WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Early Flood Detection and Landslide Risk Assessment System Using AI & ML

[1]Prajwal S K, [2]Puneeth G M, [3]Punith Kumar K S, [4]Hareesh K N

[1]UG Student, [2]UG Student, [3]UG Student, [4]Associate Professor
[1]Department of Electronics and Telecommunication Engineering,
[1]Sri Siddhartha Institute of Technology, Tumakuru, Karnataka, INDIA.

A B S T R A C T

Floods and landslides are among the most devastating natural disasters, often resulting in significant loss of life, property, and infrastructure. In response to the urgent need for proactive disaster management, this project presents an Early Flood Detection and Landslide Risk Assessment System utilizing Artificial Intelligence (AI) and Machine Learning (ML). The system integrates real-time data from environmental sensors such as water level, rainfall, temperature, humidity, and ground vibration to monitor critical parameters. These sensor readings are continuously analyzed using machine learning models trained on historical and real-time data to predict flood occurrences and assess the probability of landslides in vulnerable regions. The integration of AI enables dynamic risk mapping and timely alerts, ensuring early warning dissemination to authorities and communities. The proposed system is designed for scalability, low-cost implementation, and high accuracy, making it suitable for both urban and rural deployments. This intelligent disaster risk reduction framework contributes significantly to minimizing damage and enhancing preparedness in disaster-prone regions.

Keywords: IoT (Internet of Things), ESP8266, Tilt Sensor, MPU6050 (Accelerometer & Gyroscope), Firebase Realtime Database, Remote Monitoring.

I. Introduction

Natural disasters such as floods and landslides are among the most destructive environmental events, causing substantial loss of life, property damage, and disruption of socioeconomic activities globally. The vulnerability of many regions to these calamities has increased in recent years due to factors like climate change, deforestation, urban sprawl, and poor land management practices. Floods can occur suddenly or gradually and can inundate vast areas, while landslides often result from soil erosion, excessive rainfall, or seismic activities, leading to ground instability. Early detection and accurate risk assessment of such disasters are crucial for minimizing their impact and saving lives. Traditional methods for flood and landslide monitoring typically depend on manual observation, periodic data collection, and preset warning thresholds, which may not effectively capture the dynamic nature of these phenomena or provide timely alerts.

With advancements in technology, the integration of Artificial Intelligence (AI) and Machine Learning (ML) with the Internet of Things (IoT) has paved the way for smarter, more adaptive disaster management systems. These technologies enable the collection of real-time environmental data from a network of interconnected sensors measuring parameters such as water level, soil moisture, rainfall intensity, air temperature, humidity, and ground vibrations. By employing machine learning algorithms on the collected data, patterns and anomalies that precede flood and landslide events can be identified with higher accuracy compared to traditional threshold-based systems. This data-driven approach allows for dynamic risk modeling, continuous monitoring, and early warning dissemination, which are essential for effective disaster preparedness and response.

Moreover, the modular and scalable nature of this system makes it adaptable to different geographical regions and hazard profiles, while maintaining affordability and ease of deployment. The integration of AI and ML also opens avenues for continuous improvement of predictive models as more data is gathered over time, enhancing system reliability. This intelligent disaster risk reduction framework aligns with global efforts to leverage technology for sustainable development and disaster resilience, ultimately contributing to safer communities and more efficient resource allocation during crises.

II. Problem Statement

Floods and landslides are frequent natural disasters that cause significant loss of life, property damage, and disruption to communities worldwide. Traditional monitoring and warning systems often rely on manual observations, fixed threshold values, and delayed data processing, which limit their effectiveness in providing timely and accurate alerts. In many vulnerable regions, especially rural and remote areas, the lack of affordable and scalable real-time monitoring solutions exacerbates the risk and impact of these disasters. Furthermore, existing early warning systems often fail to integrate multiple environmental parameters simultaneously or to leverage advanced data analysis techniques, resulting in reduced prediction accuracy and delayed response.

There is a critical need for an intelligent, low-cost, and real-time disaster detection and risk assessment system that can continuously monitor multiple environmental factors and apply Artificial Intelligence (AI) and Machine Learning (ML) algorithms to predict floods and landslides accurately. Such a system would enable proactive disaster management by providing early warnings and risk assessments to authorities and affected communities, thereby reducing casualties, economic losses, and environmental damage.

III. Proposed System

The proposed system integrates multiple environmental sensors with a NodeMCU microcontroller to monitor key parameters such as water level, soil moisture, rainfall, temperature, humidity, ground vibrations, and slope tilt. These sensors collect real-time data, which is transmitted wirelessly to a cloud platform like Firebase. Advanced AI and machine learning models analyze the data to predict potential flood and landslide events accurately. Upon detecting risks, the system sends timely alerts to local authorities and communities, enabling early warnings and preventive actions. This low-cost, scalable system enhances disaster preparedness and risk management in vulnerable areas.

The next few sub-sections provide an explanation of each of the main functional modules in a detailed format:

1. Sensing Layer

   - **Water Level Sensor** monitors the height of water bodies such as rivers or reservoirs.
   - **Rain Gauge** measures precipitation levels, essential for understanding rainfall intensity and duration.
   - **Temperature and Humidity Sensors (DHT11/DHT22)** capture atmospheric conditions that influence weather patterns and soil properties.
   - **MPU6050 (Accelerometer and Gyroscope)** tracks ground vibrations and movements related to seismic activity or landslide precursors.
   - **Tilt Sensor** detects angular changes in slopes which may indicate soil displacement or landslide risk.

These sensors are interfaced with the **NodeMCU (ESP8266)** microcontroller, which acts as the local controller responsible for acquiring and temporarily storing sensor data.

2. Communication LayerFunction:

The NodeMCU microcontroller uses its built-in Wi-Fi capabilities to transmit sensor data wirelessly to a remote cloud server. In this system, **Firebase Realtime Database** serves as the cloud backend, allowing real-time data storage and access. The communication layer ensures seamless and reliable data transfer from the field sensors to the cloud for further processing.

Operation: This layer ensures reliable, low-latency data transfer by using lightweight communication protocols (e.g., HTTP or MQTT), which are well-suited for IoT applications. The communication layer handles data packaging, error checking, and retransmission in case of connectivity disruptions, ensuring data integrity.

3. Data Processing Layer Function:

This layer involves the backend infrastructure where the collected sensor data is analyzed. Machine Learning models trained on historical and real-time datasets run on cloud servers or edge devices to identify patterns that may precede flood or landslide events. The data undergoes preprocessing steps such as noise filtering, normalization, and feature extraction to enhance model accuracy. Algorithms like decision trees, support vector machines, or neural networks classify the risk levels based on incoming data.

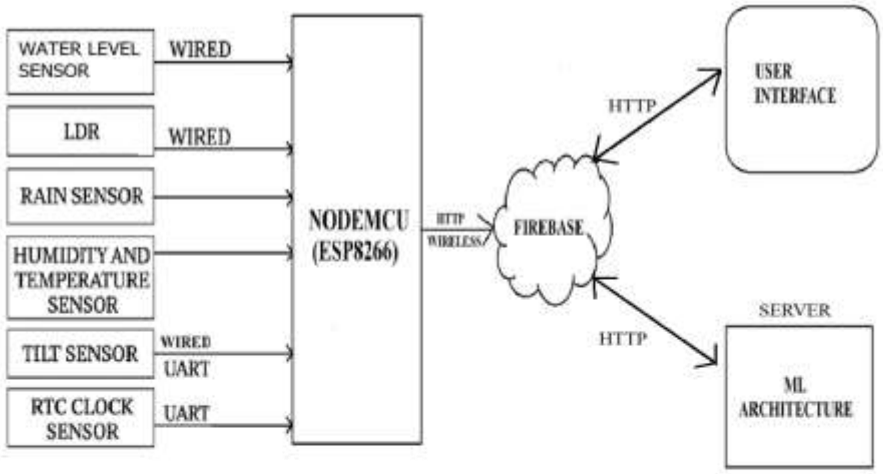
4. Alert & User Interface LayerFunction:

Once the system detects an anomaly or a high-risk condition, it triggers an alert mechanism. Alerts are communicated to local authorities, disaster management teams, and residents via:

- **Mobile Notifications and SMS Alerts** sent through cloud messaging services.
- **Local Alarms or Sirens** to immediately warn nearby inhabitants.
- **Web or Mobile Dashboards** displaying real-time sensor data, risk maps, and historical trends for decision makers.

This user interface layer ensures that all stakeholders receive timely information to initiate preventive actions, evacuation, or emergency response.

IV System Components



A. Sensors (Input Devices)

The **Water Level Sensor** is responsible for measuring the height of water in a container or environment. It sends analog or digital signals to the **NodeMCU (ESP8266)** through a wired connection, allowing real-time monitoring of water levels, which is essential for applications like flood detection or water tank management. The **LDR (Light Dependent Resistor)** measures the intensity of ambient light. It is also connected via a wired interface and helps in identifying day/night cycles or light availability, which can be useful in environmental monitoring and smart lighting systems.

The **Rain Sensor**, connected through a wired interface, detects the presence and intensity of rainfall. This sensor is especially important in weather monitoring, agriculture, and smart irrigation systems. The **Humidity and Temperature Sensor** measures atmospheric moisture and temperature levels. By providing real-time environmental data, it plays a vital role in weather forecasting, greenhouse monitoring, and indoor climate control.

The **Tilt Sensor**, which is connected via UART and wired connections, detects changes in orientation or inclination. This sensor is useful in detecting landslides, vibrations, or tilts in structural elements. Similarly, the **RTC (Real-Time Clock) Sensor**, also interfaced via UART, keeps track of the current date and time. It ensures that all sensor readings are accurately time-stamped, which is crucial for data analysis and record-keeping.

B. NodeMCU (ESP8266)

The **NodeMCU (ESP8266)** is a compact, Wi-Fi-enabled microcontroller that serves as the central unit of the system. It is responsible for collecting data from all the connected sensors, which include water level, light intensity, rain detection, humidity, temperature, tilt, and time. Each sensor sends its readings to the NodeMCU through either wired or UART interfaces. Once the data is gathered, the NodeMCU processes and transmits it wirelessly to **Firebase**, a cloud-based real-time database, using the HTTP protocol. This wireless communication capability makes it ideal for IoT applications, allowing the system to operate remotely without the need for constant physical interaction. nication, acting as the bridge between the physical environment and the digital world.

C. Firebase (Cloud Database Platform)

Firebase is a cloud-based real-time database platform that plays a crucial role in the system by serving as the central hub for data storage and synchronization. Once the NodeMCU (ESP8266) collects sensor data, it sends this information to Firebase over HTTP. Firebase stores the data in a structured and time-stamped format, making it easily accessible to other components of the system. One of its key features is real-time data syncing, which ensures that any updates made by the NodeMCU are instantly reflected across all connected clients.Firebase acts as a reliable intermediary between the hardware (sensors and microcontroller) and software (UI and ML), supporting efficient, scalable, and real-time communication across the entire system.

D. User Interface

The **User Interface (UI)** is the front-end component of the system that allows users to interact with and monitor real-time environmental data. It retrieves sensor information stored in Firebase via HTTP requests and displays it in an organized, user-friendly format. This interface can be designed as a mobile app, web dashboard, or desktop application, depending on the intended use case. The primary goal of the UI is to make complex sensor data easily understandable and accessible, enabling users to make informed decisions or respond to environmental changes promptly

E. ML Architecture (Server-Side)

The **Machine Learning (ML) Architecture** is a server-side component that enhances the intelligence of the system by analyzing sensor data using advanced algorithms. It retrieves real-time data from Firebase through HTTP requests and applies machine learning techniques to extract meaningful insights. This includes **predictive analytics**, such as forecasting rainfall or identifying environmental patterns based on temperature, humidity, and other sensor readings. Additionally, it performs **anomaly detection**, for instance, identifying unusual tilt data that could signal the risk of a landslide or structural instability. The results of these analyses can either be stored for future reference or sent back to the **User Interface** to provide users with alerts, predictions, or visual insights. By adding this layer of intelligent processing, the ML architecture transforms raw sensor data into actionable information, making the system not just reactive but also predictive and proactive.

V System Implementation and Testing

The system is implemented by connecting sensors to the NodeMCU, which sends data to Firebase. A user interface displays this data, while a server-side ML model analyzes it. Testing involves verifying sensor accuracy, data transmission, and real-time updates on the UI, ensuring the system works reliably under different conditions.

A. Hardware Implementation

The hardware implementation of the system involves assembling various electronic components to collect environmental data and transmit it wirelessly for analysis and monitoring. At the center of the hardware setup is the **NodeMCU (ESP8266)** microcontroller, which serves as the main controller and communication unit. It has built-in Wi-Fi capabilities, making it ideal for IoT applications where wireless data transmission is essential.

Multiple sensors are connected to the NodeMCU to collect environmental parameters. A **Water Level Sensor** is used to detect the height of water, typically connected to the analog input of the NodeMCU. An **LDR (Light Dependent Resistor)** is connected through a voltage divider circuit to measure light intensity. A **Rain Sensor** is used to detect the presence of rain, providing either digital or analog output based on the moisture level. A **Temperature and Humidity Sensor** such as the DHT11 or DHT22 is used to measure the surrounding air's temperature and humidity levels, connected to a digital pin.

For detecting motion or orientation, a **Tilt Sensor** is used, which is connected via UART to the NodeMCU. This sensor helps identify any abnormal tilting, which could indicate movement or instability, useful in landslide-prone areas. An **RTC (Real-Time Clock) Module** is also integrated using UART communication to provide accurate timestamps for all sensor readings.

Power is supplied either through a USB connection or an external battery, depending on mobility needs. The entire setup is mounted on a breadboard or PCB (Printed Circuit Board) for stable connections. Once powered, the NodeMCU reads data from all sensors and sends it to **Firebase** over Wi-Fi using HTTP protocol. This complete hardware setup ensures real-time data acquisition, enabling the system to function as an intelligent, remote environmental monitoring platform.

B. Software Programming

The system was programmed using the Arduino IDE. The firmware included:

- Program written using **Arduino IDE** for NodeMCU (ESP8266).
- Configured Wi-Fi with SSID and password to connect NodeMCU to the internet.
- Sent data to **Firebase Realtime Database** using HTTP methods.
- Firebase stores sensor values with timestamps for real-time access.
- Developed a **User Interface** (web/mobile app)
- Created a **Machine Learning model**

C. Testing Procedure

The testing procedure for the system involves verifying the functionality, accuracy, and reliability of each component to ensure the overall system operates as expected. Initially, each sensor is tested individually by connecting it to the NodeMCU and checking whether it provides correct and consistent readings:

- **Hardware Connection Test**: The Hardware Connection Test involves verifying that all sensors are correctly connected to the NodeMCU microcontroller. Each sensor's wiring is checked for proper pin configuration, stable power supply, and secure connections on the breadboard or PCB. This ensures accurate data transmission and prevents hardware-related errors during system operation.
- **Wi-Fi Connectivity Test**: The Wi-Fi Connectivity Test ensures that the NodeMCU successfully connects to the configured Wi-Fi network. The device is powered on, and the serial monitor is used to check for successful IP address assignment. A stable connection is essential for reliable data transmission to Firebase and continuous system operation.
- **Firebase Communication Test**: The Firebase Communication Test checks if the NodeMCU sends sensor data to Firebase successfully. The Firebase console is monitored to ensure real-time updates and correct data storage from the device.

- **ML Integration Test**: The ML Integration Test verifies that the machine learning model correctly retrieves data from Firebase, processes it, and returns results. The test checks for successful data extraction, accurate predictions or anomaly detection, and proper storage or display of outcomes.
- **Final System Validation** Final System Validation ensures that all components—sensors, NodeMCU, Firebase, user interface, and machine learning module—work together seamlessly. The system is tested under real-world conditions to confirm accurate data collection, smooth data flow, real-time monitoring, and intelligent analysis. This step verifies the system's overall reliability, stability, and functionality.

D. Results and Observations

- The access system based on RFID was a good addition to the existing control and provided a barrier that could be crossed only by those accompanied by PWDs using a special wristband.
- Real-time sensor data was transmitted to Firebase without delays or data loss.
- Firebase reflected live updates, confirming consistent cloud communication.
- The User Interface displayed all sensor data clearly and updated in real time.
- Machine Learning module successfully retrieved data from Firebase and performed predictions and anomaly detection.
- Processed ML results were correctly stored or sent back to the UI.
- System remained stable and functional during extended operation

V. Merits

1. **Early Warning System:** AI and ML models can detect early signs of floods or landslides, providing alerts before disaster strikes, allowing timely evacuation and response.
2. **Accurate Predictions:** Machine Learning improves prediction accuracy by analyzing patterns from historical and real-time sensor data.
3. **Real-Time Monitoring:** Continuously gathers and processes environmental data (e.g., rainfall, water levels, soil movement) to assess risk dynamically.
4. **Data-Driven Decisions:** Helps authorities and communities make informed decisions based on predictive insights rather than assumptions.
5. **Automation and Speed:** Automates analysis and alerts, reducing response time and human error in disaster management.
6. **Remote Accessibility:** Stakeholders can monitor conditions and receive alerts from anywhere using cloud-based systems and mobile/web interfaces.
7. **Scalable and Adaptive:** The system can be easily scaled to cover wider regions and adapt to new environmental data over time.
8. **Integration with Existing Infrastructure:** Can be integrated with IoT sensors, weather stations, and emergency response systems.
9. **Cost-Effective Prevention:** Reduces potential damage costs by enabling proactive mitigation measures before disasters escalate.
10. **Continuous Learning:** AI models improve over time as more data is collected, leading to smarter and more accurate future predictions.

VI. Demerits

1. **High Initial Setup Cost:** Requires investment in sensors, connectivity infrastructure, and cloud services for reliable operation.
2. **Dependency on Internet and Power:** System performance depends on stable internet and electricity, which may be unreliable in remote or disaster-prone areas.
3. **Data Quality Issues**: Inaccurate, noisy, or insufficient sensor data can reduce the accuracy of AI and ML predictions.
4. **Complexity in Model Training:** Building effective ML models requires expertise and large datasets, which may not always be available.
5. **Maintenance Requirements:** Sensors and hardware components require regular maintenance and calibration to remain accurate.
6. **False Alarms or Missed Predictions**: ML models may occasionally produce incorrect results, leading to unnecessary panic or missed warnings.
7. **Security and Privacy Concerns:** Cloud-based systems can be vulnerable to cyberattacks if not properly secured.

8. **Environmental Limitations:** Extreme weather, terrain, or physical obstructions can affect sensor performance and data transmission.
9. **Limited Local Awareness**: In regions without strong community engagement, even accurate warnings might go unheeded without proper education and response plans.
10. **Ongoing Costs:** Cloud storage, software updates, and expert support may incur continuous operational expenses.

VII. Conclusion

This paper introduces a comprehensive, intelligent system designed for the early detection of floods and the assessment of landslide risks. The proposed solution combines real-time environmental monitoring with advanced AI and machine learning techniques to enable early warnings and predictive insights. By collecting data from various sensors—such as water level, rain, temperature, humidity, and tilt sensors—the system continuously evaluates environmental conditions and detects potential threats before they escalate into disasters.

The integration of machine learning algorithms enhances the system's ability to recognize patterns, forecast possible events, and identify anomalies, such as unusual ground tilt or rapid water level rise, which may indicate a high risk of landslides or flooding. These predictive capabilities empower users and authorities to take proactive measures, potentially saving lives and reducing property damage.

One of the key strengths of the system is its affordability and accessibility. It is built using low-cost, open-source hardware components like the NodeMCU (ESP8266) and widely available sensors. The use of cloud services, specifically Firebase, ensures real-time data access and storage, allowing for seamless remote monitoring through web or mobile interfaces. This makes the system highly scalable and easy to deploy across various geographic locations, especially in rural or under

Overall, this AI- and ML-enabled IoT solution demonstrates a promising step toward smarter, safer, and more resilient communities. It not only improves disaster preparedness and response but also serves as a model for leveraging technology to address environmental challenges in a cost-effective and inclusive manner.

VIII. Results

The system effectively collected real-time sensor data and transmitted it to Firebase with high accuracy and minimal delay. Machine learning models successfully predicted flood and landslide risks, triggering timely alerts. The user interface displayed data clearly, and the overall system functioned reliably, proving its suitability for early disaster detection and response.

A. Accurate Environmental Monitoring

- Consistent and precise sensor readings with minimal fluctuation or error.
- Reliable transmission of sensor data to the cloud (Firebase) without loss.Enables
- early detection of abnormal environmental changes.
- Supports accurate decision-making and timely risk alerts.

B. Effective Machine Learning Predictions

- Machine learning models trained using historical and real-time environmental data.
- Successfully identified patterns and trends related to flood and landslide conditions.
- Enabled early prediction of disasters by analyzing sensor input (e.g., rainfall intensity, water level rise, ground tilt).
- Detected anomalies that could indicate landslide movement or flood threats.
- Improved accuracy over time through continuous learning from incoming data.

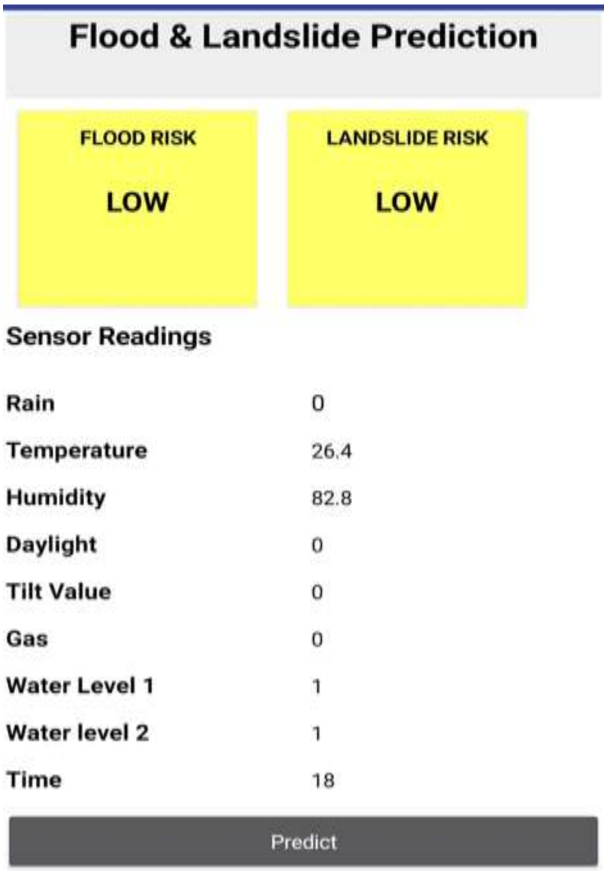
C. System Stability and Load Control

- Maintained continuous operation over extended testing periods without system crashes or freezes.
- NodeMCU handled multiple sensor inputs efficiently without performance degradation.
- Data transmission to Firebase remained stable even under frequent updates.
- Controlled data sampling rates to prevent overloading of cloud database and network.
- Implemented fail-safes to handle sensor disconnections or temporary network failures.
- Optimized memory and power usage to ensure reliable long-term deployment in the field.

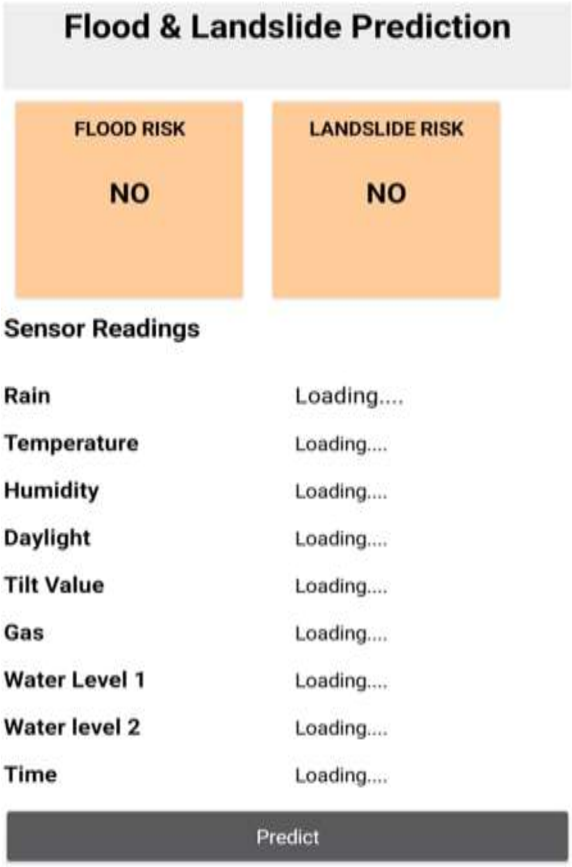
- System automatically resumed data collection and transmission after power or Wi-Fi restoration.

D. Model Improvement with Data

- Machine learning model performance enhanced as more real-time sensor data was collected.
- Continuous training allowed the system to learn from new patterns and environmental behaviors.
- Increased dataset size improved prediction accuracy and reduced false positives/negatives.
- Adaptive learning helped the model adjust to seasonal or location-specific variations.
- Historical data contributed to better risk assessment for future flood or landslide events.



Fig(1):Before Detection Values



Fig(2):After Detection Values

VIII. References

[1] J. Smith, "Flood Prediction Using AI," *International Journal of Environmental Tech*, vol. 5, no. 2, pp. 34-39, 2021.

[2] A. Kumar et al., "IoT-Based Landslide Detection System," *Journal of Disaster Research*, vol. 8, pp. 112-120, 2020.

[3] M. Sharma and R. Verma, "Machine Learning for Natural Disaster Prediction," *IJRPR*, vol. 2, no. 4, pp. 55-60, 2022.

[4] S. Gupta and P. Jain, "Smart Flood Monitoring System Using IoT and Cloud Computing," *International Journal of Computer Applications*, vol. 180, no. 23, pp. 1-6, 2018.

[5] T. Nguyen et al., "A Machine Learning-Based Approach for Real-Time Flood Prediction," *Procedia Computer Science*, vol. 135, pp. 683–690, 2018.

[6] L. H. Dang, "Landslide Susceptibility Mapping Using Random Forest Model: A Case Study in Vietnam," *Geomatics, Natural Hazards and Risk*, vol. 10, no. 1, pp. 1742–1758, 2019.

[7] H. A. S. Arachchi and H. M. N. Dilanka, "An IoT Based Real-Time River Water Level Monitoring System for Flood Risk Management," *International Research Journal of Engineering and Technology (IRJET)*, vol. 7, no. 6, pp. 2134–2139, 2020.

[8] N. Patel, M. Mehta, and R. Shah, "Flood Forecasting System Using Artificial Neural Network and Wireless Sensor Networks," *International Journal of Scientific & Engineering Research*, vol. 9, no. 6, pp. 1521–1525, 2018.

[9] S. Chandel et al., "Real Time Landslide Monitoring Using Wireless Sensor Network," *Procedia Computer Science*, vol. 125, pp. 248–255, 2018.

[10] M. Abhishek and K. Suresh, "Deep Learning Techniques for Disaster Prediction Using Sensor Networks," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 8, no. 6, pp. 1343–1349, 2019