Department of Computer Science and Engineering

Sylhet Engineering College

Quadcopters in Crisis: Enhancing Disaster Response through IoT and Machine Learning

Submitted by

Jhumki Barmon
Reg no :2019331553
4th year, 2nd semester
Department of Computer Science and Engineering
Sylhet Engineering College

Supervisor

Md. Lysuzzaman
Lecturer
Department of Computer Science and Engineering
Sylhet Engineering College

Date: 22 July 2025

Candidate Declaration

This is to certify that the work presented in this thesis, titled, "**Quadcopters in Crisis: Enhancing Disaster Response through IoT and Machine Learning**", is the outcome of the investigation and research carried out by us under the supervision of **Md. Lysuzzaman.**

It is also declared that neither this thesis nor any part of the paper has been submitted anywhere else for the award of any degree or diploma.

JHUMKI BARMON
Reg No: 2019331553
4th year, 2nd semester
Department of Computer Science and Engineering
Sylhet Engineering College

<u>Recommendation Letter from Thesis Supervisor</u>

It is with great satisfaction that I confirm the acceptance of the thesis title '**Quadcopters in Crisis: Enhancing Disaster Response through IoT and Machine Learning'**, submitted by the following student, as a partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering.

____

Signature of Supervisor

Md Lysuzzaman
Lecturer
Department of Computer Science and Engineering
Sylhet Engineering College
Date:22/07/2025

Certificate of Acceptance of the Thesis

The thesis titled "**Quadcopters in Crisis: Enhancing Disaster Response through IoT and Machine Learning'**" submitted by **Jhumki Barmon**; Student ID. **2019331553**; Session **2019-20**, to the Department of Computer Science and Engineering, Sylhet Engineering College, has been accepted as satisfactory in partial fulfilment of the requirement for the Degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents.

Internal

Nayan Kumar Nath
Lecturer
Department of Computer Science and Engineering
Sylhet Engineering College, Sylhet.

Supervisor

Md Lysuzzaman
Lecturer
Department of Computer Science and Engineering
Sylhet Engineering College, Sylhet.

Internal

Md. Rasel Ahmed
Assistant Professor
Department of Computer Science and Engineering
Sylhet Engineering College, Sylhet.

Internal

Md. Nagrul Islam
Assistant Professor
Department of Computer Science and Engineering
Sylhet Engineering College, Sylhet.

Chairman

Md. Abu Naser Mojumder
Head
Department of Computer Science and Engineering
Sylhet Engineering College, Sylhet.

Member (External)

Dr. Mohammad Shahidur Rahman
Professor
Department of Computer Science and Engineering
Shahjalal University of Science and Technology.

Acknowledgment

I express my sincere gratitude to the Department of Computer Science and Engineering at Sylhet Engineering College for providing the necessary support and resources for this research.

I am especially thankful to my supervisor**, Md.Lysuzzaman (**Lecturer, Department of CSE, Sylhet Engineering College), for his invaluable guidance, continuous encouragement, and dedicated supervision throughout the research journey. His insightful feedback and expertise have been instrumental in shaping the direction of this work.

Additionally, I acknowledge the contributions of previous researchers and scholars, whose work laid the foundation for this study. Their efforts have been a significant source of inspiration and reference in my research.

Abstract

The integration of drones into disaster response has transformed emergency operations, offering unparalleled speed, precision, and adaptability. However, the need persists for more autonomous, intelligent, and versatile systems to optimize performance in high-risk scenarios. This paper presents an advanced quadcopter system tailored for disaster rescue and environmental surveying, featuring semi-autonomous navigation, machine learning-driven decision-making, and modular payload capabilities. The system leverages cutting-edge sensors, including gas, temperature, and humidity detectors, with real-time data processing enabled by a Raspberry Pi 4B and Arduino Mega for precise flight management. A standout feature is the mechanical hand with interchangeable modules for tasks such as drilling through glass to create access points and securely gripping objects during rescue operations. The drone's firefighting capabilities are enhanced through carbon ball deployment and hosepipe transport, significantly improving efficiency in emergency situations. Machine learning algorithms facilitate real-time object detection, environmental risk analysis, and flight stability optimization, ensuring adaptive responses based on live sensor data. Complementing these features is a live video streaming system for situational awareness and an intuitive mobile application for seamless control and real-time monitoring. Designed for dynamic adaptability in evolving disaster environments, this quadcopter provides actionable insights and executes precise tasks in hazardous conditions. Its integration of IoT and machine learning redefines disaster relief operations, offering transformative potential in firefighting, search and rescue, and environmental monitoring. By enhancing autonomy, operational efficiency, and real-time decision-making, this system emerges as a critical tool for modern disaster response missions

Index Terms: Quadcopter, Disaster Response, IoT, Machine Learning, Real-Time Data Processing, Environmental Monitoring, Rescue Operations, Flight Stability Optimization

Contents

Chapter one

Introduction

Disasters, whether natural or man-made, pose serious threats to human lives, infrastructure, and the environment, often causing widespread destruction and long-term socio-economic consequences [**1**]. Events such as earthquakes, floods, hurricanes, wildfires, and industrial accidents disrupt communities, making rapid and effective emergency response crucial for saving lives and mitigating damage **[3].** However, traditional disaster management systems face limitations, including poor accessibility, communication failures, and lack of real-time information, which place human responders at significant risk [**15**].The Golden Hour—the first critical hour after a disaster—plays a decisive role in survival, and delays due to logistical constraints, hazardous conditions, or inefficient reconnaissance can drastically reduce rescue effectiveness **[4**]. Conventional ground-based search-and-rescue operations often struggle in unstable structures or difficult terrains, highlighting the need for technology-driven solutions that enhance situational awareness and accelerate response [**5**].

Unmanned Aerial Vehicles (UAVs), particularly quadcopters, have emerged as key tools in modern disaster management. IoT-based autonomous drones enable precise firefighting and real-time situational monitoring [**6**]. AI-assisted drones facilitate human detection and rescue through computer vision and audio-visual sensors, although flight time and environmental noise can limit performance [**2**]. Integration of IoT [**7**] devices, cloud computing, and GIS platforms improves data collection and decision-making during emergency operations [**8**]. Advanced object detection, multi-object tracking, and deep learning algorithms enhance UAV capabilities for locating victims, monitoring hazards, and performing high-precision reconnaissance [**9**]. Facial recognition and thermal imaging drones provide real-time monitoring for enhanced situational awareness [**10**] Learning-based and reinforcement learning controllers improve quadcopter adaptability under varying conditions, ensuring stability and reliability in dynamic disaster zones [**11**]. Multi-purpose drones extend functionality to bridge inspection, firefighting, and logistics operations, despite challenges such as payload limits and weather dependency Safety-critical real-time IoT systems integrated with machine learning enable continuous monitoring while addressing sensor reliability and processing constraints [**13**].

Further applications include methane emission monitoring in industrial sites and agricultural support such as pesticide spraying and crop health assessment [**16**][**14]**. Lightweight deep learning models deployed on drones enable real-time fire detection and hazard assessment, even under computational limitations [**17**][**18**].This research focuses on developing an AI-driven, semi-autonomous quadcopter system for high-resolution aerial surveillance, hazard identification, and survivor detection, with real-time communication to emergency teams. Key functionalities include a robotic arm for object manipulation, a hosepipe and carbon ball shooter for firefighting, and semi-autonomous surveying for urban and challenging terrains. Optional agricultural modules allow crop monitoring and pesticide spraying [**12**]. By integrating autonomous navigation, machine learning, and robotic rescue mechanisms, the system aims to reduce response times, responder safety, and optimize resource allocation, representing a significant advancement in disaster management technology.

Research Questions:

1. How can AI-driven quadcopters enhance real-time situational awareness and decision-making during disaster response operations?
2. What role do IoT-enabled sensors play in improving hazard detection, environmental monitoring, and survivor identification in disaster zones?
3. How can machine learning algorithms optimize flight stability, obstacle avoidance, and autonomous navigation in complex and hazardous environments?
4. What are the performance benefits of using modular payloads, such as robotic arms, glass-drilling mechanisms, hosepipe carriers, and carbon ball shooters, in multi-functional disaster rescue missions?
5. How does the integration of advanced video transmission systems, like the Skydroid Dual Antenna and TS135 Video Transmitter, improve real-time communication with emergency response teams?
6. How effective is the quadcopter in agricultural applications, specifically in pesticide spraying and crop monitoring, when equipped with the agricultural module?

This study seeks to address these questions, providing insights into the potential of AI-powered, semi-autonomous quadcopters as transformative tools for modern disaster management systems and agricultural applications.

Research Objectives

The objectives of this research are to:

- To design and develop a semi-autonomous quadcopter integrated with AI and IoT technologies for efficient disaster response and management.
- To enhance real-time situational awareness through the implementation of advanced video transmission systems and IoT-enabled sensors.
- To optimize the quadcopter's flight stability and autonomous navigation using machine learning algorithms in complex and hazardous environments.
- To evaluate the effectiveness of modular payloads, including robotic arms, glass-drilling mechanisms, hosepipe carriers, and carbon ball shooters, in search-and-rescue and firefighting missions.
- To improve real-time hazard detection, survivor identification, and environmental monitoring using AI-driven data processing and deep learning techniques.
- To explore the quadcopter's versatility in agricultural applications, focusing on pesticide spraying efficiency and crop health monitoring.
- To assess the system's overall impact on reducing response times, minimizing risks to human responders, and enhancing disaster resilience in urban and rural settings.

This study seeks to address these objectives, providing insights into the potential of AI-powered, semi-autonomous quadcopters as transformative tools for modern disaster management systems and agricultural applications.

Chapter Two

Literature Review:

Real-Time Fire Detection on Drones:
An end-to-end DRL-based [19] controller for quadcopters is developed, offering data efficiency and eliminating the need for manual gain tuning. The system employs an actor-critic architecture to directly map quadcopter states to motor RPMs. A quadcopter dynamics simulator was used for training, followed by deployment on a Crazyflie nano quadrotor. Key challenges include the simulation-to-reality gap, sensor noise, and system parameter variations. The solution ensures direct deployment on real hardware without fine-tuning, with effective performance in trajectory tracking tasks.

Deep RL-based Quadcopter Controller
A [20] neuroevolutionary controller for autonomous quadcopter flight is proposed, ensuring path-following and attitude stability. A multi-layer perceptron (MLP) neural network is trained with a custom evolutionary algorithm to optimize rotor speed and navigation. However, the approach relies entirely on simulations, requiring extensive training time for optimal performance. The solution uses a cost-function-based evolutionary training method to improve the controller's efficiency. Future work aims at real-world implementation and further refinement.

Gym Environment for Multi-Agent Quadcopter Control
Agilicious,an open-source [22], open-hardware quadrotor, is designed for agile, vision-based autonomous flight. Featuring a lightweight 750g frame, a 5:1 thrust-to-weight ratio, and an NVIDIA Jetson TX2 GPU for real-time AI inference, it supports modular, ROS-compatible software stacks for sim-to-real transfer through Rotors and flightmares Computational constraints, vision-based estimation errors, and aerodynamic modeling limitations present challenges, addressed through neuromorphic computing, sensor fusion, and improved aerodynamic models. Agilicious serves as a scalable, reproducible testbed for advancing high-speed drone autonomy research in academia and industry.

Agilicious: Agile Quadcopter for Vision-based Flight
The expanding[23] role of UAVs in military, medical, and agricultural sectors is examined, highlighting technological advancements and challenges. Drones support precision agriculture, real-time farm monitoring, medical supply deliveries, and military reconnaissance. However, concerns about privacy violations, regulatory hurdles, and potential misuse persist. Key challenges include forensic analysis for criminal investigations, ethical concerns in surveillance, and security risks. The study emphasizes the need for improved regulations and AI-driven safety features to balance innovation with responsible drone integration in global industries.

Drone Technology: Challenges and Opportunities

[24] Emerging drone applications and their associated challenges across military, medical, and agricultural fields are explored. Military drones enhance surveillance, reconnaissance, and precision strikes, minimizing human risk. Medical drones play a crucial role in delivering vaccines, blood, and emergency supplies to remote areas. Agricultural UAVs improve crop monitoring, irrigation, and pesticide application. Despite their benefits, challenges include cybersecurity threats, regulatory gaps, privacy concerns, and operational limitations. Suggestions include AI-enhanced navigation, improved security protocols, and global regulatory frameworks to maximize drone efficiency while mitigating risks.

Opportunities and Challenges in Drone Technology

[25] The evolution of drone technology from recreational use to commercial and industrial applications is explored. The 2015 amendment of the Civil Aeronautics Act accelerated drone adoption, followed by 2016 industrial applications such as infrastructure inspection, security, and disaster response. Japan leads in agricultural drone deployment, particularly with the RMAX for crop dusting. The global drone market is projected to reach $91 billion in the next decade, with advancements in battery swapping, fuel cells, and personal transport drones. Emphasis is placed on regulatory frameworks to ensure safe urban drone operations, supporting the industry's sustainable growth.

Neuroevolutionary Controller for Quadcopter

A flexible [21] RL-based quadcopter simulation environment is developed, supporting PID, MPC, and RL for aerodynamic modeling, collisions, and large-scale RL training. Built with PyBullet, OpenAI Gym API, and RLlib, it allows efficient multi-agent reinforcement learning (MARL). Key challenges include high computational costs, simplified aerodynamic modeling, and limited real-world transferability. Solutions involve parallelization optimization, improved aerodynamic modeling, and support for diverse drone types, making it a valuable open-source RL framework for aerial robotics research

_Drone Technology and Future Prospects

UAV [**26]** technology trends and applications in mining operations are reviewed, focusing on topographic surveys, stockpile management, safety monitoring, and slope stability analysis. UAVs provide high-resolution digital surface models (DSMs), improving planning and environmental management. Infrared cameras and laser scanners enhance data collection and operational efficiency. Benefits include cost-effectiveness and accessibility to remote areas, though high initial costs technical expertise requirements pose adoption challenges. The study concludes that UAVs are transforming mining operations, with promising future applications in automated surveying, AI-driven monitoring, and real-time data analysis.

UAV Technology Trends and Mining Applications

UAV technology [27] supported pandemic management by facilitating vaccine transport, public announcements, crowd surveillance, and disinfection spraying. AI, Machine Learning (ML), and IoT were integrated into UAV systems to improve efficiency. Key challenges included privacy concerns, battery limitations, and regulatory hurdles. Strengths of the study include comprehensive coverage, technological insights, and policy discussions. Suggested improvements include incorporating real-world case studies and solutions for privacy issues.

Drone Technology in Pandemic

Autonomous [28] Drone Racing (ADR) is explored as a benchmark for testing autonomous drone technologies. The study analyzes ADR events at IROS 2016 and IROS 2017, where drones navigated cluttered indoor environments using only onboard sensing and computing. Key techniques used include waypoint tracking and robust visual recognition of colored gates. Challenges include limited computational resources, real-time flight control, and obstacle detection. The study suggests improvements through better flight control algorithms, path-planning optimizations, and AI-driven perception models, making ADR an effective testbed for real-world autonomous drone applications.

Herbicide Spraying and Weed Identification Using Drones

[30] UAVs improve oil palm plantation management, particularly in Malaysia and Indonesia, where oil palm is a key economic crop. Drones assist in mapping, monitoring, and surveillance using multispectral, thermal, LiDAR, and hyperspectral sensors. Applications include vegetation monitoring, water stress detection, drainage assessment, and yield prediction. Challenges include high costs, limited battery life, regulatory restrictions, and technical expertise requirements. Future solutions involve affordable drone models, AI-driven analysis, and improved training programs to expand drone adoption in precision agriculture.

Drone Technology in Plantations

This review examines how UAVs[31] improve plant phenotyping and crop monitoring, helping manage drought, fertilization deficiencies, and diseases. It analyzes over 200 studies, highlighting UAV platform characteristics, sensor technologies, and machine learning integration for data processing. Applications include canopy height estimation, biomass prediction, leaf nitrogen content analysis, and pest management. Challenges include data acquisition complexity, sensor limitations, and interpretation difficulties. Future improvements involve integrating AI, multispectral imaging, and real-time data analytics to enhance precision agriculture and crop breeding efficiency.

Drone-based Imaging for Plant Phenotyping

[32] This study explores multi-UAV communication networks for disaster monitoring, focusing on data coordination, trajectory optimization, and fault tolerance. It integrates mesh and hierarchical topologies, edge computing, and AI-driven analytics to enhance real-time decision-making. Challenges include node failures, limited scalability, and infrastructure dependencies. Solutions involve adaptive mobility models, decentralized edge computing, enhanced encryption, and optimized UAV placement. The study highlights network resilience and efficiency improvements as critical for scalable UAV-based disaster response.

Autonomous Drone Racing: Challenges and Technologies

Drones [29] revolutionize weed detection and herbicide spraying, boosting efficiency by 60 times compared to traditional methods. Camera-equipped drones enable precise weed targeting, while the downwash effect enhances herbicide deposition. However, altitude, speed, weather, and spray characteristics impact performance. Drones offer a safer alternative to knapsack sprayers, reducing chemical exposure and labor costs. Future improvements focus on AI-powered weed detection, optimized spray nozzles, and better flight algorithms to further enhance precision agriculture and reduce environmental impact.

Multi-UAV Networks for Disaster Monitoring

This paper presents an IoT-enabled firefighting drone [33] designed to detect and suppress fires while reducing risks to firefighters. Equipped with a Pixhawk PX4 microcontroller, multiple gas sensors (MQ3, MQ4, MQ9, MQ135), a flame sensor, and an FPV camera, the drone provides real-time air quality monitoring and live video streaming. It employs fuzzy-based backstepping control for navigation and deploys a fire-extinguishing ball via a servo motor. Field tests in Mirpur, Dhaka confirmed its efficiency and cost-effectiveness, making it an innovative fire emergency solution.

IoT-based Firefighting Drone Development

Development [34] of a UAV-Based Framework for CH_4 Monitoring in Sludge Treatment CentresThis study introduces a UAV-based methane (CH_4) monitoring framework for sludge treatment centers, supporting the UK's Net Zero 2030 goal. A Matrice 300 RTK UAV equipped with a U10 Tuneable Diode Laser Absorption Spectroscopy (TDLAS) sensor measures vertical CH_4 concentrations in wastewater treatment plants (WWTPs). The geostatistical interpolation (Kriging) and mass balance method improve asset-specific emission estimates from anaerobic digesters and sludge storage.

IoT-based Firefighting Drone Development

This study develops an AI-powered Smart Drone Surveillance System (SDSS) [35] for security and fire detection in buildings and factories. It integrates YOLOv8 and Cascade Classifier for real-time object detection, recognizing humans, fire, and dangerous objects. The system uses a Passive Infrared (PIR) sensor and flame sensor, triggered by an ESP32 microcontroller. The drone, controlled by PID algorithms, autonomously tracks intruders, adapts movements, and avoids thrown objects. Field tests confirmed high accuracy, fast response times, and real-time video streaming, making SDSS a cost-effective, automated security solution.

IoT Cybersecurity Framework for Drones

[37] Machine Learning Techniques in the Internet of UAVs for Smart Cities ApplicationsThis study explores how Machine Learning (ML) and Artificial Intelligence (AI) optimize the Internet of UAVs (IoUAV) for smart city applications. Challenges like big data processing, cloud network management, energy efficiency, and multi-UAV communication hinder full-scale IoUAV deployment. ML-driven solutions improve real-time data processing, adaptive networking, and energy-efficient UAV operations. The study highlights how AI-enhanced autonomous decision-making can revolutionize UAV integration into public safety, urban surveillance, and disaster response. Machine Learning in UAVs for Smart Cities

This research develops an AI-powered

[38]drone-based fire detection system, integrating edge computing for real-time hazard identification. A DJI F450 drone equipped with a Raspberry Pi 5 and Pi Camera Module 3 runs YOLOv8, Detectron2, and DETR models. Knowledge distillation improves performance, with YOLOv8n achieving 95.21% accuracy. Field tests in Mirpur, Dhaka showed 89.23% real-time detection accuracy, confirming a cost-effective, efficient solution for fire surveillance and disaster management.

Fire Detection with UAVs and YOLO

This **[39]** study presents a two-stage UAV-based fire detection and geo-localization system using YOLO deep learning models. A dataset of 12,000+ images (fire, non-fire, smoke) was compiled to train YOLO-NAS, achieving mAP50 of 0.71 and an F1-score of 0.68. A stereo vision-based localization method estimates fire locations accurately. Tested on a Pixhawk-controlled quadcopter, the system enhances wildfire monitoring, disaster management, and environmental protection by enabling precise fire detection and real-time response.

Smart Drone Surveillance System with AI & IoT

[36] This study proposes an IoT-enabled cybersecurity framework for securing Internet of Drones (IoD) networks, addressing cyber threats like GPS spoofing, malware injection, and unauthorized access. Using Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Bidirectional LSTM (Bi-LSTM), the system detects intrusions with up to 91.36% accuracy based on benchmark datasets (CICIDS2017, KDDCup 99). The framework enhances authentication, secure communication, and real-time cyberattack mitigation, ensuring safer drone operations in smart cities, industrial surveillance, and defense.

Wildfire Monitoring Using Autonomous UAVs

This study introduces [**40]** an autonomous UAV-based wildfire monitoring system for remote, hard-to-reach areas. It employs leader-follower coalition formation, where fixed-wing UAVs identify fire zones and relay data, while rotary UAVs collect high-resolution imagery. This energy-efficient, scalable system reduces human intervention and enhances real-time fire surveillance. Simulations confirm near-centralized performance without requiring direct drone-to-drone communication, making it a cost-effective solution for disaster response and large-scale fire monitoring.

Internet of Drone Things (IoDT)

This study introduces the concept of the Internet of Drone Things (IoDT[**41]**), a future paradigm that integrates drones with the Internet of Things (IoT) to enhance capabilities in various sectors such as agriculture, smart cities, and industry. The paper highlights the role of advanced technologies such as computer vision, cloud computing, wireless communication, big data, and high-end security in driving the evolution of IoDT. By utilizing IoDT, sectors such as surveillance, search and rescue, and monitoring can be transformed into intelligent and autonomous systems. The study also discusses key applications, security challenges, and real-time implementations, using case studies in agriculture and smart city monitoring.

Multi-rotor Drone for Volcanic Gas Measurements

[43] This paper presents an adapted multi-rotor drone system designed for studying volcanic gas plumes, with capabilities for high-altitude, long-range SO2 concentration monitoring, plume speed measurement, and remote gas sampling. The drone serves as a stable platform for various instruments, including MultiGAS for in situ gas measurements (SO2, H2S, CO2) and MobileDOAS for spectroscopic SO2 emission rate measurements. Field-tested during campaigns in Papua New Guinea and Italy, the drone system offers an adaptive flight path with real-time data transmission, ensuring optimal plume interception, even when beyond visual line of sight. This system is particularly useful for monitoring active volcanoes like Manam, where ground-based measurements are challenging. The study highlights the drone's versatility in providing valuable data for volcanic gas flux and composition, complementing traditional sampling methods.

Multipurpose Drone for Bridge Inspection & Fire Extinguishing

This project**[44]** focuses on developing a multi-purpose quadcopter capable of performing a range of tasks, including bridge inspection, fire extinguishing, and police surveillance. The goal is to expand the functionality of drones beyond simple surveillance or video recording, offering practical solutions for safety and emergency response. The drone is designed to remotely extinguish small fires, reducing the risk to firefighters, and to assist in bridge inspections, minimizing the dangers posed to inspectors. Additionally, the drone can support law enforcement by providing aerial surveillance in critical situations. The study emphasizes the potential of multi-purpose drones in a growing market, contingent on the approval of appropriate FAA regulations.

Agriculture Drone Design Using Lightweight Materials

This research focuses on the design and development of a multipurpose agricultural **[45]** drone made from lightweight materials for tasks such as water irrigation and pesticide spraying. The drone's components were selected based on payload, weight, and design requirements, with arms made from balsa wood and stands fabricated using polyvinyl chloride (PVC) and carbon fiber. Testing showed that the drone could fly for 7 minutes with a full payload, and its endurance increased twofold when the payload was reduced by 50%. The drone is designed to handle both irrigation and pesticide spraying, reducing the need for separate drones and lowering costs. This multipurpose drone not only enhances agricultural efficiency but also helps reduce the risks associated with pesticide use, improving both productivity and safety for farmers**.**

Drones for Border Surveillance

[46] This article explores the use of drones at the southern maritime European borders, focusing on their dual role in humanitarian and security efforts. It examines how state and non-state actors, including EU agencies and humanitarian NGOs, deploy multipurpose drones for border surveillance. The article highlights that the nature of these drones varies based on how the data they collect is utilized. While drones aid in search and rescue operations, they also pose additional risks to irregular migrants. The study raises important ethical and moral concerns about the impact of drone surveillance on migrants and refugees, while also discussing issues of transparency, accountability, and the blurred lines between civilian and military, public and private sectors in their use.

Multipurpose Modular Drone with Adjustable Arms

This paper [**42**] presents an innovative, multipurpose modular drone designed with adjustable arms, allowing for quick configuration changes. The core structure consists of a universal plate combined with a circular ring that forms a rail guide to accommodate between 3 to 8 arms and legs. Each arm contains all the necessary avionics and motor drivers, enabling various rotor configurations, such as single rotor, double rotor, and amphibious setups with inflatable elements. The drone's modularity allows for different configurations in terms of the number of arms, legs, rotors, motors, and landing capabilities, making it versatile and customizable. One key innovation is the use of FDM additive manufacturing (3D printing) for producing the structural elements, reducing production costs by requiring only a few universal components. The use of desktop 3D printers to produce these parts makes the drone platform both dynamic and cost-effective.

Multi-purpose Medical Drone in Pandemics

[47] This paper presents the design and fabrication of a multipurpose medical drone aimed at addressing challenges faced by hospitals during pandemics. The drone is capable of performing two main functions: sanitizing indoor spaces to reduce the risk of infection and delivering medicine, including vaccines, to hard-to-reach areas. The drone features a quadcopter configuration with four Brushless Direct Current (BLDC) engines and propellers, providing a thrust capacity of up to 5 kg at full power. It is powered by a Lithium-Polymer (Li-Po) battery with a capacity of 33,000 mAh and 11.1 V, meeting the necessary current and voltage requirements. This cost-effective system can enhance healthcare efficiency and minimize infection risks for frontline workers

Robotic Technologies for Forest Firefighting

[49] This paper surveys the use of robotic technologies, specifically drone swarms, to enhance forest firefighting operations. It discusses the challenges faced by firefighters in prevention, surveillance, and extinguishing tasks, and explores current technological efforts to support firefighting missions. The paper proposes a concept of operation where a fleet of quadcopters works together to perform surveillance, mapping, and monitoring tasks, which would be challenging for individual drones. The system defines three operator roles—mission commander, team leaders, and team members—each with specific access to information and functions. The operators utilize virtual and augmented reality interfaces to gain an intuitive understanding of the scenario, with the mission commander overseeing the drone swarm's activities. This comprehensive approach aims to improve firefighter efficiency and safety by leveraging swarm robotics**.**

SARDO: Search-and-Rescue Drone Solution

This paper introduces SARDO**[50]**, an autonomous, drone-based search-and-rescue solution designed to efficiently locate missing persons in disaster-stricken areas, especially where cellular networks are down. The system utilizes the high penetration rate of mobile phones in society, allowing SARDO to pinpoint the location of victims without requiring infrastructure support or phone modifications. By combining pseudo-trilateration with machine learning techniques, SARDO rapidly determines the location of mobile phones with an accuracy of a few tens of meters, while minimizing battery consumption. Field tests demonstrate SARDO's effectiveness, offering a novel and efficient alternative to traditional localization methods that rely on infrastructure or onboard imaging systems. SARDO is the first solution of its kind to enable accurate victim localization using mobile phones in a search-and-rescue context.

Dronaid: Human Detection Drone for Rescue

This paper introduces DronAID**[48]**, a real-time autonomous drone system designed for human detection in disaster scenarios, such as earthquakes, floods, and terrorist attacks. These disasters often lead to high death tolls, with victims trapped under debris or inaccessible to rescue teams. DronAID aims to address the critical challenge of locating survivors quickly by utilizing a drone-based system equipped with a camera module and sensors to detect human presence. The system's mobility and ease of control allow it to reach difficult or remote areas, sending data for further investigation and action. DronAID is expected to play a crucial role in accelerating rescue operations during large-scale urban disasters, offering timely assistance to save lives**.**

Drones in Agriculture: Review and Analysis

[51] This paper provides a comprehensive review and bibliometric analysis of the role of drones, or Unmanned Aerial Vehicles (UAVs), in agriculture. It highlights the significant advancements in drone technology and its impact on farming practices, including cost savings, increased efficiency, and improved profitability. The study analyzes existing academic literature on agricultural drones, identifying key research trends and hotspots, particularly in areas such as remote sensing, precision agriculture, deep learning, machine learning, and the Internet of Things (IoT). The bibliometric analysis uncovers six major research clusters, offering valuable insights into the current state of research and suggesting potential future directions for the field.

Drones for Smart Agriculture: Technical Report

[52] This paper discusses the design and development of an agricultural drone for pesticide spraying, addressing the challenges faced by Indian farmers due to natural factors like temperature, humidity, and rainfall, as well as issues like pests, diseases, and fertilizer use. The aim is to reduce the impact of pesticide use on human health while improving crop productivity. The paper explores various architectures of Unmanned Aerial Vehicles (UAVs) and highlights their potential for automating agricultural tasks, such as pest detection and pesticide application. It also covers the integration of a sprayer module into a quadcopter system, using cost-effective components like BLDC motors, Arduino, and ESC wires to create a prototype. The proposed solution offers an intelligent, automated approach to essential agricultural operations, thereby reducing human effort and improving efficiency.

Chapter Three:Methodology

Proposed Architecture

The quadcopter drone's system architecture is designed to enhance disaster response through the integration of IoT and machine learning technologies. The system comprises three primary units: the Input Unit, the Processing Unit, and the Output Unit, all working in cohesion to optimize performance in critical scenarios.

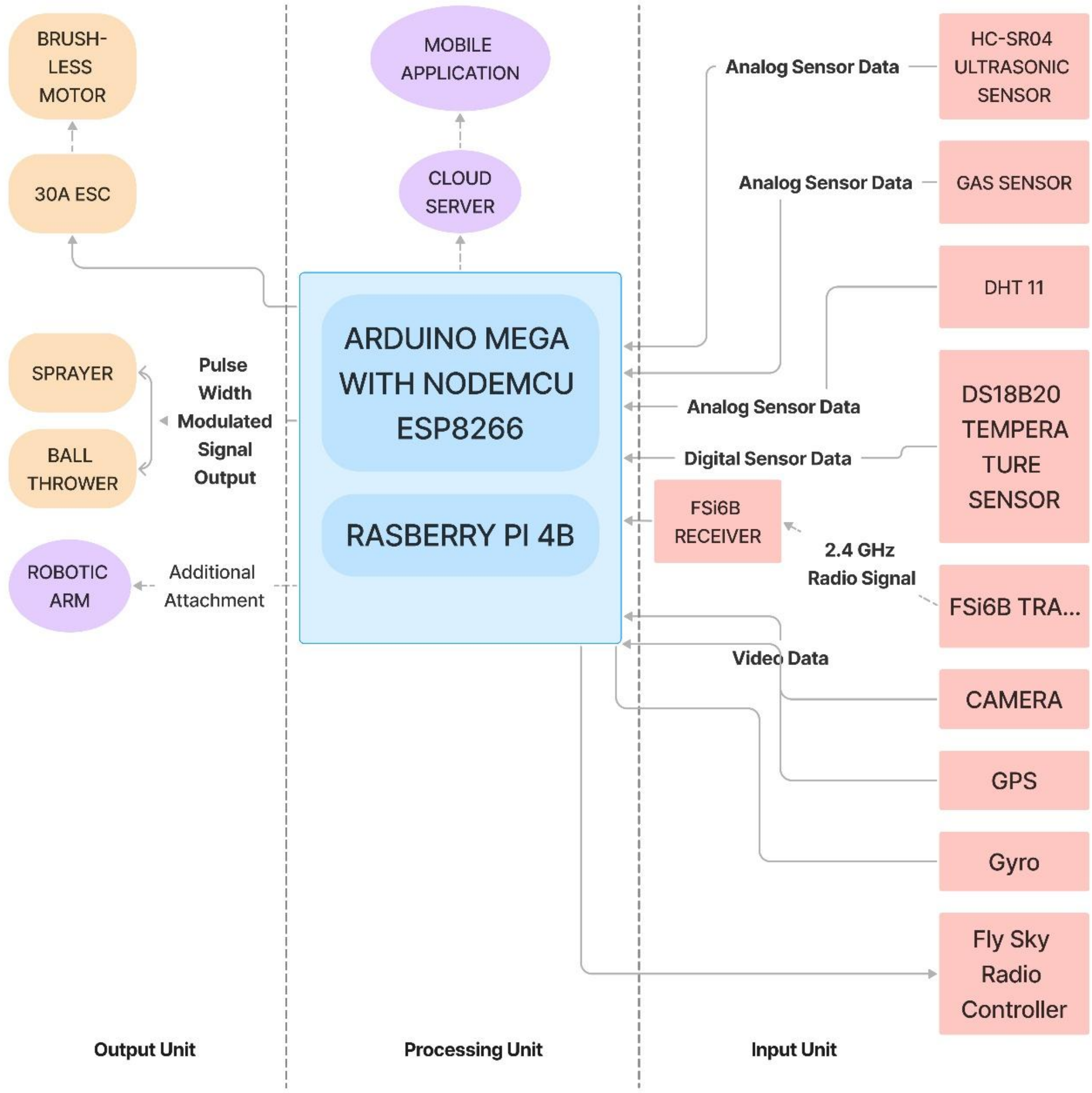


Figure 1: Block Diagram of the Drone

The central processing unit (CPU) of the quadcopter consists of an Arduino Mega microcontroller integrated with a Node MCU ESP8266 module for wireless communication and a Raspberry Pi 4B for advanced data processing and live video streaming. The live video feed, captured through an onboard camera, is transmitted via Wi-Fi to a mobile application, providing real-time visual data essential for disaster management.

Input Unit: The input unit includes various analog and digital sensors that continuously monitor environmental conditions. Key components are:

HC-SR04 Ultrasonic Sensor: Measures distance and aids in obstacle detection, ensuring safe navigation.
Gas Sensor: Detects hazardous gases in the environment, crucial during chemical spills or fires.

- **DHT11 Sensor:** Monitors temperature and humidity levels.
- **DS18B20 Temperature Sensor:** Provides precise temperature readings to assess environmental conditions.
- **GPS and Gyroscope (Gyro):** Facilitate accurate positioning and stability control.
- **Fly Sky Radio Controller with FSi6B Receiver:** Manages manual control inputs via a 2.4 GHz radio signal.

Sensor data is transmitted to the Arduino Mega, which processes the information and sends critical updates to a cloud server. This data can be accessed through the mobile application, enabling real-time decision-making.

Processing Unit: The Arduino Mega, in conjunction with the NodeMCU ESP8266, handles the core processing tasks. It manages data flow between sensors, the Raspberry Pi 4B, and the output mechanisms. The Raspberry Pi processes video data and facilitates machine learning algorithms for predictive analysis and pattern recognition in disaster scenarios.

Output Unit: The output unit comprises various actuators and devices controlled through Pulse Width Modulated (PWM) signal outputs:

- **Brushless Motors with 30A ESCs:** Provide propulsion and maneuverability.
- **Sprayer Mechanism:** Disperses water, fire retardants, or disinfectants as required.
- **Ball Thrower:** Deploys fire-extinguishing balls to targeted areas.
- **Robotic Arm:** Offers additional functionality for object manipulation, such as clearing debris or delivering supplies.

The drone's mobility is controlled via the Fly Sky FSi6 radio controller, with commands relayed to the FSi6B receiver and processed by the Arduino Mega. Adjustments to motor speeds allow for precise control of flight dynamics.

The integration of IoT enables seamless data exchange between the drone, cloud server, and mobile application. This connectivity, combined with machine learning algorithms, enhances the drone's capability to respond to disasters effectively, making it a vital tool for crisis management.

Chapter FOUR

Implementation

Electrical Design

The quadcopter's electrical architecture consists of multiple interconnected modules to ensure efficient power distribution, sensor data acquisition, and real-time video transmission. The core components include an **Arduino Mega**, a **Raspberry Pi 4B**, multiple sensors, motor controllers, and a **Skydroid 5.8G dual-antenna receiver** for live video streaming. The design ensures seamless communication between these modules while maintaining stable power distribution.

1.1Power Distribution and Motor Control

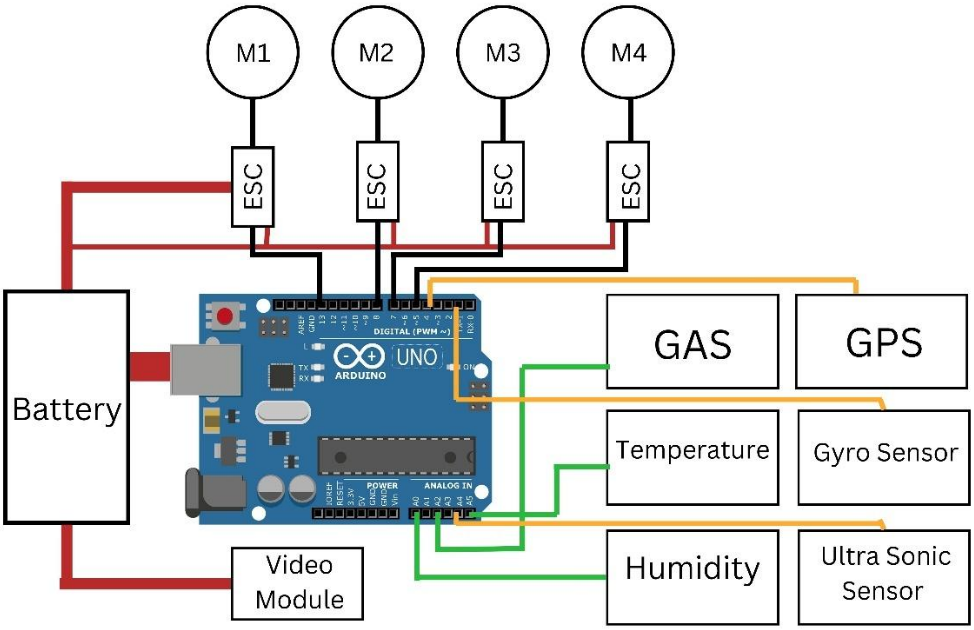


Figure 2: Motor and sensor wiring diagram with Arduino Mega.

The quadcopter is powered by a 3S 12V LiPo battery, which supplies power to all components. The 1400KV brushless motors are controlled via 30A Electronic Speed Controllers (ESCs) that regulate speed based on Pulse Width Modulation (PWM) signals from the Arduino Mega. The ESCs receive direct power from the LiPo battery and are connected to the microcontroller through digital PWM pins. The 12V-to-5V buck converter ensures that low-power components like sensors and the Raspberry Pi 4B receive appropriate voltage levels.

1.2.Sensor Integration and Data Processing

The quadcopter incorporates multiple sensors for environmental awareness and stability enhancement. The following key sensors are interfaced with the **Arduino Mega**:

- **Gas Sensor** – Detects hazardous gases in the environment.
- **DHT11 Humidity Sensor** – Measures humidity levels for environmental monitoring.
- **DS Temperature Sensor** – Provides temperature readings for situational awareness.
- **HC-SR04 Ultrasonic Sensor** – Used for obstacle detection and height estimation.
- **Gyro Sensor** – Assists in stabilization by detecting angular movement.
- **GPS Module** – Provides location tracking and navigation assistance.

1.3Real-Time Video Transmission

The quadcopter features a **dual-camera system** for real-time monitoring. The **Pi Camera v1.3** is connected to the **Raspberry Pi 4B**, which processes and transmits video through a **TS135 video transmitter**. A **Skydroid 5.8G dual-antenna receiver** is used to capture the video feed on a mobile device. This setup allows the controller to receive a live aerial view for navigation and risk assessment.

1.4.Electrical Stability and Communication

To ensure stable operation, the power distribution system is designed with proper voltage regulation. A common ground connection is maintained between the Arduino Mega, Raspberry Pi, sensors, and ESCs to prevent signal interference. The quadcopter also integrates a radio controller (FlySky 6S) for manual operation, allowing the user to override autonomous functions if needed.

2.Live object dection for emergency rescue:

***2.1Algorithm 1** YOLOv4-Based Live Object Detection*

```
function YOLOV4_LIVE_DETECTION()
    model ← LoadYOLO(weights, config)
    classLabels ← LoadCOCODataset()
    living_objects ← ["person", "bird", "cat", "dog", "horse", "sheep", "cow", "elephant", "bear", "zebra", "giraffe"]
    videoStream ← InitializeVideoCapture(webcam)
    while IsVideoOpen(videoStream) do
        frame ← ReadFrame(videoStream)
        preprocessedFrame ← PreprocessFrame(frame)
        SetYOLOInput(model, preprocessedFrame)
        outputLayers ← GetOutputLayers(model)
        detections ← ForwardPass(model, outputLayers)
        boxes, confidences, class_ids ← InitializeDetectionLists()
        for each output in detections do
            for each detection in output do
                confidence, class_id ← ExtractDetectionData(detection)
                if confidence > threshold and classLabels[class_id] in living_objects then
                    bbox ← ComputeBoundingBox(detection, frame)
                    Append(boxes, confidences, class_ids, bbox, confidence, class_id)
                end if
            end for
        end for
        filteredDetections ← ApplyNMS(boxes, confidences)
        for each validDetection in filteredDetections do
            DrawBoundingBox(frame, validDetection)
        end for
        DisplayFrame(frame)
        if UserPresses('q') then
            BreakLoop()
        end if
    end while
    ReleaseVideoStream(videoStream)
    DestroyAllWindows()
    return success
end function
```

Explanation.:

The YOLOV4_LIVE_DETECTION algorithm is designed to detect and classify living objects in real-time using the YOLOv4 deep learning model. The process begins by loading the YOLO model with pre-trained weights and configuration files, followed by importing COCO dataset class labels. A predefined list of living objects, including humans and various animals, is initialized to filter detections. The system then starts capturing video frames from a webcam. Each frame undergoes preprocessing, including resizing, normalization, and conversion into a format suitable for YOLO. The processed frame is passed through the YOLO model, where the network extracts detection outputs. The algorithm iterates through each detected object, filtering out low-confidence detections and retaining only those that belong to the predefined living objects category. Bounding boxes, confidence scores, and class IDs are stored for further processing. To eliminate redundant or overlapping detections, the Non-Maximum Suppression (NMS) technique is applied. Valid detections are then used to draw bounding boxes and labels on the frame, highlighting detected objects. The processed frame is displayed in real-time, and the system continues processing frames until the user presses the ‘q’ key, which terminates the loop. Finally, the video stream is released, and all display windows are closed. This algorithm ensures efficient and accurate real-time obdetectionmaking it useful for applications such as surveillance, wildlife monitoring

3.1Algorithm for Environmental Risk Assessment

```
function ENVIRONMENTAL_RISK_ASSESSMENT()
    Initialize sensors for gas, temperature, humidity, and air quality
    Load pre-trained machine learning model for risk prediction
    Define risk thresholds for different environmental hazards
    Start real-time data collection loop
    while system is running do
        Read sensor data (gas concentration, temperature, humidity, air quality)
        Preprocess the data (normalize, remove noise, feature extraction)
        Input the processed data into the machine learning model
        Predict risk level ← (LOW, MEDIUM, HIGH)
        if risk level == HIGH then
            Trigger emergency response system
            if gas levels hazardous then
                Alert authorities
                Activate ventilation system
            end if
            if fire risk detected then
                Deploy Carbon Ball for fire suppression
            end if
            if air pollution exceeds threshold then
                Recommend evacuation/mask usage
            end if
            if explosion risk detected then
                Shut down power sources
                Signal emergency response team
            end if
        else if risk level == MEDIUM then
            Log data and monitor closely
            Send warning notifications
        else
            Continue normal monitoring
        end if
        Send data logs to cloud dashboard for visualization and analysis
    end while
    if system receives shutdown signal then
        Terminate data collection
        Safely close all sensor connections
    end if
end function
```

The **ENVIRONMENTAL_RISK_ASSESSMENT** algorithm is designed to assess environmental hazards using real-time sensor data and machine learning predictions. It begins by initializing sensors for gas concentration, temperature, humidity, and air quality measurements. A pre-trained machine learning model is loaded to analyze the collected data and predict risk levels. Risk thresholds are defined to classify environmental conditions into LOW, MEDIUM, or HIGH danger levels. The system operates in a continuous loop, reading sensor data, preprocessing it by normalizing values and extracting features, and then feeding it into the model for risk evaluation. If the predicted risk level is HIGH, emergency response actions are triggered based on the specific hazard detected. This includes alerting authorities in case of hazardous gas levels, deploying a carbon ball for fire suppression if a fire risk is identified, recommending evacuation or mask usage in case of extreme air pollution, and shutting down power sources if an explosion risk is detected. If the risk level is MEDIUM, the system logs data, monitors conditions closely, and sends warning notifications. For LOW risk levels, normal monitoring continues. Data logs are sent to a cloud dashboard for visualization and further analysis. If the system receives a shutdown signal, it safely terminates data collection and closes all sensor connections. This structured approach ensures real-time hazard detection, timely intervention, and comprehensive data management for enhanced environmental safety.

3.2Algorithm 3 Risk Scoring Algorithm

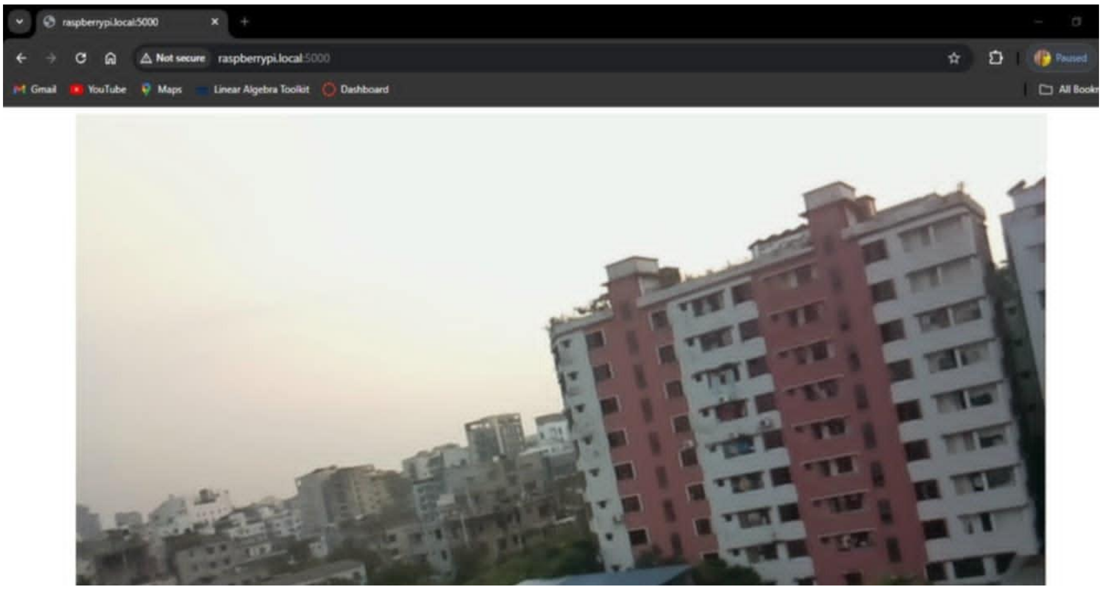
```
function RISKSCORE(sensors)
    totalRisk ← 0
    for each sensor in sensors do
        Define Xmin, Xmax for sensor
        Sscaled ← (sensor.value − Xmin) / (Xmax − Xmin)
        sensor.scaled ← Sscaled
    end for
    for each sensor in sensors do
        totalRisk ← totalRisk + (sensor.scaled × sensor.weight)
    end for
    if totalRisk ≤ 0.3 then
        riskLevel ← "Risk Level 1"
    else if totalRisk ≤ 0.6 then
        riskLevel ← "Risk Level 2"
    else if totalRisk ≤ 0.9 then
        riskLevel ← "Risk Level 3"
    else
        riskLevel ← "Risk Level 4"
    end if
    TRIGGERACTION(riskLevel)
    return totalRisk, riskLevel
end function
```

The **Risk Scoring Algorithm** assesses the risk level of a drone by analyzing its sensor data. First, it normalizes each sensor's readings using predefined minimum and maximum thresholds, scaling values between **0 and 1**. Each sensor's contribution is then weighted, and their weighted sum determines the **total risk score**. The algorithm classifies the risk into one of four levels:

- **Risk Level 1 (0.0 - 0.3):** Minimal risk, normal operation.
- **Risk Level 2 (0.3 - 0.6):** Moderate risk, potential caution needed.
- **Risk Level 3 (0.6 - 0.9):** High risk, intervention required.
- **Risk Level 4 (0.9 - 1.0):** Critical risk, emergency response needed.

The determined risk level is sent to the controller as a **popup message**, guiding necessary actions such as adjusting flight paths, returning to base, hovering, or performing an emergency landing.

4.Live Streaming for findigns fire,and victims:



4.1Live Video Streaming from Drone Camera Using Raspberry Pi:

During the flight operation of the drone, real-time video feed was successfully captured from the onboard camera and streamed directly to a local device via a web browser. This was accomplished using a Raspberry Pi as the intermediary host, which processed the video data and served it through a locally hosted Flask web server.The interface displayed in the browser (as seen in the screenshot) shows a live stream at the URL http://raspberrypi.local:5000. The "Not secure" label and the .local domain indicate that this server is running on the local network rather than over the public internet, enhancing both simplicity and privacy during development and testing phases.In this instance, the video feed shows a city skyline at sunset, streamed directly to the PC. The Raspberry Pi is interfaced with a camera module which continuously captures frames. These frames are then encoded and sent over the network using HTTP streaming, allowing the user to view the footage in real time via any web browser. This setup not only allows for monitoring the drone's flight path and environment but also demonstrates a lightweight, accessible method for edge-computing-based video transmission.This functionality is a key component of the thesis project, as it validates the feasibility of using embedded systems and local networking for live aerial surveillance, environmental monitoring, or general drone-based observation tasks without the need for cloud-based infrastructure

4.2Real-Time Drone Sensor Data Monitoring via Mobile Application

This mobile application displays real-time environmental and system data collected from various sensors mounted on the drone. The battery level is monitored using a voltage sensor to ensure flight safety. Air quality data (135.28 PPM) is captured using an MQ135 gas sensor, which detects harmful gases in the atmosphere. Temperature (32.52°C) and humidity (68%) are measured using a DHT11 sensor. The obstacle status, showing “Obstacle Ahead,” is detected by an ultrasonic sensor (HC-SR04) that measures the distance to nearby objects. All sensor data is read by a microcontroller (such as an Arduino or Raspberry Pi) onboard the drone, processed, and then wirelessly transmitted to the mobile application using Wi-Fi or Bluetooth. The app fetches this data through an API (e.g., Flask or Firebase) and presents it in a clean, user-friendly interface for real-time monitoring.

5.Enhancing drone stability using (RL algorithm) :

Deep Deterministic Policy Gradient (DDPG)

```python
def train():

    n_actions = env_no_gui_train.action_space.shape[-1]
    action_noise = NormalActionNoise(mean=np.zeros(n_actions), sigma=0.1 * np.ones(n_actions))

    train_env = make_vec_env(QuadEnv,
                             env_kwargs= dict(REAL_TIME= False, GUI= False, FLOOR = True, ONE_COMMAND= True, MODE= 'TakeOFF', EPS_TIME= 20),
                             n_envs= 1)

    callback_on_best = StopTrainingOnRewardThreshold(reward_threshold=-0,
                                                     verbose=1
                                                     )
    eval_callback = EvalCallback(env_no_gui_train, #env_no_gui_eval
                                 callback_on_new_best=callback_on_best,
                                 verbose=1,
                                 eval_freq=int(500),
                                 deterministic=True,
                                 render=False
                                 )
    model = DDPG("MlpPolicy", train_env, verbose= 1, policy_kwargs= dict(activation_fn=torch.nn.ReLU,net_arch=[512, 512, 256, 128]), action_noise=action_noise)

    model.learn(total_timesteps=1e5, callback= eval_callback)

    model.save("DDPG_TakeOFF")

    del model

    env_no_gui_train.close()

    end_time = time.time()
    print(f'Total Training Time for {1e5} steps is {end_time - start_time}')

def test_n_times(n):
    model = DDPG.load("DDPG_TakeOFF")

    env = QuadEnv(REAL_TIME= False, GUI= False, ONE_COMMAND= True, MODE= 'TakeOFF', EPS_TIME= 20)
    reward_list = []
    for i in range(n):
        obs = env.reset()
        rewards = 0
        while True:
            action, _states = model.predict(obs)
            obs, reward, dones, info = env.step(action)
            rewards += reward
            if(dones):
                break
        reward_list.append(rewards)
    env.close()

    print(f'{np.mean(reward_list)=}')

def test():
    model = DDPG.load("DDPG_TakeOFF")

    env_gui = QuadEnv(REAL_TIME= False, ONE_COMMAND= True, MODE= 'TakeOFF', EPS_TIME= 20)
    obs = env_gui.reset()
    rewards = 0
    while True:
        action, _states = model.predict(obs)
        obs, reward, dones, info = env_gui.step(action)
        rewards += reward
        if(dones):
            break
    env_gui.visualize_logs()
    env_gui.save_logs()
    env_gui.close()
```

Explanation :

The algorithm uses the Deep Deterministic Policy Gradient (DDPG) method to train a drone in a custom environment (QuadEnv) to perform a takeoff task. It first initializes a simulation environment without GUI for faster training and applies action noise to encourage exploration in the continuous action space. A DDPG model with a multilayer perceptron policy is configured and trained over 100,000 timesteps. An evaluation callback monitors the reward and stops training once a target threshold is met. After training, the model is saved and later tested in two ways: in multiple headless (no-GUI) episodes to evaluate average performance, and in a GUI-enabled mode for visual inspection and logging.

The model predicts actions in each timestep to control the drone, accumulating rewards until the episode ends.To maintain drone stability, the agent learns to continuously balance thrust and orientation through trial-and-error feedback from the environment. As it maximizes reward, it minimizes oscillation and drift, resulting in smoother and controlled takeoff behavior.

PID Algorithm:

pid.py

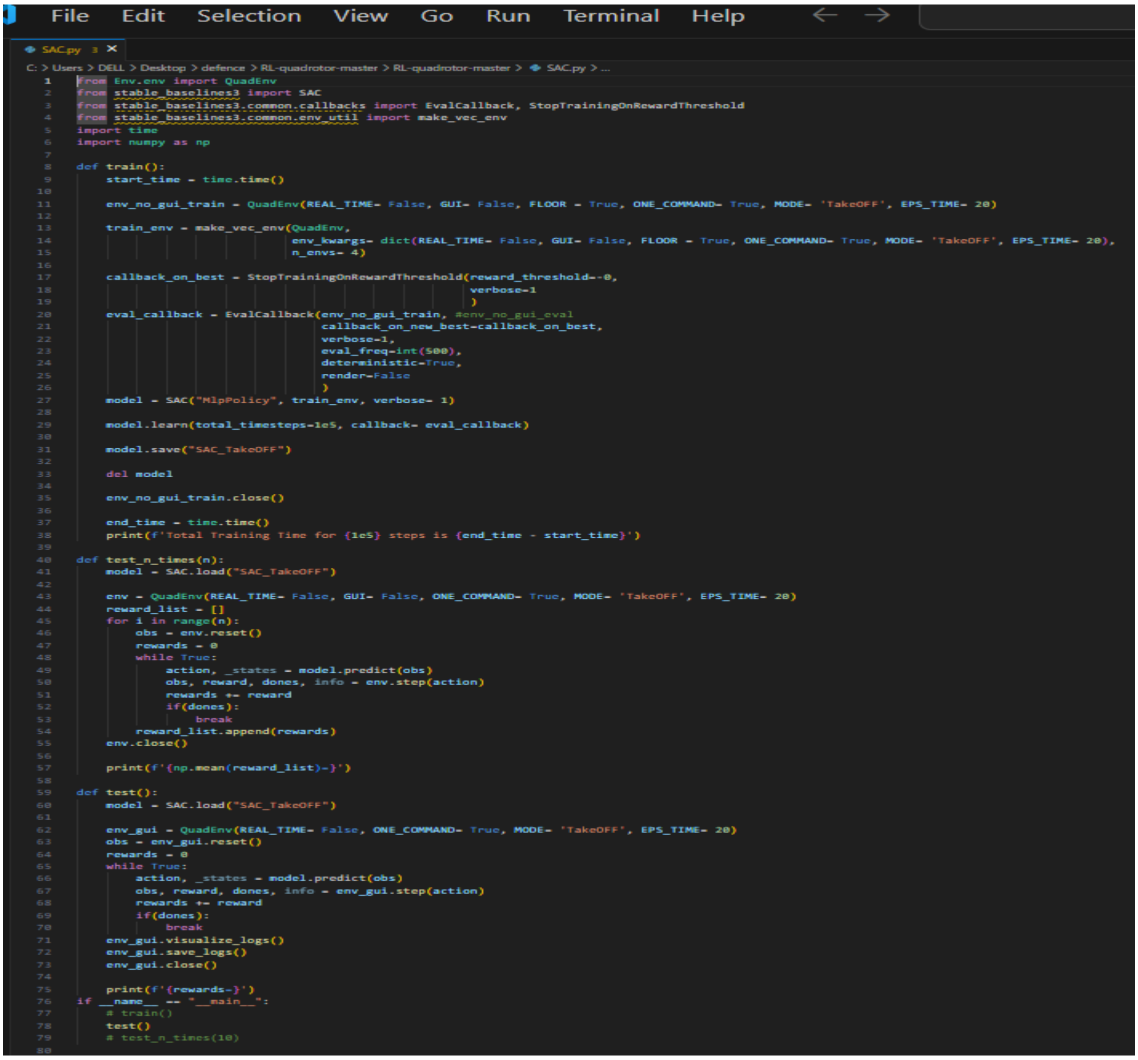
C: > Users > DELL > Desktop > defence > RL-quadrotor-master > RL-quadrotor-master > pid.py > ...

```python
from Agents.pid import PIDAgent
from Env.env import QuadEnv
import numpy as np

if __name__ == "__main__":
    env = QuadEnv(REAL_TIME=False, MODE= 'TakeOFF', EPS_TIME= 20)
    agent = PIDAgent()

    obs = env._getObservation()

    rewards = 0
    while True:

        action = agent.calculate_rpm(obs[0:3], obs[3:6], env.DesiredPos, [0,0,0], obs[6:9])
        obs, reward, done, info = env.step(np.array(action), rpm= True)
        rewards += reward

        # print(action)
        if(done): break

    env.visualize_logs()
    env.save_logs()
    env.close()

    print(f'{rewards=}')
```

EXPLANATION:

This script uses a PID-based control agent to perform a takeoff in the QuadEnvironment drone simulation. The agent receives the drone's current position, velocity, and angular rates, and computes motor RPM values to guide it toward the desired position. These RPM actions are applied in each simulation step until the episode ends. After completion, the logs are visualized and saved.
The PID controller maintains drone stability by continuously minimizing the error between the current and desired state, adjusting motor speeds to reduce overshoot and ensure smooth lift-off.

Soft Actor-Critic (SAC) Algorithm :



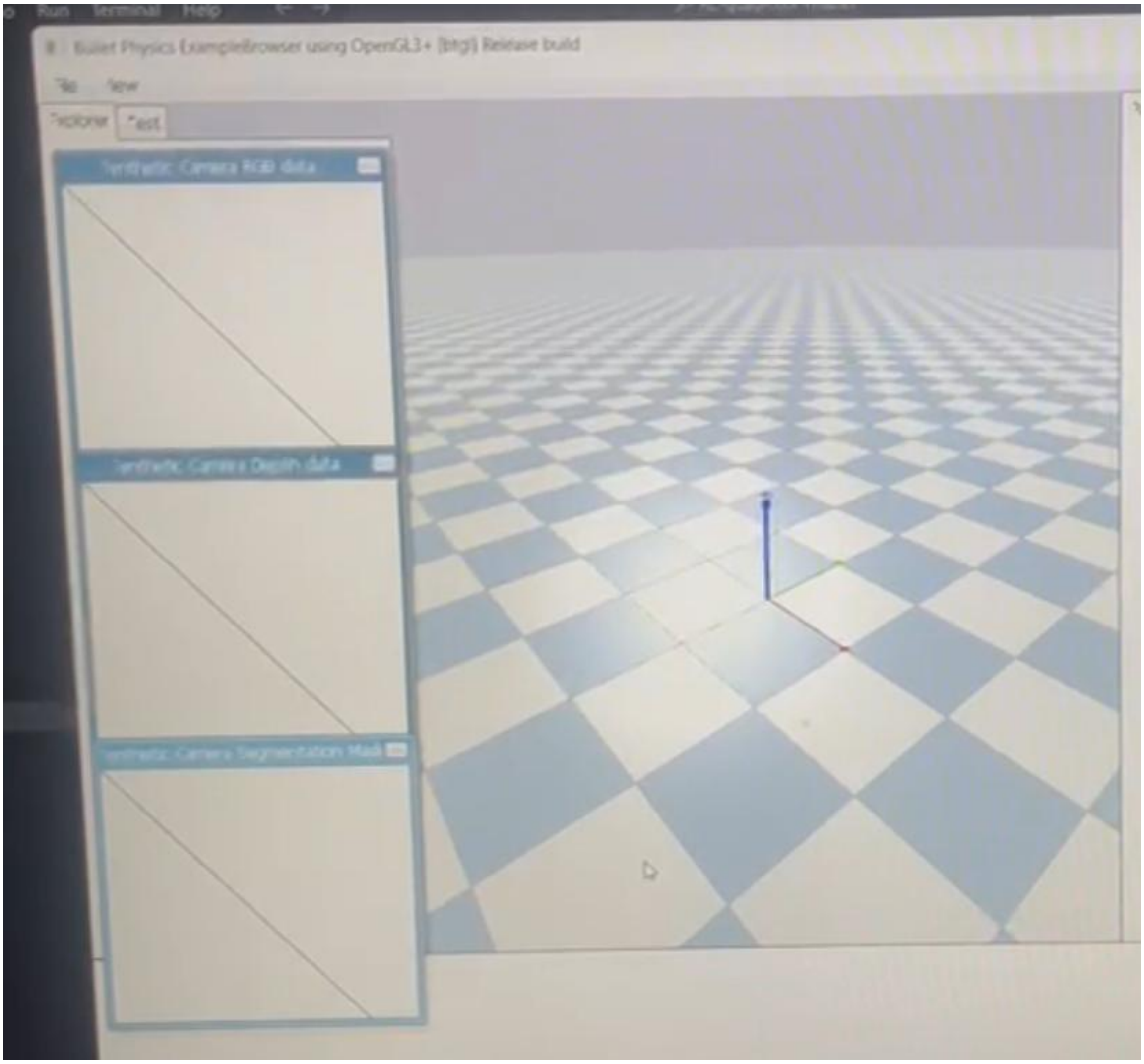
```python
from Env.env import QuadEnv
from stable_baselines3 import SAC
from stable_baselines3.common.callbacks import EvalCallback, StopTrainingOnRewardThreshold
from stable_baselines3.common.env_util import make_vec_env
import time
import numpy as np

def train():
    start_time = time.time()

    env_no_gui_train = QuadEnv(REAL_TIME= False, GUI= False, FLOOR = True, ONE_COMMAND= True, MODE= 'TakeOFF', EPS_TIME= 20)

    train_env = make_vec_env(QuadEnv,
                             env_kwargs= dict(REAL_TIME= False, GUI= False, FLOOR = True, ONE_COMMAND= True, MODE= 'TakeOFF', EPS_TIME= 20),
                             n_envs= 4)

    callback_on_best = StopTrainingOnRewardThreshold(reward_threshold=-0,
                                                     verbose=1
                                                     )
    eval_callback = EvalCallback(env_no_gui_train, #env_no_gui_eval
                                 callback_on_new_best=callback_on_best,
                                 verbose=1,
                                 eval_freq=int(500),
                                 deterministic=True,
                                 render=False
                                 )
    model = SAC("MlpPolicy", train_env, verbose= 1)

    model.learn(total_timesteps=1e5, callback= eval_callback)

    model.save("SAC_TakeOFF")

    del model

    env_no_gui_train.close()

    end_time = time.time()
    print(f'Total Training Time for {1e5} steps is {end_time - start_time}')

def test_n_times(n):
    model = SAC.load("SAC_TakeOFF")

    env = QuadEnv(REAL_TIME= False, GUI= False, ONE_COMMAND= True, MODE= 'TakeOFF', EPS_TIME= 20)
    reward_list = []
    for i in range(n):
        obs = env.reset()
        rewards = 0
        while True:
            action, _states = model.predict(obs)
            obs, reward, dones, info = env.step(action)
            rewards += reward
            if(dones):
                break
        reward_list.append(rewards)
    env.close()

    print(f'{np.mean(reward_list)=}')

def test():
    model = SAC.load("SAC_TakeOFF")

    env_gui = QuadEnv(REAL_TIME= False, ONE_COMMAND= True, MODE= 'TakeOFF', EPS_TIME= 20)
    obs = env_gui.reset()
    rewards = 0
    while True:
        action, _states = model.predict(obs)
        obs, reward, dones, info = env_gui.step(action)
        rewards += reward
        if(dones):
            break
    env_gui.visualize_logs()
    env_gui.save_logs()
    env_gui.close()

    print(f'{rewards=}')
if __name__ == "__main__":
    # train()
    test()
    # test_n_times(10)
```

EXPLANATION:

This script uses the Soft Actor-Critic (SAC) algorithm to train a drone in a simulated environment for the TakeOFF task. The environment is initialized without GUI to speed up training, and a parallelized setup with 4 environments is used for efficiency. A reward-based early stopping callback monitors performance during training. The model learns over 100,000 timesteps and is saved as "SACTakeOFF". After training, the model can be tested in two ways: across multiple headless episodes for average reward evaluation, or in a GUI-enabled simulation for visual inspection and logging. During each test episode, the model predicts continuous actions to control the drone based on current observations until the episode ends.

SAC ensures drone stability by continuously optimizing a stochastic policy that balances exploration and exploitation, leading to smooth, adaptive motor commands that minimize oscillations during takeoff

Testing Drone in virtual Environment for more stability:



Chapter Five

Result Analysis

This section presents a detailed analysis of the data collected by the **Quadcopter in Crisis** during field testing, highlighting its environmental sensing accuracy and response to controller inputs. The prototype drone currently delivers a flight duration of approximately **15 minutes** and supports a payload of **1.8 to 2.5 kg**. With future enhancements, the target is to achieve **25–30 minutes** of flight time with an expanded payload capacity of **3.5 to 4 kg**. The drone is capable of flying at a maximum altitude of **250 feet**, making it suitable for both close-range and elevated surveillance missions. The current operational range of **1.5 kilometers** is expected to reach **5 kilometers** with upgraded transmission modules.

Table I
Channel Triggers and Corresponding Drone Actions

Channel	**Trigger**	**Drone's Action / Movement**
Channel 1	Right	Right Movement
Channel 1	Left	Left Movement
Channel 2	Up	Vertical Lift
Channel 3	Up	Acceleration
Channel 3	Down	Deceleration
Channel 4	Right	Grab (Activate Robotic Arm)
Channel 4	Left	Release (Deactivate Robotic Arm)
Channel 5	Clockwise Rotation	Activates Sprayer
Channel 6	Clockwise Rotation	Circular Hovering Motion

Table I demonstrates the response behavior of the Quadcopter in Crisis based on different channel movements from the FSi86 radio controller. Channel 1 controls lateral motion—pushing the stick left reduces the speed of the left-side motors and increases the right-side motors, resulting in a leftward shift of the drone. Conversely, moving the stick right causes a rightward shift. Channel 2 controls vertical lift; pushing the stick upward increases throttle, producing downward thrust from the propellers, thus lifting the quadcopter vertically. Channel 3 regulates acceleration and deceleration, managing the drone's forward speed and slowdown. Channel 4 controls the robotic arm servo motor—a right stroke engages the gripper to grab an object, while a left stroke releases it.

Furthermore, Channel 5 initiates the sprayer motor, enabling the quadcopter to disperse liquids or dry materials such as pesticides, water, or carbon powder. Channel 6, when rotated clockwise, causes the drone to perform a controlled circular hovering motion at a fixed point, useful for target-centric spraying or observation.

Table II
Obstacle Distance, Risk Level, and Suggestions

Obstacle Distance (cm)	Risk Level	Suggestions Displayed on Mobile Application
101.50	Safe	Safe
210.99	Safe	Safe
30.12	Emergency	Avoid Obstacle
50.34	Emergency	Avoid Obstacle
351.73	Safe	Safe
76.92	Moderate	Obstacles Ahead

The Quadcopter in Crisis employs HC-SR04 ultrasonic sensors to detect nearby obstacles during flight. These sensors emit ultrasonic waves at a frequency of 40 kHz and receive the echo signal reflected from nearby objects. The distance to the obstacle is calculated using the following formula:

$$\text{Distance} = 0.5\,(\text{Speed of ultrasound wave} \times \text{Time})$$

Based on the calculated distance, the drone dynamically classifies the risk level and sends appropriate warnings to the mobile application interface. If the distance is less than 50 cm, it is considered an emergency threat, and the system displays "Avoid Obstacle." If the distance exceeds 100 cm, the drone deems the path safe, displaying "Safe" on the application. For distances between 50 cm and 100 cm, the drone treats it as a moderate risk, displaying an "Obstacle Ahead" alert to inform the user.

Air Quality Analysis

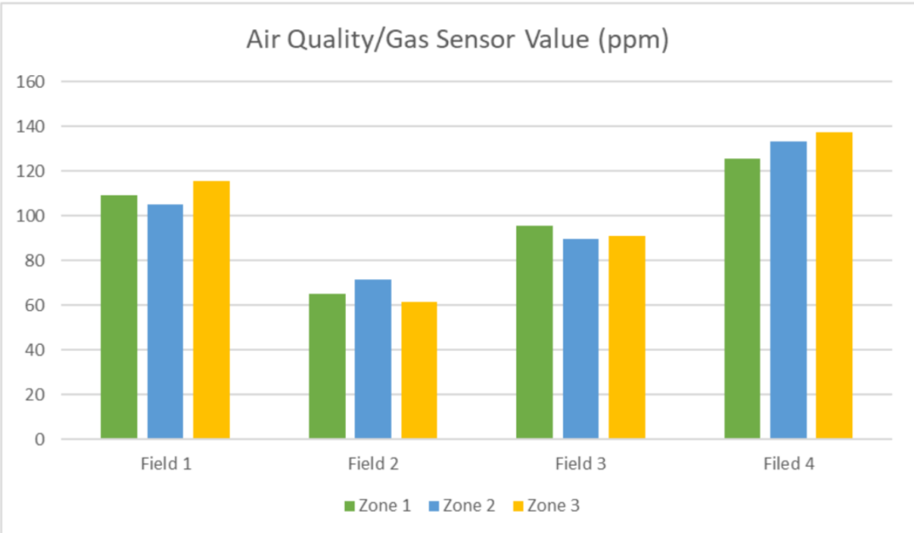


Figure 3 Air Quality / Gas sensor

The drone collects **gas sensor data** in parts per million (PPM), reflecting air contamination levels. In test zones, the highest value recorded was **131.67 PPM**, indicating severe pollution. The cleanest zone recorded **65.33 PPM**, while intermediate areas averaged around **92–111 PPM**. This sensing capability is crucial for deployment in fire zones, chemical accident sites, or collapsed structures.

Temperature Monitoring

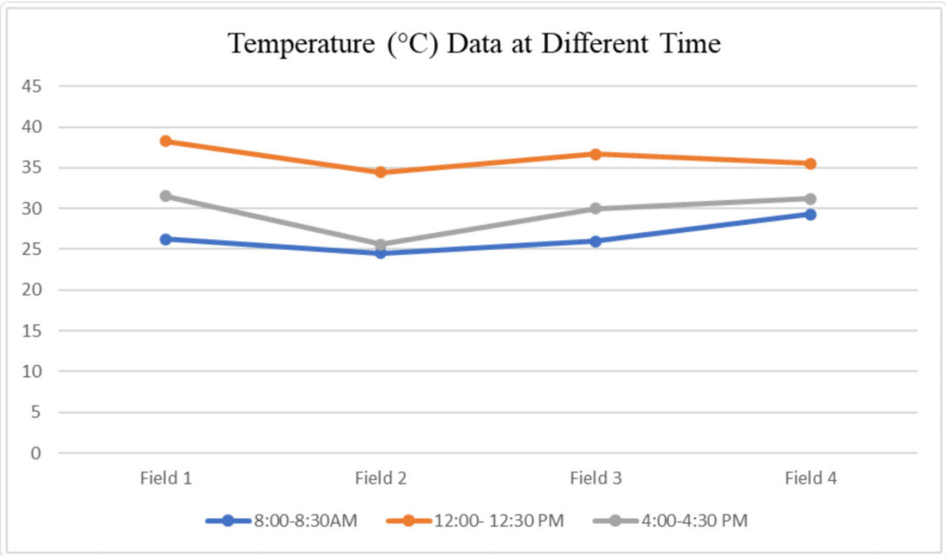


Figure 4 Temperature data at different times

The drone captured temperature readings at **three time intervals**. In the **morning (8:00–8:30 AM)**, the temperature ranged from **35°C to 40°C**. Around **midday (12:00–12:30 PM)**, it decreased to **27°C–32°C**, and in the **afternoon (4:00–4:30 PM)**, it stabilized between **26°C and 29°C**. This data is vital for assessing heat exposure risks in environments such as fire-affected zones or unstable debris fields.

Flight Stability Evaluation

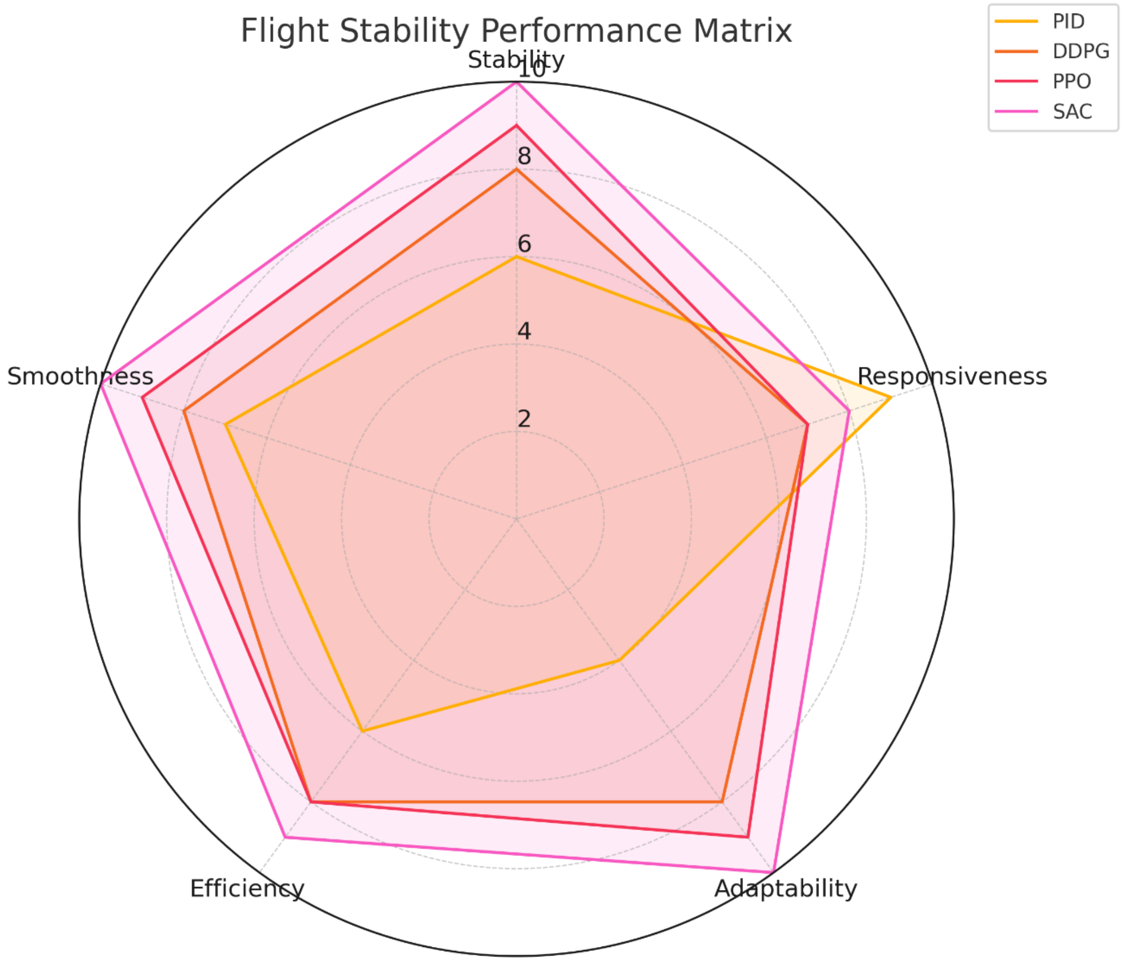


Figure 5 Flight Stability performance matrix

To assess the drone's flight stability under various control strategies, a performance matrix was developed, focusing on five key criteria: stability, responsiveness, adaptability, efficiency, and smoothness. The radar chart (see Fig. X) compares the results of traditional PID control with modern reinforcement learning algorithms—DDPG, PPO, and SAC—used during simulated training in environments such as QuadEnv. The SAC algorithm demonstrated the highest overall stability and adaptability, making it best suited for dynamic or unpredictable rescue scenarios. PPO showed strong performance in smoothness and environmental adaptability, while DDPG offered a balanced but moderately responsive behavior. In contrast, the PID controller, though highly responsive, lacked the adaptability required for real-time disturbances. These insights reinforce the suitability of RL-based controllers for future autonomous drone deployment in high-risk or unstable environment

Chapter Six

Conclusion and Future Work

The development of an AI-powered quadcopter for disaster response and emergency relief represents a substantial leap forward in the field of modern emergency management. By integrating real-time data processing, autonomous navigation, and robotic capabilities, this system significantly enhances the efficiency and effectiveness of emergency teams. It reduces response times and bolsters situational awareness in critical scenarios, which are pivotal for successful disaster response. This quadcopter's versatility in operating within hazardous and hard-to-reach areas positions it as an indispensable tool for search and rescue operations, firefighting, and environmental monitoring. Its ability to utilize advanced AI algorithms, sensor fusion, and adaptive flight control systems enables it to successfully navigate through challenging environments, locate survivors, assess damage, and deliver emergency supplies with unparalleled accuracy. Moreover, the integration of machine learning-driven object recognition ensures precise threat detection and classification, making it an invaluable asset for disaster mitigation and response efforts. The system's capacity to function autonomously or in coordination with human operators provides flexible deployment options, allowing emergency responders to focus on critical decision-making instead of manual drone operation.

Building on the foundation of this prototype, several enhancements can be pursued to transform the drone into a fully autonomous, intelligent, and robust disaster-response system. These include full autonomy implementation through advanced SLAM, GPS-denied navigation, and AI-based obstacle avoidance for independent mapping and decision-making; propeller guard integration to enhance safety in confined or unstable environments; and an autonomous robotic arm powered by computer vision and AI control for identifying, grasping, and manipulating objects during debris removal, supply delivery, or rescue tasks. Looking forward, future iterations of this AI-powered quadcopter could also integrate swarm intelligence, extended flight endurance, and enhanced real-time communication systems, further refining its capabilities and boosting disaster response effectiveness. Collaboration with emergency response agencies, government bodies, and NGOs will play a vital role in optimizing the quadcopter's features and ensuring its successful deployment in real-world disaster scenarios. Ultimately, this research sets the stage for a new era of intelligent UAVs that can play a transformative role in saving lives, minimizing the impact of disasters, and strengthening global emergency preparedness.

Chapter Seven

Reference

1. Kumaran, S., Raj, V. A., & Raman, V. R. (2023). IoT-based Autonomous Search and Rescue Drone for Precision Firefighting and Disaster Management. International Journal of Advanced Computer Science & Applications, 14(11).
2. Papyan, N., Kulhandjian, M., Kulhandjian, H., & Aslanyan, L. (2024). AI-Based Drone Assisted Human Rescue in Disaster Environments: Challenges and Opportunities. Pattern Recognition and Image Analysis, 34(1), 169-186
3. Damaševičius, R., Bacanin, N., & Misra, S. (2023). From sensors to safety: Internet of Emergency Services (IoES) for emergency response and disaster management. Journal of Sensor and Actuator Networks, 12(3), 41
4. Paik, C., & Kim, H. J. (2022). Improving object detection, multi-object tracking, and re-identification for disaster response drones. *arXiv preprint arXiv:2201.01494*.
5. Merkle, N., Bahmanyar, R., Henry, C., Azimi, S. M., Yuan, X., Schopferer, S., ... & Kraft, T. (2023). Drones4Good: Supporting Disaster Relief Through Remote Sensing and AI. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 3770-3774
6. Mostufa, S. (2020). IoT Based Facial Recognition Quadcopter Using Machine Learning Algorithm
7. Remington, R., Cordero, R., March, L., & Villanueva, D. (2014). Multi-purpose aerial drone for bridge inspection and fire extinguishing. Unpublished Thesis). Florida International University. Retrieved April, 10, 2016.
8. Jahani, H., Khosravi, Y., Kargar, B., Ong, K. L., & Arisian, S. (2024). Exploring the role of drones and UAVs in logistics and supply chain management: a novel text-based literature review. International Journal of Production Research, 1-25.
9. D. Zhang, A. Loquercio, J. Tang, T. -H. Wang, J. Malik and M. W. Mueller, "A Learning-Based Quadcopter Controller With Extreme Adaptation," in *IEEE Transactions on Robotics*, vol. 41, pp. 3948-3964, 2025, doi: 10.1109/TRO.2025.3577037
10. Zhang, D., Loquercio, A., Tang, J., Wang, T. H., Malik, J., & Mueller, M. W. (2024). A Learning-based Quadcopter Controller with Extreme Adaptation. arXiv preprint arXiv:2409.12949.
11. Bian, J., Al Arafat, A., Xiong, H., Li, J., Li, L., Chen, H., ... & Guo, Z. (2022). Machine learning in real-time Internet of Things (IoT) systems: A survey. IEEE Internet of Things Journal, 9(11), 8364-8386.
12. Hadid, S., Boushaki, R., Boumchedda, F., & Merad, S. (2024). Enhancing Quadcopter Autonomy: Implementing Advanced Control Strategies and Intelligent Trajectory Planning. Automation, 5(2), 151-175.

13. Li, W., Yang, F., Zhong, L., Wu, H., Jiang, X., & Chukalin, A. V. (2023). Attitude control of UAVs with search optimization and disturbance rejection strategies. Mathematics, 11(17), 3794.
14. Leal, I. S., Abeykoon, C., & Perera, Y. S. (2021). Design, simulation, analysis and optimization of PID and fuzzy based control systems for a quadcopter. Electronics, 10(18), 2218.
15. Zhou, L., Pljonkin, A., & Singh, P. K. (2022). Modeling and PID control of quadrotor UAV based on machine learning. Journal of Intelligent Systems, 31(1), 1112-1122
16. Abeywickrama HG, Bajón-Fernández Y, Srinamasivayam B, et al., (2023) Development of a UAV based framework for CH4 monitoring in sludge treatment centres, Remote Sensing, Volume 15, Issue 15, July 2023, Article Number 3704
17. Hoang, M. L. (2023). Smart Drone Surveillance System Based on AI and on IoT Communication in Case of Intrusion and Fire Accident. *Drones*, *7*(12), 694. https://doi.org/10.3390/drones7120694
18. Titu, M. F. S., Pavel, M. A., Michael, G. K. O., Babar, H., Aman, U., & Khan, R. (2024). Real-Time Fire Detection: Integrating Lightweight Deep Learning Models on Drones with Edge Computing. *Drones*, *8*(9), 483. https://doi.org/10.3390/drones8090483
19. Titu, M. F. S., Pavel, M. A., Michael, G. K. O., Babar, H., Aman, U., & Khan, R. (2024). Real-Time Fire Detection: Integrating Lightweight Deep Learning Models on Drones with Edge Computing. Drones, 8(9), 483
20. Do, T. D., Mung, N. X., & Hong, S. K. (2024). Deep Reinforcement Learning-based Quadcopter Controller: A Practical Approach and Experiments. arXiv preprint arXiv:2406.08815.
21. Mariani, M., & Fiori, S. (2023). Design and Simulation of a Neuroevolutionary Controller for a Quadcopter Drone. Aerospace, 10(5), 418.
22. Panerati, J., Zheng, H., Zhou, S., Xu, J., Prorok, A., & Schoellig, A. P. (2021, September). Learning to fly—a gym environment with pybullet physics for reinforcement learning of multi-agent quadcopter control. In 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 7512-7519). IEEE.
23. Foehn, P., Kaufmann, E., Romero, A., Penicka, R., Sun, S., Bauersfeld, L., ... & Scaramuzza, D. (2022). Agilicious: Open-source and open-hardware agile quadrotor for vision-based flight. Science robotics, 7(67), eabl6259
24. Shah, I. A., Laraib, A., Ashraf, H., & Hussain, F. (2024). Drone Technology: Current Challenges and Opportunities. Cybersecurity Issues and Challenges in the Drone Industry, 343-361.
25. Emimi, M., Khaleel, M., & Alkrash, A. (2023). The current opportunities and challenges in drone technology. Int. J. Electr. Eng. and Sustain., 74-89.
26. Nonami, K. (2016). Drone technology, cutting-edge drone business, and future prospects. Journal of Robotics and Mechatronics, 28(3), 262-272.
27. Lee, S., & Choi, Y. (2016). Reviews of unmanned aerial vehicle (drone) technology trends and its applications in the mining industry. Geosystem Engineering, 19(4), 197-204.
28. Mohsan, S. A. H., Zahra, Q. U. A., Khan, M. A., Alsharif, M. H., Elhaty, I. A., & Jahid, A. (2022). Role of drone technology helping in alleviating the COVID-19 pandemic. Micromachines, 13(10), 1593.

29. Moon, H., Martinez-Carranza, J., Cieslewski, T., Faessler, M., Falanga, D., Simovic, A., ... & Kim, S. J. (2019). Challenges and implemented technologies used in autonomous drone racing. Intelligent Service Robotics, 12, 137-148.
30. Meesaragandla, S., Jagtap, M. P., Khatri, N., Madan, H., & Vadduri, A. A. (2024). Herbicide spraying and weed identification using drone technology in modern farms: A comprehensive review. Results in Engineering, 101870.
31. Khuzaimah, Z., Nawi, N. M., Adam, S. N., Kalantar, B., Emeka, O. J., & Ueda, N. (2022). Application and potential of drone technology in oil palm plantation: Potential and limitations. Journal of Sensors, 2022(1), 5385505.
32. .Gano, B., Bhadra, S., Vilbig, J. M., Ahmed, N., Sagan, V., & Shakoor, N. (2024). Drone-based imaging sensors, techniques, and applications in plant phenotyping for crop breeding: A comprehensive review. The Plant Phenome Journal, 7(1), e20100.
33. .Chandran, I., & Vipin, K. (2024). Multi-UAV networks for disaster monitoring: challenges and opportunities from a network perspective. Drone Systems and Applications, 12, 1-28.
34. Jahan, N., Niloy, T. B. M., Silvi, J. F., Hasan, M., Nashia, I. J., & Khan, R. (2024). Development of an IoT-based firefighting drone for enhanced safety and efficiency in fire suppression. Measurement and Control, 00202940241238674.
35. .Abeywickrama, H. G. K., Bajón-Fernández, Y., Srinamasivayam, B., Turner, D., & Rivas Casado, M. (2023). Development of a UAV based framework for CH4 monitoring in sludge treatment centres. Remote Sensing, 15(15), 3704.
36. .Hoang, M. L. (2023). Smart Drone Surveillance System Based on AI and on IoT Communication in Case of Intrusion and Fire Accident. Drones, 7(12), 694.
37. .Ashraf, S. N., Manickam, S., Zia, S. S., Abro, A. A., Obaidat, M., Uddin, M., ... & Alsaqour, R. (2023). IoT empowered smart cybersecurity framework for intrusion detection in internet of drones. Scientific Reports, 13(1), 18422.
38. Alqurashi, F. A., Alsolami, F., Abdel-Khalek, S., Sayed Ali, E., & Saeed, R. A. (2022). Machine learning techniques in internet of UAVs for smart cities applications. Journal of Intelligent & Fuzzy Systems, 42(4), 3203-3226.
39. .Choutri, K., Lagha, M., Meshoul, S., Batouche, M., Bouzidi, F., & Charef, W. (2023). Fire detection and geo-localization using uav's aerial images and yolo-based models. Applied Sciences, 13(20), 11548.
40. .Afghah, F., Razi, A., Chakareski, J., & Ashdown, J. (2019, April). Wildfire monitoring in remote areas using autonomous unmanned aerial vehicles. In IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) (pp. 835-840). IEEE.
41. Nayyar, B.-L. Nguyen, and N. G. Nguyen, "The internet of drone things (iodt): future envision of smart drones," in First International Conference on Sustainable Technologies for Computational Intelligence: Proceedings of ICTSCI 2019, pp. 563–580, Springer, 2020.
42. .S. Brischetto, A. Ciano, and C. G. Ferro, "A multipurpose modular drone with adjustable arms produced via the fdm additive manufacturing process," Curved and Layered Structures, vol. 3, no. 1, 2016.
43. .B. Galle, S. Arellano, N. Bobrowski, V. Conde, T. P. Fischer, G. Gerdes, A. Gutmann, T. Hoffmann, I. Itikarai, T. Krejci, et al., "A multi-purpose, multi-rotor drone system for long-range and high-altitude volcanic gas plume measurements," Atmospheric Measurement Techniques, vol. 14, no. 6, pp. 4255–4277, 2021.

44. .R. Remington, R. Cordero, L. March, and D. Villanueva, “Multipurpose aerial drone for bridge inspection and fire extinguishing,” {Unpublished Thesis). Florida International University. Retrieved April, vol. 10, p. 2016, 2014.
45. .K. Balaji, V. Babu, and S. Sulthan, “Design and development of multipurpose agriculture drone using lightweight materials,” SAE International Journal of Aerospace, vol. 16, no. 01-16-02-0012, pp. 177–183, 2022.
46. .P. Loukinas, “Drones for border surveillance: multipurpose use, uncertainty and challenges at eu borders,” Geopolitics, vol. 27, no. 1, pp. 89–112, 2022.
47. .G. S. Ganesan and M. Mokayef, “Multi-purpose medical drone for the use in pandemic situation,” in 2021 IEEE Microwave Theory and Techniques in Wireless Communications (MTTW), pp. 188–192, 2021.
48. R. Tariq, M. Rahim, N. Aslam, N. Bawany, and U. Faseeha, “Dronaid: A smart human detection drone for rescue,” in 2018 15th International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT (HONET-ICT), pp. 33–37, IEEE, 2018..J.
49. , J. J. Rold´an-Gomez E. Gonz´alez-Gironda, and A. Barrientos, “A survey on robotic technologies for forest firefighting: Applying drone swarms to improve firefighters’ efficiency and safety,” Applied sciences, vol. 11, no. 1, p. 363, 2021.
50. Albanese, V. Sciancalepore, and X. Costa-P´erez, “Sardo: An automated search-and-rescue drone-based solution for victims localization,” IEEE Transactions on Mobile Computing, vol. 21, no. 9, pp. 3312–3325, 2021.
51. .Rejeb, A. Abdollahi, K. Rejeb, and H. Treiblmaier, “Drones in agriculture: A review and bibliometric analysis,” Computers and electronics in agriculture, vol. 198, p. 107017, 2022.
52. S. Kurkute, B. Deore, P. Kasar, M. Bhamare, M. Sahane, et al., “Drones for smart agriculture: A technical report,” International Journal for Research in Applied Science and Engineering Technology, vol. 6, no. 4, pp. 341–346, 2018.