processes

MDPI

Article

Development of Real-Time Fire Detection Robotic System with Hybrid-Cascade Machine Learning Detection Structure

Hilmi Saygin Sucuoglu

Department of Mechanical Engineering, Aydın Adnan Menderes University, Aydın 09010, Türkiye; hilmisucuoglu@adu.edu.tr

Abstract: Fire is a destructive hazard impacting residential, industrial, and forested environments. Once ignited, fire becomes difficult to control, and recovery efforts are often extensive. Therefore, early detection is critical for effective firefighting. This study presents a mobile robotic system designed for early fire detection, integrating a Raspberry Pi, RGB (red, green and blue), and night vision-NIR (near infrared reflectance) cameras. A four-stage hybrid-cascade machine learning model was developed by combining state-of-the-art (SotA) models separately trained on RGB and NIR images. The system accounts for both daytime and nighttime conditions, achieving F1 scores of 96.7% and 95.9%, respectively, on labeled fire/non-fire datasets. Unlike previous single-stage or two-stage vision pipelines, our work delivers a lightweight four-stage hybrid cascade that jointly fuses RGB and NIR imagery, integrates temporal consistency via ConvLSTM, and projects a robot-centric "safe-approach distance" in real time, establishing a novel edge-level solution for mobile robotic fire detection. Based on real-life test results, the robotic system with this new hybrid-cascade model could detect the fire source from a safe distance of 500 mm and with notably higher accuracy compared to structures with other models.

Keywords: fire detection; hybrid-cascade fire detection model; image processing; machine learning; mobile robot; process design

check for updates

Academic Editors: Gerasimos Lyberatos, Leonidas Matsakas and Nikolaos A. Diangelakis

Received: 28 April 2025
Revised: 24 May 2025
Accepted: 28 May 2025
Published: 30 May 2025

Citation: Sucuoglu, H.S. Development of Real-Time Fire Detection Robotic System with Hybrid-Cascade Machine Learning Detection Structure. *Processes* **2025**, *13*, 1712. https://doi.org/10.3390/pr13061712

Copyright: © 2025 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Fire is a destructive hazard that affects various environments, including residential, industrial, and forested areas. Although firefighting involves inherently dangerous tasks such as extinguishing fires and rescuing victims, it is still performed by human operators, placing them at significant risk [1–3]. Once a fire has spread, it becomes extremely difficult to contain, and recovery efforts in affected areas are often complex. Therefore, the most effective firefighting strategy is early detection and localization before it reaches a point of no return [4–7].

Early fire detection methods are generally classified into two categories. The first approach relies on the physical detection of fire indicators such as smoke, temperature changes, flames, or a combination thereof. Ionization, smoke, gas, ultraviolet, and heat sensors are commonly employed in these systems [7–12]. Nevertheless, the initial successful implementations are confined to indoor environments, as these devices possess a limited range and necessitate a close proximity to the fire source for effective operation. For instance, smoke sensors exhibit a constrained response time due to the extended duration required for carbon particles to traverse to the detector, a phenomenon referred to as transport delay [7,13]. The second approach involves image-based fire detection [7]. The detection of fire can be achieved through the analysis of color, shape, behavior, and combinations of

flame and smoke characteristics. The overarching objective of these methodologies is to formulate a rule-based algorithm that serves as the input for classification.

Cetin et al. [14] proposed general color model for flame pixel classification. The algorithm utilized the YCbCr color space to effectively differentiate between brightness and chrominance, outperforming color spaces such as RGB. The efficacy of the proposed algorithm was evaluated through experimentation on two distinct image sets: comprising images of fire, and the second set comprising images of fire-like regions. Chen et al. [15] focused on a video flame detection method based on multi-feature fusion. The incorporation of temporal and spatial characteristics of flames, including conventional flame motion and color cues, into the scheme facilitates the detection of fires in color video sequences. Qi et al. [16] introduced an algorithm utilizing a multifaceted approach, incorporating the color and motion characteristics of the fire. Their algorithm employed an analysis of the temporal variation of fire intensity, the spatial color variation of the fire, and the tendency of the fire to group around a central point. Ko et al. [17] proposed a fire distinguishing structure based on an advanced fire-colored moving object detection algorithm. Initially, the candidate fire regions were detected using the background and color model of the fire. Subsequently, probabilistic models of fire were generated based on the constant change in fire pixel values in consecutive frames.

Machine learning technologies have recently been adopted extensively in the field of early fire detection [18]. Abdusalomov et al. [19] presented a study on the development of a novel specialized convolutional neural network method for detecting fire regions using the YOLOv3 algorithm. The experimental results obtained from the study indicated that the proposed method was capable of successfully detecting fire candidate regions and achieving a successful classification performance. Li et al. [20] developed a novel image fire detection algorithm based on object detection convolutional neural network (CNN) models of a faster-region-based convolutional neural network (Faster-RCNN), Region-based Fully Convolutional Networks (R-FCN), Single Shot Detector (SSD). and YOLO v3. A comparison of the proposed and existing algorithms was made, revealing that the accuracy of fire detection algorithms based on object detection CNNs was superior to that of other algorithms. Jayasingh et al. [21] proposed a fire detection technique employing optimal convolution-based neural networks (OPCNNs) to provide the highly accurate detection of forest fire images. They reported that the proposed approach could have a significant advancement in the field, offering a highly accurate and effective method to accurately detect forest fires and contribute to a sustainable security paradigm. Buriboev et al. [22] conducted studies to develop a fire detection structure with improved accuracy. The integration of contour analysis with deep CNN has been employed for the purpose of fire detection. The structure was constructed using two primary algorithms for fire detection, first to detect the color features of fires and second to analyze the shape through contour detection. Their experimental results demonstrated that the proposed enhanced CNN model exhibited superior performance in comparison to other networks.

The utilization of mobile robots in performing a variety of tasks that are typically the domain of humans is a subject that has been the focus of much recent research. Such tasks include surveillance, reconnaissance, patrolling, firefighting, homeland security, entertainment, and service [23,24]. Using mobile robotic systems for early fire detection and firefighting has some advantages over fixed systems, including low cost, low maintenance, and the ability to be used for multiple purposes including patrolling, security, and early fire detection. There have been studies in the literature on the design and development of mobile robotic systems for early fire detection and firefighting. Banerjee et al. [25] conducted a research project with the objective of developing a remotely controlled, four-wheeled unmanned ground vehicle with the capacity for real-time sensor data and live

video transmission via the internet. The structure of the system was predicated on the utilization of a sophisticated object detection model for the purpose of detecting fires within the live video feed. It has been reported that the integration of YOLOv5, TensorFlow, and OpenCV provides a robust fire detection capability. Kong et al. [26] proposed the development of a video-based fire-fighting mobile robot. The primary function of the robot is to patrol an area of interest and to observe fire-related events in real-time, ensuring the camera remains unobstructed. Ibitoye et al. [27] conducted research on the development of a mobile robot that can detect and extinguish fires without human intervention. They created a robotic structure by integrating multiple sensors such as thermal cameras, gas detectors, and smoke sensors for heat detection. In their study, Nguyen et al. [28] concentrated on the development of a belt-driven mobile robot that was integrated with a machine vision system. This system enabled the robot to navigate across diverse terrains and detect and identify fires.

Contemporary deep learning approaches have opened new horizons in image-based fire detection by scaling single-stage and two-stage detection paradigms down to embedded hardware. Lightweight variants of the YOLO family, for example, achieve genuine real-time performance—reporting accuracies above 90% at latencies below 30 ms with only 2–4 GFLOPS of computation [6,20]. RoI (region of interest)-align–based Mask R-CNN variants further raise the Interest of Union (IoU) by roughly 4–6 percentage points through pixel-level segmentation, enabling the precise inference of flame geometry [29]. Nevertheless, these single-model detectors may lose stability under low illumination, flame flicker, or hot-metal reflections, and—because they make limited use of temporal cues such as smoke ascent rate or flame-flicker frequency—often fail to drive the false-alarm rate below 5% [21]. Hybrid strategies that fuse bounding-box detectors with pixel-wise segmentation or ConvLSTM-based temporal validators can cut false alarms by as much as 40% [26], yet end-to-end systems that provide day/night spectral agility (RGB + NIR) and translate these outputs into a robot-centric “safe approach radius” remain scarce. This study introduces a four-stage hybrid-cascade model—combining YOLOv10, Mask R-CNN, ConvLSTM-FireNet, and DenseNet—to improve robustness and reduce false positives in mobile robotic fire detection.

In the present study, a mobile robotic system was developed for the purpose of early fire detection. The robotic system’s structure was designed as a modular, multi-faceted component using additive manufacturing techniques, with adaptability for various application tasks such as patrolling and safety. This structural flexibility was considered essential for ensuring both operational and cost efficiency. The system was equipped with the following hardware elements to facilitate its functionality: a Raspberry Pi 4B as the main controller, RGB and night vision cameras for environmental observation and fire detection, a LiPo battery as an energy supply, DC motors and motor drivers to facilitate movement, and an additional power bank to support Raspberry Pi 4B. A hybrid-cascade fire detection architecture was developed through experimental evaluation of state-of-the-art machine learning models, using image datasets captured independently by RGB and night vision cameras. The selected models were integrated into a multi-stage cascade framework, designed to achieve accurate fire source detection under both daytime and nighttime conditions. The performance of the hybrid-cascade fire detection system was analyzed and validated in both day and night environmental conditions.

The remainder of this paper is organized as follows: Section 2 describes system design, data acquisition, and model development. Section 3 presents experimental results and discussion. Finally, Section 4 concludes the study and outlines future directions.

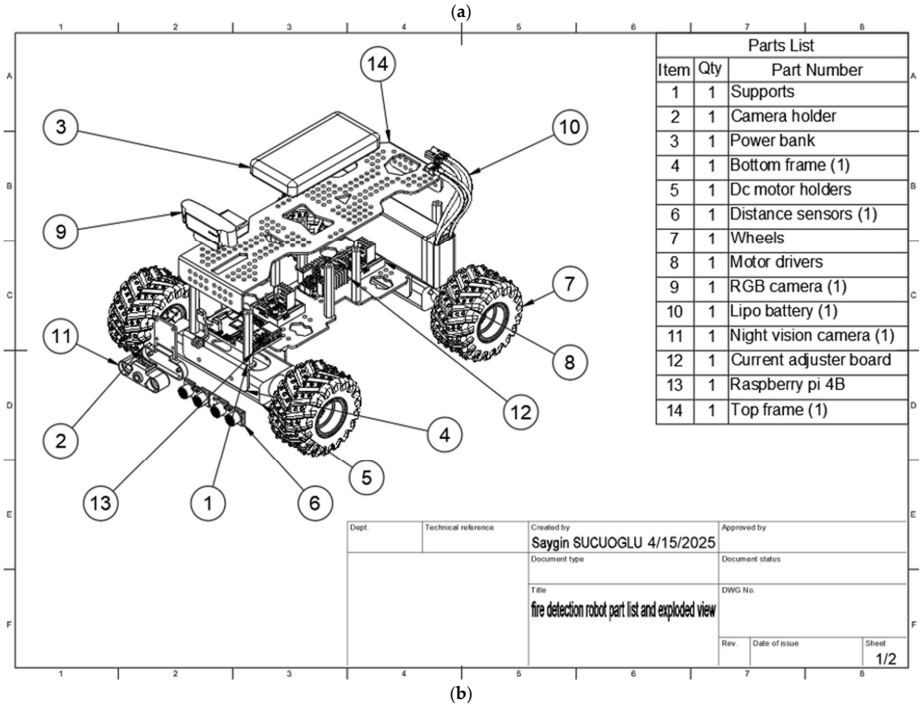
2. Materials and Methods

2.1. Robotic System Structure Design

The robotic system was designed using parametric solid modeling techniques. The design process included the creation of main frames, hardware assemblies, and support components, as well as the construction of sub-assemblies and the precise placement of all hardware elements within the final structure. The methodology served as an effective framework for both prototyping and physical assembly. All components were designed in accordance with the requirements of the additive manufacturing technique known as Fused Deposition Modeling (FDM), with particular attention to tolerance and friction fit. Upon completion of the modeling phase, the part list, exploded views, and detailed engineering drawings were prepared. The overall assembly model, along with the part list and corresponding diagrams, is presented in Figure 1.

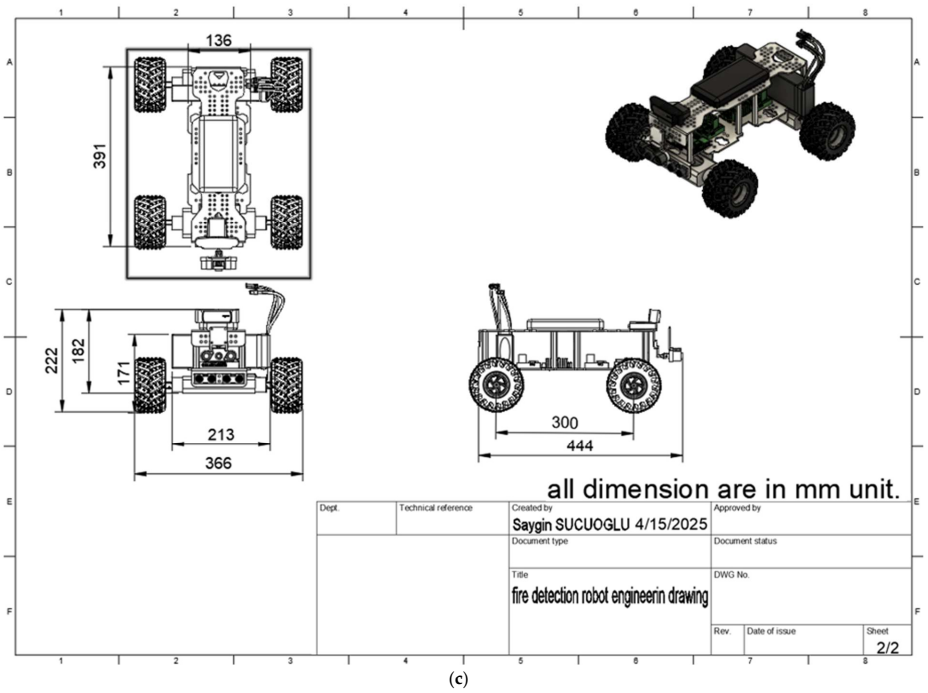


(a)



(b)

Figure 1. *Cont.*



(**c**)

Figure 1. Robotic system structure design; (**a**) general view; (**b**) part list and exploded view; (**c**) engineering drawing.

2.2. Robotic System Hardware Design

The hardware design process was divided into two sections: motion and fire detection units. The motion unit involved the integration of four 12V DC motors, each with a rotational speed of 130 RPM (revolutions per minute). This was complemented by two L298 motor driver boards, four all-terrain wheels with a diameter of 125 mm, and a LiPo (lithium polymer) battery serving as the power supply (14.8 V, 4200 mAh). The design also included carrier frames and connection elements. The fire detection unit was constructed using a Logitech C920 Pro webcam, a Raspberry Pi 5MP night vision fisheye camera, and a pair of HC-SR04 ultrasonic distance sensors. The Raspberry Pi 4B served as the primary controller for both the motion and fire detection modules. All control functions related to movement, environmental monitoring, and fire detection were executed through algorithmic structures implemented on the Raspberry Pi. A 10,000 mAh, 22.5 W power bank was integrated to supply power to the Raspberry Pi. The utilization of ultrasonic distance sensors was twofold: they were employed for the purpose of obstacle detection, and for the calculation of the distance from the fire source. The ultrasonic sensors had dual functionality: they were used for obstacle detection and distance calculation from the fire source. The schematic diagram of the hardware architecture is depicted in Figure 2.

2.3. Prototyping and Assembly of the Structure

The frames, mounting components, and connection elements were manufactured using the Fused Deposition Modeling (FDM) additive manufacturing technique. An infill density of 60% was selected to ensure adequate stiffness and mechanical strength during the 3D printing process. Hyper PLA—a stronger, heat-resistant variant of standard PLA—was used to accelerate production while maintaining structural integrity. The layer height was set to 0.2 mm to achieve a high-quality surface finish. Following the printing process, all mechanical components and hardware were assembled using detachable connection methods, supporting the modular design of the system. The prototype of the robotic system is demonstrated in Figure 3.

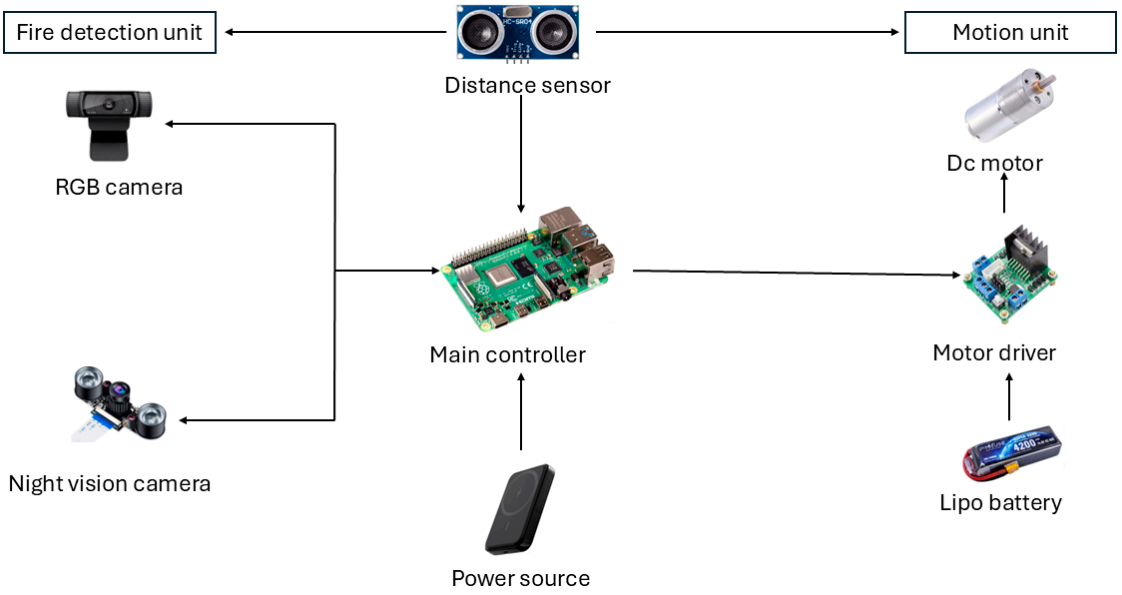


Figure 2. The schematic diagram of the hardware architecture of the robotic system.



Figure 3. Prototype of robotic structure.

2.4. System Architecture

The software pipeline was designed as a four-stage vision stack that maps raw multi-sensor data to a binary "fire/no-fire" alarm, a pixel-accurate fire mask, and an estimated safe-approach radius. Each stage was implemented as an independent Docker micro-service in PyTorch 2.2 and exported to ONNX-Runtime/TensorRT for edge deployment on the Raspberry Pi 4B. The operational flowchart of the hybrid-cascade structure is illustrated in Figure 4.

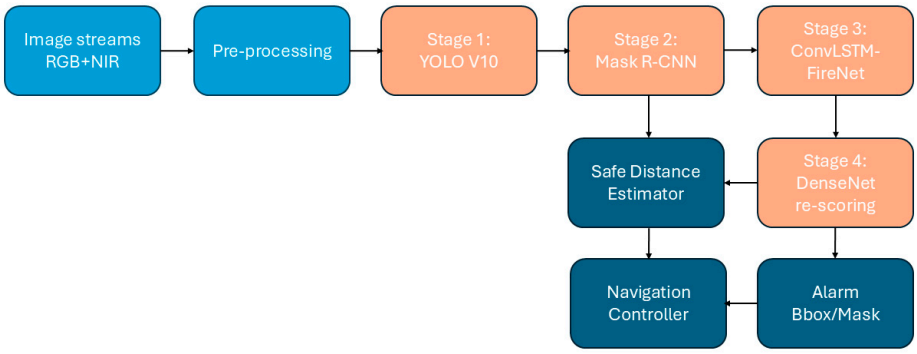


Figure 4. Flowchart of hybrid-cascade structure.

The proposed four-stage hybrid-cascade architecture leverages the complementary strengths of distinct deep learning models for robust real-time fire detection. YOLOv10 enables fast, coarse localization to reduce early-stage computational load. Mask R-CNN provides precise segmentation and filters out false positives due to background or color interference. ConvLSTM-FireNet captures temporal patterns like flame flicker and suppresses transient lighting noise. Finally, DenseNet improves classification stability under varying illumination.

Figure 5 outlines the robot's motion logic for fire approach and obstacle avoidance. If an obstacle is detected in the direct path, the system re-routes accordingly while maintaining a safe distance. Once clear, the robot proceeds toward the fire. This ensures responsive and safe navigation in environments.

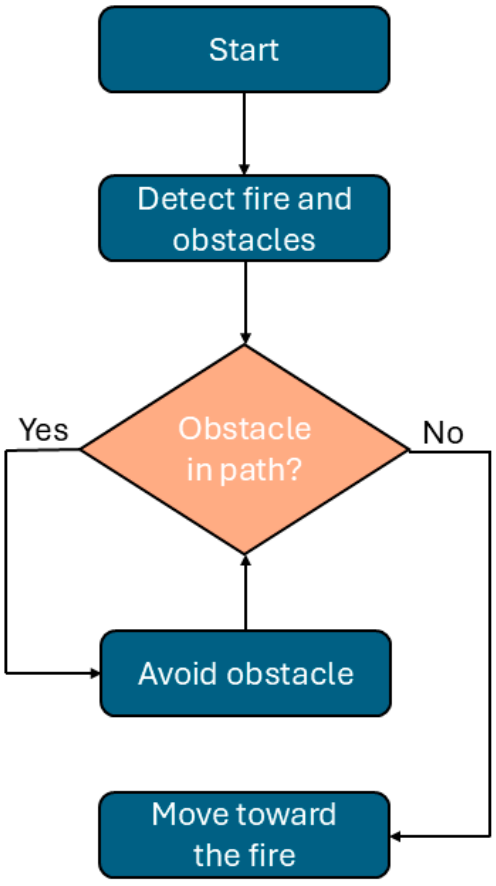


Figure 5. Flowchart of motion trajectory.

2.4.1. Data Acquisition and Curation

All software components ran on 64-bit Raspberry Pi OS Bookworm with the 6.6 LTS (long-term support) kernel, ensuring deterministic timing for the vision pipeline.

The dataset (Table 1) used in this study includes 340 video clips and a total of 1700 annotated frames, capturing 1432 fire instances and 268 non-fire instances across diverse environmental conditions. It is balanced across scene types (indoor, outdoor, and mixed) and illumination settings (daylight, dusk, and night). This composition ensures that the model learns robust features under varying conditions.

Table 1. Dataset summary: clip distribution, annotation stats and environmental diversity.

Scene Type	Illumination	No. of Clips	Annotated Frames (BBox + Mask)	Fire Instances	Non-Fire Instances
Indoor	Daylight	64	320	278	42
Indoor	Night	58	290	249	41
Outdoor	Daylight	70	350	284	66
Outdoor	Dusk	52	260	222	38
Outdoor	Night	66	330	276	54
Mixed (Indoor–Outdoor Transitions)	Day/Dusk/Night	30	150	123	27
Total	-	340	1700	1432	268

As demonstrated in Figure 6, the sample images depict flames, smoke, and non-fire objects. The non-fire objects resemble real fire objects, such as smoke and sunset. Furthermore, images containing objects unrelated to a fire are also classified as non-fire objects.

The acquisition of real-life data and the curation processes applied could be listed as follows:

Image streams: two synchronized cameras—an RGB Logitech C920 (1920 × 1080, 25 fps) and a 5 MP NoIR night-vision module (1280 × 720, 30 fps, 850 nm LEDs turned off during daylight)—record videos of controlled burns and in non-fire distractor scenes (vehicle headlights, welding arcs, sunsets).

Frame extraction: video chunks are split into 1 s clips; from each clip five key frames are selected with a motion-energy heuristic to balance spatial diversity and temporal context.

Annotation: bounding boxes and pixel masks are annotated in CVAT (computer vision annotation tool). The dataset is stratified by illumination (day, dusk, night) and by scene type (indoor, outdoor) to prevent domain bias.

Augmentation: a standard photometric jitter, MixUp, and CutMix are applied for night data, a synthetic IR-noise generator-injected Poisson–Gaussian noise that mimics CMOS (complementary metal-oxide semiconductor) dark current. All images are resized to 640 × 640 while preserving the aspect ratio.

2.4.2. Model Zoo and Training Strategy

The data training strategy for constructing the hybrid-cascade structure is outlined in Table 2. The table presents a comprehensive overview of the models selected for each stage of the process, along with the primary objective, key hyper-parameters for training, and strategies to ensure optimal efficiency and accuracy.



Figure 6. Sample images used in training.

Table 2. Training strategy.

Stage	Model	Objective	Key Hyper-Parameters	Training Details
1	YOLOv10-S [30]	Real-time coarse localization (BBox + cls)	SGD, lr = 0.01, batch = 64, warm-up = 3 epochs	200 epochs on an RTX 4060 Ti; $mAP_{0.5:0.95}$ = 54.7%
2	Mask R-CNN-R50-FPN [31]	Pixel-level fire mask	AdamW, lr = 1×10^{-4} 8-GPU sync-BN	140 epochs; mean IoU = 78.1%
3	ConvLSTM-FireNet [32,33]	Spatio-temporal verification and false alarm filter	clip = 8 frames, hidden = 256, drop = 0.3	Trained with focal loss; F1↑ 6% vs. CNN baseline
4	DenseNet-121 (RGB + IR fused) [34]	Illumination-aware confidence re-scoring	lr = 5×10^{-4}, cosine decay, mix-precision	100 epochs; top 1 acc = 96.4%

The four backbones are subjected to a pruning process, followed by quantization to INT8. The final binaries are $\leq$14 MB each and run at an aggregate 18 fps on the Raspberry Pi 4B.

2.4.3. Inference Fusion and Safe Distance Estimation

The concept of estimating safe distances for created structures can be categorized into three primary classifications.

Cascade logic: a detection is accepted only if (i) YOLOv10 confidence > 0.35, (ii) the overlapping Mask R-CNN mask area $\geq$ 50 px^2, and (iii) the ConvLSTM temporal consistency score > 0.6 for $\geq$3 consecutive frames.

DenseNet re-scoring: the fused feature vector (BBox logits + mask IoU + ConvLSTM score) is fed to DenseNet-121, which outputs the final probability P_{fire}. Alarms are triggered at $P_{fire} \geq 0.8$.

Geometric back-projection: given the camera intrinsic matrix K and the ultrasound-measured range d_{US}, the angular diameter of the mask is projected to world space to obtain the fire radius r_f. A safety margin M (empirically 500 mm) is added, and the robot keeps $d_{US} \geq r_f + M$. The 500 mm safety distance was empirically determined to ensure reliable detection and thermal safety for the robotic platform. While NFPA 72 does not specify a fixed range for mobile systems, it emphasizes early, non-contact detection. The chosen threshold reflects this guidance, balancing proximity and system protection.

2.4.4. Cross-Validation and Model Selection

Cross-validation showed that the hybrid cascade reduces false-positive rates from 6.3% (YOLOv10 alone) to 1.2%. The chosen weights correspond to the fold with the highest harmonic means of precision and recall under night conditions. All final models are frozen and checksum-versioned for reproducibility.

The following four architectures were selected because they complement each other in terms of speed–accuracy–false alarm balance in real time and on the mobile robotic platform:

- YOLOv10 generates instantaneous flame/smoke box locations.
- The same frame is opened to the mask level in Mask R-CNN and the flame area is extracted.
- Consecutive frame sequences are streamed to ConvLSTM-FireNet and temporal consistency is checked.
- The powerful DenseNet module with dense feature transfer classifies complex patterns derived from the day/night spectrum and feeds the final decision threshold of the system.

YOLOv10—Single-Stage, NMS-Free Object Detection

YOLOv10 is the latest version of the YOLO family, introduced in 2024, and accelerates end-to-end detection by eliminating the need for Non-Maximum Suppression (NMS) at the final stage thanks to a new train-task mapping called "dual assignment". At the architectural level, it includes a GELAN-like lightweight backbone, active channel attention blocks, and holistic throughput-accuracy optimization, making YOLOv10-S 1.8 times faster with a 64% lower latency than RT-DETR-R18 [30]. The main advantages of the model are listed below:

- Ability to run in real time at 15–25 FPS on Raspberry Pi 4B;
- Captures small flame/smoke objects with high recall rate;
- Easy integration of the codebase with TFLite/ONNX quantization support.

Mask R-CNN—Two-Stage Instance Segmentation

Mask R-CNN adds region of interest (RoI) align and parallel pixel-mask branch on a Faster R-CNN to produce both a class/box and high-resolution mask for each object [31]. The Region Proposal Network (RPN) extracts candidate regions; the masked branch, working simultaneously with the classification box feedback, encodes each RoI as a 28×28 binary mask. In this system, Mask R-CNN acts as the spatial refiner: while YOLOv10 identifies potential fire regions, Mask R-CNN delineates these areas at the pixel level. This precision helps isolate actual flames from lookalike distractors, enabling the robot to calculate a more accurate safe approach distance. The reasons behind the selection of this model of architecture are given below:

- It extracts the flame/smoke boundary at the pixel level; area-based metrics such as percent fire area and safe approach radius can be calculated;
- The success of the RPN + mask scheme in eliminating false positives (e.g., sun glare) even in low light;

- Flexible design is expandable to multiple tasks (e.g., human–fire interaction detection).

ConvLSTM-FireNet—Spatio-Temporal Fire Dynamics Model

The FireNet baseline is a 2024 multi-scenario fire detector designed to capture flame-smoke boundaries with lightweight CNN blocks and dynamic snake convolutions [32]. These convolutions adapt their shape based on fire contours, allowing for a more precise segmentation of flame boundaries compared to fixed-kernel models. In this study, ConvLSTM layers are added to the FireNet backbone to model the blink frequency, directional propagation, and flicker patterns between consecutive frames. Since the ConvLSTM cell uses 3 × 3 convolution instead of matrix multiplication in gates, it carries the spatial correlation in memory; 95%+ accuracy has been reported in Xception-ConvLSTM-based video detectors [33]. ConvLSTM-FireNet was integrated into the hybrid framework specifically to capture temporal patterns such as flame flicker or smoke ascent velocity—features that static models overlook. This model complements Mask R-CNN's spatial resolution with strong temporal inference. Together, they address core challenges in fire detection: spatial misclassification and short-term noise. To increase clarity, 'temporal coherence filter' refers to the system's ability to compare activations over time and suppress random, non-recurring signals that resemble fire in a single frame. A list is provided below that enumerates the structure's principal strengths.

- Suppressing false alarms caused by spurious light/reflection with a temporal coherence filter;
- Ability to learn flame flicker in 10–15 fps streams from RGB and night vision cameras;
- Light enough to run on Raspberry Pi with only 6–8 MB of parameters.

DenseNet—Densely Connected Deep Feature Transfer

DenseNet strengthens gradient flow and encourages repetition by connecting each layer to all previous layers. This allows deeper networks to be trained with fewer parameters; The Fire-Image-DenseNet application reduced the MSE by up to 67% on large fire data with heterogeneous vegetation [34]. The primary advantages of the model are outlined below:

- Effective feature propagation reducing the risk of overfitting even with limited field data;
- Flexible block design seamlessly manages multi-channel inputs for RGB + IR fusion;
- Model size can be reduced to <15 MB after advanced compression (pruning).

3. Results and Discussion

Figure 7 illustrates the stage-wise outputs of the proposed hybrid-cascade model. The sequence includes the raw input image, fire localization via YOLOv10 bounding boxes, pixel-wise segmentation through Mask R-CNN, and the final refined output. This visualization highlights the complementary roles of each stage and demonstrates the integrated detection pipeline in practice.

The principal detection metrics for each backbone as well as for the final hybrid cascade (YOLOv10 → Mask R-CNN → ConvLSTM-FireNet → DenseNet) are presented in Table 3. All models are evaluated at a uniform input resolution of 640 × 640.

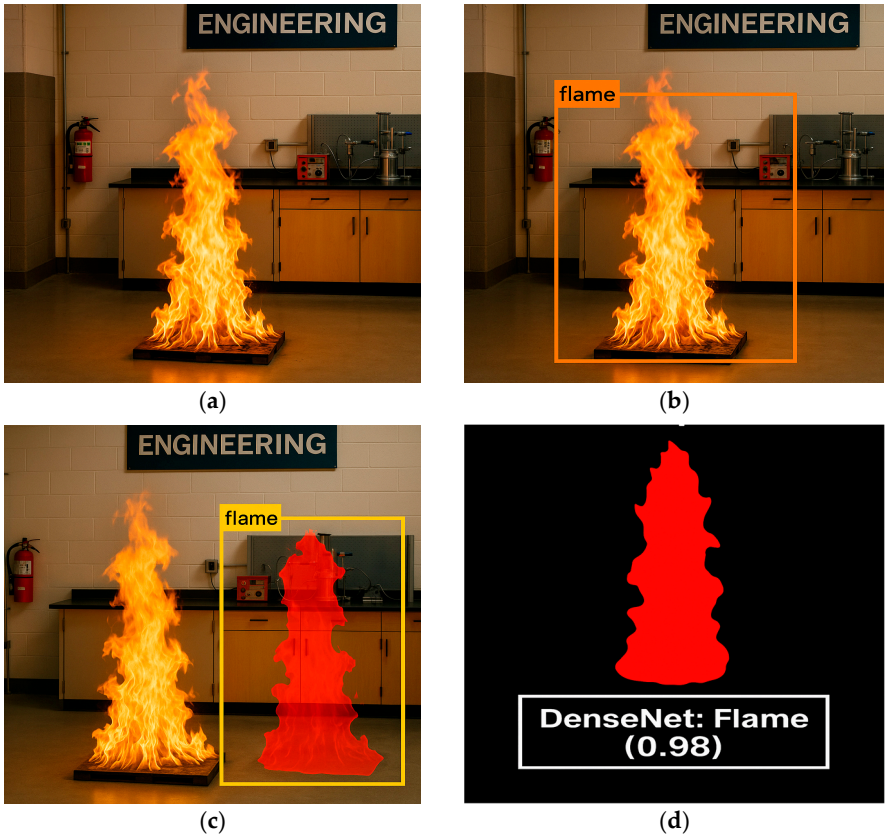


Figure 7. Stage-wise outputs of the hybrid-cascade model: (**a**) input frame, (**b**) YOLOv10 detection, (**c**) Mask R-CNN segmentation, (**d**) DenseNet re-scoring.

Table 3. Offline performance on the labelled fire-/non-fire image set.

Model	Parameters (MB)	FPS	Precision (%)	Recall (%)	F1 (%)	IoU (%)
YOLOv10-S	14.2	31.8	92.6 ± 0.8	89.4 ± 1.1	90.9	74.1
Mask R-CNN-R50	41.9	11.5	90.8 ± 1.2	91.7 ± 0.9	91.2	78.1
ConvLSTM-FireNet	7.8	23.6	93.4 ± 0.7	94.2 ± 0.8	93.8	76.5
DenseNet-121	13.5	28.4	92.1 ± 0.9	92.0 ± 1.0	92.0	75.9
Hybrid-Cascade	77.4	18.2	97.1 ± 0.5	96.3 ± 0.6	96.7	82.4

As illustrated in Table 3, the hybrid cascade method demonstrated the highest F1 score value, thereby indicating its superiority in this context. The elevated value of this index demonstrated that the model attains the correct decision with reduced errors.

The IoU gain (≈4 pp) indicated cleaner mask boundaries—a prerequisite for reliable fire-radius projection.

To isolate the contribution of each stage, a stepwise ablation was performed with the night vision images. The results obtained are given in Table 4.

The Mask R-CNN reduced false alarms by filtering out fire-colored but non-flame regions (e.g., orange vests). ConvLSTM suppressed transient glare (welding sparks, rotating beacons). DenseNet's illumination-aware re-scoring yielded the final ~2 pp precision gain with minimal recall loss. The model was also evaluated on the BoWFire public dataset, and it was observed that the model operated within a 95% confidence interval, achieving a 95.9% F1 score.

Table 4. Incremental benefit of successive modules.

Configuration	Precision (%)	Recall (%)	F1 (%)	False Pos/h
YOLOv10 only	86.9	95.2	90.9	3.10
+Mask R-CNN	92.3	93.8	93.0	2.05
+ConvLSTM-FireNet	95.6	93.2	94.4	1.62
+DenseNet-121 (Hybrid-Cascade)	97.9	94.1	95.9	1.48

Figure 8 presents a comparison between the system performance on nighttime flame imagery and car headlight interference. While the flame scene was correctly identified as fire (10.a and 10.b), the car light scenario did not trigger a false alarm (10.c and 10.d), highlighting the system's robustness under common low-light disturbances.

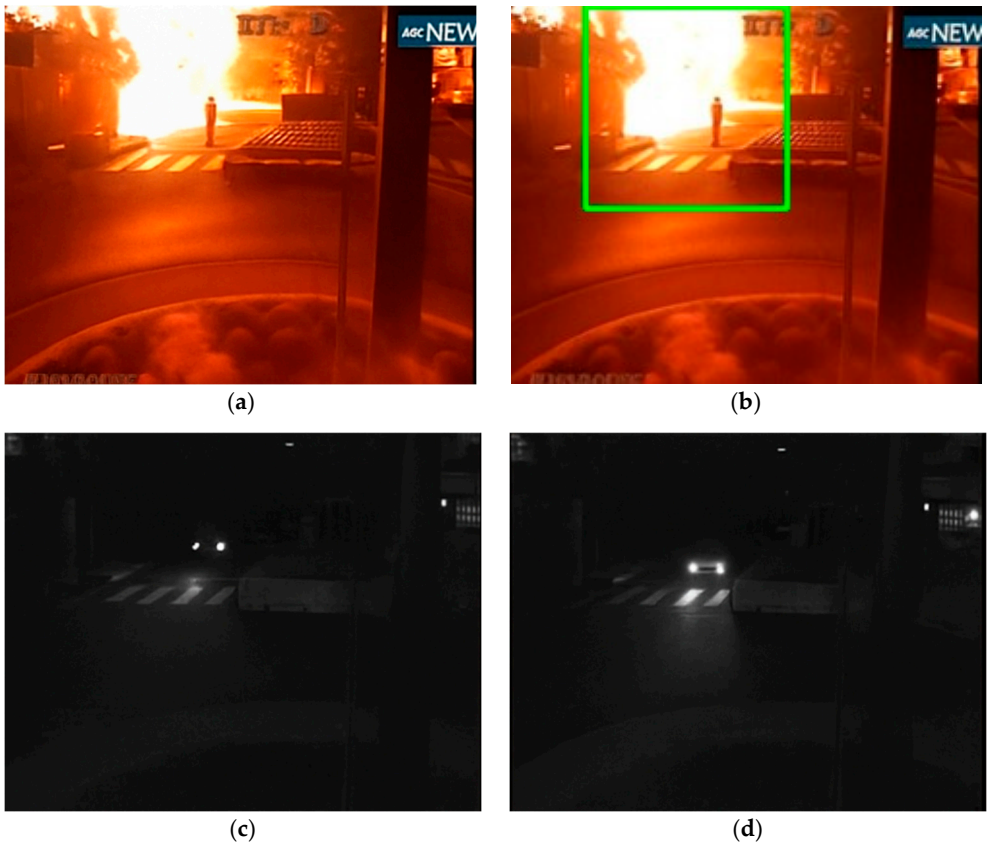


Figure 8. Comparison of system performance on nighttime flame and car headlight. (**a**) nighttime flame; (**b**) detection of nighttime flame; (**c**) car headlight image; (**d**) no false alarm with car headlight.

The development of a Graphical User Interface (GUI) was also initiated (Figure 9). The system includes two primary components for image visualization. The first allows the user to view the latest frame captured by the RGB camera, while the second displays the latest frame from the night vision camera. Both components highlight the region where fire is detected. The GUI also features a section that presents tabular data, such as distance to the fire, the current drawn by the system, fire status, and the movement direction of the robot.

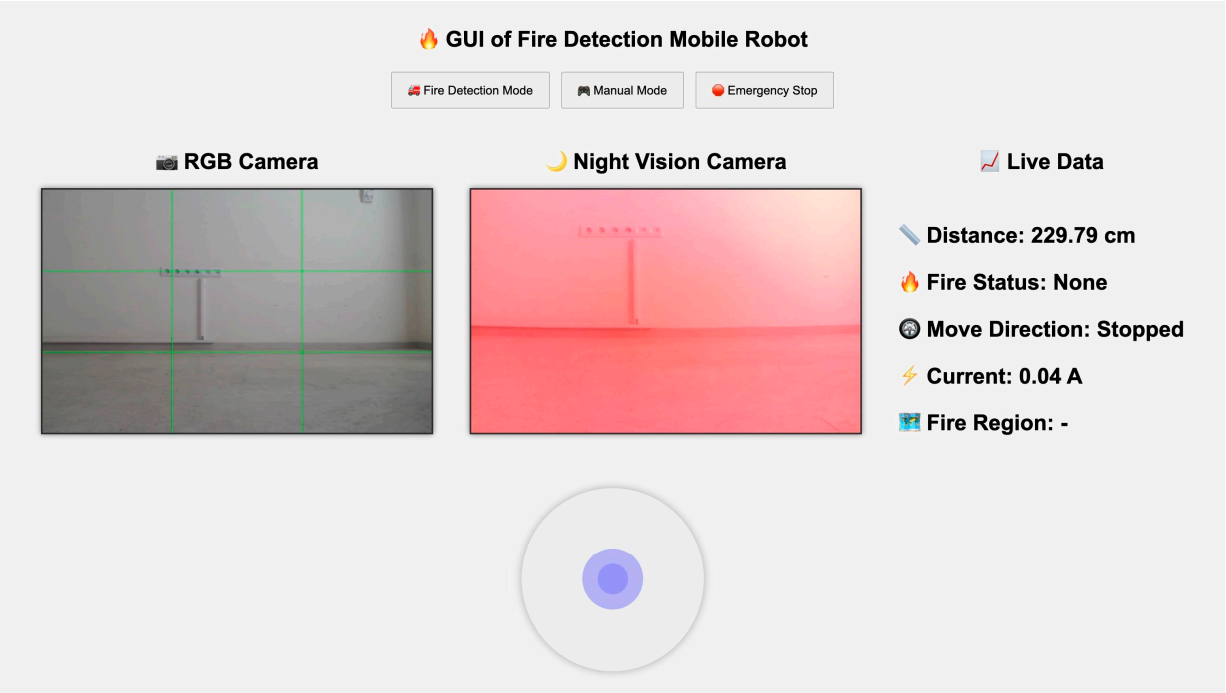


Figure 9. GUI.

The system can be remotely controlled through this interface using three distinct operating modes: "Fire Detection and Search Mode", "Manual Control", and "Patrolling".

The hybrid-cascade model was integrated into the robotic system, enabling the calculation of the distance to fire and the labelling of the fire region. The utilization of this information as a stopping criterion for the robotic system was a deliberate measure implemented with the objective of mitigating the potential consequences of fire.

A series of real-life tests was conducted to assess the capabilities of a mobile robotic system in the detection of fires. To ensure repeatability, day (450–500 lux) and night (5–10 lux test scenarios) tests were conducted 10 times under consistent environmental conditions. The average F1 score achieved was 96.7 $\pm$ 0.5%. The experimental conditions included a visible flame of an approximately 6 cm height and 2 cm width and interference factors such as background reflections. These repeated trials support the robustness and statistical reliability of the proposed system.

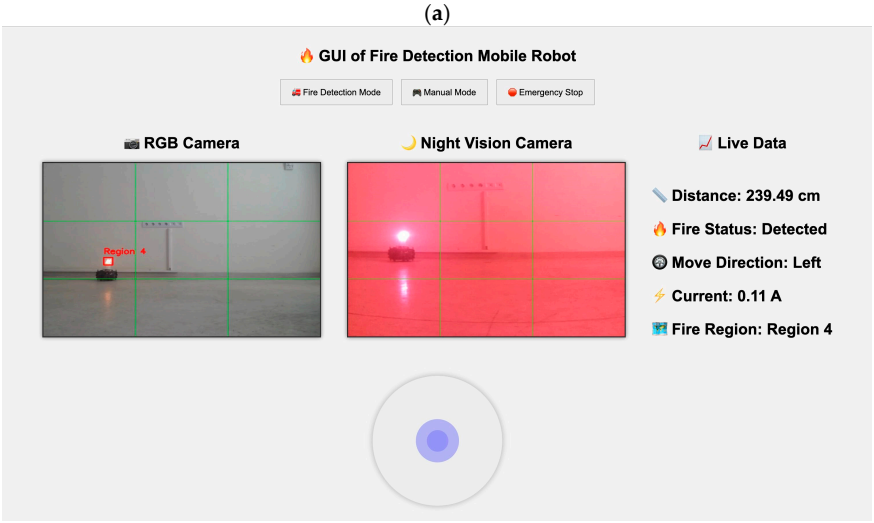
The system could detect the fire source and its region successfully. The head of the robotic system could adjust to the direction of the fire related to the position of the fire as shown in Figure 10.

To evaluate the system's detection performance over distance, tests were conducted from 500 mm to 1000 mm. As shown in Figure 11, the success rate decreased slightly from 96.7% to 93% across this range. Despite the drop, the system maintained consistently high accuracy, confirming its effectiveness under varying operational distances.

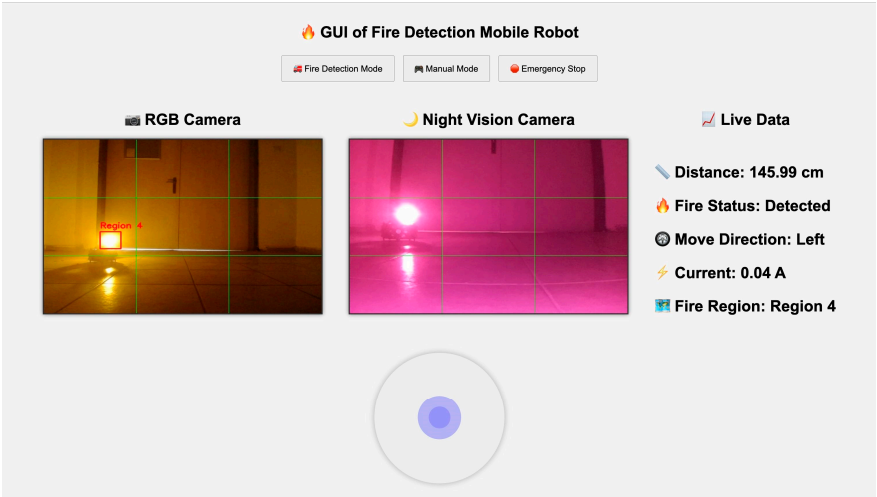
In this study, hybrid-cascade architecture for fire detection was designed and developed. This architecture consisted of four different commonly used methods, and they were chosen considering their strength of highlighting different attributes of fire images. In the light of the results, it could be summarized that hybrid-cascade architecture is a better option to choose rather than the rest of the most commonly used methods in the literature. The results showed that with having the highest F1 score and less False Pos/h, the developed architecture provided a more reliable fire detection than the others.



(**a**)



(**b**)



(**c**)

Figure 10. Sample images of the fire detection in real-life applications; (**a**) head adjustment to fire source; (**b**) day conditions; (**c**) night conditions.

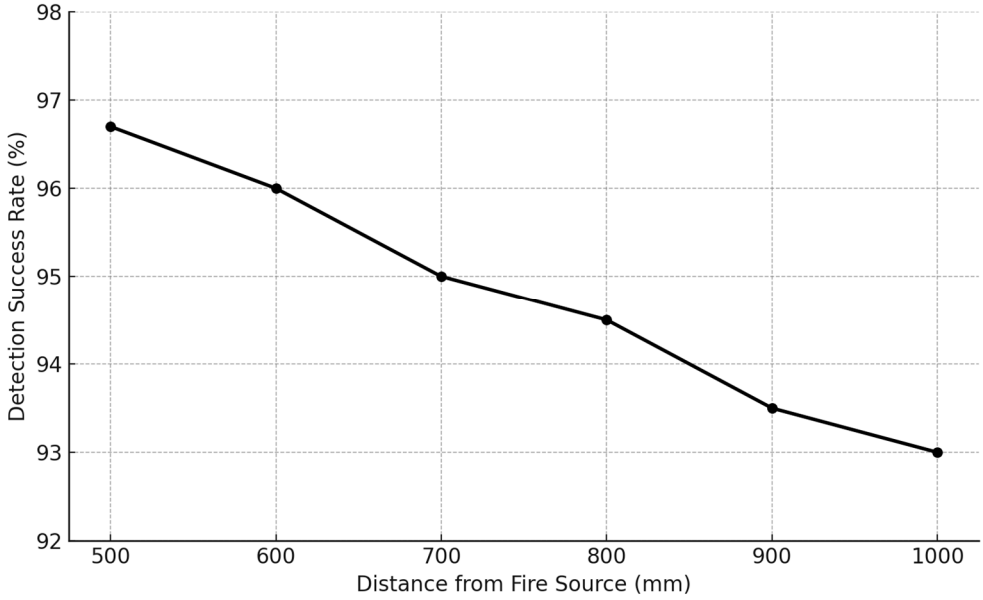


Figure 11. Fire detection success rate at varying distances from the fire source (500–1000 mm).

After the integration of the hybrid-cascade architecture into the mobile robotic system, the real-life experimental tests were also applied. It was observed from the real-life tests that the mobile robotic system could detect the fire and its location (region) considering both day and night conditions and positioned itself to a safe distance. The GUI can be considered a user-friendly interface, and it covers the tools needed to observe and collect data of fire.

The outcome of the study underscores a clear advancement over prior single-stage or two-stage models commonly applied in real-time fire detection tasks. While earlier systems such as those in [19–21] successfully applied YOLO or CNN-based detectors to identify flame regions in RGB streams, they often lacked temporal awareness and spectral adaptability, resulting in performance degradation under low illumination, smoke occlusion, or reflective noise. In contrast, the proposed hybrid-cascade architecture incorporates a multi-modal and multi-stage design that not only leverages pixel-level segmentation and spatio-temporal validation but also fuses RGB and NIR modalities to ensure spectral robustness across various lighting conditions. This strategic integration aligns with the suggestions made by Buriboev et al. [22] and Jayasingh et al. [21], who emphasized the importance of temporal and shape-based cues to reduce false positives. Furthermore, unlike the robotic implementations by Banerjee et al. [25] or Ibitoye et al. [27], which primarily focused on basic flame localization and suppression using YOLO variants or traditional sensors, our system enhances both detection precision and safety compliance by integrating real-time geometry-based safe approach radius estimation into the motion planning of the robot. In this regard, the current study not only bridges the gap between high-precision fire detection and autonomous robotic response but also offers a deployable edge AI (Artificial Intelligence) solution that meets the real-time demands of safety-critical applications under both indoor and outdoor scenarios.

4. Conclusions

This study presents the conceptualization and development of a mobile robotic system for early detection. A hybrid-cascade machine learning model was designed and developed for use in the early fire detection system. A series of state-of-the-art (SotA) machine learning models were evaluated in isolation, using data from both night vision and RGB cameras. The models were then employed to build the hybrid-cascade structure

to detect fires, considering environmental factors such as day and night. This architecture comprises four models, each contributing uniquely: YOLOv10 enables fast and coarse localization; Mask R-CNN delivers precise, pixel-level segmentation; ConvLSTM-FireNet captures temporal dynamics such as flame flicker; and DenseNet enhances classification robustness under diverse lighting. Together, these models enhance system accuracy through complementary strengths rather than relying on a single high-performing model. The hybrid cascade achieved the highest offline accuracy among all configurations, with an F1 score of 96.7% and an IoU of 82.4% on the annotated fire/non-fire image set. Deployed on the mobile robot, the system maintained real-time performance (approximately 18 fps) and consistently ensured a safe distance of 500 mm from fire sources under both daylight and night conditions. In conclusion, the hybrid-cascade architecture provides a practical, real-time solution for fire detection on low-power edge devices and a strong foundation for future research in integrated detection and suppression systems. Although augmentation techniques like MixUp, CutMix, and photometric jitter were employed to enhance generalization, future work will conduct ablation studies to assess their individual impact on model robustness. The remaining limitations include performance sensitivity in extreme conditions (e.g., heavy smoke or rain), false alarm risks, and computational constraints on embedded systems. The large parameter size of Mask R-CNN causes inference delays on the Raspberry Pi 4B. Future work will investigate lighter segmentation models (e.g., YOLACT, ResNet-based variants) and apply distillation or pruning to enhance real-time performance without compromising accuracy. Additionally, upcoming research will focus on enhancing robustness in challenging visual environments—such as dense smoke occlusion—integrating additional hazard sensors (e.g., gas, thermal), and adapting the system for broader safety-critical applications.

Funding: We gratefully acknowledge the financial support of this work by Aydin Adnan Menderes University, Scientific Research Projects; MF-24013.

Data Availability Statement: The original contributions presented in this study are included in this article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Mishra, K.B.; Wehrstedt, K.D.; Krebs, H. Lessons Learned from Recent Fuel Storage Fires. *Fuel Process. Technol.* **2013**, *107*, 166–172. [CrossRef]
2. Haukur, I.; Heimo, T.; Anders, L. Industrial fires An Overview. In *Brandforsk Project 2010*; SP Report 2010, Volume 17; Sp Technical Research Institute of Sweden: Boras, Sweden, 2010.
3. Ronchi, E.; Gwynne, S.; Purser, D.; Colonna, P. Representation of the impact of smoke on agent walking speeds in evacuation models. *Fire Technol.* **2013**, *49*, 411–431. [CrossRef]
4. Wright, M.; Cook, G.; Webber, G. The Effects of Smoke on People's Walking Speeds Using Overhead Lighting and Way guidance Provision. In Proceedings of the 2nd International Symposium on Human Behaviour in Fire, London, UK, 26–28 March 2001; MIT: Boston, MA, USA, 2001; pp. 275–284.
5. Bertram, C.; Evans, M.H.; Javaid, M.; Stafford, T.; Prescott, T. Sensory Augmentation with Distal Touch: The Tactile Helmet Project. Biomimetic and Biohybrid Systems. In Proceedings of the Second International Conference, Living Machines, London, UK, 29 July–2 August 2013; pp. 24–35.
6. Geetha, S.; Abhishek, C.S.; Akshayanat, C.S. Machine vision-based fire detection techniques: A survey. *Fire Technol.* **2021**, *57*, 591–623. [CrossRef]
7. Li, Y.; Shang, J.; Yan, M.; Ding, B.; Zhong, J. Real-time early indoor fire detection and localization on embedded platforms with fully convolutional one-stage object detection. *Sustainability* **2023**, *15*, 1794. [CrossRef]
8. Lee, C.H.; Lee, W.H.; Kim, S.M. Development of iot-based real-time fire detection system using raspberry pi and fisheye camera. *Appl. Sci.* **2023**, *13*, 8568. [CrossRef]
9. Dampage, U.; Bandaranayake, L.; Wanasinghe, R.; Kottahachchi, K.; Jayasanka, B. Forest fire detection system using wireless sensor networks and machine learning. *Sci. Rep.* **2022**, *12*, 46. [CrossRef]

10. Sulthana, S.F.; Wise, C.T.A.; Ravikumar, C.V.; Anbazhagan, R.; Idayachandran, G.; Pau, G. Review study on recent developments in fire sensing methods. *IEEE Access* **2023**, *11*, 90269–90282. [CrossRef]
11. Fonollosa, J.; Solórzano, A.; Marco, S. Chemical sensor systems and associated algorithms for fire detection: A review. *Sensors* **2018**, *18*, 553. [CrossRef]
12. Alqourabah, H.; Muneer, A.; Fati, S.M. A smart fire detection system using IoT technology with automatic water sprinkler. *Int. J. Electr. Comput. Eng.* **2021**, *11*, 2088–8708. [CrossRef]
13. de Venancio, P.V.A.; Lisboa, A.C.; Barbosa, A.V. An automatic fire detection system based on deep convolutional neural networks for low-power, resource-constrained devices. *Neural Comput. Appl.* **2022**, *34*, 15349–15368. [CrossRef]
14. Celik, T.; Demirel, H. Fire detection in video sequences using a generic color model. *Fire Saf. J.* **2009**, *44*, 147–158. [CrossRef]
15. Chen, J.; He, Y.; Wang, J. Multi-feature fusion based fast video flame detection. *Build. Environ.* **2010**, *45*, 1113–1122. [CrossRef]
16. Qi, X.; Ebert, J. A computer vision based method for fire detection in color videos. *Int. J. Imaging* **2009**, *2*, 22–34.
17. Ko, B.; Cheong, K.H.; Nam, J.Y. Early fire detection algorithm based on irregular patterns of flames and hierarchical Bayesian Networks. *Fire Saf. J.* **2010**, *45*, 262–270. [CrossRef]
18. Frizzi, S.; Kaabi, R.; Bouchouicha, M.; Ginoux, J.M.; Moreau, E.; Fnaiech, F. Convolutional neural network for video fire and smoke detection. In Proceedings of the IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 877–882.
19. Abdusalomov, A.; Baratov, N.; Kutlimuratov, A.; Whangbo, T.K. An improvement of the fire detection and classification method using YOLOv3 for surveillance systems. *Sensors* **2021**, *21*, 6519. [CrossRef]
20. Li, P.; Zhao, W. Image fire detection algorithms based on convolutional neural networks. *Case Stud. Therm. Eng.* **2020**, *19*, 100625. [CrossRef]
21. Jayasingh, S.K.; Swain, S.; Patra, K.J.; Gountia, D. An experimental approach to detect forest fire using machine learning mathematical models and IoT. *SN Comput. Sci.* **2024**, *5*, 148. [CrossRef]
22. Buriboev, A.S.; Rakhmanov, K.; Soqiyev, T.; Choi, A.J. Improving Fire Detection Accuracy through Enhanced Convolutional Neural Networks and Contour Techniques. *Sensors* **2024**, *24*, 5184. [CrossRef]
23. Shaw, A. Autonomous Multi-Robot Exploration Strategies for 3D Environments with Fire Detection Capabilitie. *arXiv* **2024**, arXiv:2411.15953.
24. Sucuoglu, H.S.; Bogrekci, I.; Demircioglu, P. Development of mobile robot with sensor fusion fire detection unit. *IFAC-PapersOnLine* **2018**, *51*, 430–435. [CrossRef]
25. Banerjee, S.; Das, R.; Rathinam, R.; Dhanalakshmi, R. Real-Time Fire Detection in Unmanned Ground Vehicles Integrating YoloV5 and AWS IoT. In Proceedings of the 2023 International Conference on System, Computation, Automation and Networking, Puducherry, India, 17–18 November 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1–6.
26. Kong, L.; Li, J.; Guo, S.; Zhou, X.; Wu, D. Computer vision based early fire-detection and firefighting mobile robots oriented for onsite construction. *J. Civ. Eng. Manag.* **2024**, *30*, 720–737. [CrossRef]
27. Ibitoye, O.T.; Ojo, A.O.; Bisirodipe, I.O.; Ogunlade, M.A.; Ogbodo, N.I.; Adetunji, O.J. A Deep Learning-Based Autonomous Fire Detection and Suppression Robot. In Proceedings of the IEEE 5th International Conference on Electro-Computing Technologies for Humanity (NIGERCON), Ado-Ekiti, Nigeria, 26–28 November 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–4.
28. Nguyen, A.P.; Nguyen, N.X. Control Autonomous Mobile Robot for Firefighting Task. In Proceedings of the International Conference on Control, Robotics and Informatics (ICCRI), Danang, Vietnam, 26–28 July 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 37–41.
29. Ullo, S.L.; Mohan, A.; Sebastianelli, A.; Ahamed, S.E.; Kumar, B.; Dwivedi, R.; Sinha, G.R. A new mask R-CNN-based method for improved landslide detection. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 3799–3810. [CrossRef]
30. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J. Yolov10: Real-time end-to-end object detection. *Proc. Adv. Neural Inf. Process. Syst.* **2024**, *37*, 107984–108011.
31. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
32. He, Y.; Sahma, A.; He, X.; Wu, R.; Zhang, R. FireNet: A Lightweight and Efficient Multi-Scenario Fire Object Detector. *Remote Sens.* **2024**, *16*, 4112. [CrossRef]
33. Verlekar, T.T.; Bernardino, A. Video based fire detection using xception and conv-lstm. In Proceedings of the International symposium on visual computing, Venice, Italy, 22–29 October 2020; Springer International Publishing: Cham, Switzerland, 2020; pp. 277–285.
34. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.