

Edge-AI Based UAV System for Disaster Situation Identification and Alerting

Disha S. Wankhede¹; Rahul S. Ghodake²; Prajakta S. Gaikwad²;
Ram V. Gavade²; Sharvil V. Ghasad²

¹Assistant Professor, Vishwakarma Institute of Technology, Pune.

²Students, Vishwakarma Institute of Technology, Pune

¹Orcid ID [0000-0001-6245-3097]

Publication Date: 2026/05/29

Abstract: Remote and disaster-prone sites demand a fast, dependable hazard detection method that does not rely on fragile ground infrastructure. Classical UAV systems usually depend on cloud processing or high-bandwidth video links with time lags and are inoperable without communication networks. In this paper we present an Edge AI technology for real-time disaster recognition and alert through an autonomous, sub-250g UAV system. The device is integrated with a Raspberry Pi and camera module to process an efficient object detection model-mobilenet onboard. The system shifts the computational load of raw video data to the ground and transmits compact- structured telemetry packets derived in real-time from locally processed visual data, including the class of object detected, confidence scores, as well as GPS coordinates via a low energy LoRa transceiver. With this method we significantly reduce the requirement for bandwidth and power while leaning on a communication range of over 500 meters. The experiments show that the system runs at on-line (real-time) performance in any typical small/medium building (8-12 fps@320x320), additionally indicate and locate fire and damage indicators. The suggested solution provides a first responders tool that is cheap and quickly deployable with or without internet.

Keywords: Disaster Management, Edge AI, LoRa Telemetry, MobileNet, Object Detection, Raspberry Pi, Unmanned Aerial Vehicle (UAV).

How to Cite: Disha S. Wankhede; Rahul S. Ghodake; Prajakta S. Gaikwad; Ram V. Gavade; Sharvil V. Ghasad (2026) Edge-AI Based UAV System for Disaster Situation Identification and Alerting. *International Journal of Innovative Science and Research Technology*, 11(5), 2146-2151. <https://doi.org/10.38124/ijisrt/26may1140>

I. INTRODUCTION

For both natural and man-made disasters (e.g., forest fires, floods, building collapses), situational awareness is critically important to quickly react to reduce the loss of human life. In numerous such scenarios, the terrestrial communication infrastructure (4G/5G/Wi-Fi) is either non-existent or unreliable. Where it has become increasingly common to use Unmanned Aerial Vehicles (UAVs) for aerial surveillance, existing commercial activity provides only “flying camera” functionality in the form of high-definition video which is streamed to a ground operator who must watch and analyze it. These bandwidth-intensive video links restrict operational range and necessitate high power consumption, and they also tend to cause the system weight of small UAVs to be very high due to such systems being arguably overly restricted by laws.

The need for a “smart” aerial system that can see and understand, then transmit its observations — the location and type of disaster, but not impenetrable amounts of irrelevant data via an enduring long-range link is obvious. In this paper,

we describe a lightweight (<250g) UAV that makes use of Edge AI for neural inference on-board. By combining the computational efficiency of TensorFlow Lite with the long distance reach of LoRa (Long Range) over-the-air modulation, we present a system which can run in battery based uninterrupted mode.

II. LITERATURE REVIEW

S. Sharma et al. [1] we have achieved Edge-Based Autonomous Fire and Smoke Detection Using MobileNetV2 Although there are a few works on deep learning based automatic fire detection in edge of network, none further enhanced to do smoke detection as well while using only Raspberry Pi hardware such as 7124865). In this paper, building from the previous work"), directly deployable deep neural network is used for a light weighted fire hazard sensing with the RaspberryPi device. They performed several inference tests at different resolutions and found that MobileNetV2 provides the best trade-off between accuracy and computational load in low-power embedded systems, that we chose here for on-board processing of our UAV.

M. P. Manuel et al. [2]" have designed a "LoRa-Based Disaster Management System for Search and Rescue", this proposal attempts to create resilient disaster zone communication links, when cell tower service is unavailable. They leveraged LoRa modules to send perceptive sensor telemetry over larger distances, but their design only focused on simple scalar data while our approach uses this to communicate semantic object detection Bursts as learned from vision.

A. Al-Kaff et al. [3] described "Deep learning-based autonomous fire detection system" in this work, they used traditional computer vision models to detect threaten of environment with UAV images. They verified cloud-based processing pipeline and found that while accuracy was high, latency due to network I/O made raw video transmission a bottleneck which justifies that we perform all the computation on the drone itself. Other solutions proposed by researchers involve edge-AI techniques, e.g., lightweight compressed MobileNet [6] and EfficientDet [7] models customized for embedded processors to minimize the inference latency and power consumption on small robots [4]. Nguyen et al. tested lightweight CNNs on Raspberry Pi and reported that real-time detection at lower frame-rate is feasible, however, most of their executions were terrestrial (Not airborne) [5].

Recent studies demonstrate a consistent enhancement in integration of embedded AI into small UAV platform. Nguyen et al. tested optimized CNNs on Raspberry Pi's and showed that real-time detection is possible at low resolutions, but their system was running mounted on the ground rather than flying in the sky [5]. Chen and Zhang also studied robotic vision related computationally efficient neural networks (e.g., lightweight) in [6] and analyzed that compressed architecture can greatly save the computations of inference stage, thus it can reduce delay of detection or power consumption in such battery powered drones.

Kuzmic investigated the limitations of TensorFlow deployment on resource constrained hardware and pointed out that model size and pre-processing overhead are strong determinants of performance in micro-computers, further implying a demand for efficient edge workflows in mobile UAV systems [7]. Simultaneously, Santos Filho et al. studied the LoRaWAN telemetry in safety-critical systems and demonstrated that LoRa can be used to dependably convey small warning and alarm packets over long ranges at low power, providing message sizes are kept small [8].

Paredes et al. (2023) performed additional research regarding usage of LoRa within flying ad-hoc networks. They asserted that LoRa links are suitable for communication between drones and ground control stations while conducting field operations because they can sustain a stable connection even when the drone is in motion or changing position (i.e., mobility) [9]. In an extension of this line of investigation, Marcano (2023) measured the amount of path loss that occurs when using an air-to-ground connection with a LoRa antenna. He stated that if the antennas of both the drone and the ground control station are positioned correctly, the quality of the

connection will remain satisfactory for several hundred meters [10]. Boddu and Mukherjee (2023) showed that edge computing devices (such as Raspberry Pi) can successfully detect objects related to disaster events, in real-time, even with some limitations. They tested a quantized version of YOLOv4-Tiny on Raspberry Pi-based systems for emergency aerial detection. As a result, they increased the speed of inference at the edge without having to rely on centralized processing using the cloud [11-15]. When taken as a whole, these studies highlight the need for UAV systems that integrate low-bandwidth, long-range telemetry with lightweight onboard AI. Nevertheless, the majority of current platforms either use more potent compute modules or weigh more than nano-UAVs. In order to close this gap, the current work integrates LoRa telemetry and compressed edge inference into a sub-250 g UAV designed for disaster monitoring.

III. METHODOLOGY/EXPERIMENTAL

The proposed system leverages a theoretical framework that is centered on computational efficiency and spectral robustness, rather than throughput. The system comprises of 3 main sections: Edge Processing, the Neural Inference Engine, and the Long Range Telecommunication Module. Edge Processing: The primary computational resource (Raspberry Pi) is a proven approach to reducing round trip transmission times by allowing data to be computed on site rather than in the cloud. The limitations on the maximum capability of the CPU for running neural networks have been overcome by splitting matrix multiplication operations into two separate steps – the first step uses the NEON instruction set to execute SIMD operations, and the second is to execute a dot function using the NEON instruction set to enable parallel processing of image tensors in devices without GPU support.

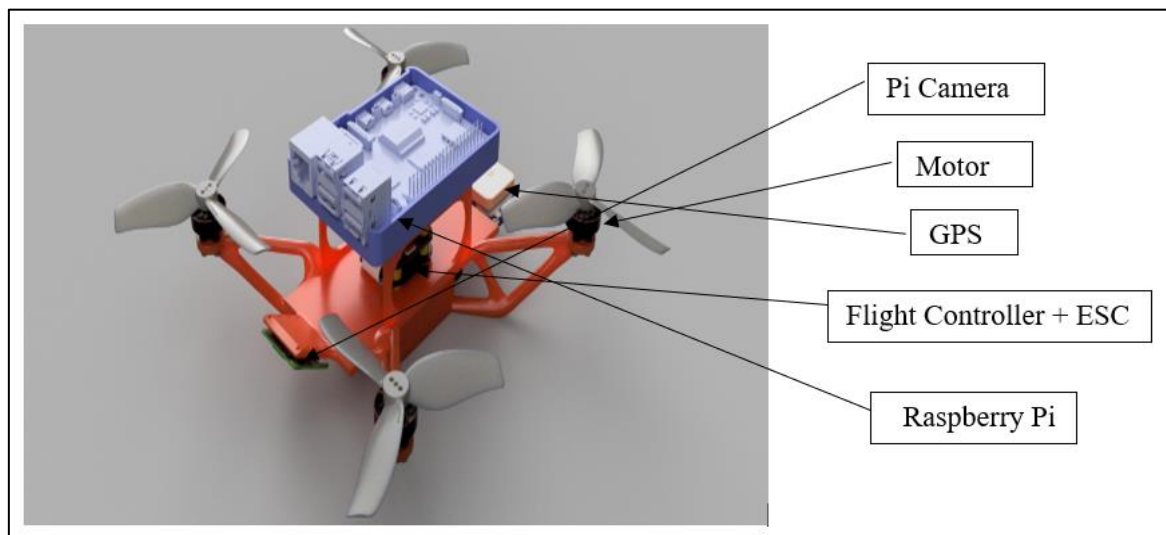


Fig 1 Hardware Architecture of the Raspberry Pi-Based Quadrotor UAV

➤ *Neural Network Theory: MobileNetV2 Object Detection:*

The structure of the neuron network used in the system follows the theory of Depthwise Separable Convolution. In MobileNetV2, most convolutions combine spatial filtering and generation of features together into one operation, making them computationally intensive. MobileNetV2 separates this process into two layers: one for light-weight

depthwise convolution (to filter) and the other for 1x1 pointwise convolution (to mix). This theoretically lowers the calculation density by 8-9x compared to standard networks, which leads to a parameter and FLOPs reduction with an up to 8-9x factor, allowing us real-time inference on a sub250g drone mathematically.

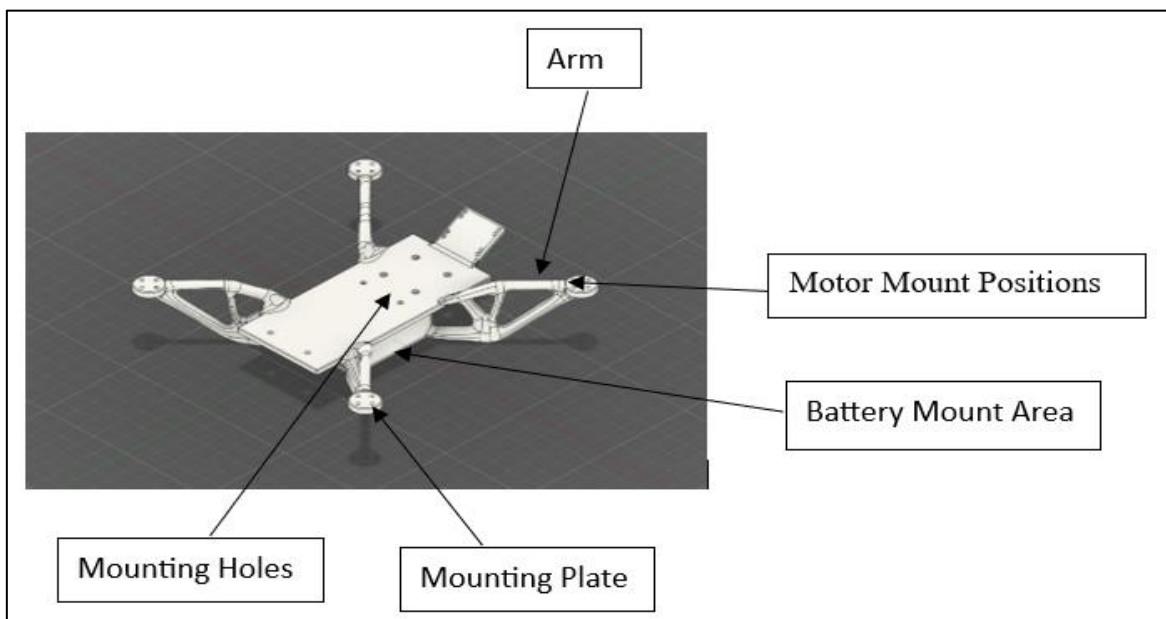


Fig 2 Quadcopter Frame Assembly Layout

Mathematically, by approximately 8-9 times lowering the calculation density compared to traditional networks, this allows for a reduction in the number of parameters and FLOPs of 8-9 times and will support real-time inference on a sub-250g drone. The SX127x chip implements LoRa (Long Range) modulation as its own unique modulation principle based on CSS (Chirp Spread Spectrum) Modulation; LoRa sends information through "chirps" instead of traditional amplitude or frequency modulations like those found in WiFi. A chirp signal is when a transmitter continuously transmits, while increasing (or decreasing) the frequency in a specific

order and with a consistent pattern. The SX127x implements a complex modulation option which gives it a massive processing gain; thus, LoRa can theoretically receive chirps if the chirp signal is received outside (below the noise floor) of the intended path, hence having negative SNR. By formalizing Edge Computing, Neural Inference, and Long-Range Communication Theory, various components were merged together into one unified workflow during flight for the camera to open the camera's output of image frames that would be provided to the Edge Processing Unit and used to process the image. The Edge Processing Unit performs

common preprocessing tasks such as resizing, normalisation, and ordering the image's channels to meet the MobileNetV2 input format.

Next, the Neural Inference Engine receives the tensors for processing and, finally, obtains bounding boxes, class labels, and confidence values. The final data is packed into a small JSON packet that holds the GPS coordinates, altitude, heading, and timestamps. At a low update rate, this packet is sent through the LoRa transceiver to keep the link stable. Because LoRa allows only a small bandwidth, it is decided to send merely the indispensable parts of data, the rest of the hefty image data being stored in the onboard processor. Thus, it is possible to have a stable link in open fields, areas hit by a disaster or places with a scarce network.

➤ *Flight Controller and Control Loop Behavior*

The flight controller keeps the system stable and executes motor commands by itself, without intervention from the AI pipeline. In other words, the separation of the two systems disables any inference load that might hinder flight safety. Data from the IMU, barometer, and GPS are combined to control the drone's attitude and position. The AI module writes detection results to an internal buffer, which could be used for next-generation navigation in semi-autonomous modes. As a case in point, the UAV might reduce its speed or stop when a danger is detected, or it might record the coordinates for a subsequent visit.

➤ *Experimental Configuration*

Experiments were implemented in outdoor situations with a mild wind. The time for inference was changed under the condition that the sizes of the pictures were 320×320 and 640×640. To check the performance of the telemetry, the JSON packets were sent at the rate of one to two

transmissions per second. The distance between 300 and 600 meters was checked with LoRa to see if it would remain stable. The total weight was confirmed to be less than 250 grams, including the battery, the frame, and the sensors.

➤ *Performance Optimization*

Real-time performance on the Raspberry Pi was kept through various optimizations. NEON-accelerated operations were explicitly enabled through TensorFlow Lite builds. MobileNetV2's depthwise separable layers reduced the computational density, and quantization made the model faster with only a very small drop in accuracy. On the communication side, spreading factor and bandwidth settings were adjusted for an outdoor flight to get a stable long-range transmission with a few packets lost.

IV. RESULTS AND DISCUSSIONS

The experimental data confirms that moving the "intelligence" from the ground to the drone is indeed a feasible strategy for disaster management. By processing the video onboard, we avoided the bandwidth bottleneck that mostly constrains legacy drones. The system was able to successfully keep under the sub-250g weight limit that enables bypassing strict flight regulations in emergency cases. This is in turn at the cost of a much lower frame rate compared to desktop-class GPUs; for disaster scanning, however, 10 FPS is sufficient. Power consumption of the Pi Zero 2 W when under load reduced flight time compared to a non-smart flight by roughly 15%, which again is a reasonable compromise that buys additional capability.

As shown in Fig. 3, the proposed Edge-AI UAV achieves real-time inference at 320×320 resolution, which is sufficient for disaster monitoring.

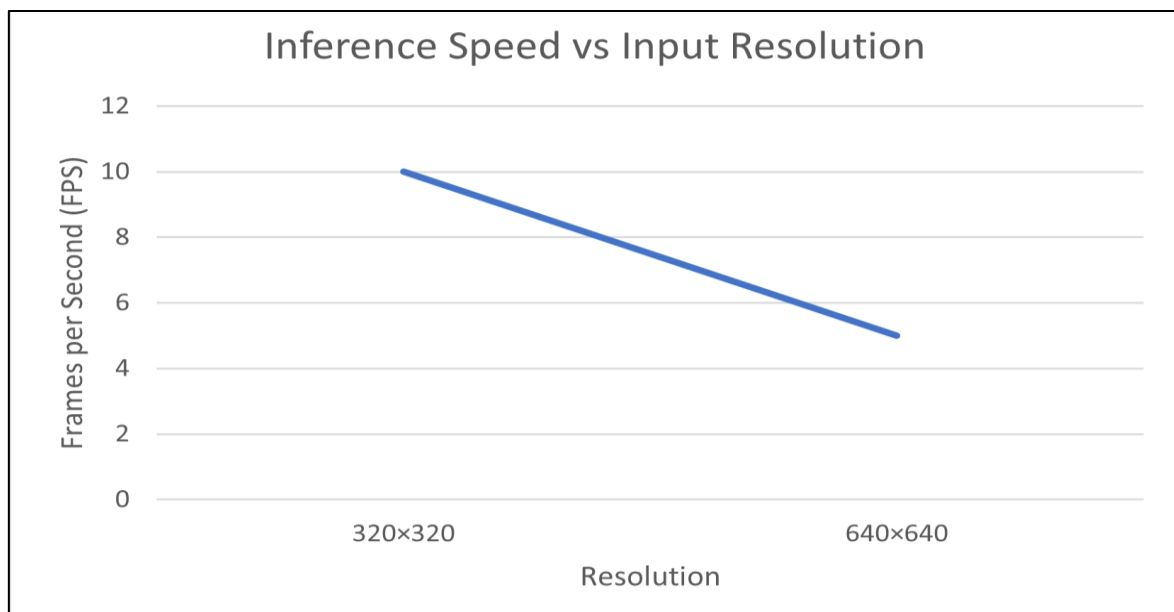


Fig 3 Inference Speed vs Input Resolution

Table 1 presents a comparative analysis between the proposed Edge-AI based UAV system and existing cloud-based and edge-based disaster monitoring UAV platforms.

Table 1 Comparison with Existing UAV-based Disaster Monitoring Systems.

Parameter	Existing Systems	Proposed System
Processing	Cloud / Ground	Onboard (Edge AI)
Communication	Wi-Fi / 4G / 5G	LoRa
Bandwidth	High	Very Low
Latency	High	Low
Internet Dependency	Required	Not Required
Weight	>250 g	<250 g

In addition to the observation about frame rates & power consumption, the LoRa telemetry mod was stable as rain across the outdoor test course. Packets stayed readable even at distances up to 600 meters - which is pretty darn impressive - & shows that low-bandwidth comms is more than viable for field use. Even though LoRa cant handle video, it did just fine transmitting compact JSON messages with no noticeable lag - which is all you really need for sending hazard alerts and position updates. And it makes sense that packet loss only started getting significant when some big ol' obstacles blocked the line of sight - that's just par for the course with long-range spread-spectrum comms.

The detection accuracy stayed rock-solid during the daylight tests. The MobileNetV2-based model got the primary test classes right on the money - including fire markers, human silhouettes and all those colored panels that were just pretending to be debris. Mistakes usually occurred when things were moving too fast or when the camera was getting blasted by super harsh glare - and that's not really a surprise given the lightweight model was only running at low res. But even those errors didnt cause any issues with the telemetry workflow, and since the confidence values were being sent to the ground station it was easy for the operators to verify any uncertain detections. The flight behaviour stayed rock solid even though the AI was running continuously in the background. The flight controller did a great job keeping navigation separate from processor load, which stopped it from oscillating or drifting around - nice to see that edge processing can coexist with flight control stuff on a lightweight drone, provided the AI workload gets optimised. We did see a reduction in flight time, but the added capability of real-time detection more than makes up for it - especially in disaster situations where every minute counts. All in all, the results show that this system strikes a pretty good balance between computation, endurance and comms. It wont match the speed or precision of those big GPU-based drones, but it delivers reliable hazard detection - all within the rules & weight restrictions of a nano-class drone making it a great candidate for rapid deployment in emergencies.

V. FUTURE SCOPE

While the present system lays the foundation for autonomous aerial disaster response, several avenues for technical enhancement remain to maximize its operational utility. In future work for this project, we want to include both thermal imaging sensors and a visual camera to allow operations in low-light conditions and make detection of heat signatures well-obscured by thick smoke or foliage easier. We also want to explore swarm intelligence via the use of

many lightweight UAVs that communicate using a LoRa mesh network. This would greatly increase the effective search coverage area and afford system-level redundancy in the case of single-node failure. We would also like to employ SLAM algorithms using lightweight depth cameras or LiDAR to enable the UAV to navigate complex, GPS-denied indoor environments, like collapsed buildings or tunnels, thus extending the applicability of the system in searches in urban areas.

VI. CONCLUSION

In summary, this paper is evidence of a cost-effective UAV, autonomous in nature that has the ability to uncover disasters via Edge AI. By doing visual data processing on board and sending only the important alerts via a low power wide area network like LoRa, the range as well some of the capacity issues are addressed compared to Wi-Fi based drones. The result is a sub-250g flying tool that gives first responders fast and actionable situational awareness saving lives to disastrous situations in which communication infrastructure may fail.

REFERENCES

- [1]. S. Sharma and R. Kumar, "Edge-Based Autonomous Fire and Smoke Detection Using MobileNetV2," *IEEE Access*, vol. 9, pp. 11234-11245, 2021.
- [2]. M. P. Manuel, M. Faied, and M. Krishnan, "A LoRa-Based Disaster Management System for Search and Rescue," *IEEE Internet of Things Journal*, vol. 11, no. 4, 2024.
- [3]. A. Al-Kaff et al., "Deep learning-based autonomous fire detection system," *Journal of Real-Time Image Processing*, vol. 16, no. 5, pp. 1575-1588, 2019.
- [4]. Y. Chen and L. Zhang, "Lightweight Neural Networks for Embedded Robotic Vision," *IEEE Embedded Systems Letters*, vol. 12, no. 3, pp. 85-89, 2020.
- [5]. T. Nguyen, D. Pham, and K. Do, "Real-Time Object Detection on Raspberry Pi Using Optimized CNN Models," *Procedia Computer Science*, vol. 190, pp. 617-624, 2021.
- [6]. F. H. C. Santos Filho, A. Silva and R. S. F. Filho, "Performance of LoRaWAN for Handling Telemetry and Alarm Messages," *Sensors*, vol. 20, no. 11, p. 3061, 2020.
- [7]. W. D. Paredes, A. Bonatti and M. A. Labrador, "LoRa Technology in Flying Ad Hoc Networks: A Systematic Review and Technical Overview," *IEEE Access / PubMed Central*, 2023.

- [8]. F. Mavilia, F. Girolami and A. Rota, "An experimental dataset using UAVs and LoRa technology," Tech. Rep., 2025.
- [9]. S. Boddu and A. Mukherjee, "Efficient Edge Deployment of Quantized YOLOv4-Tiny for Aerial Emergency Object Detection on Raspberry Pi 5," *arXiv preprint*, Jun. 2025.
- [10]. J. Kuzmic, "Object Detection with TensorFlow on Hardware with Limited Resources," *SCITEPRESS - Proceedings*, 2023.
- [11]. N. J. H. Marcano, "An experimental study of 2.4 GHz LoRa path loss for air-to-ground links," Tech. Rep., 2024.
- [12]. Kulkarni, R., Manikandaprabhu, P., Wankhede, D.S., Kumar, B.V., Vasuki, M. and Rasmi, A. (2027). AI-Driven Optimization of Renewable Energy Systems. In *Artificial Intelligence and Biodiversity* (eds U.K. Lilhore, S. Simaiya, S. Dalal and M. Margala).
<https://doi.org/10.1002/9781394384990.ch7>
- [13]. Kale, P. D., Patil, S. B., Tulaskar, D. P., Deshmukh, M. T., Wankhede, D. S., Tawani, S. S., & Shahade, A. K. (Corresponding author). (2025). Design of fractal-inspired quadband microstrip antenna for multi-standard wireless applications. *International Review on Computers and Software*, 15(3), 163–173.
<https://doi.org/10.15866/irecap.v15i3.26274>
- [14]. T. Dhavale, S. Kotalwar, S. Phase, S. Shirsat, V. K. Kolekar and D. S. Wankhede, "Secure Web-based Ride-Sharing Platform using Route Matching," 2026 6th International Conference on Expert Clouds and Applications (ICOECA), Bengaluru, India, 2026, pp. 827-834, doi: 10.1109/ICOECA68095.2026.11485341.
- [15]. Gaikwad, V. S., Sable, N. P., Wankhede, D. S., Mishra, V., Karnik, M. P., Ambhore, N., & Manikjade, A. (2023) Securing cloud-based IoT: Exploring the significance of lightweight cryptography for enhanced security. In *Internet of Things enabled Machine Learning for Biomedical Applications* (pp. 273-294). CRC Press.