

Iot-Based Smart Bike And Helmet System For Automatic Accident Detection And Emergency Alert

Ezhil Arasan

Dept of Computer Application

Abstract- *R Two-wheeler accidents often prove fatal because victims cannot call for help after a crash. Existing approaches such as manual reporting, mobile-based crash apps, GPS trackers, and basic helmet sensors are unreliable when the rider is unconscious, the phone is damaged, or the application is not activated. This paper proposes an IoT-enabled smart safety system that combines an accident-detecting bike unit with an interactive smart helmet. The bike module uses an accelerometer, gyroscope, and vibration sensor to detect sudden impacts and falls. When a possible accident is identified, the helmet initiates a short voice or button-based confirmation with the rider. If no response is received within a set time, the system retrieves the GPS location and uses a GSM module to send an emergency SMS to family members, guardians, or the nearest police station. A web-based monitoring dashboard built using PHP (Laravel), MySQL, and RESTful APIs supports administration, rider registration, and accident-log analysis. The proposed threshold-based decision algorithm reduces false alarms while ensuring rapid emergency response, offering a cost-effective and scalable solution to improve two-wheeler road safety.*

Keywords: IoT, smart helmet, accident detection, GPS, GSM, emergency alert system, accelerometer, embedded systems

I. INTRODUCTION

Road accidents involving two-wheelers are a leading cause of death and serious injury worldwide, particularly in countries where motorcycles serve as a primary mode of transport. A major contributor to fatalities is the delay in providing medical assistance: riders are frequently unable to call for help due to unconsciousness or severe injury, and bystanders may not notice or respond promptly. Existing solutions—manual reporting, smartphone crash-detection apps, GPS trackers, and basic helmet tilt sensors—are unreliable when the phone is damaged, the application is inactive, or the sensor cannot distinguish a real crash from normal riding shocks.

This work proposes an IoT-based smart bike and helmet system that automatically detects accidents using motion sensors, verifies the rider's condition through an interactive helmet, and—if no response is received—transmits

the rider's GPS location via SMS to predefined emergency contacts and authorities. By integrating sensor fusion, rider-confirmation logic, and GSM/GPS communication with a web-based monitoring platform, the system aims to minimize emergency response time, reduce false alarms, and provide a low-cost, scalable road-safety solution.

A. Objectives

- Design a system that automatically detects accidents using accelerometer, gyroscope, and vibration sensors.
- Develop a reliable method to identify sudden impacts, abnormal tilt, and collisions with high accuracy.
- Integrate a smart helmet that interacts with the rider to verify safety condition after an accident.
- Implement a confirmation mechanism using push buttons or voice input.
- Automate emergency SMS alerts with GPS location when the rider does not respond.
- Reduce false alarms through intelligent decision logic and confirmation steps.

B. Scope

The work covers integration of sensor-based accident detection with a helmet interface, embedded software for sensor processing and decision-making, and an SMS-based alert mechanism using GPS and GSM modules. Optional extensions include voice interaction, cloud connectivity, and web-based monitoring via APIs. The current scope is limited to prototype-level implementation and controlled testing, excluding large-scale deployment or direct integration with national emergency services.

II. EXISTING SYSTEM AND LIMITATIONS

Current accident-response approaches include: (1) manual reporting by bystanders, which depends on human presence and fails in low-traffic or remote areas; (2) mobile crash-detection applications relying on phone sensors and connectivity, which are unreliable if the device is damaged, uncharged, or the app is inactive; (3) GPS tracking devices that provide location only, without accident-detection

capability; and (4) basic helmet sensors (tilt/shock switches) that lack processing intelligence and often misclassify normal vibrations as accidents.

The principal disadvantages of these systems are: dependence on manual reporting, ineffectiveness when the rider is unconscious, failure of mobile apps if the phone is damaged or switched off, delayed emergency response, inability of GPS-only devices to detect accidents, high false-alert rates in basic sensors, dependence on continuous internet connectivity for app-based solutions, high cost of advanced commercial systems, and poor coordination among emergency services.

III. PROPOSED SYSTEM

The proposed system combines sensor-based detection, rider validation through an interactive helmet, and automated GSM/GPS-based emergency communication, supported by a Laravel/MySQL web application for monitoring and administration.

A. Accident Detection and Rider Validation

The bike unit continuously monitors motion using an accelerometer, gyroscope, and vibration sensor. When sensor readings exceed predefined thresholds—indicating a sudden impact, abnormal tilt, or fall—the microcontroller signals the helmet unit. The helmet then activates a speaker–microphone (or push-button) interface and asks the rider a simple confirmation question. A response within the allotted time is treated as a non-critical event, and the emergency alert is cancelled, reducing false alarms.

B. Emergency Communication and Monitoring

If no response is received, the system assumes the rider may be unconscious or seriously injured and activates the emergency protocol. The GPS module retrieves real-time coordinates, and the GSM module sends an SMS containing the rider's identity, accident status, and location link to registered emergency contacts, guardians, and/or nearby police and ambulance services. Because GSM communication does not require internet access, message delivery remains reliable in remote areas.

C. Advantages

- Enables faster emergency response and works even if the rider is unconscious.
- Reduces delay in medical assistance and dependency on bystanders.

- Provides accurate, real-time location sharing to responders.
- Avoids panic and unnecessary alerts via smart confirmation logic.
- Cost-effective, user-friendly, and suitable for all two-wheeler riders.

IV. SYSTEM ARCHITECTURE

Fig. 1 shows the overall architecture. The bike unit integrates an accelerometer, gyroscope, vibration sensor, GPS module, GSM module, and power supply, all interfaced to a central microcontroller (Arduino/ESP32/NodeMCU). The smart helmet unit contains a separate controller (Arduino Nano) connected to a speaker, microphone, status display, and response buttons. The two units communicate over a wired/wireless IoT link. On confirmed emergencies, alerts are routed to family/guardians, the nearest police station, and ambulance services.

B. Data Flow

At Level 0, the system exchanges registration data, sensor configuration, and emergency-contact management with the Admin; rider information, bike details, GPS requests, and safety/alcohol alerts with the Bike Owner; and accident notifications, GPS coordinates, and emergency SMS with Emergency Contacts and Police/Ambulance Services. At Level 1, the process decomposes into eight sub-processes: User and System Management, Sensor Data Monitoring, Accident Detection and Validation, Helmet Interaction, GPS Location Tracking, Emergency Alert Generation, GSM Communication, and Monitoring/Report Management, supported by User, Sensor Data, Accident Log, GPS Tracking, and Emergency Alert Report data stores. Level 2 further refines these into granular tasks such as accelerometer/gyroscope data collection, MQ-3 alcohol detection, sudden-fall and impact-threshold analysis, confirmation logic, false-alert reduction, GPS signal reception and coordinate extraction, Google Maps link generation, and SMS dispatch to family, police, and ambulance services.

C. Use Case and Class Design

The use-case model identifies four actors—Admin, Bike Owner, Emergency Contact, and Police/Ambulance Service—with core use cases including Detect Accident, Track GPS Location, Detect Alcohol Consumption, Validate Rider Response, Generate Emergency Alert, Send Emergency SMS, Store Accident Logs, and Provide Emergency Assistance. The class diagram comprises entities such as Admin, Owner, Emergency Contact, Accident Detection, GPS

Tracking, Alcohol Detection, and Emergency Alert, each encapsulating attributes and operations (e.g., Manage Users(), Send Emergency SMS(), Generate Alert(), View Booking Details()).

V. REQUIREMENTS AND DATABASE DESIGN

A. Hardware and Software Requirements

Hardware: Arduino/ESP32/NodeMCU main controller, Arduino Nano helmet controller, GSM/GPRS module, GPS module, rechargeable battery with voltage regulator, buzzer, optional OLED/LCD display, push buttons, and bike/helmet mounting enclosures. Software: Arduino IDE/PlatformIO with Embedded C/C++ for firmware; Serial and MQTT/HTTP protocols for communication; AT commands for SMS control; TinyGPS++ for NMEA parsing; PHP with the Laravel framework and MySQL (via XAMPP/WAMP) for the backend; and HTML, CSS, JavaScript, and Bootstrap with RESTful APIs and the Google Maps API for the frontend and geolocation.

B. Database Design

The relational schema comprises seven tables. Table I summarizes the principal entities and their key attributes.

Table	Key Fields	Description
Admin	Admin_ID (PK), Name, Email, Password, Phone	Stores administrator credentials and contact details
Owner	Owner_ID (PK), Name, Email, Password, Bike_Number	Bike owner/ rider profile and vehicle registration
Emergency Contact	Contact_ID (PK), Owner_ID (FK), Name, Relationship, Number	Registered emergency contacts per owner
Accident Detection	Accident_ID (PK), Owner_ID (FK), Date, Time, Location, Alert Status	Logs each detected accident event
GPS Tracking	GPS_ID (PK), Owner_ID (FK), Latitude, Longitude, Tracking_Time	Real-time location records
Alcohol Detection	Alcohol_ID (PK), Owner_ID (FK), Alcohol_Level, Detection_Time, Warning_Status	MQ-3 sensor readings and warnings
Emergency Alert	Alert_ID (PK), Accident_ID (FK), Contact_Number, SMS_Message, Alert_Time, Delivery_Status	SMS dispatch records and delivery status

VI. DETECTION AND ALERT ALGORITHM

The system employs a threshold-based detection and validation algorithm. Accelerometer, gyroscope, vibration, and MQ-3 alcohol sensor values are continuously monitored. If the alcohol level exceeds a threshold, a warning is displayed and vehicle start is inhibited. If an impact or sudden fall is detected, the helmet activates a rider-confirmation prompt and waits for a response. A valid response cancels the alert; otherwise, the system fetches the GPS location, sends an emergency SMS via the GSM module, and stores the alert information in the database.

Algorithm 1: Accident Detection and Emergency Alert 1. Initialize sensors, GSM, and GPS modules. 2. WHILE system active: a. Read accelerometer, gyroscope, vibration, MQ-3 data. b. IF alcohol level > threshold THEN warn rider and disable start. c. IF impact OR sudden fall detected THEN activate helmet confirmation; wait for response; IF response received THEN cancel alert; ELSE fetch GPS location, send emergency SMS, store alert record. 3. END WHILE

VII. SYSTEM IMPLEMENTATION

The implementation comprises a Smart Helmet web application (PHP/Laravel backend, MySQL database, REST APIs), an Admin Dashboard for managing riders, sensors, accident logs, police, hospital, and ambulance records, and an Owner Dashboard for rider registration, emergency-contact management, device configuration, and ride-history viewing. The Accident Detection module continuously evaluates accelerometer, gyroscope, and vibration data against threshold values to filter out false positives from normal riding events such as speed breakers or braking. The Rider Validation module manages the speaker/microphone or push-button confirmation dialogue. The GPS Location Tracking module supplies coordinates for accident logging and Google Maps visualization, while the Emergency Alert and Communication module formats and dispatches SMS messages to family members, police, and ambulance services via the GSM module, independent of internet availability.

VIII. SYSTEM TESTING

Testing was performed at unit, integration, system, performance, and user-acceptance levels. Unit tests verified correct accelerometer readings, successful GSM SMS transmission, and accurate GPS coordinate reception. Integration tests confirmed that helmet activation follows sensor-based detection, that GPS coordinates are correctly embedded in outgoing SMS, and that all modules cooperate during a full trigger sequence. System tests validated end-to-end behavior for simulated accidents, automatic alerting on no response, and alert cancellation on valid rider response. Performance tests confirmed smooth operation under continuous sensor input and multiple simultaneous alerts, while user-acceptance testing confirmed usability and correct message delivery. All fifteen test cases (TC01–TC15) passed. Minor issues identified during testing—sensor-sensitivity false detections, SMS-sending delay, and GPS fetch latency—were resolved, improving detection accuracy, message speed, and location accuracy respectively.

IX. RESULTS

The prototype dashboard displays aggregate statistics (total riders, accidents detected, safe alerts, and emergency cases) along with a chronological log of accident reports showing rider identity, GPS coordinates, map links, status (Safe/Pending/Emergency), and timestamps. A bike-simulation interface visually demonstrates collision detection, a voice-monitor countdown for rider response, and automatic transition to an emergency state with location plotted on an interactive map when no response is received. A model-training view illustrates feature-importance analysis (impact force, vibration, tilt angle, acceleration, gyroscope) and accident-class distribution (normal, minor, major) derived from a sample dataset, supporting future machine-learning-based refinement of the threshold logic.

X. CONCLUSION AND FUTURE ENHANCEMENT

This paper presented an IoT-based smart bike and helmet system that integrates accelerometer, gyroscope, and vibration-sensor data with an interactive helmet and automated GSM/GPS-based emergency alerting. The threshold-based detection and rider-validation algorithm reduces false alarms while ensuring that, when the rider is unresponsive, precise location information reaches emergency contacts and responders without manual intervention. The accompanying Laravel/MySQL web platform provides administration, monitoring, and reporting capabilities. Overall, the system offers a cost-effective, scalable approach to reducing emergency-response time and improving two-wheeler road safety.

Future enhancements include integrating health-monitoring sensors (heartbeat, temperature, oxygen saturation) to assess the rider's physiological condition post-accident, and incorporating power-optimization or solar-assisted charging to extend operational endurance without frequent recharging.

REFERENCES

- [1] J. Han, Y. Liu, Z. Li, Y. Liu, and B. Zhan, "Safety helmet detection based on YOLOv5 driven by super-resolution reconstruction," *Sensors*, vol. 23, no. 4, p. 1822, Feb. 2023.
- [2] C. Huang, X. Zhou, X. Ran, Y. Liu, and W. Deng, "Coevolutionary competitive swarm optimizer with three-phase for large-scale complex optimization problem," *Inf. Sci.*, vol. 619, pp. 2–18, Jan. 2023.
- [3] Z. Li et al., "Toward efficient safety helmet detection based on YOLOv5 with hierarchical positive sample selection and box density filtering," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–14, 2022.
- [4] L. Wang et al., "Investigation into recognition algorithm of helmet violation based on YOLOv5-CBAM-DCN," *IEEE Access*, vol. 10, pp. 60622–60632, 2022.
- [5] K. Han and X. Zeng, "Deep learning-based workers safety helmet wearing detection on construction sites using multi-scale features," *IEEE Access*, vol. 10, pp. 718–729, 2022.
- [6] W. Jia et al., "Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5 detector," *IET Image Process.*, vol. 15, no. 14, pp. 3623–3637, Dec. 2021.
- [7] N. Li et al., "Incorporate online hard example mining and multi-part combination into automatic safety helmet wearing detection," *IEEE Access*, vol. 9, pp. 139536–139543, 2021.
- [8] F. W. Siebert and H. Lin, "Detecting motorcycle helmet use with deep learning," *Accident Anal. Prevention*, vol. 134, Jan. 2020.
- [9] Raj Kamal, *Embedded Systems: Architecture, Programming and Design*, McGraw Hill Education, 2017.
- [10] V. Madiseti and A. Bahga, *Internet of Things: A Hands-On Approach*, Universities Press, 2014.