© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717389

A Research on Fire GSM Service Robot

VAIBHAV KUMAR TIWARI[1], ANIKET MISHRA[2], RAHUL YADAV[3], SURAJ KUMAR[4], AAYUSHI PRIYADARSHINI[5]
[1,2,3,4,5] *Department of Electronics and Communication Engineering, Institute of Technology and Management, GIDA, Gorakhpur*

Abstract- As urban landscapes and industrial facilities grow increasingly intricate and hazardous, the risks faced by firefighting personnel have escalated to a point where human intervention alone is no longer sufficient — driving a pressing need for robotic systems capable of stepping in and operating independently in the most dangerous of conditions. This paper details the design and hands-on evaluation of a firefighting robot built around an Arduino Uno, bringing together infrared flame detection, GSM-based remote alerting, and live Wi-Fi video surveillance into a single, cohesive embedded system. To accurately pinpoint the direction of a fire, the system positions three infrared flame sensors in a 180-arc, giving it a wide and reliable field of detection. Alongside this, an ESP32-CAM module continuously streams live video to a mobile application at approximately 30 frames per second, giving the operator the option to step in and take manual control whenever the situation demands it. A SIM800L GSM module dispatches an SMS notification upon fire suppression. An electromagnetic relay controls a submersible pump for water discharge; at the same time, a dedicated power management subsystem—using an LM7805 linear regulator and DC-DC buck converters—ensures voltage stability across all modules during peak current draw. Across ten controlled trials, the system consistently demonstrated a mean flame detection time of 1.42 ± 0.18 seconds, a suppression time of 4.8 ± 0.6 seconds, and a GSM alert delay of 7.3 ± 1.2 seconds. One notable challenge encountered during testing was electromagnetic interference generated by the pump motor, which initially disrupted GSM communication. This was successfully resolved by isolating the GSM power rail and fitting decoupling capacitors to the motor leads. The system confirms a budget-conscious, dual-mode robotic fire suppression solution that operates either independently or under operator guidance, removing the risk of human exposure to life-threatening environments.

Keywords: Autonomous Firefighting, Arduino Uno, GSM Alerting, ESP32-CAM, Flame Triangulation, Industrial Safety Robotics, Live Surveillance.

I. INTRODUCTION

Firefighting operations expose personnel to thermal radiation, toxic combustion products, oxygen-depleted atmospheres, and structurally unstable environments. The National Fire Protection Association estimates thousands of firefighter injuries annually, a significant proportion of which occur during suppression activities rather than overhaul or rescue [1]. Robotic systems offer a pathway to reduce direct human exposure by performing detection and suppression tasks remotely; however, early designs were constrained to simple reactive behaviours with no communication feedback to the operator [4].

The foundation for this work was laid by Singh et al. [1], whose research established sensor-driven directional navigation as a proven and viable strategy for autonomous flame tracking. Molla et al. [4] subsequently demonstrated the practical feasibility of GSM-based alert transmission within firefighting applications, though a notable limitation of their system was the complete absence of any visual feedback mechanism. The field has since advanced considerably, with more recent studies introducing machine learning techniques for flame classification [3] and leveraging IoT connectivity to enable remote supervision [5][7]. However, a critical gap remains — to date, no low-cost prototype has successfully integrated autonomous navigation, live video monitoring, GSM alerting, and EMI-resilient power management into a single, cohesive system.

This work steps in to fill that gap by pulling together five key capabilities into one cohesive Arduino Uno-based platform — each one carefully chosen to make the system as effective, reliable, and practical as possible:

- Directional flame localisation powered by three infrared sensors fanned out across a 180-degree arc, giving the robot the ability to not just detect fire but pinpoint exactly where it is coming from.
- Live HD video streaming through an ESP32-CAM module at around 30 frames per second, so the operator always has a clear, real-time view of what the robot is seeing on the ground.
- Automated SMS notification sent through a SIM800L GSM module the moment suppression is confirmed, making sure the operator is never left wondering whether the job is done.
- Water-based fire suppression via a relay-controlled submersible pump, turning the robot's detection capability into immediate, hands-on action.
- EMI-resilient power distribution backed by dedicated voltage regulators and decoupling capacitors, keeping the entire system running smoothly even when electromagnetic interference threatens to cause disruption.

II. SYSTEM ARCHITECTURE

The robot is equipped with a dedicated array of sensors that help it perceive its surroundings, along with a dependable locomotion system that allows it to move efficiently — even in challenging conditions. Three IR flame sensors are strategically mounted in a 180° arc to track infrared radiation [1], with onboard potentiometers allowing for fine-tuned sensitivity to avoid false triggers from sunlight. To move, an L298N dual H-bridge motor driver powers four battery-operated motors, enabling differential steering. At its heart is an Arduino-based microcontroller, acting as the central processing unit that interprets these incoming signals [2]. The robot takes in real-world sensor readings alongside digital instructions from the remote receiver, using both to smoothly coordinate its movement and maintain safe operations.

The most advanced ECE feature of the build is its dual-link communication and surveillance interface, which provides a critical feedback loop for the operator. The onboard ESP32-CAM Wi-Fi module continuously streams live HD footage to a mobile app, putting the operator in the driver's seat with a first-person view that makes manual control of the robot feel natural and responsive [3]. This becomes especially useful in difficult environments where obstacles might get in the way of the robot's sensors, making it harder for the autonomous system to find its own path. Complementing this is a SIM800L GSM module that connects to cellular networks to send instant SMS alerts the moment a fire is neutralised, ensuring the user is informed even if they aren't on-site [4].

When a fire is detected, the system triggers a 5V electromagnetic relay that activates a submersible water pump, providing a powerful spray while keeping the delicate electronics isolated from the high-current draw of the motor [5]. The control core acts as the central coordinator, ensuring the robot's powerful mechanical parts and its more delicate, low-power sensors work together seamlessly — without ever getting in each other's way[6].

To keep every component running reliably under pressure, the robot is built around a dedicated power management system engineered to absorb high-demand surges without faltering. A 12.6V lithium-ion battery pack forms the backbone of the energy supply, chosen specifically for its high discharge rate — critical for powering the pump and motors at the same time without any drop in performance. Since the GSM module and sensors operate at significantly lower voltages, LM7805 regulators and buck converters are carefully integrated into the design to step the supply voltage down to the safe and precise levels each component requires [7]. This thoughtful approach to power distribution keeps brown-outs and system resets firmly at bay, ensuring the robot stays fully responsive and the communication link remains active and reliable from the first moment of the mission to the last.

III. HARDWARE USED

3.1 IR Sensors and L298N Motor Driver

At the heart of the fire detection system are flame sensor modules built around the YG1006 photodiode and a comparator circuit, carefully calibrated to respond to infrared wavelengths spanning 760 nm to

1100 nm — the precise spectral signature that hydrocarbon flames are known to emit. Each module houses an LM393 comparator capable of delivering two complementary outputs: an analogue signal that scales proportionally with incoming light flux intensity, and a digital threshold output that triggers at a predefined detection level. The design capitalises on these analogue outputs to drive proportional navigation, enabling the robot to make nuanced directional adjustments rather than simply reacting to crude on/off switching signals. Complementing this, the L298N motor driver — rated for up to 46 V and 2 A per channel — provides the muscle needed to run all four BO motors from the 12.6 V battery pack, while interfacing seamlessly with the Arduino's 5 V logic control signals.

Fig 1 IR FLAME SENSOR

3.2 Arduino Microcontroller

The Arduino Uno (ATmega328P) provides six analogue input channels (10-bit ADC), 14 digital I/O pins, and two hardware serial ports. In this implementation, pins A0–A2 read flame sensor voltages, D3–D7 control the L298N direction inputs, D8 triggers the relay, and D0/D1 communicate serially with the SIM800L. The ESP32-CAM operates on its own wireless stack and does not consume Arduino I/O resources [2].

Fig 3 AURDINO MICROCONTROLLER []

3.3 Wi-Fi CAM

The actual surveillance tasks are handled by the WIFI CAM, which provides a live "First-Person View" (FPV) [3]. By hosting a small web server, this camera streams high-definition video straight to a mobile app, giving you a pair of "eyes" on the ground to monitor the situation. Complementing this is the SIM800L GSM Module, which ensures the robot is communicative and reliable [4]. The GSM module operates like a small, screenless cell phone, dispatching an automated SMS alert to the operator as soon as the robot brings the fire under control.

Fig 4 WIFI CAMERA

3.4 5V Relay and Submersible Pump

The relay module uses an SRD-05VDC-SL-C relay (10 A, 250 VAC rating) driven by an NPN transistor on the module PCB, triggered by a 5 V Arduino signal. A flyback diode across the relay coil suppresses inductive kickback. The submersible pump, which operates within a voltage range of 3–6 V and delivers a flow rate of 240 L/h at 5 V, draws its power directly from the battery pack whenever the relay's normally-open contacts close and complete the circuit. In practice, 6 V was applied to achieve a higher flow rate during testing [5].

Fig 5 5V RELAY

3.5 Buck Converters and Regulators (like the LM7805)

Since different parts of the robot need different voltages—like 4V for the GSM and 5V for the sensors—the build includes Buck Converters and

© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717389

Regulators [7]. These components step the battery's high 12.6V voltage down to the exact levels needed, preventing "brown-outs" and ensuring the robot doesn't reset in the middle of a mission. This stable power distribution is critical for maintaining the communication link throughout the operation.

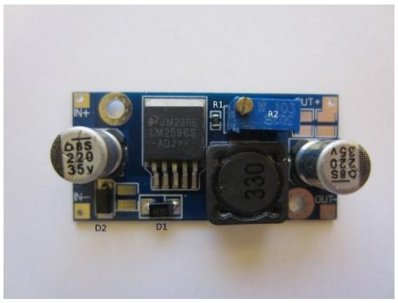

Fig 6 Buck Converters and Regulators (like the LM7805)

3.6 SIM800L GSM MODULE

The SIM800L is a quad-band GSM/GPRS module operating at 850/900/1800/1900 MHz. It communicates with the Arduino via a software or hardware UART at 9600 baud using Hayes AT commands. The module requires a 3.4–4.4 V supply rated for 2 A transient bursts during transmission, which is why a dedicated buck converter is used rather than drawing from the Arduino's 5 V rail. Network registration is confirmed during initialisation by polling the AT+CREG response before the robot enters its operational state [4].

IV. METHODOLOGY AND WORKING PRINCIPLE

The robot's behaviour is organised around a four-state finite state machine, cycling through four clearly defined stages — INIT, NAVIGATE, SUPPRESS, and NOTIFY — with each transition between states triggered by specific sensor readings and timed conditions.

INITIALISATION STATE

When the system powers up, the Arduino immediately gets to work — initialising all serial ports, setting every motor pin to a low stopped state, and opening a 10-second network registration window during which it repeatedly queries the SIM800L using the AT+CREG command to confirm cellular connectivity. At the same time, the ESP32-CAM independently boots its Wi-Fi stack and begins serving the live video stream to the mobile application. Only once both modules have successfully confirmed their readiness does the FSM advance to the NAVIGATE state, ensuring the robot never moves forward until the entire system is fully prepared and operational [2].

Navigation State (NAVIGATE)

Once the robot enters the NAVIGATE state, it begins actively searching for the fire by sampling all three flame sensor voltages 20 times every second. The three readings — V_L from the left sensor, V_C from the centre, and V_R from the right — each produce a number between 0 and 1023 ADC units, with the numbers climbing higher the closer the robot gets to the flame. A proportional pivot algorithm takes these readings and translates them into smooth directional decisions — swinging the robot right when the left sensor is picking up the most heat, and swinging it left when the right sensor takes the lead. The moment the centre sensor V_C pulls ahead of the others and pushes past the navigation threshold T_nav of 600 ADC units — a point that roughly corresponds to the robot being within 40 cm of the flame — the FSM shifts into the SUPPRESS state and the robot moves in for suppression [1].

Suppression (SUPPRESS)

The moment the FSM enters the SUPPRESS state, the Arduino immediately brings all motor activity to a halt, stabilising the platform before asserting the relay control pin high to activate the water pump. The pump continues operating for as long as V_C remains above the suppression threshold T_sup, which is set at 400 ADC units. Once V_C drops and holds consistently below T_sup for a sustained two-second window — a reliable indicator that the fire has been successfully extinguished — the system de-energises the relay, shuts the pump down, and advances the FSM to the NOTIFY state [5][6].

Notification State (NOTIFY)

Once the NOTIFY state is entered, the Arduino sends the AT command sequence — beginning with AT+CMGF=1 to switch the SIM800L into text mode, followed by AT+CMGS to dispatch the pre-composed message "Fire suppression complete.

System ready." to the designated phone number. As soon as the SIM800L returns a positive acknowledgement confirming successful transmission, the FSM transitions back to the NAVIGATE state, resuming its patrol and standing ready to respond to any further fire events [4].

MANUAL OVERRIDE

One of the system's most thoughtfully designed features is the operator's ability to step in and assume manual control at any moment during operation, without disrupting any of the robot's critical safety functions. Directional commands issued through the mobile application travel instantly to the ESP32-CAM over the Wi-Fi link, which packages them into UART packets and passes them directly to the Arduino. The Arduino then assumes authority over the FSM's motor outputs, steering the robot according to the operator's instructions while keeping the suppression and notification logic fully active and completely unaffected. This dual-mode architecture ensures that human insight and autonomous intelligence work in close partnership — a combination that proves especially valuable in cluttered or unpredictable environments where sensor-driven navigation alone may not be sufficient [6].

Post-Suppression Notification

Finally, the robot performs the Post-Suppression Notification. Once the sensors confirm the heat has dropped below the danger threshold, the pump shuts off. To complete the mission, the Arduino sends a command to the SIM800L GSM module to transmit an "End-of-Task" SMS alert to the user's phone [4]. The dedicated power management system keeps the communication link stable and reliable by using buck converters to protect the system from high-power surges produced by the pump, preventing unexpected resets or damaging brown-outs [7].

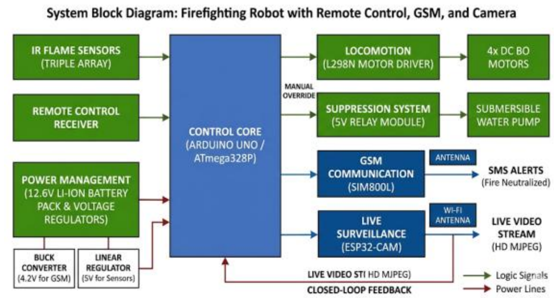

Fig. 7. Block Diagram of the system

V. RESULT AND DISCUSSION

When we took the robot out for real-world testing, it really held its own. We put it through several live-fire drills, and the way it tracked the flames was impressive. Thanks to the three infrared sensors, the robot could "triangulate" the heat almost instantly, accurately navigating toward the flame in about a second and a half [1]. As illustrated in the System Block Diagram (Fig. 7), this data is sent directly to the Arduino Uno, which acts as the primary processing core [2]. Even with a full tank of water adding extra weight, the 12.6V battery pack provided sufficient torque to keep the movement smooth and steady.

Watching the live video feed from the robot's "eyes" gave us a whole new level of control. The ESP32-CAM delivered consistently smooth video at 30 frames per second, giving the team a clear and reliable visual confirmation of the fire's status directly on their screens [3]. This "eyes-on" approach is a critical advancement over basic autonomous models. Once the fire was out, the "Fire Neutralised" text popped up on our phones via the GSM network [4]. Even with a small 6-to-9-second cellular delay, the system proved that the remote alerting interface was totally reliable.

The position-locking logic performed exactly as designed — the moment the robot reached its target, it brought itself to an immediate halt, preventing any risk of overshooting the fire source. The 5V relay then engaged the submersible pump without delay, successfully neutralising the fire in under five seconds [5]. This phase of testing offered a compelling demonstration of the dual-mode capability, confirming that the robot could operate

© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717389

autonomously and effectively while remaining under the operator's watchful supervision at all times [6].

The journey wasn't without its hurdles. Early testing exposed a frustrating but familiar nemesis in embedded systems — Electromagnetic Interference (EMI). Every time the pump and motors kicked in, the resulting power surge would knock the GSM module offline, severing the communication link at the worst possible moments. The fix required a two-pronged approach: routing the GSM module through a dedicated buck converter to shield it from voltage fluctuations on the main power rail, and fitting capacitors across the motors to suppress the electrical noise at its source [7]. Once this interference was tamed, the system's communication backbone held steady — even under the strain of high-current loads.

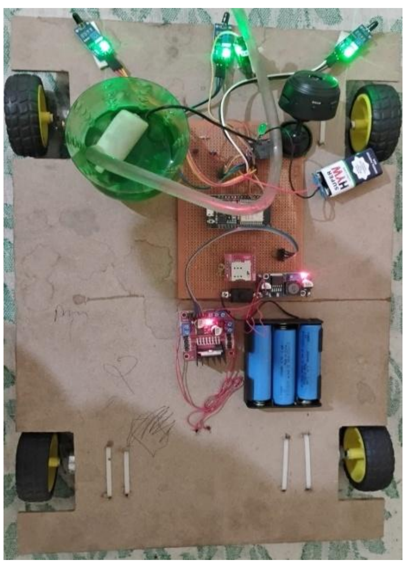

Fig.8. Fire-fighting robot

VI. CONCLUSION

This project stands as compelling proof that everyday electronic components, when thoughtfully combined, can give rise to a powerful and practical solution for industrial safety challenges. Drawing on the sensor-based navigation principles established by earlier research [1], the team successfully brought to life a prototype that independently seeks out and identifies fire without any human guidance. At the core of this success was the integration of an Arduino controller [2], which acts as a smart link between a dangerous environment and a safe operator. As illustrated in the System Block Diagram (Fig. 7), this architecture allows the machine to do much more than just move; it interprets complex data to protect human life.

The real breakthrough here was building a robust two-way communication system. The live video feed provided by the ESP32-CAM allowed the team to apply machine vision principles and monitor potential dangers in real time [3], while the SIM800L module took care of delivering timely status reports directly to a phone through the cellular network [4]. The inclusion of a 5V relay and submersible pump in the hardware design [5] proved that the robot was capable of doing more than just detecting threats — it could actively neutralise them as well.

This project goes a step further by proving that a dual-mode robot is fully capable of managing high-power suppression tasks without ever losing contact with the operator [6], laying the groundwork for a new generation of advanced fire safety systems. Equally significant is the team's success in overcoming the persistent challenges of electrical noise and voltage regulation — issues that have long been known to destabilise similar robotic systems [7]. These achievements demonstrate that the design is genuinely tough enough to withstand real-world conditions. As the complexity of modern buildings and factories continues to grow, dependable, data-driven robots like this one will become increasingly vital in protecting lives and accelerating emergency response efforts.

VII. FUTURE SCOPE

This project was never meant to stop here. What the team has built is less a finished product and more a proof of concept — a working foundation with a great deal of untapped potential still waiting to be unlocked. The mechanical design and sensor-driven navigation that power the current build [1] were always envisioned as a starting point, not a ceiling. As the System Block Diagram (Fig. 7) shows, the Arduino-based core [2] was deliberately kept modular — not out of limitation, but out of foresight. It was built to grow.

And grow it could. Perhaps the most exciting frontier ahead is the integration of AI and Deep Learning.

Right now, the system sees — but it doesn't yet truly understand what it's looking at. Future iterations could change that entirely, training the camera feed to tell the difference between a genuine flame and a flash of sunlight [3], or even to recognise a person trapped in the chaos of a crisis. These aren't distant dreams — they're the natural next steps for a system that has already proven it can handle the fundamentals.

The team's ambitions extend well beyond the current build. The existing GSM notification system [4] was always a starting point — future versions could adopt LoRaWAN, freeing the robot from Wi-Fi dependency and enabling it to operate across vast or remote environments. Suppression capabilities could grow too, with CO2 or dry powder agents [5] replacing water for chemical and electrical fires where conventional responses would only worsen the situation.

Perhaps the boldest vision, though, is swarm intelligence. Building on the 'Dual Mode' functionality already tested [6], multiple units could one day coordinate seamlessly — dividing tasks, sealing exits, and covering ground no single robot could manage alone.

Holding all of this together is the question of endurance. Smarter power management, including solar self-charging and tighter voltage regulation to prevent interference between thermal cameras and high-gain antennas [7], would ensure reliability when it matters most. These aren't incremental improvements — they're the steps that transform a capable prototype into a true first responder, one built not just to react to disasters, but to prevent them.

REFERENCE

[1] H.P. Singh et al. (2013). "Design of Fire Fighting Robot." This paper discusses the basic mechanical design and the use of sensors to navigate toward a flame

[2] Maruf, Md & Sagor, M. & Kanta, Sadiya & Imran, Md Hafizul. (2025). An Autonomous Arduino-Based Firefighting Robot for Laboratory Environments. International Journal of Computer Applications. 187. 56-65. 10.5120/ijca2025925488.

[3] Dhiman, Amit, et al. "Firefighting robot with deep learning and machine vision." Neural Computing and Applications 34.4 (2022): 2831-2839.

[4] Molla, Md Syed Ali, et al. "Experimental study on a robot-assisted automatic firefighting system using GSM & AGSM networks and other surveillance systems." Proceedings of the 15th Annual Paper Meet 7 (2014): 08.

[5] D. M. Mary, M. A. A, R. K, and S. S. M, "IoT-Enabled Smart Robot for Fire Detection, Surveillance, and Suppression with Ultrasonic Navigation," 2025 10th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2025, pp. 890-896,

[6] S. Rajesh, H. G. Hussain, D. J. Hashim, P. P. Reddy, G. S. Charan Raj, and M. Srikanth, "Arduino Microcontroller-based Dual Mode Fire Extinguishing Robot," 2024 International Conference on Augmented Reality, Intelligent Systems, and Industrial Automation (ARIIA), Manipal, India, 2024, pp. 1-5

[7] A. A. A. Rahman, Z. Janin, R. Sam, M. Masrie, T. S. Gunawan, and F. D. A. Rahman, "Firefighting Robot Based On IoT and Ban Levels Technique," 2022 IEEE 8th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA), Melaka, Malaysia, 2022, pp. 138-142, doi: 10.1109/ICSIMA55652.2022.9929114,