FIRE PHOENIX- AN ADVANCED FIRE DISASTER MANAGEMENT SYSTEM

By

George Tonmoy Roy
Roll: 1907114
&
Masudur Rahman Rabby
Roll: 1907113

Supervisor:
Md. Repon Islam
Lecturer
Dept. of Computer Science and Engineering
Khulna University of Engineering & Technology

Signature

Department of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna 9203, Bangladesh
November 2023

Acknowledgments

First and foremost, I must be grateful to and wish to acknowledge my insightful indebtedness to Md. Repon Islam, Lecturer, Department of Computer Science and Engineering and the supervisor of this project. His knowledge in this field influenced me to carry out this project up to this point. His endless endurance, scholarly guidance, continual encouragement, constant and lively supervision, constructive criticism, and priceless suggestions made it possible to come to this phase. Without his inspiration, enthusiasm, and encouragement, this work could not have been completed. Last, but by no means least, I thank Allah for the talents and abilities I was given that made it possible to undertake this project.

Authors

Abstract

This project combines a mobile application, a fire-fighting robot car, and an administrative control website to improve fire detection, response, and monitoring. The mobile app uses image recognition algorithms to detect fire incidents and transmits data to the administrative control website. The fire-fighting robot car, equipped with a camera, object avoidance sensor, pumper mechanism, and temperature sensor, autonomously navigates to the fire location using GPS coordinates. The administrative control website provides real-time monitoring of the robot's activities, allowing administrators to accept user-reported incidents, initiate the robot's response, and oversee its operations remotely. This project demonstrates the potential of integrating diverse technologies for fire management and exemplifies the practical implementation of autonomous systems in emergency scenarios.

Contents

List of Figures

1 Introduction

1.1 Background

Traditional firefighting methods and emergency response systems face challenges in swiftly addressing fire disasters, especially in remote or hazardous areas. Communication gaps between affected individuals and responders often lead to delays in assessment and action. Additionally, maneuvering firefighting vehicles through complex environments poses risks to human responders and limits access. Our project aims to revolutionize fire disaster management by integrating a web-based admin panel, a user-centric Android app, and an autonomous firefighting robot with live surveillance. This integration seeks to establish real-time communication, facilitate quick response, and ensure the safety of responders while effectively combating fire emergencies.

1.2 Objectives

The project objectives encompass developing an interconnected system where an Android app facilitates real-time reporting of fire incidents to an admin panel. The primary goal is to synchronize this data flow with a specialized fleet of firefighting robots. These robots, equipped with obstacle avoidance technology and live streaming capabilities, are remotely controllable via the admin panel. The project aims to create a seamless integration between user-generated reports and immediate robot deployment, significantly reducing response time to fire emergencies. This innovative approach aims to enhance firefighting efficiency, augmenting human safety, and enabling rapid, precise actions in critical situations.

1.3 Scope

The project demonstrates its scope in critical scenarios necessitating rapid, remote, and precise fire disaster management. It is particularly impactful in emergencies such as forest fires, urban conflagrations, or industrial incidents where prompt responses are pivotal. The system's ability to receive user reports via an Android app, control firefighting robots, and stream live visual data empowers authorities to orchestrate immediate, accurate, and efficient firefighting operations. This comprehensive approach aids in minimizing damage, enhancing response time, and ensuring more effective disaster mitigation strategies.

1.4 Unfamiliarity of the problem

The complexity and unfamiliarity of the problem stem from the intricate nature of fire disaster management, particularly in scenarios demanding real-time response and intervention. Handling diverse challenges, such as the unpredictability of fire spread, the integration of robotics in disaster management, and the seamless coordination between users, a web-based admin panel, and an Android app, poses significant challenges. Additionally, ensuring the reliability and precision of live-streamed data from the ESP32 cam module adds a layer of complexity to the project. Addressing these multifaceted aspects while ensuring robustness and effectiveness in disaster management forms the core challenge of this undertaking.

1.5 Project planning

Project planning entails meticulous coordination across various facets: devising a comprehensive roadmap, setting clear milestones, and allocating resources efficiently. The project delineates a structured approach, encompassing phases such as conceptualization, system design, implementation, and testing. Each phase requires meticulous planning, considering hardware integration, software development, and user interface design. Additionally, establishing an agile framework for iterative improvements and flexibility in responding to unforeseen challenges remains pivotal in the project planning process. The robust planning framework ensures the seamless execution and successful accomplishment of each project phase within stipulated timelines.

2 Related Works

2.1 Related works

There are several works on fire disaster management systems. They have used nearly the same approach and methodology we used.

A work [1] introduces the QRob, a compact firefighting robot designed to extinguish fires without exposing firefighters to danger. Equipped with an ultrasonic sensor and flame sensor, QRob can detect fires up to 40 cm away and can be monitored using a camera connected to a smartphone or remote device. Its lightweight water pump ensures low noise and high effectiveness.

Another research [2],[5] aims to detect and control small-scale fires using sensors, cameras, IoT, and Wi-Fi modules, using Thermite and Fire Robot.

[3] Also, one research included a face detection technique where the robot will detect a person and send information to the remote server.

Coming to the streaming part of the robot [4] a CCD (Charge Coupled Device) camera can be also used instead of an ESP cam module housed over the top to survey real fire situations.

3 System Design

Fire Phoenix is a system, that has a mobile app that detects fire from images and sends fire images and location, a firefighting robot car that has a camera for monitoring the robot, a sensor for object avoidance, a pumper remove fire, and a temperature sensor for robot condition monitoring, GPS for tracking the robot to go to the fire destination that the app user will send. An admin control website to monitor the robot's data and also other control over the app and robot. If a user sends data of fire, the admin accepts that and starts the robot to go to that location and then it will automatically remove fire and come back

.

3.1 Analysis of the system

This system analysis consists of three flowcharts: the robotic system's operational workflow, the web application's user interface hub, and the mobile application's interface. The first flowchart outlines the robotic system's decision-making and action execution processes, providing insights into efficiency, responsiveness, and adaptability. The second flowchart focuses on the web application's user journey, data processing, and communication channels. The third flowchart outlines the mobile application's interaction with the system, addressing accessibility, responsiveness, and feature integration issues.

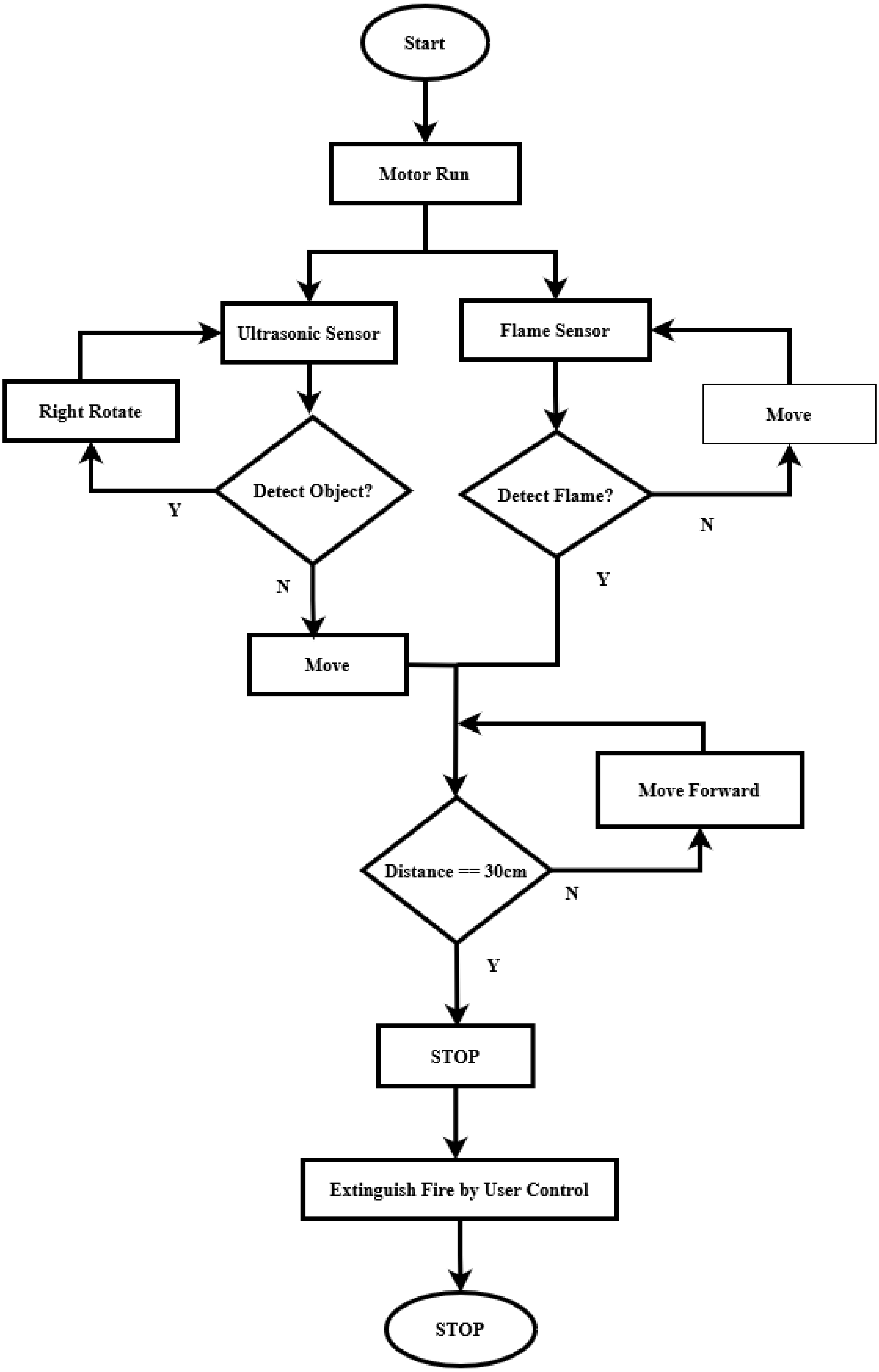

Figure 3.1: Flow Chart of robot working principle.

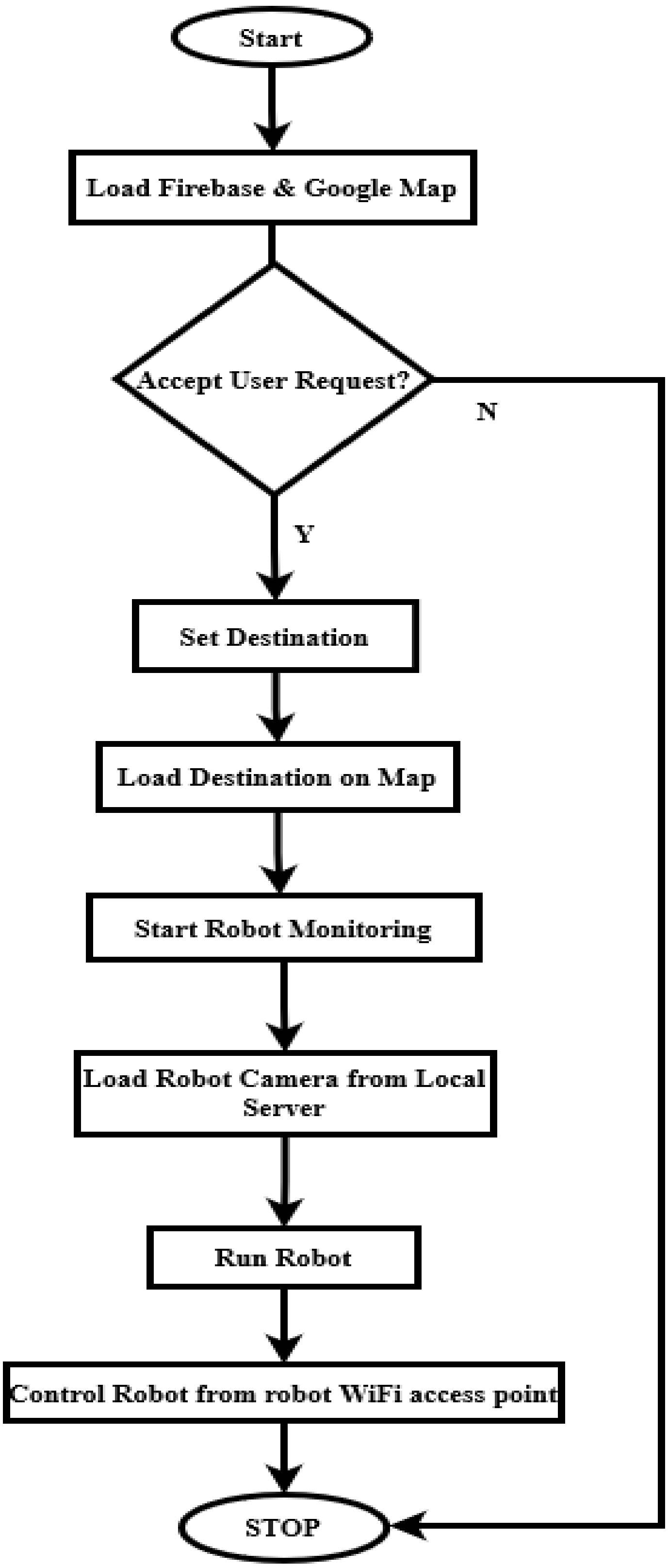

Figure 3.2: Flow Chart of Web application working principle.

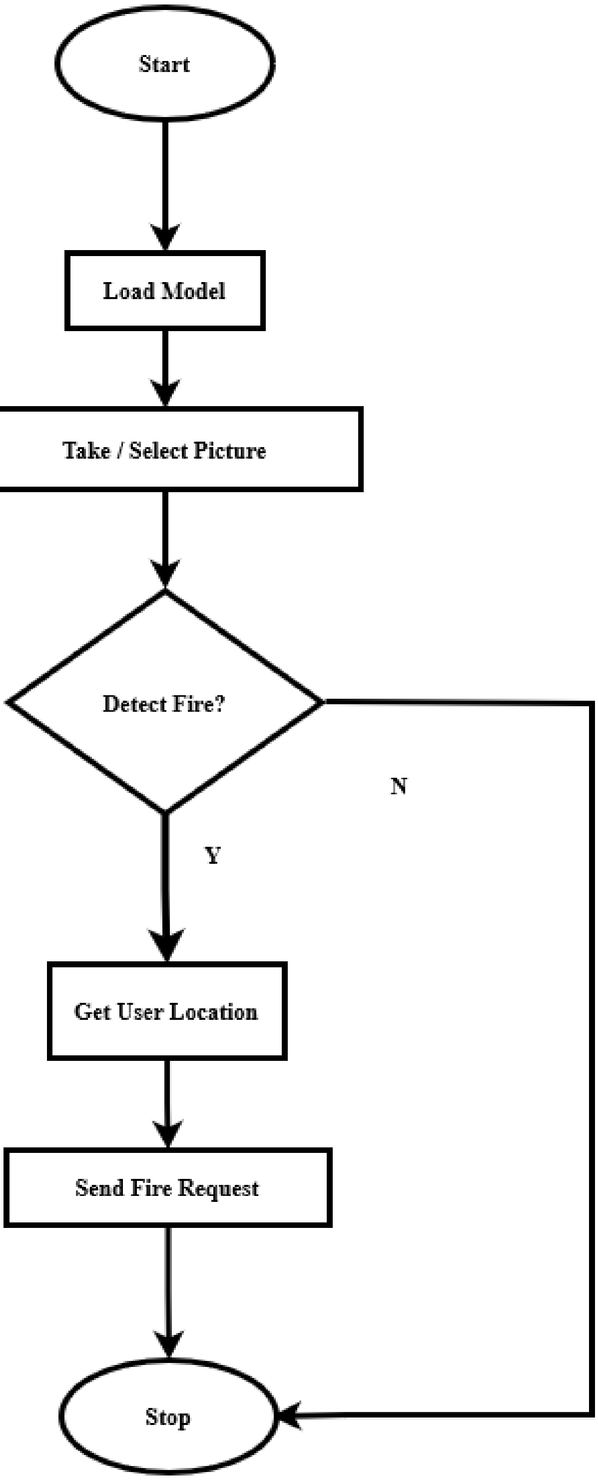

Figure 3.3: Flow Chart of Mobile application working principle.

3.2 System architecture

The system is designed for fire disaster management, and have 2 interface and a hardware tool. So, there will 3 phases of the system structure. These are given below.

Mobile Application:

- User authentication.
- The deep learning model for fire detection.
- Get the current location from the user device.
- Take a picture or select a picture from the user to detect fire.
- Send a request to the admin of the fire station to extinguish the fire.

Web Application:

- Firebase for user requests and robot data monitoring.
- Take user request of fire in a list consisting of image and fire location.
- Accept or decline the request, based on fire existence.
- A map view for monitoring the robot's location and the destination to go.
- A box, where the data of destination.
- Another box for monitoring the robot's location data, temperature, etc.
- A camera view to monitor the robot over the internet.

Robot:

- Controlled by a web-based application.
- Detect objects from ultrasonic sensors.
- Use the camera to sense the front of the camera and send it to an online host NodeJS server to monitor from a web application.
- Detect fire by using a flame sensor
- Extinguish fire by using a water pumper and a water tank of the robot and spread out the water based on the needs.

3.3 Tools used

In our project, for mobile applications, we use Android Studio for IDE and Firebase database. We also use the Kaggle dataset, and Google Teachable Machine for the fire detection model, and use Java language for development. For web applications, we use Visual Studio Code as IDE, Firebase database, and HTML, CSS, and JavaScript as primary language. For a robot, we use Arduino IDE and C / C++ language for development and Firebase, WebSocket.

3.3.1 Android Studio

Android Studio is the official integrated development environment (IDE) for Android app development. In Android Studio, we built our mobile application with the integration of the TensorFlow Lite model for detecting fire.

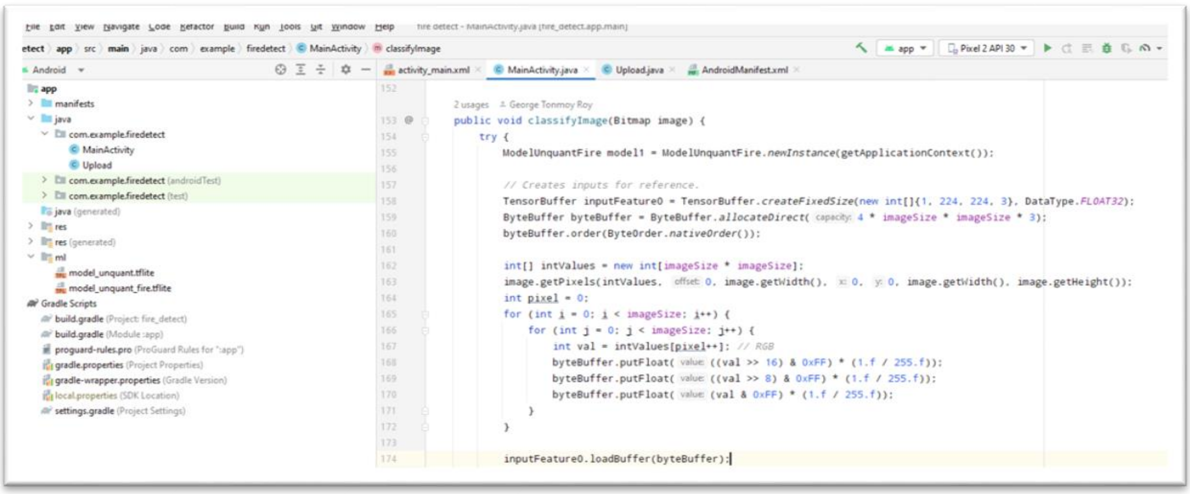

Figure 3.4: Integrating TF Lite Model in Android

3.3.2 Firebase

Firebase is a mobile and web application development platform provided by Google. We use the real-time database for monitoring robot sensor data and get user requests from the user's mobile device to the admin webpage.

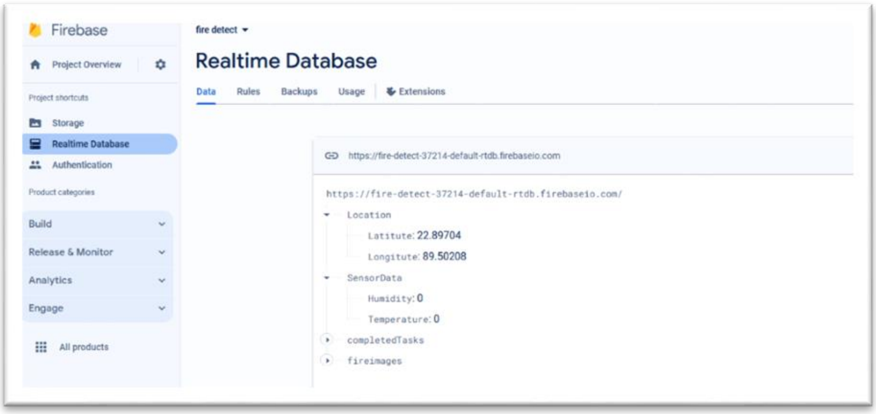

Figure 3.5: Sensor Data and Location Storing in Firebase

3.3.3 Google Teachable Machine

Google Teachable Machine is an experiment that makes machine learning more accessible. It allows users to train a model with their images and gestures without writing code. We use this built-in way to get the model of our model for fire detection.

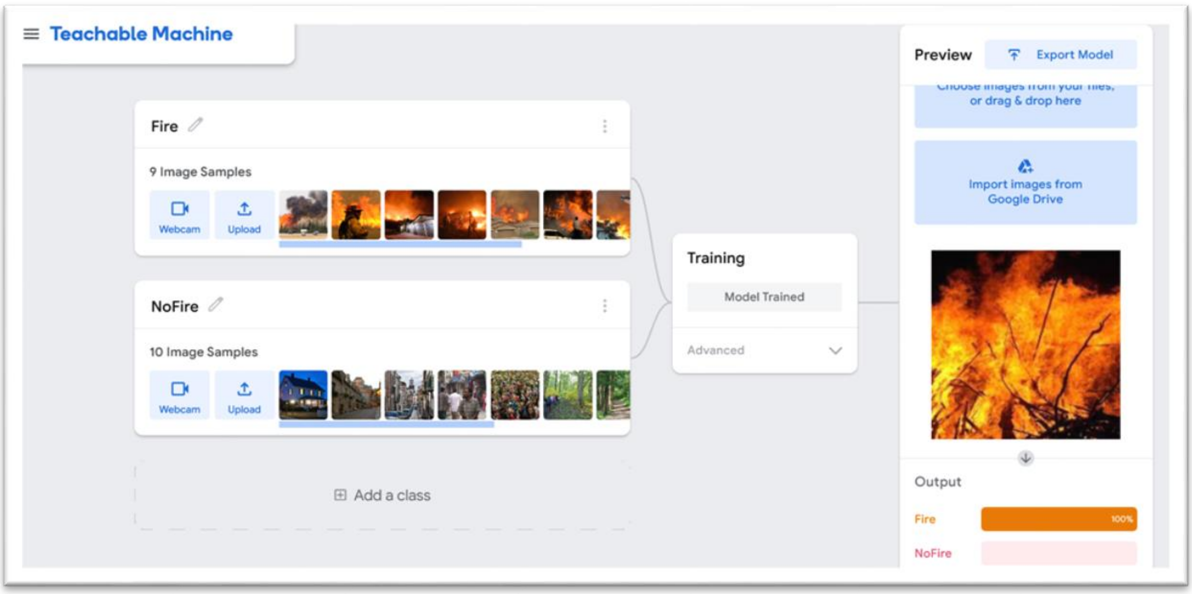

Figure 3.6: Creating TF Lite Model for Fire Detection

3.3.4 Visual Studio Code

Visual Studio Code (VS Code) is a lightweight, open-source code editor developed by Microsoft. It supports various programming languages and features built-in Git integration, syntax highlighting, intelligent code completion, and extensions for further customization. It is widely used for web and general-purpose development.

3.3.5 HTML, CSS, JavaScript

HTML (Hypertext Markup Language), CSS (Cascading Style Sheets), and JavaScript form the core technologies for building and designing websites. HTML structures content, CSS styles presentation, and JavaScript add dynamic behavior, creating interactive and visually appealing web pages.

3.3.6 Arduino IDE

Arduino IDE is an open-source software used for programming Arduino boards. It provides a simple integrated development environment for writing, compiling, and uploading code to Arduino microcontrollers. We use Arduino IDE for hardware implementation. Use C language for coding in the IDE.

Figure 3.7: Microcontroller Programming with Arduino IDE

3.3.7 C / C++

C and C++ are powerful, general-purpose programming languages. C is known for its efficiency and is the foundation for many other languages, while C++ extends C with object-oriented programming features. We use C language for robot-controlling logic.

3.4 System Interface

3.4.1 Mobile Application Interface

For the detection of fire user can take or upload pictures of surrounding via the mobile application. It will predict the data with deep learning model. The data can be sent to the system administration.

Figure 3.8: Android App for Users to Send the Data

3.4.2 Web Application Interface

The robot control and video streaming is integrated here. The user data like location, request for service is available in this interface.

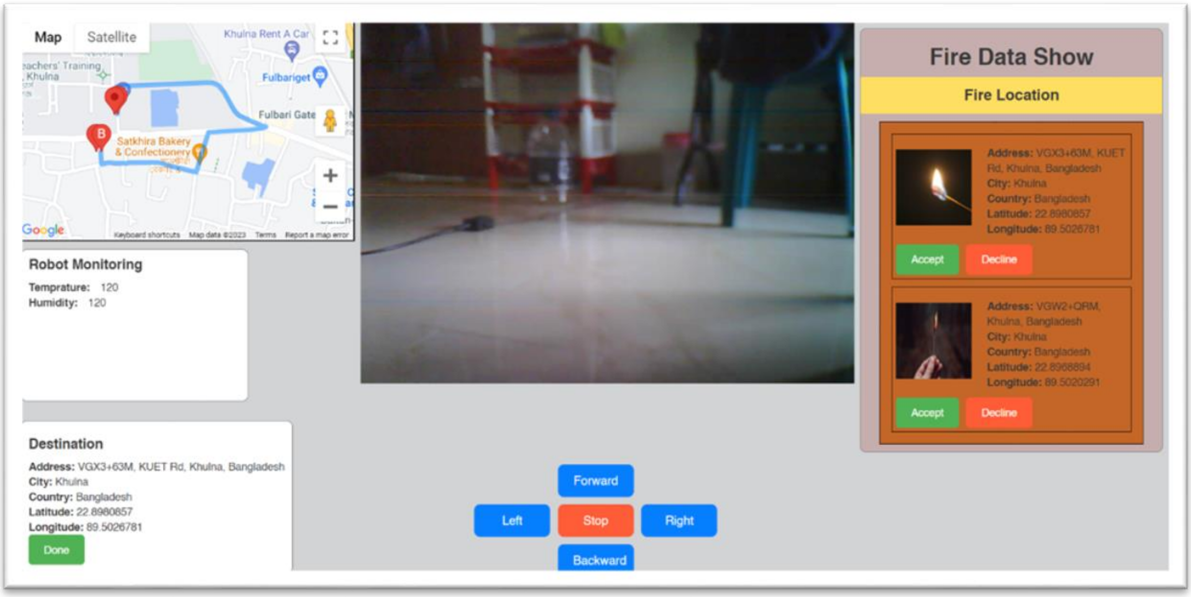

Figure 3.9: Admin Control Using Web Application

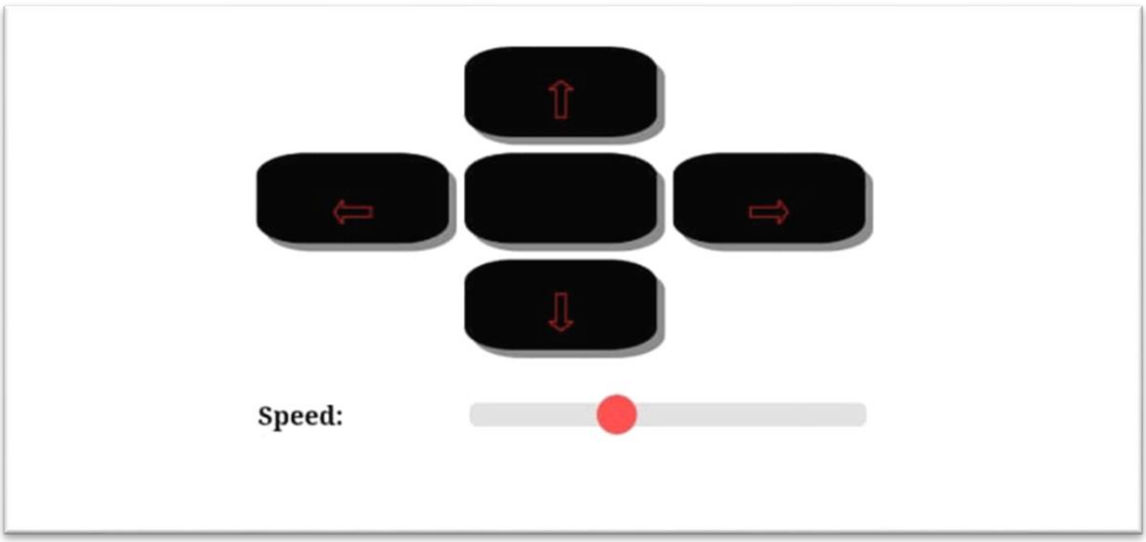

Figure 3.10: Car Controlling in Web Application

3.4.3 Firefighting Robot Car

The car is designed with suitable sensors and other equipment for controlling the fire disaster situation, tracking the user location, sent via request.

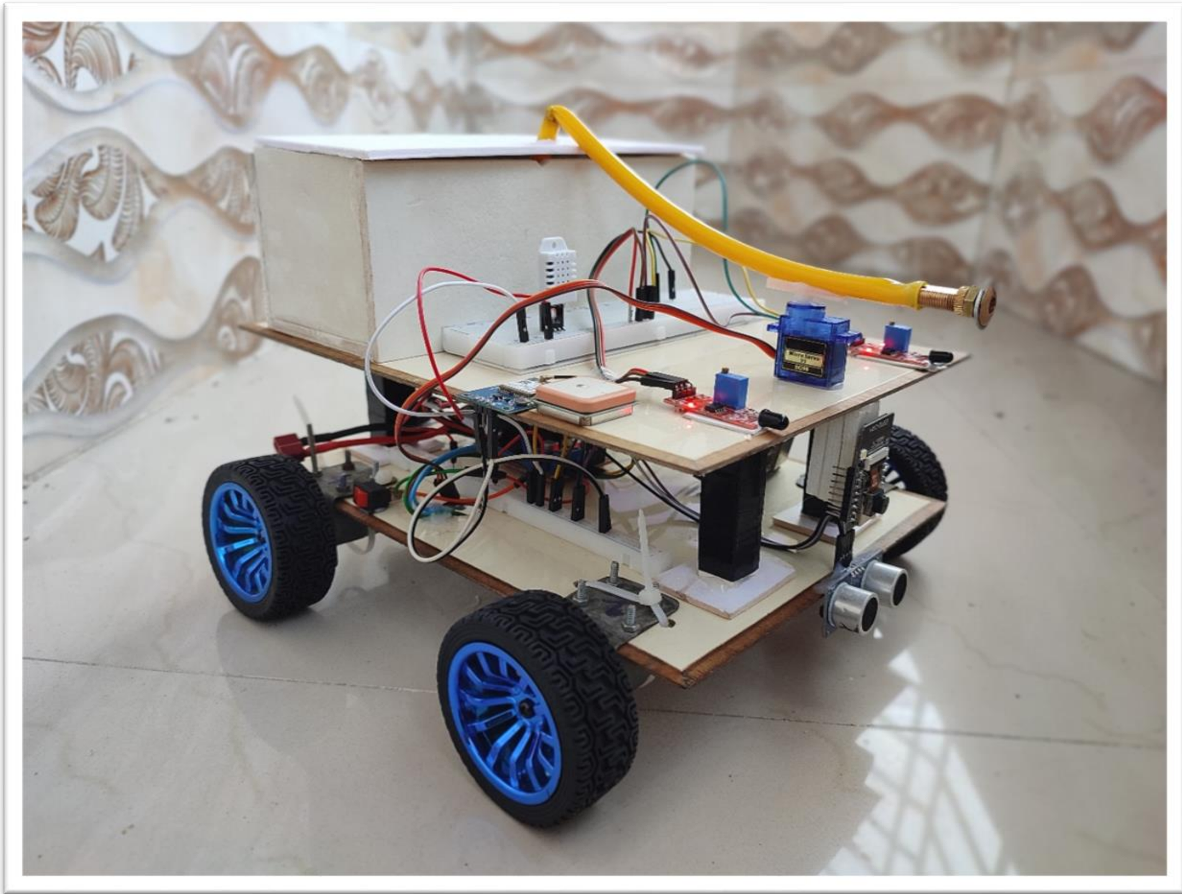

Figure 3.11: Robot Prototype

4 Project Implementation

4.1 System implementation

The system implementation for your fire-fighting robot project could cover various aspects related to the deployment and integration of hardware and software components. Here's a breakdown of potential topics for the system implementation phase

4.1.1 Hardware Integration:

- **Robot Assembly:** The robot has a Camera module (esp32 cam), Ultrasonic sensor (HC-SR04), Flame sensor (KY-026 IR), Gear motor (25GA 370 400rpm), GPS module (NEO-06m), Motor Driver (L298N), Power Supply (Lipo Battery 30C 1500mah), Esp32S3 Dev Module, Breadboard. The robot connection is established with an esp32 microcontroller with a breadboard.
- **Power Management:** The robot is powered with a 12V DC power supply from Lipo Battery to the motor driver and then it powers all the components.
- **Communication Protocols:** The communication with the robot and web application in 3 different ways. One is a NodeJS server for camera stream, Firebase for sensor monitoring, Local web server for controlling the car.

4.1.2 Software Development:

- **Mobile App:** The mobile app contains a deep learning model to detect fire from any image. It has user authentication and also can take pictures or select pictures.
- **Admin Control Website:** The web app has a Firebase connection, a NodeJS connection, and in another web app a local webserver of car control. It has a map view to track the location of the robot, a control mechanism of the robot, user request panel of fire.

- **Deep Learning Model:** For fire detection, we use deep learning TensorFlow Lite Model, created in Google teachable machine. We give the dataset from our own and also from Kaggle. This is a built-in model. That's why sometimes it gives error prediction

.

4.2 Morality or ethical issues

The project involves a mobile app and firefighting robot to address fire-related incidents. However, there are ethical concerns to consider. Privacy concerns arise from the use of the app to capture and send images and location data, which requires secure handling. Data security is crucial for storing and transmitting sensitive information, and algorithmic bias can introduce false positives or negatives. Autonomous robot behavior raises ethical questions about accountability and decision-making. Emergency response accuracy is crucial, and the technology's reliability is essential. Equitable access is crucial, and the environmental impact of the technology should be minimized. Transparency and accountability are essential, and community engagement is crucial for addressing concerns and ensuring cultural sensitivity. Human control and intervention are also essential, especially in critical situations. To create a technology that aligns with societal values, prioritizes safety, and respects individual rights, developers, stakeholders, and policymakers must address these ethical considerations through regular ethical reviews and ongoing dialogue with relevant stakeholders.

4.3 Socio-economic impact and sustainability

The fire-fighting robot project has a socio-economic impact that includes community safety, job creation, education, and accessibility to emergency services. It contributes to community safety by providing a rapid and automated response to fire incidents, reducing property damage, and improving emergency response times. The project also creates employment opportunities, requiring specialized skills in robotics, software development, and system administration. The introduction of advanced technologies empowers individuals with valuable skills. The project addresses challenges related to accessibility to emergency services, positively impacting underserved communities. It also considers the project's environmental impact, scalability, resource efficiency, and long-term viability. It ensures social inclusivity, equity, cultural sensitivity, user engagement, and ethical considerations. The project's approach considers privacy and transparency, ensuring its positive impact on both the immediate community and the broader societal context.

5 Conclusion

Future research should focus on parallel computing and communication framework design for improved system responsiveness, scalability, efficiency, and predictive analysis for disaster containment.

5.1 Conclusion and challenges faced

Our project embodies a significant leap in fire disaster management systems, streamlining response mechanisms through advanced technology integration. Throughout its development, several challenges emerged, notably in optimizing real-time communication between the admin panel, user app, and the firefighting robot. Balancing user-friendly interfaces with intricate functionalities posed another hurdle. These challenges, while demanding, provided invaluable insights and opportunities for innovation. Despite these obstacles, the project’s success demonstrates the immense potential for technological solutions in revolutionizing emergency responses. The continued challenge lies in refining these systems to adapt to evolving disaster scenarios, ensuring swift, efficient, and reliable mitigation strategies.

5.2 Future Study

In the future, parallel computing and the optimal model for the design of the project’s communication framework stands as promising areas for further study. Enhancing the system's responsiveness and scalability through parallel processing could revolutionize real-time data exchange between the admin panel, user app, and the firefighting robot. Exploring and refining algorithms to streamline and prioritize communication channels might amplify system efficiency during peak demand, ensuring swift and seamless information flow during critical fire incidents. Additionally, delving into machine learning integration for predictive analysis could empower the system to anticipate fire spread, enabling preemptive measures for more effective disaster containment.

.

References

[1] *Aliff, M., Samsiah, N., Yusof, M.I., & Zainal, A. (2019). Development of Fire Fighting Robot (QRob). International Journal of Advanced Computer Science and Applications.*

[2] *Sawant, M., Pagar, R., Zutshi, D., Sadhukhan, S. (2021). Fire Detection and Controlling Robot Using IoT. In: Patnaik, S., Yang, XS., Sethi, I. (eds) Advances in Machine Learning and Computational Intelligence. Algorithms for Intelligent Systems. Springer, Singapore.*

[3] *Saritha, C., & Harish, B. (2020). Firefighting Robot with Face Detection System. International Journal of Engineering Research and, 9.*

[4] *Mittal, S., Rana, M.K., Bhardwaj, M., Mataray, M., & Mittal, S. (2018). CeaseFire: The Fire Fighting Robot. 2018 International Conference on Advances in Computing, Communication Control and Networking (ICANN), 1143-1146.*

[5] *Al Rakib, M. A.; Rahman, M. M.; Anik, M. S. A.; Masud, F. A. J.; Rahman, M. A.; Hossain, M. S.; Abbas, F. I. Fire Detection and Water Discharge Activity for Fire Fighting Robots Using IoT. EJENG* ***2022****, 7, 128-133*

[6] https://teachablemachine.withgoogle.com/train/image

[7] https://github.com/un0038998/WiFiTank/tree/main/WiFi_Tank