-04 during the first 10% of training (12 epochs) and then smoothly decays back toward 5e-06 over the next 70% of training (84 epochs) using a cosine-like annealing schedule. These hyperparameters were chosen based on extensive benchmarking with NVIDIA TAO Toolkit recommendations and empirical evaluation to ensure stable training, faster convergence, and deployment compatibility.

Dataset categorization: the augmented and real-world image dataset categorization for training and testing operations is summarized in Table 2. The table provides comprehensive dataset information used for training and testing our model.

TABLE 2 The augmented and real-world image dataset categorization for training and testing our proposed DetectNet model.

Category	Training images			Testing images			Total images in dataset
	Augment	Real	Total	Augment	Real	Total	
Fire	763	624	1,387	98	155	253	1,640
Smoke	667	546	1,213	52	95	147	1,360
Total			2,600			400	3,000

To guarantee that the model learns fire and smoke patterns from both synthetic and real-world images, a balanced and diverse training environment is created by carefully structuring this dataset. Further, we guarantee no overlap between the two sets by randomly distributing the images into 90% training and 10% test sets (90:10 split) as given below:

Training set: comprises 2,600 images (90%), including:

- *Fire instances* - 763 augmented and 624 real-world images.
- *Smoke instances* - 667 augmented and 546 real-world images.

Test set: is made up of 400 (10%) randomly selected images, which do not overlap with the training images, as follows:

- *Fire instances* - 98 augmented, and 155 real-world images.
- *Smoke instances* - 52 augmented and 95 real-world images.

The model evaluation results, performance metrics, and comparative analysis with baseline detectors are presented in the following sections.

4.1 Training and testing the DetectNet_v2 model

The training process began with a pre-trained DetectNet_v2 architecture. The pre-trained model is originally trained on large-scale, general-purpose datasets such as KITTI and COCO (Tsang, 2020), providing good object detection capabilities on relatively low-resolution inputs (3×368×640). To meet fire and smoke detection systems requirements, the model is greatly improved using the TAO Toolkit, which offers a simplified but effective framework for performing transfer learning and fine-tuning. The TAO-optimized framework, in conjunction with domain-specific training and higher input resolution, allows us to retrain the model on our custom fire and smoke dataset, enabling it to effectively learn specific visual features such as diminishing smoke in complex scenarios, mist, fog, or clouds. This refined model was trained on higher-resolution input images (3×720×1280) to capture finer detail, such as thin smoke trails or early-stage flames in complex industrial environments, necessary for alerting the concerned authorities. These performance enhancements prepare the model to work with NVIDIA DeepStream and TensorRT edge inference frameworks. Our DetectNet_v2 framework is evaluated across multiple test conditions, and its effectiveness is demonstrated in Figure 8. In indoor conditions (Figure 8a), outdoor conditions (Figure 8b), and on an unseen dataset (Senthil, 2025) (Figure 8c), our model accurately identifies true-positive and false-negative instances for both fire and smoke cases. Notably, our model exhibited strong generalization on unseen data, confirming its reliability for live surveillance applications and its suitability for real-world deployment. These findings indicate that the recommended methodology is well-suited for real-time implementation in intelligent surveillance systems, guaranteeing prompt fire hazard identification with low computational overhead.

4.2 Baseline model evaluation

Our proposed DetectNet_v2 model's training and validation losses during the evaluation process are presented in Figure 9. The model's validation loss curve in Figure 9a exhibits a steady downward trend, reaching the fourth decimal place after approximately 120 training indicating a stable convergence and robust optimization. Whereas the model's efficient learning and strong generalization capabilities are confirmed by the training loss curve's sharp reduction to 0.00008 within just 35 epochs, as observed in Figure 9b. To provide a neutral and unbiased comparison with our proposed DetectNet_v2, we retrained SSD MobileNet_v2 and Faster R-CNN (Inception_v2) benchmark detectors on our custom fire and smoke dataset using identical experimental conditions like input resolution of 1280×720, KITTI annotation format, and a 90:10 training-to-testing split.

Various model performance parameter comparisons are listed in Table 3 below. In contrast to Faster R-CNN and MobileNet_v2, our suggested DetectNet_v2 produced low validation loss and training loss, with faster convergence occurring within fewer total training steps. These improvements can be attributed to architectural differences and model optimization processes. To ensure model efficiency and stability, we continued to train the DetectNet_v2 until it converged.

Following retraining, we further evaluated the performance of our proposed DetectNet_v2 using a confusion matrix containing 253 fire and 147 smoke test images (refer to Table 2).

The confusion matrix, in Figure 10, highlights the DetectNet_v2's classification performance on the unseen fire and smoke test dataset. The model's effectiveness on diverse industrial scenarios is confirmed through 95.6% fire and 92% smoke detection accuracy results. These outcomes reveal the model's ability to correctly identify true instances while maintaining a low false prediction (<3.5%). On the other hand, low false positives and negatives indicate strong precision (94.8% for fire, 93.0% for smoke) and recall (95.3% for fire, 92% for smoke) outputs.

The Faster R-CNN (Pincott et al., 2022) and MobileNet_v2 (Pincott et al., 2022) baseline models are validated with test conditions discussed in 4.2. Our models' classification performance values obtained from the confusion matrix and other state-of-the-art methods' performance results are reported in Table 4. Our DetectNet_v2-based detector produces the highest fire and smoke

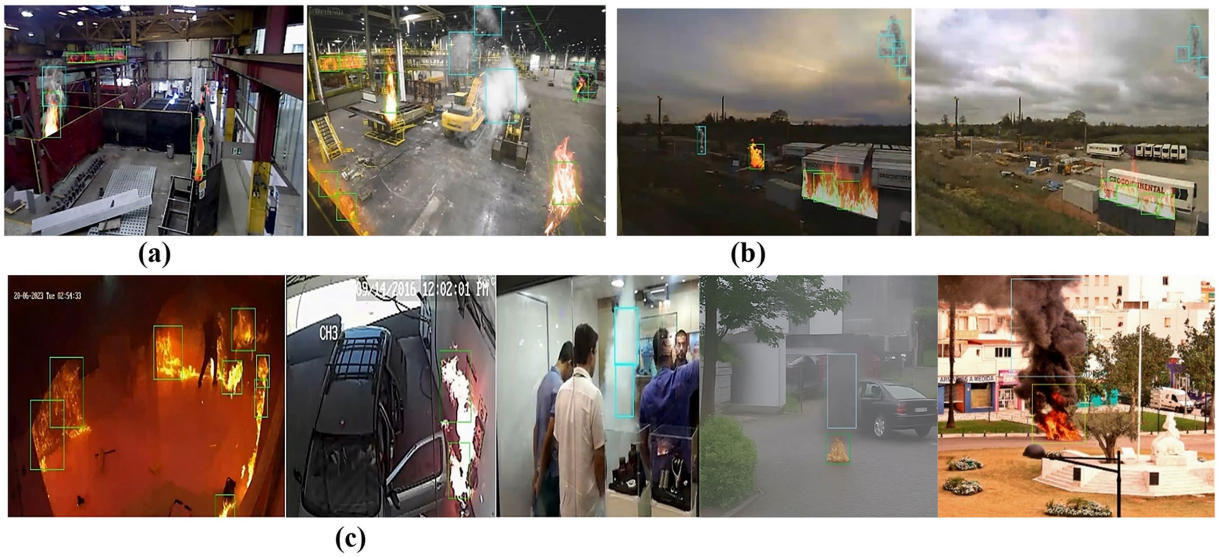


FIGURE 8
Visual examples of fire (green bounding boxes) and smoke (blue bounding boxes) detections by the proposed DetectNet_v2 model across different environments: **(a)** indoor industrial settings, **(b)** outdoor industrial scenarios, and **(c)** unseen test images (Senthil, 2025).

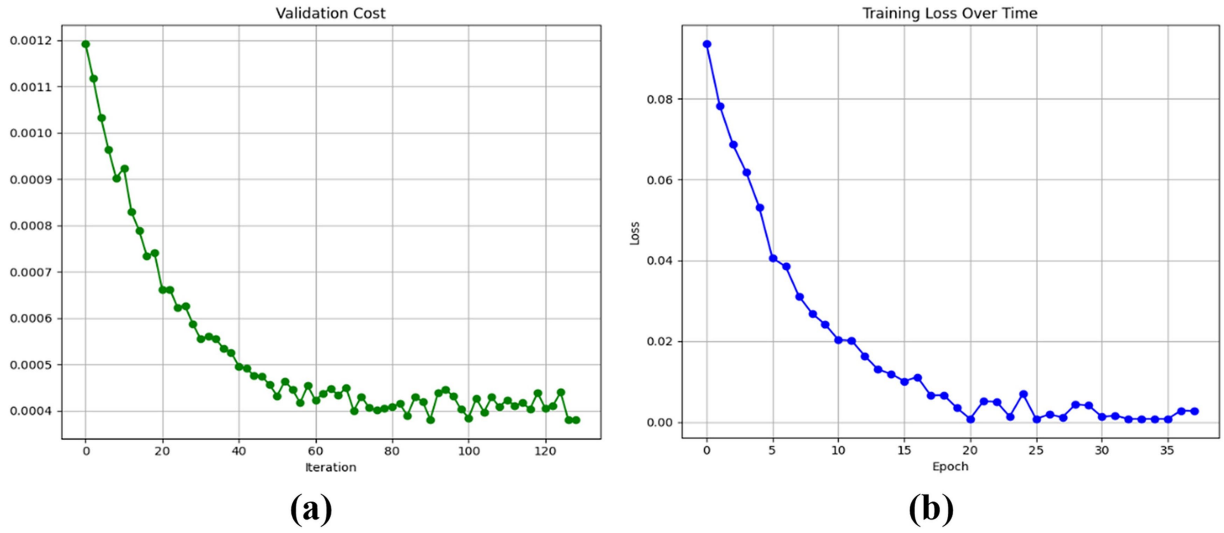


FIGURE 9
Training behavior of the proposed DetectNet_v2 model: **(a)** smooth and consistently converging validation loss curve, and **(b)** training loss curve indicating efficient learning with minimal overfitting.

TABLE 3 **DetectNet_v2, faster R-CNN, and SSD MobileNet_v2 model performance parameter comparisons.**

Performance parameters	Proposed DetectNet_v2	Faster R-CNN with Inception_v2 Jadon et al. (2020)	SSD MobileNet_v2 Jadon et al. (2020)
Total steps	42,102	90,973	36,348
Training duration	3 h 28 min, 50 s	5 h 32 min, 40 s	7 h 49 min, 30 s
Average loss	0.0141	0.084	1.912
Minimum loss	0.00008	0.0032	0.731

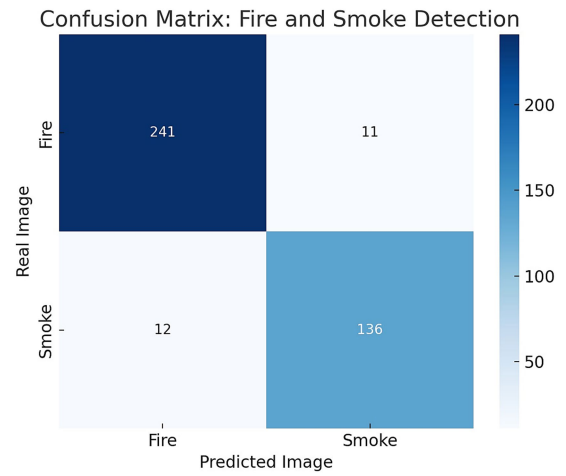


FIGURE 10
Confusion matrix showing the classification results of the proposed DetectNet_v2 model on the unseen test data.

detection accuracies and F1-scores over the other reported benchmark models, demonstrating its capability for reliable and real-time fire and smoke detection in industrial environments.

Figure 11 illustrates the proposed fire and smoke detection performance across multiple evaluation metrics. A detailed performance comparison of our proposed DetecNet_v2 over YOLOv8s, Faster R-CNN, SSD MobileNet_v2, DeepCNN, and AlexNet benchmark models reveals that our model consistently outperforms the state-of-the-art methods in accuracy, precision, and F1-scores.

4.3 Ablation study of pre-trained vs. fine-tuned and TAO optimized DetectNet_v2

A detailed ablation study was conducted to evaluate the effects of fine-tuning, pruning, and quantization on the performance of DetectNet_v2 across different training and optimization stages. The results are summarized in Table 5. We start with a COCO-pretrained DetectNet_v2 model with an initial size of 93.3 MB. This baseline model is fine-tuned (unpruned) by training on our custom fire and smoke dataset at 120 epochs. Pruning is done to eliminate unnecessary weights and minimize the size of the model after it has been trained, using a ResNet-18 backbone. This step also helps us to achieve higher mAP@0.5:0.95 performance. As a result of this step, the model size is reduced from 43 MB to 37.5 MB (12.7% reduction) and is further retrained to recover accuracy. We use the Quantization-Aware Training (QAT) method with INT8 precision included in the NVIDIA TAO Toolkit v5.0 to optimize the model for real-world edge deployments. To guarantee evaluation integrity and avoid data leakage before QAT, we carefully shortlisted 300 class-balanced (calibrated) fire and smoke image samples (from our training set) representing diverse illumination, indoor/outdoor scenarios. We use TensorRT 8.6.1 on an NVIDIA RTX 2000 Ada GPU for evaluating live inference stream performance and achieving the model benchmarking with a batch size of 1 to represent real-world deployment scenarios. The model is deployed as an optimised INT8 TensorRT engine after initially being exported from the TAO Toolkit. The performance evaluation was carried out across 1000 consecutive frame runs, with each frame having a resolution of 1280 × 720 (FP32 RGB). We use COCO-style mean Average Precision (mAP@0.5:0.95) to measure the model's performance across a range of IoU thresholds from 0.5 to 0.95 to thoroughly evaluate fire/smoke detection stability. With a mAP@0.5 of 94.26% and a mAP@0.5:0.95 of 85.4%, our proposed system demonstrates excellent accuracy in fire/smoke localization and classification across various environmental conditions. A comparison of the FP32, pruned FP32, and INT8 models in Table 5 reveals a maximum accuracy drop of <1.5 %, with mAP@0.5:0.95 continuously over 85%, indicating outstanding performance retention after quantization.

The histogram in Figure 12 illustrates the latency distribution per frame over 1000 inference passes, showcasing a mean inference time of 42.5 ms and a standard deviation of ±3.8 ms, indicating a narrow and consistent runtime spread.

This performance demonstrates the real-time frame processing limit of 24 frames per second (FPS), independent of the input video source frame rate, which is essential for deployment on NVIDIA Jetson devices with the TensorRT engine. Further, we carried out frame-level analysis on 400 test images (See Table 2) and observed that only 14 frames produced false positives, resulting in a false alarm rate of only 3.5%. These numbers demonstrate the model's agility and dependability, particularly in crowded and visually complex industrial conditions where minimising false detections is crucial for real-world implementation. We measure the classification performance of the DetectNet_v2 fire and smoke detector model using ROC and Precision-Recall (PR) curves as shown in Figure 13. ROC curves in Figure 13a reveal high area under the curve (AUC) values of 0.954 for fire and 0.922 for smoke, demonstrating superior discrimination capacity and a low false positive rate over a range of thresholds. On the other hand, PR curves in Figure 13b produce AUC scores of 0.949 for fire and 0.916 for smoke, striking an excellent balance between precision and recall. The performance metrics listed in the confusion matrix discussed in Section 4.1 (Figure 10) further validate these results and discuss how resilient and reliable the model is in correctly identifying fire/smoke incidents, keeping very low misclassifications.

4.4 Edge-device validation

To validate the deployment readiness on edge devices, we comprehensively benchmark the INT8-optimized DetectNet_v2 discussed in Section 4.3 on NVIDIA Jetson Xavier NX (16 GB) and Jetson Orin Nano (8 GB) using DeepStream SDK 6.3 and TensorRT 8.6.1 (NVIDIA, 2025).

We set a batch size of 1 to run a single-stream inference on the 1280 × 720 resolution input stream. Results in Figure 14 show that Jetson Xavier NX achieved 22.3 FPS with an average latency of 45 ms, while Jetson Orin Nano reached 19.4 FPS after a 52 ms latency. The Xavier NX was able to consume 1.4 GB of memory, drawing a maximum of 12.8 W of power. The Orin Nano ended up using 1.7 GB of memory, restricting maximum power consumption to 9.2W. These features, along with both devices achieving 18 FPS, make them ideal

TABLE 4 Proposed fire and smoke detection performance comparison with state-of-the-art models.

Model	Class	Accuracy	Precision	Recall	F1-score
Proposed DetectNet_v2	Fire	95.6%	0.95	0.953	0.952
	Smoke	92%	0.93	0.919	0.922
YOLOv8s (Kong et al., 2024)	Fire and smoke	91%	0.906	0.851	0.878
Faster R-CNN with INCEPTIONV2 (Jadon et al., 2020)	Fire	94.1%	0.89	0.96	0.92
	Smoke	91.7%	0.90	0.91	0.91
SSD MobileNet_v2 (Jadon et al., 2020)	Fire	91.5%	0.87	0.89	0.88
	Smoke	88.2%	0.85	0.88	0.86
DeepCNN Muhammad et al. (2019)	Fire	94.5	0.86	0.97	0.91
AlexNet (Krizhevsky et al., 2017)	Fire	94.39	0.85	0.92	0.88

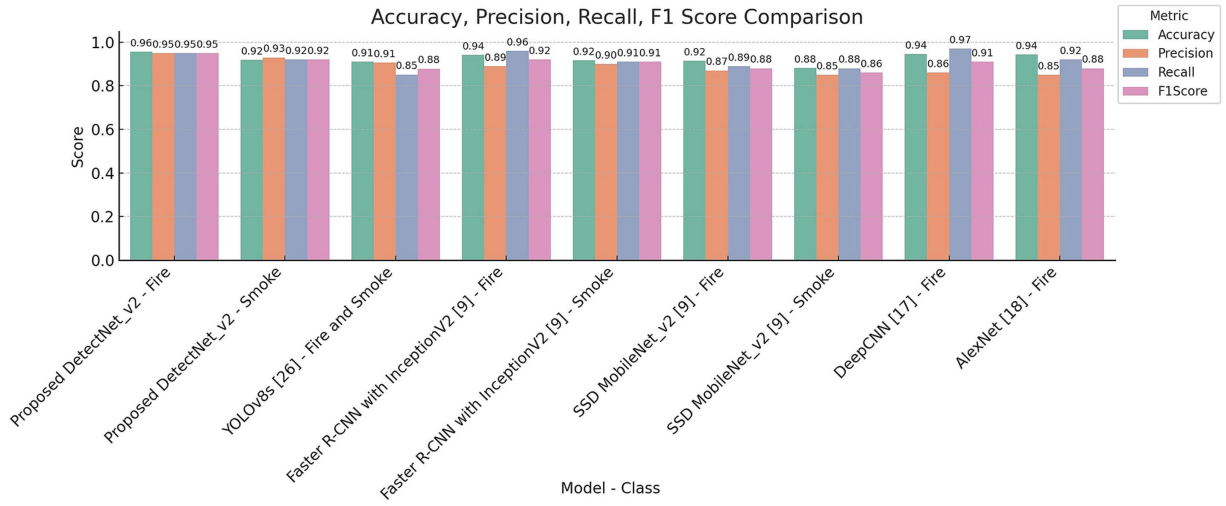


FIGURE 11
Proposed model performance metric comparisons with state-of-the-art methods.

TABLE 5 Pre-trained vs. fine-tuned and retrained DetectNet_v2 (ResNet 18) model comparison.

Model variant	Fire accuracy (%)	Smoke accuracy (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Model size (MB)
Fine-tuned FP-32 (unpruned)	96.42	92.94	94.9	86.7	43
Pruned, retrained with QAT INT8	95.26	91.92	94.2	85.4	37.5

for implementing AI models in resource-constrained (energy and space) applications.

4.5 Failure cases and human-in-the-loop deployment

Figure 15 provides insight into some difficulties encountered by our model during real-world validation by demonstrating the practical constraints using six visual representations grouped into false negative and false positive cases. Figure 15a illustrates a false negative instance that occurred due to fire reflections during the early-stage fire ignitions were confused as fire-like objects, or bright light reflections from a white board resembling smoke, were incorrectly classified as smoke. False positive instances depicted in Figure 15b occur due to similar-looking objects, for instance, welding sparks or steam, that get wrongly classified as fire or smoke. These failure cases are strongly influenced by factors like poor illumination conditions, blurry frames, visually confusing images, or lack of domain knowledge, strongly affecting

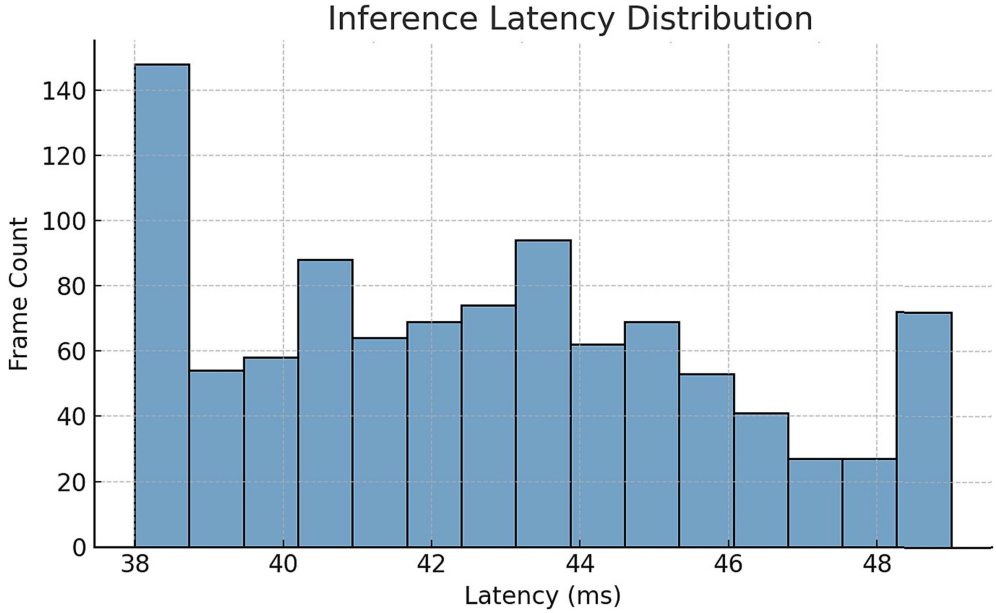


FIGURE 12
Inference latency distribution histogram for 1,000 frame runs.

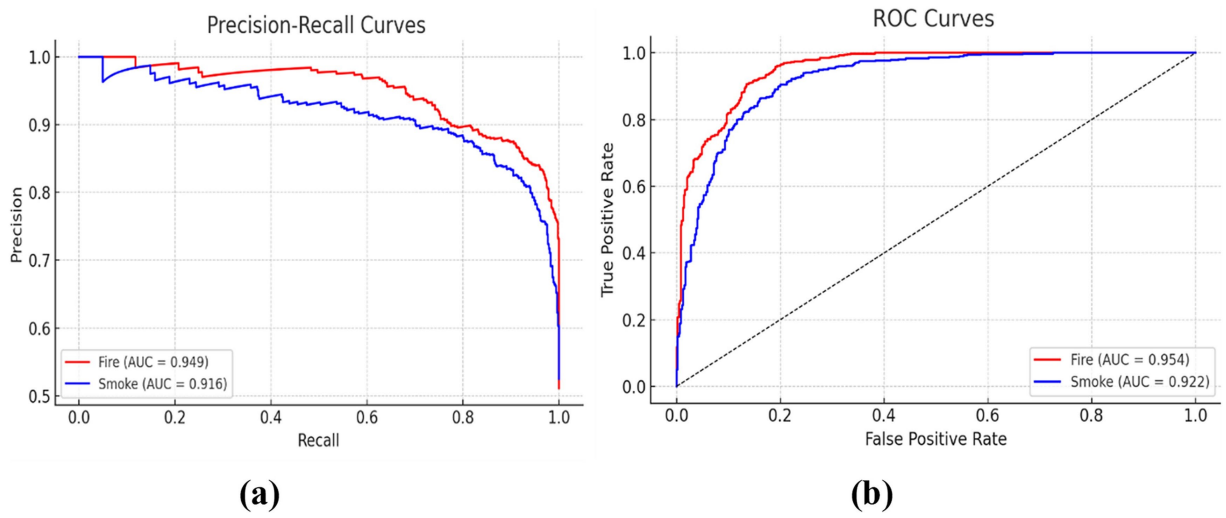


FIGURE 13
The proposed fire and smoke DetectNet_v2 model's classification performance with high area under the curve (AUC) values based on **(a)** ROC curves and **(b)** precision-recall curves.

the reliability of the proposed systems. Training our proposed DetectNet_v2 on diverse datasets certainly has helped to mitigate many of the above-mentioned issues, but a certain amount of risk still exists while deploying in real-world scenarios. To address these issues, a human-in-the-loop (HITL) validation mechanism can be interfaced with conventional safety infrastructure, such as suppression modules or alarm panels, through edge ports or cloud relay, ensuring backward compatibility. Thus, creating a two-tier safety architecture is required to improve safety and decision trustworthiness. HITL interventions work especially well when there are low-confidence detections, visual ambiguities (such as steam, fog, welding sparks, or reflections), sensor failures, and hardware degradation (such as reduced or blurred frames). In such circumstances, the human operator verifies AI-generated alerts before making an alert decision. Overall, HITL assists operators in multi-stream surveillance systems to filter wrong alarm decisions or human oversight errors after correct threat validations. As this method strikes the right balance between automation and supervision, it is particularly well-suited for safety-critical areas like power plants, warehouses, and industrial sites where missed detections or false alarms could have dire repercussions.

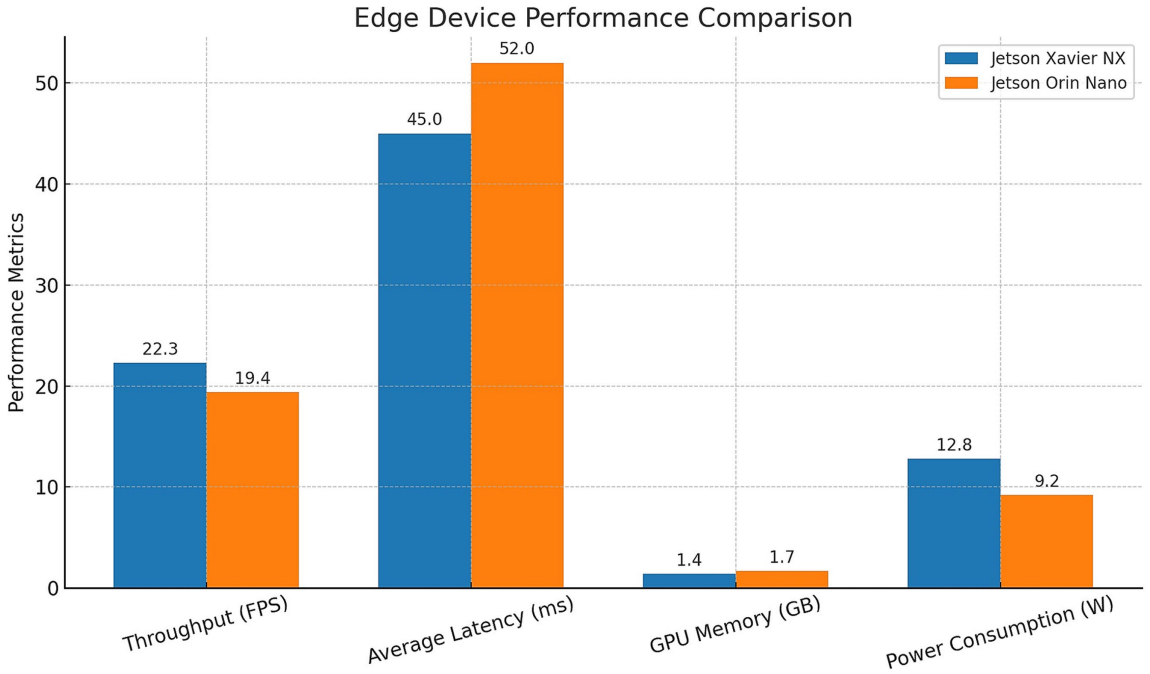


FIGURE 14
NVIDIA Jetson Xavier NX and Jetson Orin Nano (edge device) performance comparison.

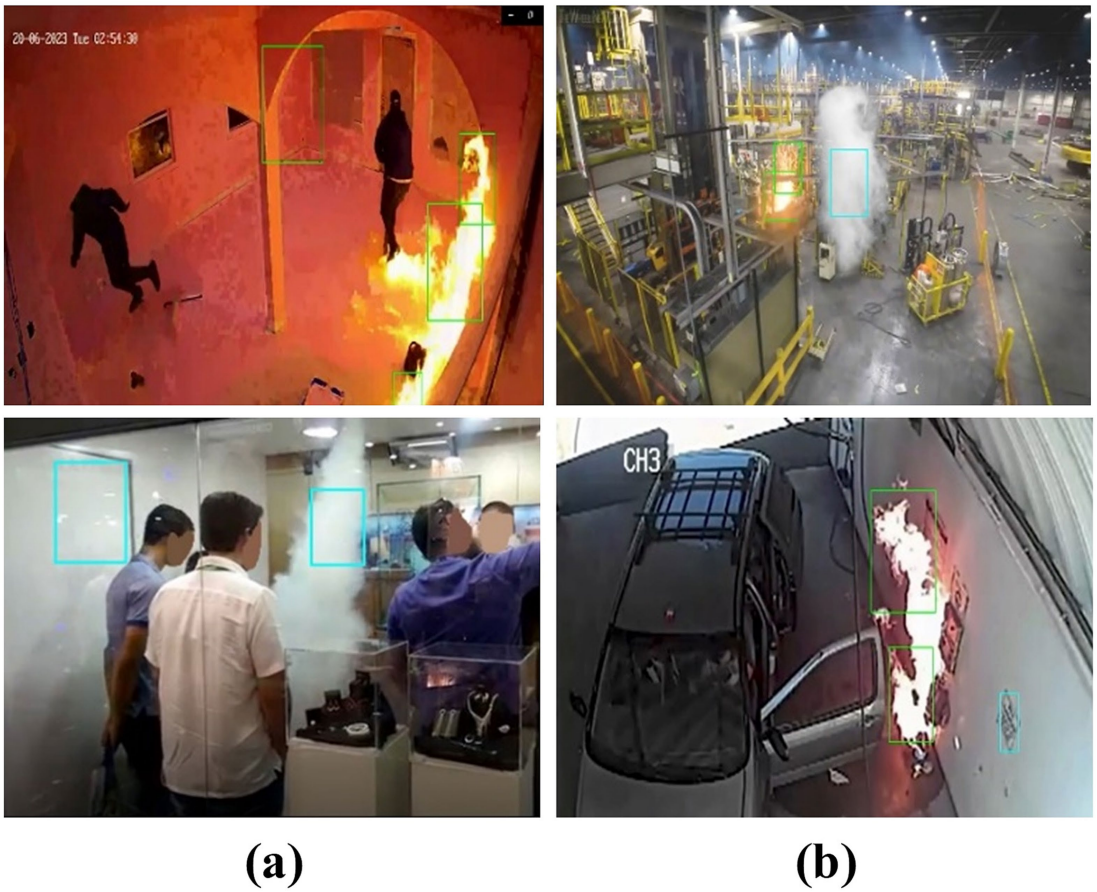


FIGURE 15
Observed failure cases, **(a)** false negative, **(b)** false positive.

5 Conclusion and future scope

In this work, an optimized DetectNet_v2 model with a ResNet-18 backbone is used to develop a real-time, vision-based fire and smoke detection system for both indoor and outdoor industrial environments. Our system leverages pruning and Quantization-Aware Training (QAT) operations on a custom dataset containing 3,000 real-world and augmented fire/smoke images, producing a high detection accuracy (95.6% for fire, 92% for smoke), maintaining a low inference latency of 42 ms. These results enabled us to further validate deployment readiness on edge devices like Jetson Xavier NX and Orin Nano and examine their actual throughput and power efficiency requirements. Evaluation metrics such as mAP@0.5:0.95 (87.4%), low false-alarm rates (3.5%), and ROC/AUC scores further confirm the effectiveness and readiness of models for real-world deployment. This study lays a solid platform for readily deployable AI-driven fire suppression systems and offers reliable, scalable, and context-aware surveillance solutions.

Future research must concentrate on the development of newer model compression and acceleration strategies to optimize edge device performance on 1080p or 4K high-resolution video streams. System's reliability and generalizations can be further improved via diversified datasets constructed by gathering large industrial setup images captured across various geographical locations. The integration of temporal information from video sequences to capture fire and smoke progression patterns may be carried out to boost the model's confidence score. Additional strategies, such as multi-modal sensor fusion by combining vision with heat, gas, or smoke sensors, can enhance system reliability by minimizing false alarms. Finally, motion blur, poor contrast, or visual ambiguity problems encountered from complex scenes during fire and smoke detection tasks can lead to serious problems when working in safety-critical applications. Future systems should explore the possibilities of adopting human-in-the-loop (HITL) verification frameworks into the existing automated detection systems to build a safer planet.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

UD: Supervision, Investigation, Writing – review & editing, Software, Writing – original draft, Resources, Validation, Project administration, Visualization, Methodology, Formal analysis. GM: Data curation, Resources, Funding acquisition, Project administration, Writing – review & editing, Supervision, Formal analysis, Investigation, Writing – original draft, Software. SA: Methodology, Formal analysis, Validation, Visualization, Writing review & editing. SS: Writing – review & editing, Resources, Data curation, Project administration, Conceptualization. HM: Visualization, Writing – original draft, Validation, Data curation, Conceptualization, Writing – review & editing, Software, Investigation, Methodology. YK: Supervision, Project administration, Writing – review & editing, Methodology, Software, Investigation, Data curation, Visualization. YD: Software, Investigation, Writing – original draft, Visualization, Formal analysis, Validation, Data curation, Supervision.

Funding

The author(s) declare that no financial support was received for the research and/or publication of this article.

Acknowledgments

We sincerely thank Ajay Kabadi of DocketRun Tech. Pvt. Ltd. Hubballi, Karnataka, India, for collaborating in carrying out this research work.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The authors declare that no Gen AI was used in the creation of this manuscript.

Publisher's note

References

Accidental Deaths and Suicides in India (ADSI). (2022). Available online at: https://data.gov.in/catalog/accidental-deaths-suicides-india-adsi-2022 (Accessed August 08, 2025).

Avazov, K., Mukhiddinov, M., Makhmudov, F., and Cho, Y. I. (2022). Fire detection method in Smart City environments using a deep-learning-based approach. *Electronics* 11:73. doi: 10.3390/electronics11010073

Beji, V. (2025). Deep Learning Spring 2025: CIFAR 10 classification. Available online at: https://kaggle.com/competitions/deep-learning-spring-2025-project-1 (Accessed August 08, 2025).

Çetin, E., Dimitropoulos, K., Gouverneur, B., Grammalidis, N., Osman Günay, Y., Hakan Habiboğlu, B., et al. (2013). Video fire detection – review. *Digit. Signal Process.* 23, 1827–1843. doi: 10.1016/j.dsp.2013.07.003

Deshpande, U. U., Michael, G. K. O., Araujo, S. D. C. S., Deshpande, V., Patil, R., Chate, A., et al. (2025a). Computer-vision based automatic rider helmet violation detection and vehicle identification in Indian smart city scenarios using NVIDIA TAO toolkit and YOLOv8. *Front. Artif. Intell.* 8:1582257. doi: 10.3389/frai.2025.1582257

Deshpande, U. U., Shanbhag, S., Koti, R., Chate, A., Deshpande, S., Patil, R., et al. (2025b). Computer vision and AI-based cell phone usage detection in restricted zones

of manufacturing industries. *Front. Comput. Sci.* 7:1535775. doi: 10.3389/fcomp.2025.1535775

Deshpande, U. U., Shanbhag, S., Patil, R., Chate, R. A. A., das Chagas Silva Araujo, S., Pinto, K., et al. (2025c). Automatic two-wheeler rider identification and triple-riding detection in surveillance systems using deep-learning models. *Discov. Artif. Intell.* 5:104. doi: 10.1007/s44163-025-00263-3

Feng, X., Xie, R., Sheng, J., and Zhang, S. (2016). Population statistics algorithm based on MobileNet_v2. *J. Physics Conf. Series* 1237:22045. doi: 10.1088/1742-6596/1237/2/022045

Filonenko, D. C., Hernández, and Jo, K. -H. (2018). Fast smoke detection for video surveillance using CUDA. *IEEE Trans. Ind. Inform.* 14, 725–733. doi: 10.1109/TII.2017.2757457

Fires in India: Learning Lessons for Urban Safety (2020). Available online at: https://nidm.gov.in/PDF/pubs/Fires_in_India_2020.pdf (Accessed August 08, 2025).

Gagliardi, A., and Saponara, S. (2020). AdViSED: advanced video smoke detection for real-time measurements in antifire indoor and outdoor systems. *Energies* 13:2098. doi: 10.3390/en13082098

Gov, Fire Alarms-Property Management (2020). Available online at: https://www.london-fire.gov.uk/safety/property-management/fire-alarms/

Healey, G., Slater, D., Lin, T., Drda, B., and Goedeke, A. D. (1993). "A system for real-time fire detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (605–606). IEEE.

Jadon, A., Varshney, A., and Ansari, M. S. (2020). Low-complexity high-performance deep learning model for real-time low-cost embedded fire detection systems. *Procedia Comput. Sci.* 171, 418–426. doi: 10.1016/j.procs.2020.04.044

Kong, D., Li, Y., and Duan, M. (2024). Fire and smoke real-time detection algorithm for coal mines based on improved YOLOv8s. *PLoS One* 19:e0300502. doi: 10.1371/journal.pone.0300502

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017, 2017). ImageNet classification with deep convolutional neural networks. *Commun. ACM* 6, 84–90. doi: 10.1145/3065386

Muhammad, K., Ahmad, J., Lv, Z., Bellavista, P., Yang, P., and Baik, S. W. (2019). Efficient deep CNN-based fire detection and localization in video surveillance applications. *IEEE Trans Syst Man Cybern Syst* 49, 1419–1434. doi: 10.1109/tsmc.2018.2830099

NVIDIA. (2025). Available online at: https://docs.nvidia.com/tao/tao-toolki/t/text/overview.html (Accessed August 08, 2025).

Pincott, J., Tien, P. W., Wei, S., and Calautit, J. K. (2022). Indoor fire detection utilizing computer vision-based strategies. *J. Building Eng.* 61:105154. doi: 10.1016/j.jobe.2022.105154

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (779–788).

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L. C. (2018). "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (4510–4520).

Saponara, S., Elhanashi, A., and Gagliardi, A. (2021). Real-time video fire/smoke detection based on CNN in antifire surveillance systems. *J. Real-Time Image Proc.* 18, 889–900. doi: 10.1007/s11554-020-01044-0

Sathishkumar, V. E., Cho, J., Subramanian, M., and Naren, O. S. (2023). Forest fire and smoke detection using deep learning-based learning without forgetting. *Fire Ecol.* 19:165. doi: 10.1186/s42408-022-00165-0

Senthil, M. (2025). BoWFire, Kaggle. Available online at: https://www.kaggle.com/datasets/malligasenthil/bowfire (Accessed August 08, 2025).

Tien, P. W., Wei, S., Calautit, J. K., Darkwa, J., and Wood, C. (2020). Occupancy heat gain and prediction using deep learning approach for reducing building energy demand. *J. Sustain. Develop. Energy Water Environ. Syst.* 9, 1–31. doi: 10.13044/j.sdewes.d8.0378

Toptaş, B., and Hanbay, D. (2020). A new artificial bee colony algorithm-based color space for fire/flame detection. *Soft. Comput.* 24, 10481–10492. doi: 10.1007/s00500-019-04557-4

Tsang, S.-H. (2020). Review: G-RMI - winner in 2016 COCO detection (object detection). Available online at: https://towardsdatascience.com/review-g-rmiwinner-in-2016-coco-detection-object-detection-af3f2eaf87e4

Wang, X. (2013). Intelligent multi-camera video surveillance: A review. *Pattern Recogn. Lett.* 34, 3–19. doi: 10.1016/j.patrec.2012.07.005

Wei, S., Tien, P. W., Calautit, J. K., Wu, Y., and Boukhanouf, R. (2020). Vision-based detection and prediction of equipment heat gains in commercial office buildings using a deep learning method. *Appl. Energy* 277:115506. doi: 10.1016/j.apenergy.2020.115506

Wu, S., and Zhang, L. (2018). "Using popular object detection methods for real time Forest fire detection," *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, 2018, 280–284.

Xiong, Z., Caballero, R. E., Wang, H., Finn, A. M., Lelic, M. A., and Peng, P.-y. (2007). Video-based smoke detection: Possibilities, techniques, and challenges. Presented at the IFPA, fire suppression and detection research and applications—A technical working conference (SUPDET). Available online at: https://www.academia.edu/30284548/Video_Based_Smoke_Detection_Possibilities_Techniques_and_Challenges (Accessed August 08, 2025).

Zhang, Q., Xu, J., Xu, L., and Guo, H. (2016). "Deep convolutional neural networks for forest fire detection," in *2016 International Forum on Management, Education and Information Technology Application* (568–575). Atlantis Press.

Zhou, Z., Shi, Y., Gao, Z., and Li, S. (2016). Wildfire smoke detection based on local extremal region segmentation and surveillance. *Fire Saf. J.* 85, 50–58. doi: 10.1016/j.firesaf.2016.08.004