Grenze International Journal of Engineering and Technology, January Issue

Emergency Response Network using Motion Sensors & Adaptive Location-based Power Management

Salman Faris AV[1], Ajay Krishna[2], Mohammed Badusha[3], Vipin[4] and Yadu P Dev[5]
[1-5]Department of Computer Science and Engineering,
Sreepathy Institute of Management and Technology, Vavanoor, India
Email: {salmanfaris.av, ajay.krishna, mohammed.badusha, vipin}@simat.ac.in, yadu@simat.ac.in

***Abstract*— In the digital era where connections are growing, problem-solving technologies are becoming essential. Emergency response is one of the major problems among them. The efficiency of the emergency response system is lacking. This work aims to overcome this issue by creating the Emergency Response Network (ERN), a mobile application to improve how accidents are detected and managed. Many people suffer from accidents due to poor emergency systems, especially in areas with limited help available. In remote areas where immediate assistance is lacking, ERN aims to overcome the gap between accident occurrence and professional medical care. By making use of various sensors and communication technologies, the ERN aims to detect accidents automatically and quickly send distress signals to nearby certified first aid responders. Using a threshold-based algorithm, the ERN detects sudden movements indicating accidents with the help of motion sensors. This approach is inspired by insights from academic research papers and technological innovations. This work is improved by including the integration of motion sensors, GPS tracking, and advanced technologies.**

***Index Terms*—Emergency Response, Accident Detection, Motion Sensors, Power Management, Mobile Application, Real-Time Assistance.**

I. Introduction

As technologies have advanced, the emergency response system still struggles to provide quick assistance due to the lack of alert systems and trained first aid responders. To address this issue, this work aims to create a mobile application designed to detect accidents and alert nearby responders. The app was developed using Flutter, a cross-platform app development framework that simplifies the development process with pre-built libraries and functions. The backend of the system is powered by Firebase, a cloud-based service provided by Google. It offers Firebase Firestore to store user data and handle authentication. Firebase Realtime Database functionality was also used to create a custom method for sending alerts to responders in real time, ensuring a quick response during emergencies.

The core of the ERN is its accident detection mechanism, which utilizes the smartphone's built-in motion sensors, such as the accelerometer and gyroscope. The accelerometer measures linear motion, while the gyroscope measures the orientation of the phone. The integration of geolocation services helps the ERN accurately detect the location of both the user and the responder and also aids in implementing a power management system based on the user's movement. The combination of these technologies allows the ERN to provide first aid assistance quickly, reducing the gap between accident occurrence and medical attention.

Grenze ID: 01.GIJET.11.1.537

II. RELATED WORKS

Accident detection has become an essential area of research due to the increasing number of accidents and the need for timely responses. In many cases, delay in responding to accidents can significantly affect survival rates, making rapid detection and notification systems vital. Various studies have focused on improving accident detection systems using sensors, machine learning, and smartphone technologies.

Felisberto et. all. [1] proposed a solution aimed at monitoring the elderly using sensor fusion. Their system combines data from wireless sensors placed on the user to detect movements and accidents. While their work primarily focuses on elderly monitoring, it highlights how sensor fusion can be used for movement and accident detection with high precision, especially for detecting falls. Mateen et. all. [2] introduced the concept of smart roads, where accidents are detected and alerted to nearby vehicles using roadside sensors. Their Accident Alert Light and Sound (AALS) system minimizes the risk of multiple-vehicle collisions (MVCs) by notifying other vehicles of accidents in poor visibility conditions. This approach avoids the need for each vehicle to have accident detection equipment. Ghazalli et. all. [3] presented MotoSOS, a mobile application that detects accidents for motorcycle riders using motion sensors like the accelerometer and gyroscope. Their work focuses on sending SMS alerts to emergency contacts when an accident is detected, especially in rural areas where bystanders may not be available to help. Agrawal et. all. [4] developed an Android application that detects accidents using the phone's inbuilt sensors, such as accelerometers and rotational sensors. When an accident occurs, the system captures images from the phone’s camera and sends an alert with the GPS location to pre-stored contacts. Their system also adapts the detection threshold for different types of vehicles. Maruyama and Ohashi [5] focused on accident prediction using visual attention models. Their method uses a deep learning approach to predict accidents based on differences between visual attention and the focus of expansion (FOE) in vehicle-mounted camera images. Their system provides an early prediction of accidents, even in cases where the motion of risk factors is minimal. Raza et. all. [6] worked on detecting driver behavior that can lead to accidents using smartphone motion sensors. They developed an LR-RFC method, combining logistic regression and random forest classifier to extract features from sensor data and predict risky driving behavior. Their approach achieved a high-performance score, emphasizing the role of early detection in accident prevention. Tian et. all. [7] proposed an automatic car accident detection system using cooperative vehicle infrastructure systems (CVIS) and machine vision. Their model, YOLO-CA, detects accidents by analyzing images from roadside devices, improving accident detection even under various weather and traffic conditions. Fogue et. all. [8] designed a novel system to improve response times in traffic accidents. Their system uses communication technologies installed in vehicles to automatically detect accidents and notify emergency services. By providing detailed information about the accident, they aim to reduce the time required to deploy the appropriate resources. Amin et. all. [9] developed an accident detection and reporting system using GPS, GPRS, and GSM technology. Their system monitors vehicle speed and detects accidents based on sudden speed changes. Once an accident is detected, the system sends an alert to a service center with the vehicle’s location, helping rescue services reach the accident site promptly. Dogru and Subasi [10] utilized machine learning techniques to develop a traffic accident detection system using simulated data from vehicular ad-hoc networks (VANETs). Their approach applies random forest classifiers to detect accidents based on vehicle speed and position data, achieving high accuracy in distinguishing between accident and normal driving conditions. Kamijo et. all. [11] focused on monitoring traffic flow at intersections and detecting potential accident-prone situations using a spatio-temporal Markov random field (MRF) model. Their system tracks vehicles at intersections and predicts behaviors that may result in collisions, offering a high success rate in tracking and event recognition. Alwan and Alshaibani [12] presented a smartphone-based accident detection system that focuses on low-speed car accidents. Their system differentiates between low-speed driving and activities like walking or running, ensuring accurate detection of accidents at lower speeds. Once an accident is detected, detailed information, including images and location, is sent to emergency responders. Wu et. all. [13] examined how smartphone motion sensors, like accelerometers and gyroscopes, can classify physical activities. Their research focuses on using sensor data to track daily activities, which could be extended to accident detection based on abnormal movements detected by the sensors.

III. METHODOLOGY

The Emergency Response Network (ERN) system is used to detect accidents and immediately alert nearby responders by combining real-time data, mobile sensors, and efficient backend systems. The architecture of ERN consists of multiple interconnected modules. Login enables secure access for users and responders through

Firebase Authentication. Once logged in, Geolocation Services track the user's position in real time, while the Accident Detection module uses motion sensors to detect sudden movements. When an accident occurs, the Distress Signal Sending System automatically alerts nearby responders. These responders are verified through a Verification process and registered via Registration for Responders. Energy consumption is managed by the Location-Based Power Management system, which only activates sensors when necessary to save battery life. Figure. 1 depicts the proposed architecture. The pseudo code for the system is given in Figure.2.

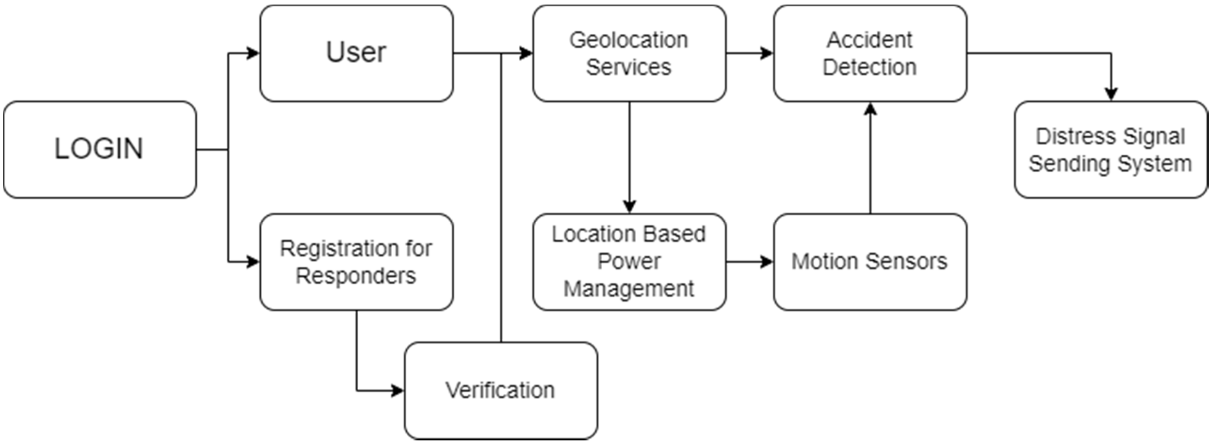

Figure 1. Basic System Architecture

A. Mobile Application Development

The ERN app was created using Flutter, a framework for building mobile applications that work across different platforms. Flutter provides many built-in tools and libraries that simplify the design and development process. For accessing the motion sensor data, the sensors_plus package was utilized in this work, which provides the X, Y, and Z values of both the accelerometer and gyroscope. To access location information, the geolocator package was used, which gives the latitude and longitude of the phone. Since Flutter supports cross-platform development, a single codebase was written for both Android and iOS devices, although the current version is deployed for Android only. Flutter's simplicity helped streamline the development process, maintaining a consistent look and feel across devices. With this approach, the app efficiently allows users to sign up, log in, and send alerts without dealing with complex user interface issues.

B. Backend Infrastructure

For the backend, this app used Firebase, which is a cloud-based service from Google. Firebase offers several features that are very useful for mobile apps, such as real-time databases and secure user authentication. In this work, Firebase Authentication with email and password was used to handle secure login and signup processes. The integration of Firebase Storage allows users to upload and store profile images. Firebase Firestore stores essential user and responder details such as names, phone numbers, blood types, and any medical conditions that could be crucial for responders to know in an emergency. Additionally, this work uses Firebase Realtime Database to track the locations of both users and responders. This database allows for the implementation of a real-time alert system, where an alert is sent to the nearest responders as soon as an accident is detected. This combination of Firebase services ensures that the app functions smoothly and securely, with all critical data handled efficiently.

C. Accident Detection Mechanism

Accident detection is at the core of the ERN system. The application uses the smartphone's in-built motion sensors, such as the accelerometer and gyroscope, to detect unusual movements that might indicate an accident. The accelerometer measures the linear motion of the phone, while the gyroscope tracks its orientation. The system uses a threshold-based algorithm that processes the data from these sensors. If the sensors detect movement consistent with an accident, the ERN system sends an automatic distress signal to responders within a certain radius. To find the threshold value for an accident, MotoSOS [3] was referenced, and it was found that the average accelerometer values for different scenarios are shown in Table I. The results indicate that 72~73 m/s^2 is the maximum value for an accident scenario. Additionally, the maximum value for the gyroscope in normal situations is around -2 and 2, therefore exceeding this value indicates an accident.

TABLE I. ACCELEROMETER EXPERIMENT RESULTS

Experiment Situation	**Maximum Value (m/s^2)**
Normal ride	36.92
Simulated	72.69
Throw	65.52
Desk drop	38.79

C. Energy Efficiency

Energy management is a key concern for the ERN application, particularly in emergency situations where long battery life is essential. The system needs to access the user's location at all times, so disabling the location service to save energy is not an option. To address this, the application incorporates a location-based energy-saving feature that conserves battery life by activating motion sensors only when the user is in motion. The system calculates the user's speed and sets a threshold, if the user is moving above 5 m/s, the system considers the user to be traveling, and motion sensors remain active. If the speed falls below this threshold, the system deactivates the motion sensors to save battery life.

To calculate the user's speed, the application first determines the distance between the user's previous location and current location using the Haversine formula (1).Then sum all the distances to get a total distance (2). This distance is then divided by the time elapsed to find the rate of movement (3), which is the user's speed.

$$d = 2r \cdot \arcsin\left(\sqrt{sin^2\left(\frac{\Delta\emptyset}{2}\right) + cos(\emptyset_1)\cdot cos(\emptyset_2)\cdot sin^2\left(\frac{\Delta\lambda}{2}\right)}\right) \qquad (1)$$

Where d represents the distance, r is the Earth's radius, and ϕ and λ represent the latitudes and longitudes of the two locations.

$$D_{total} = \sum_{i=1}^{n-1} d_i \qquad (2)$$

Where d_i represents the distance between two consecutive points and n is the number of recorded location points.

$$v = \frac{d}{\Delta t} \qquad (3)$$

Where v represents the speed of the user and Δt is the time difference between the two consecutive points.

This approach ensures that the phone's battery is not drained unnecessarily while the user is stationary, preserving power for when it is needed most. Additionally, this feature can be enabled or disabled by the user based on their preference.

```
1. Login or Signup
  IF new user THEN
     - Perform Signup
  ELSE
     - Perform Login
  END IF

2. IF user is a responder THEN
     - Proceed with responder registration
     - Perform verification for responders
  END IF

3. Profile Setup
  - Get user details (name, contact info, blood type, medical conditions, etc.)

4. Get the location coordinates of both users and responders
  - Use geolocation services to track the current location (latitude, longitude)
```

```
5. IF battery-saving feature is enabled THEN
     - Calculate speed based on location changes using Haversine formula
     IF user speed > 5 m/s THEN
       - Enable motion sensors (accelerometer, gyroscope)
     ELSE
       - Disable motion sensors
     END IF
   ELSE
     - Always enable motion sensors
   END IF
6. Continuously monitor accelerometer and gyroscope values
   IF values exceed threshold THEN
     - Show "Accident Detected" pop-up with a timeframe to cancel
     - Wait for user input (Cancel or Proceed)

     IF user cancels accident alert THEN
       - Stop the distress signal sending process
     ELSE
       - Find responders within a certain radius using location coordinates
       - Send distress signal to nearest responders with user details (location, medical info)
     END IF
   END IF
```

Figure 2. Pseudo code of the system

IV. RESULTS

As testing the ERN to detect accidents and send distress signals required simulating an accident, the built-in emulator feature in Android Studio was used to simulate a traveling scenario. The threshold values were reduced, as achieving the real threshold value was not possible without a high-impact event. The system's accuracy, speed, and overall effectiveness in detecting accidents and sending distress signals were evaluated.
Below are the key results from the testing:

A. User Login and Profile Setup

The application uses Firebase authentication to handle user registration and login processes securely. During testing, users were able to sign up using their email and password as shown in Figure. 3, with their credentials securely stored in Firebase. After logging in, users completed their profiles by providing important personal information, such as their medical details (e.g., blood type, medical conditions), contact information, and an optional profile picture as shown in Figure. 4. This profile information was saved in Firebase Firestore, ensuring that it could be quickly accessed when needed. The well-organized profile setup also plays a critical role during an emergency, as the stored data (such as medical conditions) helps responders provide more informed and effective assistance.

B. Accident Detection

The accident detection relies on smartphone motion sensors such as the accelerometer and gyroscope. During the testing phase, the Android Studio emulator was used to simulate a traveling scenario, and the threshold values were artificially reduced to mimic the impact of an accident, as achieving real-life accident forces was impractical without a high-impact event.
The accident detection algorithm continuously monitored the motion sensor data, analyzing changes in acceleration and orientation. Once the threshold was crossed, indicating a possible accident, the system flagged the event and prepared to send a distress signal. The results showed that the system effectively detected these simulated accidents with a high level of accuracy.

C. Distress Signal Sending

After detecting an accident, the ERN application prompts the user with a pop-up notification that includes an audible alert shown in Figure. 5.
This notification provides the user with a 20-second window to cancel the distress signal if it was a false alarm. If the user does not cancel the signal within the time frame, the app automatically sends a distress signal to pre-registered nearby responders Figure. 6. The distress signal contains critical information, such as the user’s profile

details and current GPS location of the incident. This seamless process ensures that the signal is sent quickly, even if the user is unable to manually request help.

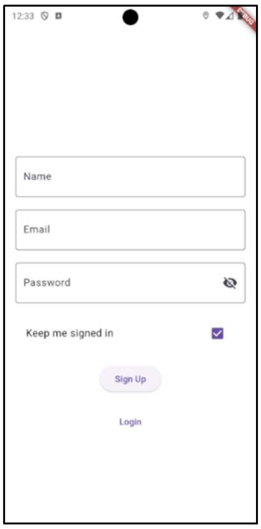

Figure 3. Login/Signup

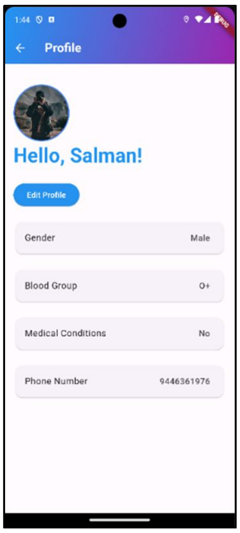

Figure 4. Profile Setup

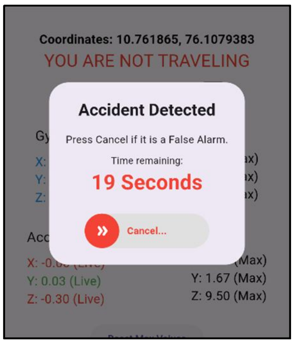

Figure 5. Accident detected pop-up

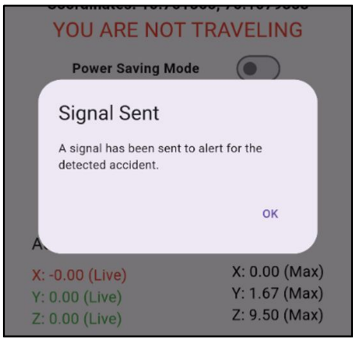

Figure 6. Distress signal send pop-up

Figure 7. Responder emergency alert

Figure 8. Google-map view of accident location

D. Responder Notification and Location Sharing

Once a distress signal is triggered, nearby responders registered in the ERN system receive a push notification alert almost instantly, as demonstrated in Figure. 7. This ensures that responders are immediately aware of the emergency and can begin the process of providing assistance. The notification includes the user's relevant information, such as their profile details and real-time GPS coordinates.

To help responders quickly locate the user, the app integrates Google Maps, as shown in Figure. 8. This feature allows responders to accurately pinpoint the accident location and navigate to the site with ease. During the tests, the location-sharing feature worked effectively, ensuring that responders could swiftly reach the accident site after receiving the distress signal. This combination of fast alerts and accurate navigation significantly improves the response time in emergency situations.

V. CONCLUSION

Through extensive testing and virtual simulations, the Emergency Response Network (ERN) has proven to be an effective solution in improving emergency response systems. The app enables quick accident detection and distress signal dispatch, utilizing the powerful combination of motion sensors and real-time location tracking. This ensures that help arrives promptly in critical moments where every second counts, potentially saving lives and reducing the severity of injuries.

Looking forward, there is immense potential for expanding this system. Right now, ERN relies on threshold-based detection, but this work can take it further by integrating machine learning models, making accident detection smarter and more accurate over time. By analyzing patterns and learning from real-world data, the app could predict accidents more reliably. Adding additional sensor inputs, like microphone data, could further enhance detection accuracy, capturing audio cues from collisions or sudden impacts.

Incorporating background running capabilities would ensure continuous monitoring, even when the app isn't active, making the system even more responsive. Additionally, by refining the notification system and integrating it with local authorities, the ERN could expand its reach, ensuring that both first responders and official services are alerted faster and more efficiently. These future developments are aimed at making the app more robust and far-reaching, pushing the boundaries of what technology can do to protect and save lives.

REFERENCES

[1] Felisberto, F.; Fdez.-Riverola, F.; Pereira, A. "A Ubiquitous and Low-Cost Solution for Movement Monitoring and Accident Detection Based on Sensor Fusion". Sensors 2014, 14, 8961-8983.

[2] Mateen A, Hanif MZ, Khatri N, Lee S, Nam SY. "Smart Roads for Autonomous Accident Detection and Warnings. Sensors". 2022; 22(6):2077.

[3] Hajar Izzati Mohd Ghazalli, Muhammad Izaiddin Hassan, Zuhri Arafah Zulkifli, and Siti Nuramalina Johari, "MotoSOS: Accident Detection for Motorcycle Riders using Motion Sensors", ARCA, vol. 15, no. 1, pp. 9–19, Oct. 2023.

[4] Agrawal, Akshay, Anand Khinvasara, Mitali Bhokare, Sumit Kaulkar and Prof. Y. K. Sharma. "ACCIDENT DETECTION SYSTEM APPLICATION." (2013).

[5] Y. Maruyama and G. Ohashi, "Accident Prediction Model Using Divergence Between Visual Attention and Focus of Expansion in Vehicle-Mounted Camera Images," in IEEE Access, vol. 11, pp. 140116-140125, 2023.

[6] A. Raza et al., "Preventing Road Accidents Through Early Detection of Driver Behavior Using Smartphone Motion Sensor Data: An Ensemble Feature Engineering Approach," in IEEE Access, vol. 11, pp. 138457-138471, 2023.

[7] D. Tian, C. Zhang, X. Duan and X. Wang, "An Automatic Car Accident Detection Method Based on Cooperative Vehicle Infrastructure Systems," in IEEE Access, vol. 7, pp. 127453-127463, 2019.

[8] M. Fogué, P. Garrido, F. J. Martinez, J. -C. Cano, C. T. Calafate and P. Manzoni, "Automatic Accident Detection: Assistance Through Communication Technologies and Vehicles," in IEEE Vehicular Technology Magazine, vol. 7, no. 3, pp. 90-100, Sept. 2012.

[9] M. Syedul Amin, J. Jalil and M. B. I. Reaz, "Accident detection and reporting system using GPS, GPRS and GSM technology," 2012 International Conference on Informatics, Electronics & Vision (ICIEV), Dhaka, Bangladesh, 2012, pp. 640-643.

[10] N. Dogru and A. Subasi, "Traffic accident detection using random forest classifier," 2018 15th Learning and Technology Conference (L&T), Jeddah, Saudi Arabia, 2018, pp. 40-45.

[11] S. Kamijo, Y. Matsushita, K. Ikeuchi and M. Sakauchi, "Traffic monitoring and accident detection at intersections," in IEEE Transactions on Intelligent Transportation Systems, vol. 1, no. 2, pp. 108-118, June 2000.

[12] Alwan, Zainab & Alshaibani, Hamid. (2015). "Car Accident Detection and Notification System Using Smartphone". International Journal of Computer Science and Mobile Computing (IJCSMC). 4. 620-635.

[13] Wu W, Dasgupta S, Ramirez EE, Peterson C, Norman GJ. "Classification accuracies of physical activities using smartphone motion sensors". J Med Internet Res. 2012 Oct 5;14(5):e130. PMID: 23041431; PMCID: PMC3510774.