Jurnal Kejuruteraan 37(6) 2025: 3063-3074
https://doi.org/10.17576/jkukm-2025-37(6)-40

Autonomous Blazebot: A Real-Time Fire Detection and SMS Alert System Using AI and GSM Technology

Eugene Tan Wei Ren[a], Faarih Farhan Mohd Nasir[a]*, Khairul Azlan A Rahman[a] & Ahamad Zaki Mohamed Noor[b]

[a] *School of Engineering, University of Wollongong (UOW) Malaysia,*

Utropolis Glenmarie, Jalan Kontraktor U1/14, Seksyen U1, 40150 Shah Alam, Selangor, Malaysia.

[b] *Fakulti Kecerdasan Buatan dan Keselamatan Siber, Universiti Teknikal Malaysia Melaka,*

76100 Hang Tuah Jaya, Melaka, Malaysia.

**Corresponding author: farhan.m@uow.edu.my*

Received 4 December 2024, Received in revised form 19 May 2025
Accepted 19 June 2025, Available online 30 September 2025

ABSTRACT

Indoor fire incidents pose a significant threat to both life and property, particularly in areas that are not regularly monitored or are isolated. Traditional fire detection systems, which typically rely on smoke, temperature, or gas sensors, tend to be passive, prone to false alarms, and incapable of providing intelligent, real-time alerts to users in remote locations. This paper addresses these challenges by introducing the development of an autonomous Blazebot for fire detection and alert systems using GSM technology. The proposed system integrates a lightweight YOLOv8n (You Only Look Once version 8 nano) deep learning model, deployed on a Raspberry Pi 4, to continuously recognize flames visually through a USB camera. Once a flame is confirmed, the system sends an alert signal to an Arduino Mega 2560, which then activates a SIM900 GSM module to send SMS notifications to designated recipients. The system successfully identified flame sources with a minimum size of 2500 pixels at distances up to 350 cm, achieving optimal accuracy between 200 and 250 cm. The average delay for SMS transmission was recorded at 10.01 seconds after detection. These findings demonstrate the viability of a cost-effective, real-time, vision-based fire detection and communication system suitable for settings without internet access or constant human oversight.

Keywords: Fire-fighting robot; SMS alert; YOLOv8n; real-time vision; hazard response; fire detection

INTRODUCTION

In recent years, the increasing frequency and severity of indoor fire hazards have underscored the critical need for more intelligent and responsive fire safety systems. Many domestic and industrial fire incidents result in delayed responses due to inadequate early-warning mechanisms and the limitations of conventional alarm-based systems (Kodur et al. 2020). Traditional fire alert solutions, such as smoke alarms and heat detectors, are passive and rely heavily on human presence for interpretation and action. This delay in response can lead to catastrophic outcomes, especially in remote or unattended environments (Shrivastava et al. 2024; Shaharuddin et al. 2023). The rising concern over such incidents has prompted growing interest in the development of autonomous firefighting robots that can proactively detect, react to, and suppress fires without direct human intervention (Wang et al. 2024; Singhal et al. 2025).

In Malaysia, indoor fires have become a major public safety concern, especially in densely populated urban areas. From 2015 to 2020, more than 40,000 fire incidents were reported annually, with a large percentage occurring indoors, affecting homes, factories, and commercial spaces. The increase in these incidents is largely due to faulty electrical wiring, improper use of appliances, and insufficient fire prevention measures. In 2022, the Fire and Rescue Department of Malaysia documented 28,580 fire incidents, resulting in 124 deaths and billions of ringgits in property damage. While traditional firefighting methods remain effective, they often put firefighters in life-threatening situations as they confront extreme heat, toxic smoke, and unstable structures. (Fire and Rescue Department of Malaysia, 2023).

Autonomous firefighting robots offer numerous advantages over conventional fire detection methods. These robots are designed to operate in high-risk environments where human access is limited or dangerous (Abimbola, 2023; Kong et al. 2024; Samkari et al. 2020; Kuznetsov et al. 2022). Blazebot, the autonomous platform proposed in this study, embodies this next generation of firefighting solutions. It integrates real-time fire detection with immediate alert and suppression capabilities using embedded Artificial Intelligence (AI) and communication modules. Through the combination of a YOLOv8n object detection model and GSM-based SMS transmission, Blazebot ensures that fires are not only detected with high accuracy but also reported instantly to relevant responders, thereby reducing the overall response time.

Despite advances in robotic systems, several gaps remain in current implementations. Many existing firefighting robots are not equipped with intelligent communication systems and lack the ability to notify users autonomously. The limitations of current designs lie in their reliance on local indicators such as buzzers or lights which may not be effective in all emergency scenarios (Aliff et. al. 2019; Bahamid et al. 2020). Furthermore, conventional fire alarms are typically static, do not track flame sources dynamically, and offer no remote notification. The absence of seamless integration between AI-based fire detection and GSM communication in mobile robotic platforms presents a significant constraint in fire safety technology (Maraveas et al. 2021). This research addresses these challenges by developing a system that leverages AI for fire recognition and instantly triggers alert messages via a SIM900 GSM module bridging the gap between real-time fire response and user awareness.

The primary objective of this project is to develop a self-contained robotic platform capable of autonomously detecting and responding to fire incidents. Blazebot is designed to perform three main functions which are continuously monitor its environment using a camera feed processed by the YOLOv8n detection algorithm, confirm and align itself toward the source of fire using servo-based turret movement and deploy a water spray system while simultaneously sending SMS alerts to predefined emergency contacts. To meet these objectives, the system incorporates robust sensing, precise motor control, and a GSM communication interface that transmits alerts immediately upon fire confirmation. This functionality ensures that fire suppression can begin even before human operators are aware of the danger.

The key technical contribution of this work lies in the real-time integration of computer vision and mobile alert systems on a single robotic platform. Unlike traditional setups that separate detection from communication, Blazebot performs both tasks autonomously. Once a fire is detected and validated based on confidence thresholds and bounding box size, the Raspberry Pi sends a signal to the Arduino Mega 2560, which activates the water pump and transmits an SMS alert through the SIM900 module. This modular communication pathway not only enhances fire suppression efficiency but also supports timely human intervention by minimizing communication delays.

The proposed system stands apart from existing firefighting robots by introducing a low-cost, real-time detection and alert mechanism tailored for indoor deployment.

Blazebot offers a clear advantage over conventional solutions by combining autonomous vision-based detection with immediate GSM-based notification enabling users to be alerted even when offsite or away from the robot's operational area. Blazebot's applications span across residential homes, warehouses, and indoor industrial facilities, where early detection and rapid response are crucial for damage prevention and occupant safety. This study demonstrates the potential of such systems to revolutionize fire safety practices by improving detection speed, automating suppression, and enhancing communication during emergencies.

LITERATURE REVIEW

AUTONOMOUS FIRE DETECTION SYSTEMS

Autonomous fire detection systems have transformed traditional fire monitoring approaches by incorporating machine learning and real-time sensor analytics (Shrivastava et al. 2024). Conventional methods that rely solely on smoke or heat detection are often insufficient in dynamic environments, particularly in mobile robotics (Gaur et al. 2019). These traditional systems also lack the ability to distinguish between fire and similar thermal or visual stimuli, often leading to false alarms. Robotics-based fire detection addresses these shortcomings by combining multiple sensing modalities with intelligent processing capabilities (Aliff et al. 2019; Jalani et al. 2019).

Recent developments have introduced vision-based detection powered by deep learning, which has proven to be particularly effective. Algorithms such as YOLO (You Only Look Once) are capable of processing camera inputs frame-by-frame, detecting objects such as flames based on colour, motion, and shape features (Jain et al. 2022; Carta et al. 2023). YOLOv8n, a lightweight variant designed for low-latency embedded applications, allows real-time fire detection even on limited platforms like the Raspberry Pi. In the context of Blazebot, this model processes live video feed from a USB webcam and identifies flames using bounding boxes and confidence scores, which are then used to guide suppression actions. This AI-based approach ensures fast, precise detection with a significantly reduced false positive rate (FPR) compared to traditional sensor only systems (Jain et al. 2022).

FIRE SUPPRESSION MECHANISMS AND NAVIGATION IN FIREFIGHTING ROBOTS

At the beginning, firefighting robots such as Arduino-based mobile platforms primarily relied on primitive flame sensors or infrared modules to detect fire presence. However, their strengths were in their fundamental setup relying on basic flame sensors, for instance, while they suffered from reliability and detection range, especially in the presence of changing light conditions or reflective surfaces. The application of innovation in fire-fighting robotics has produced systems like the Thermite RS1 and Colossus, which can be operated remotely, are heat resistant, and include thermal imaging cameras. However, most of these solutions are costly, physically big, manually controlled and do not have AI driven fire recognition or self-sufficient suppression mechanisms (Wang at el. 2024; Tanyıldızı, 2023).

When it comes to mobility and navigation, conventional systems usually prefer primitive line-following or scripted movement controls, which greatly reduced their capacity to adapt to different conditions (Almasri et al. 2015). Blazebot uses ultrasonic sensors that offer real-time obstacle detection to allow the robot to navigate complex and obstructed areas. Consequently, Blazebot can be used in indoor environments, such as office buildings, dormitories, or warehouse aisles, because of possible obstructions between it and the fire source. Blazebot's motion is based on sensor inputs that allow the robot to continue forward movement and constantly change direction with respect to turret data, creating a constant loop of surveillance and reaction.

COMMUNICATION AND ALERT SYSTEMS

Timely alerting is a core component of any fire safety solution. While most stationary systems trigger alarms through sound or visual signals, these are ineffective in unoccupied zones. Wireless communication technologies, particularly Global System for Mobile Communication (GSM) based messaging, are a practical solution for mobile firefighting platforms (Obeidat et al. 2021; Gragasin et al. 2024; Kanagamalliga & Rajalingam, 2025). Blazebot incorporates a SIM900A GSM module that enables autonomous alerting via SMS, ensuring that emergency messages reach relevant personnel even when the robot operates in isolation.

In addition, the integration of a GSM module for sending fire alert notifications is a major enhancement in functional responsiveness (Kanagamalliga & Rajalingam, 2025; Anju et al. 2024). Unlike conventional systems limited to visual or audible messages, the SMS notifications of Blazebot make it possible to notify users remotely. This ability is especially useful for controlling fires in places not under supervision or access outside normal business hours. This new communication aspect connects robotics independence with human oversight, which guarantees timely surveillance and effective response during urgent situations.

METHODOLOGY

The methodology adopted in this research is divided into four structured stages: prototype development, system architecture implementation, alert system design, and experimental setup for evaluating both fire detection and SMS notification delay. Each component of the methodology is derived from the actual development phases of Blazebot, which integrates hardware design, embedded software, real-time AI-based vision processing, and GSM-based notification logic.

PROTOTYPE DEVELOPMENT

This stage was crucial for planning the robot's layout and optimizing the placement of electronic components within a confined physical frame. Figure 1 illustrates the preliminary 3D CAD model of Blazebot, created using SolidWorks, a widely used mechanical design software. The chassis is modelled using standard 30x30 mm aluminium extrusions, which offer modularity, strength, and lightweight characteristics suitable for mobile robotics. The 3D model clearly marks each section with annotated components such as the Arduino Mega microcontroller, L298N motor drivers, Raspberry Pi unit, relay module, SIM900A GSM module, and the MG995 servo brackets. This digital design process enabled the designer to simulate mounting feasibility, wiring clearances, and load distribution before proceeding to hardware assembly.

The use of SolidWorks also allowed for interference checking between parts, ensuring that high-current motor wires were separated from low-voltage control lines, which is vital for avoiding electromagnetic interference (EMI). The turret mounting location was designated at the front section, aligning the camera and water nozzle along a shared pan-tilt axis. The CAD model served as a precise reference throughout the physical build process, ensuring all parts were placed according to design tolerances and that structural symmetry was preserved.

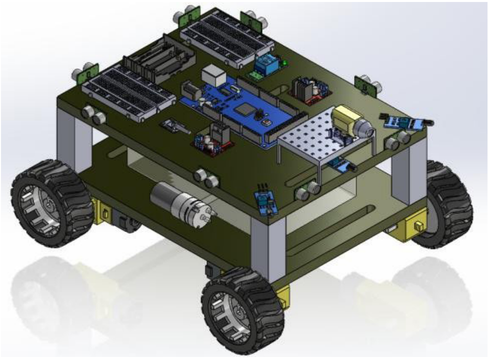

FIGURE 1. SolidWorks Model of Blazebot

Figure 2 showcases the initial skeleton build of Blazebot, constructed directly based on the CAD model using extruded aluminium profiles. This phase marked the transition from digital modelling to physical prototyping. The skeleton houses the electronic components and sets up the foundational mechanical structure for testing motor movement, obstacle detection, and turret actuation. At this point, the wiring is still visible and externally routed for easy troubleshooting and adjustments. The main control boards (Arduino Mega and Raspberry Pi) are mounted on a layered deck system, allowing vertical separation of high-power modules and low-voltage control circuits.

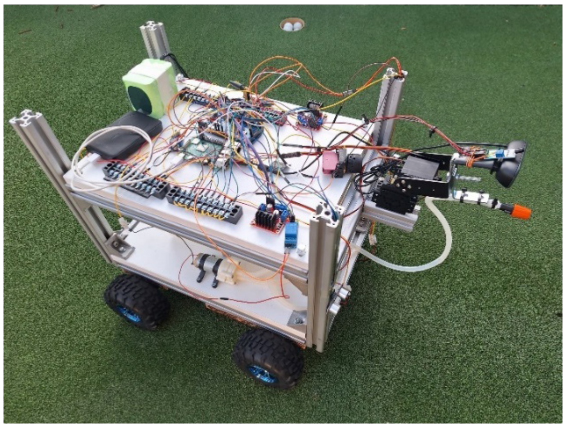

FIGURE 2. Skeleton Frame of Blazebot

Noticeably, the MG995 pan-tilt servos are installed at the front, forming the base of the turret system. These servos enable two-axis control horizontal (pan) and vertical (tilt) which will later guide the Logitech C270 camera and water nozzle to track and suppress fires. Additionally, two ultrasonic sensors are mounted on the front-left and front-right of the robot for autonomous obstacle detection and navigation. A relay-controlled diaphragm pump is also secured within the frame, connected to a flexible hose that will later attach to the nozzle. This phase primarily focused on verifying system logic, including Bluetooth communication, servo movement, ultrasonic response, and pump switching, before completing the aesthetic and protective enclosures.

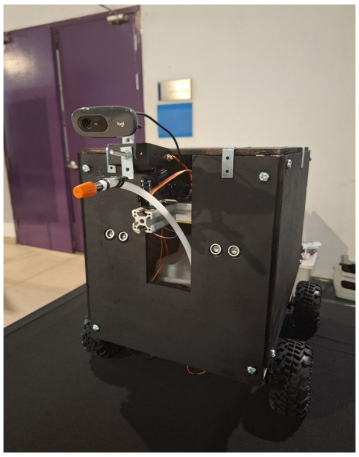

FIGURE 3. Blazebot Prototype

Figure 3 displays the fully assembled and operational version of Blazebot, where all internal modules are housed within a rigid outer casing for field deployment. The robot is now equipped with the Logitech C270 webcam, mounted firmly on the turret, which serves as the visual sensor for flame detection using the YOLOv8n AI model running on the Raspberry Pi. The water nozzle is precisely aligned with the camera's field of view to ensure that the turret can spray water accurately at the detected flame centre. The servo-controlled turret allows real-time orientation adjustments as the fire shifts position, guided by AI detection feedback.

In this stage, the electronic wiring has been secured, and the enclosure protects internal modules from mechanical damage, water exposure, and environmental interference. The 3030-aluminum frame remains as the backbone of the structure, providing rigidity and impact resistance. At the rear or lower panel (not visible in the image), the SIM900A GSM module is connected to the Arduino Mega 2560, allowing the system to send SMS alerts when fire is detected or extinguished.

This version includes optimized wheel placement for better manoeuvrability and stability, with all critical systems including obstacle avoidance, turret alignment, pump activation, and SMS communication fully integrated and tested. The completed robot reflects a comprehensive integration of mechanical engineering, embedded systems, and artificial intelligence. It demonstrates the successful translation of digital design into a physical autonomous firefighting robot capable of detecting fire visually, avoiding obstacles, notifying responders via GSM, and extinguishing flames using an onboard water system.

SYSTEM ARCHITECTURE

The Arduino Mega 2560 is the primary microcontroller responsible for executing all low-level hardware operations in the Blazebot system. Its pins are configured to control motion, servo-based turret positioning, water spraying, and sensor data processing. Two MG995 servo motors are mounted on pins 44 and 45, designated for pan and tilt control, respectively. These servos orient the camera and nozzle horizontally and vertically based on real-time fire location inputs received from the Raspberry Pi. The water spraying system is controlled through a 12V diaphragm pump connected to a relay module, which is triggered via pin 26. This relay is active LOW, meaning it turns on when the digital pin is pulled LOW and shuts off when set HIGH, preventing accidental activation as shown in Figure 4.

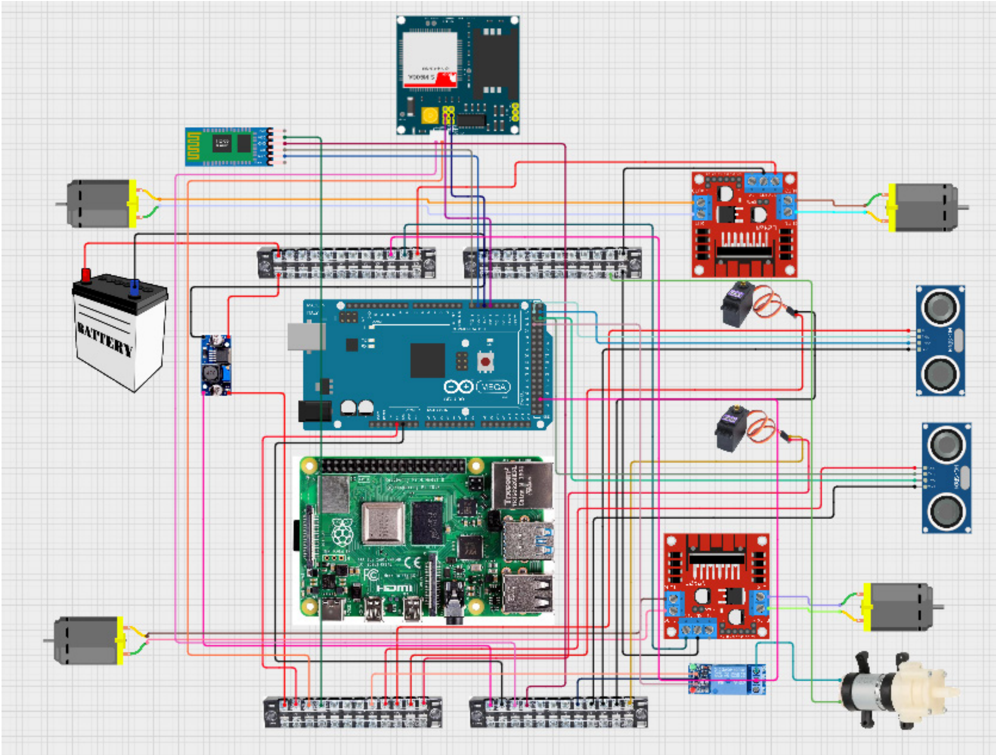

FIGURE 4. System Circuitry

For environmental navigation and obstacle avoidance, two ultrasonic sensors (HC-SR04) are mounted at the front-left and front-right of the robot. The left sensor uses digital pins 22 (Trigger) and 23 (Echo), while the right sensor uses pins 24 (Trigger) and 25 (Echo). These sensors allow the Arduino to measure the distance of obstacles in real time. If either detects an object below 58 cm, the robot halts, reverses for 2 seconds, and then turns away using differential motor control. The motor control system consists of two L298N motor drivers one for the left side and one for the right side. PWM speed control is achieved using pins 3 (ENA) and 11 (ENB). Directional movement is handled by four H-Bridge control pins per motor bank: IN1 through IN4 (pins 4–7) for the left motors and pins 8–12 for the right motors. This setup enables Blazebot to move forward, reverse, or turn left/right with precision based on sensor input or command signals.

Serial communication between the Arduino and Raspberry Pi is handled through Serial1, where the Arduino receives UART signals from the Pi through pin 19 (RX1) and optionally transmits feedback via pin 18 (TX1). This interface allows the Arduino to respond instantly to commands like "PAN:x,TILT:y" for turret alignment or "SMS_FIRE" to initiate the suppression sequence. All real-time actuation motor control, servo adjustments, and pump triggering is centralized in the Arduino, which continuously checks for updated commands or environmental data within the main loop. The connections for Arduino Mega shown in Table 1.

The Raspberry Pi 4 operates as the AI computation and vision-processing unit in Blazebot, handling tasks related to flame detection and communication. It interfaces with the Arduino Mega via UART (Universal Asynchronous Receiver/Transmitter) serial communication, sending signals over GPIO14 (TXD, physical pin 8) to deliver instructions such as pan-tilt coordinates or fire alerts. Optionally, GPIO15 (RXD, physical pin 10) may be used to receive status feedback from the Arduino, though this is not mandatory for basic operation. This UART link forms the backbone of coordination between vision-based detection and physical actuation.

TABLE 1. Arduino Mega 2560 Connections

Component	Arduino Mega Pin(s)	Purpose
Pan Servo	44	Horizontal turret rotation
Tilt Servo	45	Vertical turret movement
Water Pump Relay	26	Activates water spray (active LOW)
Left Ultrasonic Trigger	22	Triggers ultrasonic pulse (left sensor)
Left Ultrasonic Echo	23	Reads return signal (left sensor)
Right Ultrasonic Trigger	24	Triggers ultrasonic pulse (right sensor)
Right Ultrasonic Echo	25	Reads return signal (right sensor)
Motor Driver ENA	3	PWM speed control for left motors
Motor Driver ENB	11	PWM speed control for right motors
Left Motor IN1	4	H-Bridge control for left motor
Left Motor IN2	5	H-Bridge control for left motor
Left Motor IN3	6	H-Bridge control for left motor
Left Motor IN4	7	H-Bridge control for left motor
Right Motor IN1	8	H-Bridge control for right motor
Right Motor IN2	9	H-Bridge control for right motor
Right Motor IN3	10	H-Bridge control for right motor
Right Motor IN4	12	H-Bridge control for right motor
Python Serial (from Pi)	Serial1 (TX1=18, RX1=19)	Receives PAN/TILT and SMS_FIRE commands
SIM900A (GSM)	Serial2 (TX2=16, RX2=17)	Sends AT commands and SMS alerts
Bluetooth HC-05	Serial3 (TX3=14, RX3=15)	Manual/auto mode switch and movement

To ensure safe system shutdown and prevent SD card corruption, a normally closed (NC) pushbutton is connected to GPIO4 (physical pin 7). When this button is released, the state change is detected by a Python script on the Raspberry Pi, which initiates a graceful shutdown process. The Pi is powered via a regulated 5V DC input through physical pin 2, supplied by a buck converter that can deliver at least 3A of current to handle camera streaming and AI inference loads. Pin 6 is connected to ground (GND), ensuring a stable electrical reference shared between the Pi and Arduino for signal consistency.

For vision input, the Raspberry Pi is equipped with a Logitech C270 USB camera, which streams live video directly to the Pi's USB port. This video is processed in real time using a lightweight YOLOv8n model, trained specifically for fire detection. When the model identifies a flame with confidence above a predefined threshold and minimum bounding box area, the Pi sends two things: (1) a pan and tilt alignment command to centre the fire in the turret's field of view, and (2) an "SMS_FIRE" message over UART to instruct the Arduino to activate the water pump and send an SMS. The Raspberry Pi has no direct access to the relay or GSM module; instead, it delegates these responsibilities to the Arduino for execution. This division of labour allows the Raspberry Pi to focus solely on high-level decision-making while ensuring that time-sensitive controls (motors, sensors, pump) remain in the microcontroller's domain. The connection for Raspberry Pi 4 can refer to Table 2.

TABLE 2. Raspberry Pi 4 Connection

Function	Component	GPIO Pin	Physical Pin	Direction	Remarks
UART TX	To Arduino Mega (RX1)	GPIO14 (TXD)	8	OUTPUT	Sends serial instructions (e.g., PAN, TILT, STOP, RESUME)
UART RX	From Arduino Mega (TX1)	GPIO15 (RXD)	10	INPUT	(Optional) Receives feedback or status signals if implemented
Shutdown Button (NC pushbutton)	Safe Power-Off Input	GPIO4	7	INPUT	Detects release of shutdown button; triggers system shutdown
5V Power Input	Power Supply via Buck Module	-	2	INPUT	5V DC input (min 3A) from regulated buck converter
GND	Common Ground	GND	6	-	Shared ground for GPIO logic and external button/LED circuits
USB Camera	Fire Detection Visual Input	-	USB port	INPUT	Provides real-time video stream for YOLOv8n flame detection

ALERT SYSTEM ARCHITECTURE

The alert system architecture integrates computer vision, serial communication, and GSM messaging into a cohesive workflow as shown in Figure 5. The process begins when the Raspberry Pi captures a video stream from the USB camera and processes each frame using the YOLOv8n object detection model. The model identifies potential flame regions using bounding box parameters, and if the flame area exceeds a specified threshold and confidence score, a "SMS_FIRE" string is sent to the Arduino via UART. Once the Arduino receives this command, it sets the "fireCameraDetected" flag to true, initiates suppression via relay activation, and sends an SMS using AT commands through the SIM900A module.

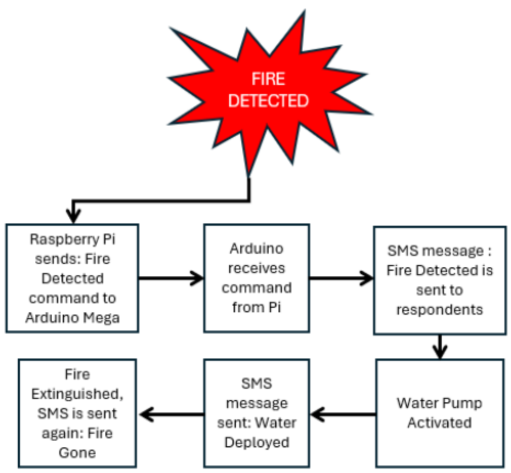

FIGURE 5. SMS Notification Alert System for Blazebot

The Arduino composes the SMS using a predefined message string, typically "FIRE DETECTED AT UOW MALAYSIA, SPRAYING WATER", and transmits this through a sequence of AT instructions: enabling text mode (AT+CMGF=1), specifying the phone number (AT+CMGS), inputting the message content, and finally sending the termination character (ASCII 26). The SIM900A responds with acknowledgment, and the system returns to monitoring mode. If the fire signal is not updated within seven seconds (cameraDetectionTimeout), the Arduino deactivates the flag. If the pump remains active beyond 15 seconds (pumpHoldTime) without reconfirmed fire detection, the pump is turned off and a second SMS, "FIRE EXTINGUISHED", is sent. This layered communication protocol ensures fail-safe operation and timely updates.

EXPERIMENT SETUP

The evaluation for performance of Blazebot's fire detection and notification system, a controlled testing environment was established. Fire scenarios were simulated under varied lighting and background conditions to test detection consistency. The YOLOv8n model parameters were fine-tuned for the Logitech C270 webcam, using a confidence threshold of 0.6 and a bounding box pixel area exceeding 2500 pixels as validation criteria. These values were chosen based on preliminary trials that balanced detection sensitivity with false positive suppression.

Each successful fire detection instance was timestamped, and the time taken for the corresponding SMS to be received by a standard mobile phone was recorded. This SMS delay was measured over multiple iterations to determine average latency, with specific emphasis on network variability, GSM module response, and Arduino-GSM synchronization. These results were later used to validate the system's responsiveness and reliability under practical use for Blazebot testing as shown in Figure 6.

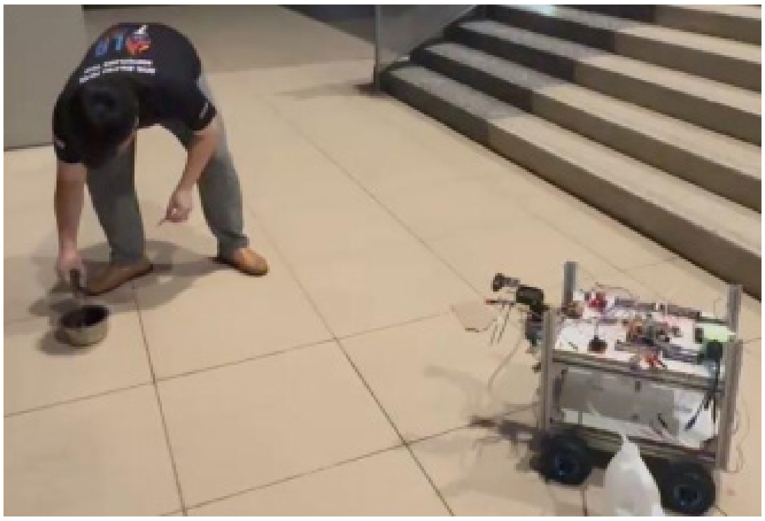

FIGURE 6. Blazebot Testing for Fire Detection

RESULTS AND DISCUSSION

FLAME DETECTION AND SUPPRESSION AT DIFFERENCE DISTANCES

Blazebot was tested for its flame detection and suppression capabilities at distances between 100 cm to 400 cm. Blazebot responded instantly, locking onto and extinguishing the fire with water in just 6 seconds at 100cm. When the flames were at 150 cm and 200 cm, Blazebot extinguished them in 9.0 and 13.4 seconds, respectively mostly due to increased movement required as shown in Table 3. Flames at such distances were quickly detected because they were placed both in the camera's field of view and YOLOv8n's processing area. Distance above 250 cm, the robot only started to detect the fire when it got closer to the flame. To reduce the errors of non-flame sources, the system only detected flames whose bounding boxes were greater than 300 px², serving as a filter against false positives such as flashlights or glare (Refer Table 4).

TABLE 3. Distance of Fire and Extinguishing Time

Distance(cm)	Time Taken Each and extinguis Fire(s)	Fire Extinguish	Fire Detected at Start
100	6.0	YES	YES
150	9.0	YES	YES
200	13.4	YES	YES
250	17.2	YES	NO
300	21.2	YES	NO
350	23.9	YES	NO
400	29.3	YES	NO

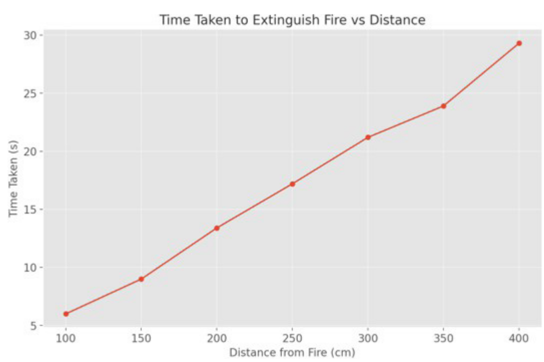

FIGURE 7. Time Taken Extinguishing vs Distance

Distance at 350 cm, the Blazebot system recorded the highest successful detection and extinguishing time of 23.9 seconds compared to the distance 400 cm, the system was unable to detect the fire, verifying its maximum effective range under the current setup (Refer Table 3 and Figure 7). The extension of the delay in the detection process beyond 250 cm can be explained by the fact that flame covers a smaller part of the camera field of view, thus, harder for bounding box algorithms to find the fire. The Blazebot had to move forward to make the fire's bounding box greater than the detection threshold (≥ 300 px²) (Refer Table 4), which led to a delay in the system's response. For instance, at the distance of 300 cm, the extinguish time had 21.2 seconds, mostly due to the delayed flame recognition.

The calculated time to reach the fire, ignoring detection delay, follows the relationship $t = d/v$ where $v = 17.5$ cm/s. The time that was estimated for 300 cm operation is approximately 17.14 seconds, which matches the measured response after turret realignment and AI validation lags have been taken into consideration. From these results, it seems that the most effective operating range for BlazeBot to detect and suppress fire under present conditions is approximately 350 cm.

VISION DETECTION FOR FIRE

The Blazebot system utilizes a YOLOv8n model running on a Raspberry Pi to perform flame detection based on image frames captured by the Logitech C270 USB webcam. Detection is triggered when the bounding box surrounding a flame exceeds 2500 pixels and the confidence score surpasses 0.6 as shown Figure 8.

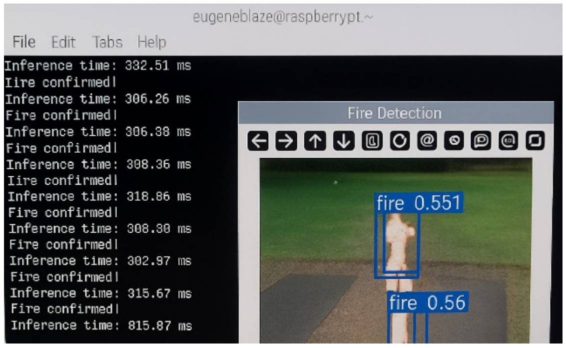

FIGURE 8. Fire AI Detection using YOLOv8n

The visual observation of the amount of flame influenced the degree to which the system could recognize the flame. Bounding boxes under 300 px² were equally rejected in trials, which is the detection limit. At 310 px², the detection became accurate and worked steadily at sizes over 400 px² (Refer Figure 9 and Table 4). This means that the current detection threshold successfully balances between precision and sensitivity in the model. By eliminating smaller flames, the system limits false alarms but weakens long-range detection. Some potential improvements to the next versions of the system may include changing thresholds for detection or adding higher resolution data sources to increase the detection efficiency without sacrificing accuracy.

TABLE 4. Fire Detection Area

Trial	Flame Area (px²)	Fire Detected
1	180	NO
2	220	NO
3	250	NO
4	280	NO
5	310	YES
6	330	YES
7	360	YES
8	400	YES
9	440	YES
10	480	YES

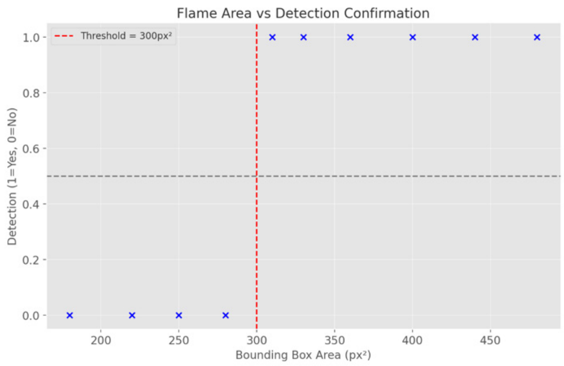

FIGURE 9. Graph of Effect of Bounding Box Area on Flame Detection Confirmation

The graph in Figure 9 shows the correlation between bounding box area of the detected flame and confirmation of a valid fire detection. The horizontal axis shows the bounding box area in pixel squared (px2) and the vertical axis shows the detection decision, where 1 means that the flame is confirmed and 0 means rejection. A small or insignificant detection caused by noise, reflection or small light disturbance is filtered out by the threshold of 300 px2, indicated by the red dashed vertical line. The data points on the left of the threshold (areas below 300 px2) are reliably rejected as false or unreliable detections, and points on the right side of the threshold are above and are accepted as valid flame detections. This clear separation reflects the effectiveness of the region-based thresholding method to improve detection accuracy by avoiding false positives and ensuring that only relevant flame events are detected.

These results indicate that Blazebot is highly effective for medium-range detection scenarios, especially in enclosed indoor environments such as offices, dormitories, or storage rooms. The filtering logic based on bounding box size helps reduce misclassification caused by light reflections or small dynamic elements.

COMMUNICATION SYSTEM PERFORMANCE

The communication component of Blazebot was tested by evaluating the responsiveness and reliability of the SMS-based alert system. The SIM900A GSM module, integrated through Serial2 on the Arduino (Refer Table 1), was able to transmit text messages within an average delay of 10.01 seconds from the moment of detection confirmation, with a range of responses from 9.72 seconds to 10.29 seconds.

This delay included the time required for camera detection smoothing (YOLOv8n) (~2.2 seconds), UART command transmission (< 0.3 seconds), and the GSM network execution (~7.5 seconds). In more favourable signal conditions, the delay reduced to as low as 9.72 seconds, while poor coverage extended it to nearly 10.29 seconds as shown in Table 4 and Figure 10.

Manual responsiveness was also analysed using the Dabble Bluetooth gamepad which connected on Serial3 (Refer Table 1) that allowed the users to switch either manual and automatic modes by using Start and Select buttons and override the movements of Blazebot which can allowed real-time control during field testing, remote control during emergencies or navigating out of complex environments.

Upon power-up, the system configures all hardware, sets servos to default positions, and disables motors and pump. The robot remains idle until the user selects either autonomous or manual mode via Bluetooth input. In manual mode, user can directly issue certain movement commands, such as directional adjustments and servo angle settings, while automatic mode operates under the assistance of artificial intelligence. Once a mode is chosen, it proceeds to patrol, detect, extinguish, or respond as programmed.

The latency between button presses and corresponding motor or servo response was negligible (within 100 ms), confirming that the control logic is suitable for rapid adjustments in manual mode.

TABLE 5. Communication System Performance

Test	Detection to SMS delay (s)	SMS sent	SMS received
1	9.89	YES	YES
2	9.80	YES	YES
3	10.06	YES	YES
4	9.82	YES	YES
5	9.72	YES	YES
6	10.29	YES	YES
7	9.95	YES	YES
8	10.01	YES	YES
9	9.72	YES	YES
10	9.89	YES	YES
11	10.02	YES	YES
12	9.77	YES	YES
13	10.08	YES	YES
14	9.88	YES	YES
15	9.94	YES	YES

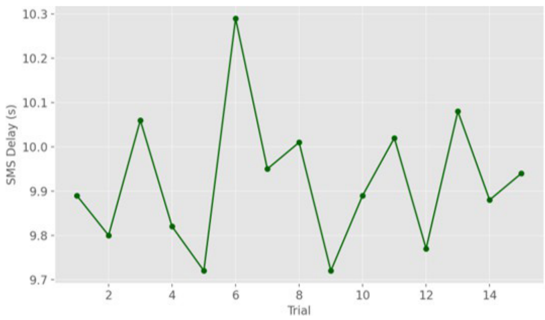

FIGURE 10. SMS Delay Time

The system's ability to send SMS messages during fire detection and upon fire suppression adds a significant layer of safety communication. The first SMS, labelled "FIRE DETECTED AT UOW MALAYSIA, SPRAYING WATER", is sent immediately after the detection of fire. If no fire is confirmed again after 15 seconds, a second SMS "FIRE EXTINGUISHED" is sent. These messages were successfully received in all test runs, showing no dropped packets or delivery failures as shown in Figure 11.

FIGURE 11. SMS Sent by Blazebot

Although this response is not immediate, it gives adequate time for early warning in residential or unattended industrial setups. The alert messages were successfully delivered, which confirms the stability of the UART-to-GSM pipeline for communication. The SIM900A module performed AT command operations for message delivery with consistency and without error.

CONCLUSION

This study successfully developed and tested Blazebot, an autonomous fire detection and suppression robot equipped with real-time AI-based visual detection and GSM-based emergency notification. The integration of the YOLOv8n flame detection model running on a Raspberry Pi allowed for high-accuracy fire recognition under standard lighting conditions. The system successfully identified flame sources with a minimum size of 2500 pixels at distances up to 350 cm, achieving optimal accuracy between 200 and 250 cm. The average delay for SMS transmission was recorded at 10.01 seconds after detection.

Manual operation via Bluetooth was also implemented, offering full override capability, which is useful during system testing or emergency intervention. The architecture supports state-driven transitions between patrol, detection, and suppression modes, ensuring effective and autonomous performance. The study demonstrated that low-cost, AI-enhanced mobile robots could be a viable solution for early-stage fire suppression in indoor environments. The system's real-world testing showed that all modules from AI detection for communication and actuation functioned harmoniously, validating the design for future scalability and deployment.

Despite its success, the current implementation has limitation. The SMS system is entirely dependent on GSM network availability and does not currently support fallback options like Wi-Fi alerts. Therefore, for future development, the alerting and communication systems of Blazebot could also be expanded by integrating with cloud-based platforms such as Firebase, Blynk, or AWS IoT which useful in factories or commercial buildings where several units may be deployed simultaneously.

ACKNOWLEDGEMENT

This research is partial financial support from UOW Malaysia. Their support has made this important research viable and effective.

DECLARATION OF COMPETING INTEREST

The authors declare no competing interest.

REFERENCES

Abimbola, M. B. 2023. Robots for fighting fires: A comparative analysis. *TMP Universal Journal of Research and Review Archives* 2(4): 133-138.

Aliff, M., Yusof, M., Samsiah, N., & Zainal, A. 2019. Development of fire fighting robot (QRob). *International Journal of Advanced Computer Science and Applications* 10(1): 142-147.

Almasri, M., Elleithy, K., & Alajlan, A. 2015. Sensor fusion based model for collision free mobile robot navigation. *Sensors* 16(1): 24.

Anju, M. I., Sharma, A., & Pranathi, P. 2024. Real-time fire surveillance and SMS alert system with ruggedized gsm module. In *2024 International Conference on IoT Based Control Networks and Intelligent Systems (ICICNIS):* 616-620. IEEE.

Bahamid, A., Ibrahim, A. M., Ibrahim, A., Zahurin, I. Z., & Wahid, A. N. 2020. Intelligent robot-assisted evacuation: A review. *Journal of Physics: Conference Series* 1706(1): 012159.

Carta, F., Zidda, C., Putzu, M., Loru, D., Anedda, M., & Giusto, D. 2023. Advancements in forest fire prevention: A comprehensive survey. *Sensors* 23(14): 6635.

Fire and Rescue Department of Malaysia. 2023. Fire and rescue statistics, Malaysia. https://data.gov.my/dashboard/fire-and-rescue [5November 2024].

Gaur, A., Singh, A., Kumar, A., Kulkarni, K. S., Lala, S., Kapoor, K., & Mukhopadhyay, S. C. 2019. Fire sensing technologies: A review. *IEEE Sensors Journal* 19(9): 3191-3202.

Gragasin, M. N. G., Cusi, K. G. D., Calle, A. M. J. M., Calabio, E. B., Enriquez, C. L. P., Dablo, I. O., & Limos-Galay, J. A. 2024. Arduino Uno fire alarm system with Android connectivity using GSM Module, Temperature, and Smoke Sensor. *International Journal of Research* 8(4): 9-17.

Jain, V. K., & Jain, C. 2022. Fire and smoke detection with deep learning: A review. *Journal on Digital Signal Processing* 10(2): 22.

Jalani, J., Misman, D., Sadun, A. S., & Hong, L. C. 2019. Automatic fire fighting robot with notification. *IOP Conference Series: Materials Science and Engineering* 637(1): 012002.

Kanagamalliga, S., & Rajalingam, S. 2025. Innovative fire and gas recognition system featuring remote monitoring and automated alerts. *Procedia Computer Science 252*: 7-14.

Kodur, V., Kumar, P., & Rafi, M. M. (2020). Fire hazard in buildings: Review, assessment and strategies for improving fire safety. *PSU Research Review* 4(1): 1-23.

Kong, L., Li, J., Guo, S., Zhou, X., & Wu, D. 2024. Computer vision based early fire-detection and firefighting mobile robots oriented for onsite construction. *Journal of Civil Engineering and Management* 30(8): 720-737.

Kuznetsov, G., Kopylov, N., Sushkina, E., & Zhdanova, A. 2022. Adaptation of fire-fighting systems to localization of fires in the premises. Energies 15(2): 522.

Maraveas, C., Loukatos, D., Bartzanas, T., & Arvanitis, K. G. 2021. Applications of artificial intelligence in fire safety of agricultural structures. *Applied Sciences* 11(16): 7716.

Obeidat, M. A., Mansour, A. M., Hamadneh, T., & Abdullah, J. 2021. Remotely controlled smart home system using gsm and iot. In *2021 International Conference on Information Technology (ICIT):* 748-753. IEEE.

Samkari, Y., Guedri, K., Oreijah, M., Munshi, S., & Azam, S. 2020. Designing and testing of a smart firefighting device system (LAHEEB). *International Journal of Robotics and Automation (IJRA)* 9(2): 143-152.

Shaharuddin, S., Abdul Maulud, K. N., Syed Abdul Rahman, S. A. F., Che Ani, A. I., & Pradhan, B. 2023. The role of IoT sensor in smart building context for indoor fire hazard scenario: A systematic review of interdisciplinary articles. *Internet of Things (Netherlands)* 22*(2023): 100803.*

Shrivastava, A., Gogoi, A., Shahi, S., & Chaitanya, S. 2024. IoT enabled real time fire monitoring and response in urban areas. *Information Sciences and Technological Innovations* 1(1): 19-27.

Singhal, A., Singh, S., Siddharth, S., & Rajalakshmi, T. 2025. Autonomous Fire Fighting Robotic Vehicle. In *2025 11th International Conference on Communication and Signal Processing (ICCSP)* (pp. 177-181). IEEE.

Tanyıldızı, A. K. 2023. Design, control and stabilization of a transformable wheeled fire fighting robot with a fire-extinguishing, ball-shooting turret. *Machines* 11(4): 492.

Wang, M., Chen, X., & Huang, X. 2024. Robotic firefighting: A review and future perspective. *Intelligent Building Fire Safety and Smart Firefighting*, 475-499.