FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 2 | Pages 618-629 | e-ISSN: 2347-470X

IoT-Based Short-Term Water Level Monitoring and Prediction Using Random Forest with MQTT QoS Evaluation

Satryo Budi Utomo*

Electrical Engineering, University of Jember, Jember, Indonesia; Email: satryo@unej.ac.id

***Correspondence:** *Satryo Budi Utomo, krammohanee@gmail.com*

ABSTRACT- Flood-related water level variations are among the most frequent hydrometeorological challenges with high frequency and impact in Indonesia, so an early warning system capable of providing fast, accurate, and real-time information is needed. An Internet of Things (IoT)-based experimental framework is proposed for short-term water level monitoring and prediction. Ultrasonic sensors are used to continuously measure water surface elevation, while NodeMCU functions as the main controller to process data and send information to the MQTT broker using a publish-subscribe architecture. The Random Forest model is trained using sequential water level data (time-series) and evaluated using MSE, RMSE, MAE, and R^2 parameters. The results show that the Random Forest model is able to produce high prediction accuracy with an R^2 value of 0.995488, indicating the model's strong ability to follow the temporal pattern of water level changes. In addition, the performance of the MQTT protocol is analyzed through QoS parameters including throughput, delay, jitter, packet loss, and Round-Trip Time (RTT). The test results show a packet loss value of 0.6% on the publisher-broker path and 1% on the broker-subscriber path, which is categorized as very good for IoT communication. Delay and jitter values are also in a stable range, thus supporting a decision support system for early warning applications. The integration of machine learning methods and the MQTT protocol in this study offers an experimentally evaluated and lightweight implementation framework for an IoT-based short-term water level monitoring and early warning support system. These findings indicate that the combination of Random Forest and MQTT has the potential to be an effective approach to support local monitoring and early warning decision-making through accurate short-term predictions and reliable data transmission. The contribution of this study is experimental, integrative, and evaluative in nature, focusing on system-level performance assessment rather than the development of new algorithms or theoretical models.

Keywords: Laboratory-Scale Early Warning Prototype, Internet of Things (IoT); Random Forest, MQTT Protocol, Ultrasonic Sensor (HC-SR04).

ARTICLE INFORMATION
Author(s): Satryo Budi Utomo;
Received: 25/11/2025; **Accepted**: 29/01/2026; **Published**: 30/06/26;
E- ISSN: 2347-470X;
Paper Id: IJEER 2511A13;
Citation: 10.37391/ijeer.140237
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-14/ijeer-140237.html

Publisher's Note: FOREX Publication stays neutral with regard to jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

Floods are hydrometeorological disasters whose frequency has increased significantly due to global climate change and urban growth [1], [2], [3]. Their impacts not only threaten public safety but also disrupt infrastructure, economic activities, and public services [4], [5], [6]. This situation emphasizes the need for a monitoring system that can provide information quickly, accurately, and easily accessible in real time to support responsive decision-making. [7], [8], [9]. The development of reliable water level monitoring systems to facilitate timely decision-making in flood-prone regions is urgently required, especially in developing countries.

Previous research in the field of the Internet of Things (IoT) shows that many flood monitoring systems only focus on direct water level measurements without predictive capabilities [10], [11], [12], [13]. Such systems are effective for detection but inadequate for providing early warning before critical conditions occur [14], [15]. Furthermore, some studies still utilize demanding communication protocols such as HTTP, which are not optimal for low-power IoT devices [16], [17], [18]. These limitations highlight a research gap related to the integrating short-term water level monitoring and prediction into a single lightweight system that operates in real time [19], [20], [21].

On the other hand, machine learning-based studies have demonstrated high performance in predicting water level dynamics, but most require large computing infrastructure and stable networks [22], [23], [24]. Many previous prediction models rely on large datasets and complex hydrological data sources, making them difficult to implement on small-scale IoT devices [25]. Some studies also fail to consider data communication performance, even though prediction accuracy must be accompanied by reliable information delivery [26], [27], [28]. Machine learning approaches for short-term water level prediction still require optimization in terms of system efficiency and integration with IoT networks.

The Message Queue Telemetry Transport (MQTT) protocol is widely recognized as an efficient solution for Internet of Things (IoT) applications because of its low bandwidth requirements and lightweight publish–subscribe architecture [29], [30], [31], [32], [33]. While previous studies have shown satisfactory MQTT performance in environmental monitoring, systematic evaluations under time-critical conditions are limited [34], [35], [36]. Most existing research focuses on basic data delivery, with few comprehensive analyses of Quality of Service (QoS) parameters such as delay, jitter, throughput, and packet loss [37], [38], [39]. This gap underscores the need for studies that rigorously assess MQTT's end-to-end performance to ensure reliability in time-sensitive IoT-based environmental monitoring systems.

Conceptually, the experimental framework evaluated in this study is designed following an integrated path of sensors, edge, communication, and prediction. Water level measurements are performed at the sensing layer using low-power proximity sensors that generate time-series data at fixed sampling intervals. This data is then processed at the edge device through a lightweight preprocessing mechanism before being forwarded to the network layer. Data transmission uses a publish-subscribe-based communication scheme, and message delivery performance is analyzed based on Quality of Service (QoS) parameters such as latency, jitter, throughput, and packet loss rate. At the application layer, the received time-series data is used by a machine learning-based regression model to evaluate the accuracy of short-term water level predictions. Unlike approaches that optimize each component separately, this framework enables a system-level evaluation of the interrelationships between sensing resolution, communication reliability, and predictive performance under the constraints of the edge network.

This study presents an experimental framework that integrates a Random Forest regression model for short-term water level prediction with MQTT-based data communication in an IoT ecosystem. Instead of proposing a new communication algorithm or protocol, this study evaluates an integrated system-level framework of established machine learning and lightweight communication mechanisms in a network-constrained edge environment. This study also provides insights into the interaction between prediction accuracy, data sampling characteristics, and communication Quality of Service (QoS). Specifically, the analysis focuses on the influence of communication latency, packet loss, and sampling interval on the timeliness and performance of short-term water level prediction in an edge-based IoT system. These findings are expected to provide a reference in designing and developing efficient machine learning-IoT architectures for real-time water level monitoring in network-constrained environments.

The main contribution of this research lies in the development and evaluation of an Internet of Things (IoT)-based experimental framework designed for short-term water level time series monitoring and prediction in network-constrained edge computing environments. This study not only assesses the predictive performance of machine learning models but also performs a system-level analysis of prediction accuracy against communication performance through Quality of Service (QoS) parameters in the MQTT protocol, thus enabling an integrated evaluation of computational and data transmission aspects. In addition, the experimental results provide empirical insights into the relationship between data sampling characteristics, communication latency, and packet loss rates, which can then be used as a general reference in the design and evaluation of lightweight and reliable machine learning–IoT architectures for real-time water level monitoring applications.

2. MATERIALS AND METHODS

An experimental evaluation approach is used to assess the performance of an Internet of Things (IoT) framework in water level monitoring and prediction. The methodology is designed as a layered experimental evaluation pipeline consisting of sensing, edge processing, machine learning modeling, and network performance evaluation. The analysis focuses on two main dimensions: the predictive capability of machine learning models based on water level time series data and the reliability of MQTT communication under limited edge network conditions. Water level data is collected using ultrasonic sensors at the sensing layer, then time-stamped and lightly processed on an edge device that serves as a data acquisition unit and communication gateway to maintain sampling interval consistency and temporal synchronization. At the application layer, the accuracy of time series-based predictions is evaluated using edge-processed data, while the network layer is analyzed through Quality of Service (QoS) parameters to assess the timeliness and reliability of data transmission. All hardware and software components are treated as evaluation instruments, rather than as elements of a new system architecture design, so the analysis focuses on end-to-end performance from data acquisition to prediction validation. This layered approach allows for a systematic and replicable analysis of the interrelationships between sensor measurement quality, edge processing, prediction accuracy, and communication performance, as summarized in the experimental schematic in *figure 1*.

2.1. Materials and Equipment

The hardware components are selected as representative IoT devices to evaluate the sensing stability as well as communication feasibility in short-term water level monitoring under edge network constraints. The HC-SR04 ultrasonic sensor is used to measure the distance between the water surface and the sensor mounting point, with an accuracy of ±3 mm and a detection angle of 15°. The NodeMCU ESP8266 microcontroller is used as a data processing center as well as a wireless transmission module via a 2.4 GHz Wi-Fi connection. In addition, the DS3231 Real-Time Clock (RTC) module is used to maintain data retrieval time synchronization, while the MQTT broker is run using the ThingSpeak cloud platform and the Mosquitto local server for QoS testing.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 2 | Pages 618-629 | e-ISSN: 2347-470X

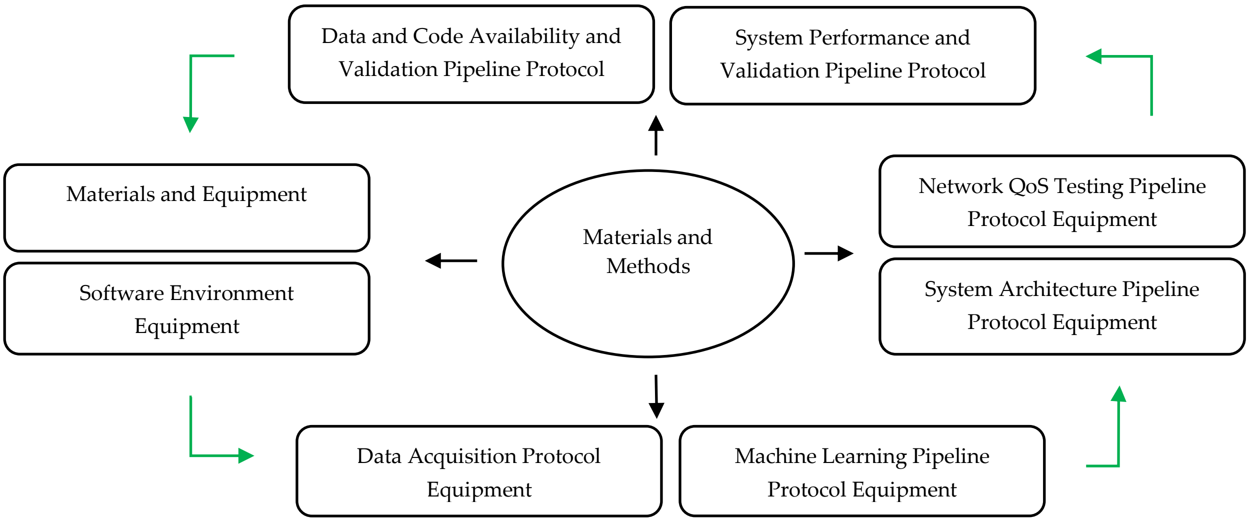


Figure 1. Research Method Scheme

2.2. Software Environment

Programming on the microcontroller was done through the Arduino IDE by utilizing the PubSubClient library as an MQTT communication interface. Machine learning model development and evaluation were carried out on the Google Colab platform using Python 3.10, while data analysis was supported by several libraries such as scikit-learn, pandas, numpy, and matplotlib. MQTT network traffic monitoring and QoS parameter measurements were performed using Wireshark on a Windows environment. All scripts, including data processing, Random Forest implementation, and NodeMCU code, will be made public through the GitHub repository after this article is accepted, without any access restrictions. *Figure 2* shows the architecture of the IoT–MQTT-based water level acquisition and prediction system. The software is used to support the evaluation of prediction accuracy and reproducible communication performance, not to introduce a new framework.

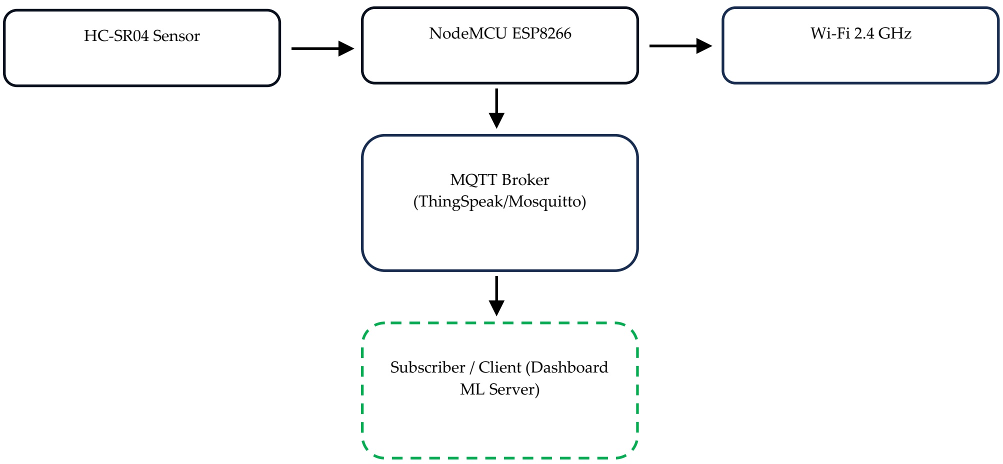


Figure 2. IoT–MQTT-Based Water Level Acquisition and Prediction System Architecture

2.3. Data Acquisition Protocol

This data collection protocol was developed to evaluate the validity of a fixed-interval time-series sampling approach to support short-term water level prediction. Testing was conducted under controlled experimental conditions to ensure repeatability and consistency of the measurement results. Data collection was conducted in a laboratory environment with water level settings varied gradually between 0 cm and 50 cm. The ultrasonic sensor was placed at a fixed position relative to the water surface and took measurements every second, while the NodeMCU forwarded the readings to the MQTT broker *via* a publish-subscribe mechanism. All collected data was stored in a CSV file containing timestamps, measured distance values, and information on environmental conditions during the experiment. This dataset will be made public through the Zenodo or Figshare repositories and equipped with a Digital Object Identifier (DOI) that will be included when the manuscript enters the review stage. The data acquisition protocol schematic can be seen in *figure 3*.

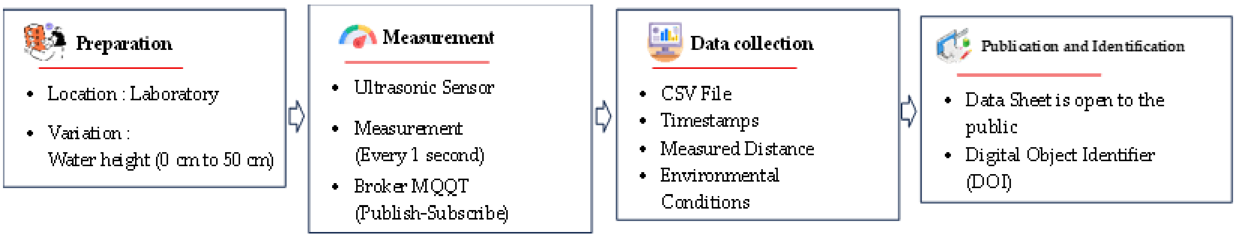


Figure 3. Data Acquisition Protocol Schematic

2.4. Random Forest Model Development

The Random Forest algorithm is used as an ensemble-based regression approach to predict water level dynamics by exploiting temporal patterns from sensor-based time series observations. This model is chosen because it represents an ensemble learning framework capable of assessing prediction accuracy and model stability, particularly in short-term water level prediction in edge-based Internet of Things (IoT) environments. By combining multiple decision trees, Random Forest provides reliable and suitable prediction performance for evaluating short-term water level dynamics. The dataset is divided into 80% training data and 20% testing data using a training-testing split strategy to ensure objective model performance evaluation. Tree growth and node splitting are controlled by Mean Squared Error (MSE) minimization, as defined in Equation (1). Furthermore, key hyperparameter configurations, such as the number of trees and maximum tree depth, are carefully selected to balance prediction performance and computational efficiency, thus keeping the model design in line with practical limitations of edge-level deployments.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (1)$$

Where y_i is the actual water level value and $\hat{y}_i$ is the model prediction output. The entire decision tree is built as many as 200 trees (T=200) with a maximum depth of 15 levels. The final prediction of the Random Forest model for an input x is obtained by averaging the output of all trees, which can be expressed in *equation (2)*.

$$\hat{f}(x) = \frac{1}{T}\sum_{i=1}^{n} f_t(x) \quad (2)$$

The evaluation metrics used include Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and coefficient of determination R^2. Each can be observed in *equations (3), (4), (5)*.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \mid y_i - \hat{y}_i \mid \quad (3)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (4)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (5)$$

The entire training and testing process was run using Python 3.10 with the scikit-learn library. To ensure the research's replicability, all modeling scripts and parameter configurations were documented and stored for public release.

2.5. IoT System Architecture and MQTT Protocol (System Architecture)

An Internet of Things (IoT) system architecture is developed to evaluate the reliability of data streams and message delivery behavior at the sensing, communication, and application layers within an edge-based monitoring framework. The system is designed using a three-layer architecture that separates data acquisition, message handling, and user access functionality. In the data acquisition layer, an HC-SR04 ultrasonic sensor is used in conjunction with an ESP8266 NodeMCU microcontroller to measure distance and perform pre-transmission data processing. The NodeMCU publishes formatted water level data to the MQTT topic /sensor/water level/ using a publish-subscribe communication paradigm. The intermediate layer is handled by an MQTT broker, which distributes messages and ensures reliable delivery to all subscribed clients. In the application layer, the ThingSpeak dashboard and monitoring interface receive the transmitted data for real-time visualization and advanced analytical processing. Communication reliability is evaluated through quality of service (QoS) performance measurements at three different QoS levels, allowing a comparative analysis of message delivery behavior under varying reliability configurations. Additionally, the number of decision trees and the maximum tree depth in the Random Forest model are specifically configured to balance predictive performance and computational load, in line with operational constraints on edge deployments.

2.6. MQTT Performance Evaluation (Network QoS Testing)

MQTT protocol performance testing focused on several metrics considered important in lightweight IoT-based communication systems. Measured parameters included throughput, delay, jitter, packet loss, and Round-Trip Time (RTT). All network traffic data was recorded using Wireshark, with filters limited to MQTT packets for more targeted analysis. Two network conditions were used for comparison: a stable Wi-Fi network and a network intentionally disrupted through packet injection and bandwidth reduction. All observations were stored as PCAP files and will be publicly

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 2 | Pages 618-629 | e-ISSN: 2347-470X

available in a research repository to support the reproducibility of this study. The analyzed Quality of Service (QoS) metrics are used to evaluate the extent to which communication performance affects the timeliness and reliability of short-term water level prediction results delivered to end users. The assessment of Quality of Service (QoS) in this study is positioned as a fundamental component of the machine learning–Internet of Things (IoT) continuum, rather than as a discrete evaluation of network performance. Communication latency, packet loss, and jitter are critical variables that directly impact the timeliness and integrity of sensor data transmitted to predictive modules, thereby affecting the operational efficacy of short-term water level forecasting. By evaluating prediction accuracy under empirically measured QoS conditions, the proposed framework enables a systematic examination of how network reliability constrains or enhances machine learning-based inference in real-time monitoring systems. This QoS-aware evaluation pipeline supports the interpretation of prediction performance within realistic edge-network conditions, rather than presuming optimal data availability.

2.7. System Performance Evaluation (System Performance and Validation)

System validation was conducted in stages to ensure that each component functioned as designed and delivered consistent output. Retesting was conducted in three separate sessions to assess sensor stability, transmission reliability, and the accuracy of the predictive model built with Random Forest. In addition to direct observations of network performance, model accuracy was retested using five-fold cross-validation as an additional step to assess the consistency of predictions on data the model had never seen before. Because this study was entirely device-based and did not involve human or animal participants, no ethics approval was required.

2.8. Data and Code Availability

To facilitate replication and reuse of the research results, the entire dataset, modeling scripts, and network monitoring code will be released in an open repository via GitHub and scientific repository platforms such as Zenodo or Figshare. Published files include the raw data in CSV format, PCAP files of network captures, and all model configurations used during the training process. There are no licensing or access restrictions, so interested parties can use or modify the data as needed. Additional information, such as system circuit diagrams and device demonstration footage, will be included as supplementary materials upon acceptance for publication.

3. RESULTS

3.1. Water Level Sensor Measurement Results

The performance of the HC-SR04 ultrasonic sensor was assessed using two complementary stages. In the accuracy test, the sensor produced an error of 0% to –1% over a range of 55 cm – 57 cm. This value indicates that the sensor's performance is within acceptable tolerance limits and is capable of providing stable distance readings without significant deviation. These results align with the findings of Hanan et al. [40], who reported that the HC-SR04 is able to consistently detect water levels when combined with the ESP8266 module, even under conditions of gradually varying water levels.

In the dynamic testing phase, the sensor was tested through continuous readings of approximately 1,800 samples at 1-second intervals and complete timestamp recording. The measurement results showed that the distance between the sensor and the water surface was in the range of 54 cm to 60 cm, consistent with the initial test conditions. The fluctuations of 1 cm to 2 cm that occurred are typical of ultrasonic sensors, primarily due to the influence of water surface conditions, the angle of incidence of wave reflection, and minor changes in the surrounding environment. The study by Djalilov et al.[41] demonstrated a similar phenomenon, where an ultrasonic sensor-based water level detection system experienced deviations of no more than 1.5% across a range of water level values, thus confirming that small variations in readings are part of the sensor's characteristics, not a form of instability.

Some of the measurement data is shown in *table 1*. The data show that the sensor readings converged back to a dominant value of 55 cm to 57 cm, indicating stable sensor output throughout the test. No anomalous patterns or extreme spikes were found that could indicate device malfunction. This consistency aligns with the research of Sze et al. [42], who compared various ultrasonic sensors and found that the HC-SR04 exhibited high stability over the mid-range, although it was susceptible to small fluctuations due to environmental factors.

Table 1. Ultrasonic Sensor Reading Data

Time	Measured Distance (cm)
22/05/2025 05:23	56
22/05/2025 05:23	57
22/05/2025 05:23	56
22/05/2025 05:23	57
22/05/2025 05:23	56
22/05/2025 05:23	57
22/05/2025 05:23	56
22/05/2025 05:23	55
22/05/2025 05:23	55
22/05/2025 05:23	55

Overall, the measurement results show that the HC-SR04 is able to maintain stable performance in both static and dynamic conditions. The dominant reading range between 55 cm and 57 cm indicates that the sensor does not experience drift and remains responsive throughout the test. The small fluctuations that occur are still within the normal tolerance limits of ultrasonic sensors, as also reported by previous studies. Thus, the sensor reading dataset is declared valid and suitable for use as a basis for training the Random Forest model in the predictive modeling stage.

3.2. Performance of Random Forest Model in Water Level Prediction

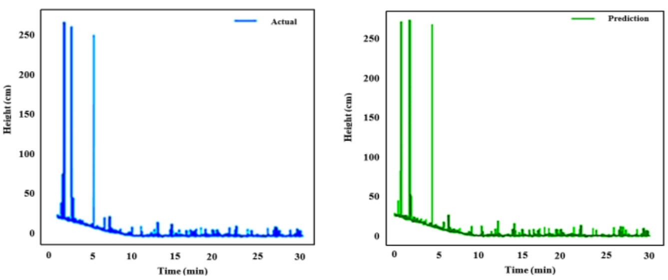
This study systematically evaluates the predictive performance of Random Forest regression for short-term water level prediction using time-series data obtained under controlled sensing conditions. The dataset underwent rigorous cleaning and temporal alignment to maintain internal consistency. An 80:20 train test split was implemented to assess the model’s ability to generalize to previously unseen data. The statistical and temporal characteristics of the sensor observations were preserved throughout the evaluation, ensuring that the reported results accurately reflect actual water level dynamics and sensor performance. Consequently, the resulting prediction performance provides a robust benchmark for future research on machine learning based short-term water level prediction under similar sensing resolutions and sampling protocols.

Model performance was evaluated using four main parameters, Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2). The results of all these parameters are shown in *table 2*. The MSE value of 0.982067, RMSE of 0.990993, and MAE of 0.175011 indicate a very low level of prediction error compared to the range of actual data variation. In addition, the R^2 value of 0.995488 indicates that the model is able to explain more than 99% of the data variation, so its performance can be categorized as very accurate and stable.

Table 2. Random Forest Model Evaluation Performance

Score Metrics			
MSE	**R^2**	**RMSE**	**MAE**
0.982067	0.995488	0.990993	0.175011

The visualization of the relationship between predicted and actual values shown in *figure 4* shows that the predicted curve closely follows the original pattern. At most observation points, the predicted line appears to coincide with the actual value line, indicating that the model is able to follow fluctuating changes in the sensor data. At certain points, especially when sudden fluctuations occur due to ultrasonic sensor noise, the model still provides a consistent prediction response without significant deviations. This demonstrates that Random Forest has good noise tolerance due to the ensemble process that combines multiple decision trees.



(a) (b)

Figure 4. Relationship between (a) actual value and (b) predicted value

Overall, the evaluation results show that Random Forest demonstrates high predictive performance within the controlled experimental conditions of this study. High accuracy, low error rates, and the consistent fit of the prediction curves to actual values demonstrate this model's reliability as a key component in an IoT-based local short-term water level monitoring and early warning support system. Integrating the prediction model with an MQTT-based data delivery system allows for real-time and accurate water level information to be presented to users. With this performance, the Random Forest model is highly suitable for use in the implementation phase of the water level monitoring and prediction framework in this study. While high prediction accuracy is achieved under controlled conditions, the practical value of these predictions in real-time monitoring relies on the reliability and latency of the communication infrastructure. This is assessed in the following QoS analysis.

The high prediction accuracy in this study is attributed to the use of a fixed one-second sampling interval during data acquisition. The high temporal resolution allows the Random Forest model to capture the dynamics of water level changes in detail and mitigates the effects of temporal aliasing that often occurs with less frequent sampling intervals. Although variations in sampling intervals were not explicitly tested, the low error values indicate that a tight sampling interval directly improves short-term prediction performance. These findings emphasize that sampling interval selection is crucial for balancing prediction accuracy and system responsiveness in IoT-based monitoring applications.

3.2.1. Comparison with Baseline Prediction Model

Table 3 compares the performance of three water level prediction methods: Linear Regression, ARIMA, and Random Forest, using MAE, RMSE, and R^2 metrics. All three models were tested on the same dataset, sampling interval, and data sharing scheme, so differences in performance reflect model characteristics, not variations in the data or evaluation. Linear Regression performed the lowest with an RMSE of 8.49 cm, an MAE of 7.23 cm, and an R^2 of 0.709. This R^2 value indicates the model only explains approximately 70.9% of water level variation. While capturing global trends, the high RMSE and MAE values highlight the model's limitations in representing nonlinear dynamics and local fluctuations due to the physical characteristics of the water surface and sensor noise. These results confirm that the assumption of a linear relationship between time-lagged variables is inadequate for water level prediction with high temporal resolution.

Table 3. Comparison of Water Level Prediction Model Performance

Model	RMSE	MAE	R^2
Linear Regression	8,49	7,23	0,709
ARIMA	2,46	1,42	0,975
Random Forest	0,99	0,17	0,995

The ARIMA model performed better than Linear Regression, with an RMSE of 2.46 cm, an MAE of 1.42 cm, and an R^2 of 0.975. A high R^2 value indicates ARIMA's ability to explain most of the time series variation, especially for very short-term predictions. However, ARIMA's RMSE, which was more than double that of Random Forest, indicates that this model is sensitive to sudden fluctuations and measurement noise. This is consistent with ARIMA's characteristics, which rely on the assumptions of stationarity and a linear temporal structure, so its performance decreases when sensor data is unstable or contains unrepresented nonlinear variations. Conversely, Random Forest performed best, with an RMSE of 0.99 cm, an MAE of 0.17 cm, and an R^2 of 0.995. An RMSE approaching one centimetre indicates a very small prediction error compared to actual water level variations, while a low MAE indicates consistent predictions across nearly all test time points. A coefficient of determination close to one confirms this model's ability to accurately represent data variation.

The advantage of Random Forest lies in its ensemble learning mechanism, which combines multiple decision trees to model nonlinear relationships and complex interactions between time lag values. Aggregation between trees also suppresses noise, making the model more robust to ultrasonic sensor fluctuations than conventional statistical approaches. Unlike Linear Regression and ARIMA, which rely on structural assumptions, Random Forest adaptively adjusts the feature space partitioning based on the actual data distribution, making it more robust for high-temporal-resolution sensor monitoring. This performance difference has a significant impact on the implementation of IoT-based early warning systems. Large error values in Linear Regression can lead to delayed or inaccurate water level estimates, thus reducing system reliability. ARIMA, although more accurate, remains susceptible to noise, which affects prediction stability under non-ideal network and sensor conditions. Conversely, Random Forest's high accuracy and stability make it more suitable for integration in real-time IoT–MQTT pipelines, where sensor data must be processed quickly and reliably despite resource constraints and network quality.

Thus, the results in *table 3* not only demonstrate the comparative performance advantage under the evaluated experimental setup but also provide empirical evidence that ensemble learning offers better generalization and robustness to the characteristics of real-world sensor data. These findings strengthen Random Forest's position as the most appropriate prediction model for short-term water level monitoring and prediction systems in edge-based IoT environments with limited network quality and sensor stability.

3.2.2. Feature Importance Analysis

To improve the interpretability of the Random Forest model in predicting water levels, a feature importance analysis was performed based on the temporal structure of the data. The features analyzed were lagged water level values at several previous times, representing the temporal dependence of sensor data. This analysis was interpretive, evaluating the relative contribution of each lag feature to the stability and accuracy of the predictions, without claiming absolute quantitative values for feature importance. This approach was chosen to maintain consistency between model interpretation and the limitations of the experiment with a one-second sampling interval. The analysis results in *table 4* indicate that features with the closest time lags, specifically lag-1 (t−1) and lag-2 (t−2), made the most significant contributions to the predictions. This finding confirms that water level changes are primarily influenced by previous conditions over a short time span, consistent with the gradual nature of small-scale hydrological dynamics with strong temporal dependence.

The contribution of features decreased with increasing lag distance. The lag-3 feature remained moderately influential, while lag-4 and lag-5 contributed little to the predictions. This pattern suggests that long-term historical information does not add significant value to short-term predictions with high time resolution. These findings support the results in *section 3.2.1*, where Random Forest outperforms the baseline model. By adaptively exploiting multiple time lags through ensemble learning, Random Forest can capture relevant temporal relationships without relying on the assumptions of linearity or stationarity of the data. This time lag-based feature importance analysis not only improves model interpretability but also demonstrates that predictive accuracy is driven by the meaningful temporal structure of the data and is consistent with the observed phenomena.

Table 4. Relative Importance of Time Lag-Based Features in the Random Forest Model

Time Lag Feature	Contribution Rate	Interpretation
Lag-1 (t−1)	Water level 1 second prior	Very High
Lag-2 (t−2)	Water level 2 seconds prior	High
Lag-3 (t−3)	Water level 3 seconds prior	Moderate
Lag-4 (t−4)	Water level 4 seconds prior	Low
Lag-5 (t−5)	Water level 5 seconds prior	Very Low

3.3. Visualization of Predictions and Actual Data

A visualization of the relationship between water level distance and time is presented to demonstrate the dynamics of water level changes during the testing process. The graph in *figure 5* illustrates how the distance value read by the HC-SR04 ultrasonic sensor gradually decreases as the water volume in the container increases. The resulting graph pattern indicates that the sensor is able to consistently record both small and large changes in the water level. This visualization serves as an initial validation step that the data obtained has a logical pattern and is in accordance with the physical conditions of the test.

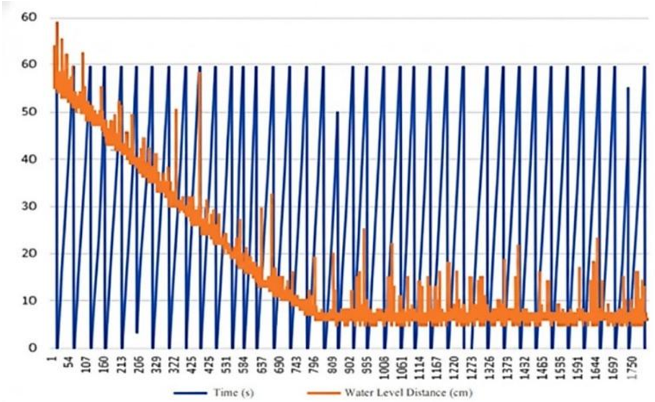


Figure 5. Graph of the relationship between water height (cm) and time (s)

At the beginning of the graph, the sensor distance ranges from approximately 50 cm to 56 cm, indicating that the container is nearly empty, leaving a large distance between the sensor and the water surface. As time passes, the graph shows a fairly smooth downward trend, indicating a gradual increase in water volume. As the tank nears full capacity, the distance decreases to around 9 cm to 10 cm, the lowest limit before the water level reaches a safe distance from the sensor. This gradual change is consistent with the testing mechanism and supports the assumption that the sensor is functioning optimally. Small fluctuations in the graph are a common characteristic of ultrasonic sensors and are caused by the influence of an unstable water level or environmental disturbances during the measurement process. However, there are no extreme spikes or unusual patterns, indicating that noise in the data is within acceptable limits and does not disrupt the overall pattern. This data consistency confirms the validity of the dataset used as a reference for training the predictive model in the previous subsection.

Overall, *figure 5* shows that the HC-SR04 sensor is capable of accurately capturing water level changes and is responsive to water surface dynamics. The clearly and consistently measured distance reduction pattern indicates that the sensor data is of good quality for further processing in a machine learning-based prediction system. This visualization also demonstrates that the sensor is performing optimally within the required measurement range, providing a strong foundation for integrating an IoT-based water level monitoring and prediction framework.

3.4. QoS Analysis for MQTT Protocol

Quality of Service (QoS) testing was conducted to evaluate the performance of the MQTT protocol in data communication between sensor nodes (publishers), brokers, and subscribers. This evaluation aims to determine whether the observed communication performance is adequate to support the timely deployment of machine learning-based water level predictions in a real-time monitoring environment. QoS evaluation is crucial for assessing whether the system can transmit water level information in real time and without interruption. The four main parameters analyzed were throughput, delay, packet loss, and jitter. All tests were conducted using three different IP addresses acting as publisher, broker, and subscriber.

The measurement results from the publisher to broker path are shown in *table 5*. A throughput of 36 bit/s with 0.6% packet loss indicates stable data transmission despite minor packet loss. A delay of 224ms and jitter of 2ms are within acceptable limits for IoT-based monitoring applications. These conditions indicate that the primary transmission path from the sensor to the broker is operating consistently and responsively.

Table 5. QoS Measurement Results from Publisher to Broker (192.168.43.127)

No	QoS Parameter	Measurement Results	Category
1.	Throughput	36 bit/s	Average
2.	Packet loss	0,6 %	Very good
3.	Delay	224 ms	Average
4.	Jitter	2 ms	Very good

The results of the broker-to-subscriber path testing are shown in *table 6*. Throughput increased to 84 bit/s, while the packet loss rate was 1% and the delay was 95.2ms, indicating that communication from the broker to the user device was more stable and faster. A jitter value of 1ms indicates good data transmission stability. Thus, the broker-subscriber path has higher reliability than the previous path.

Table 6. QoS Measurement Results Broker → Subscriber (35.201.97.85)

No	QoS Parameter	Measurement Results	Category
1.	Throughput	84 bit/s	Good
2.	Packet loss	1%	Very good
3.	Delay	95.2 ms	Very good
4.	Jitter	1 ms	Very good

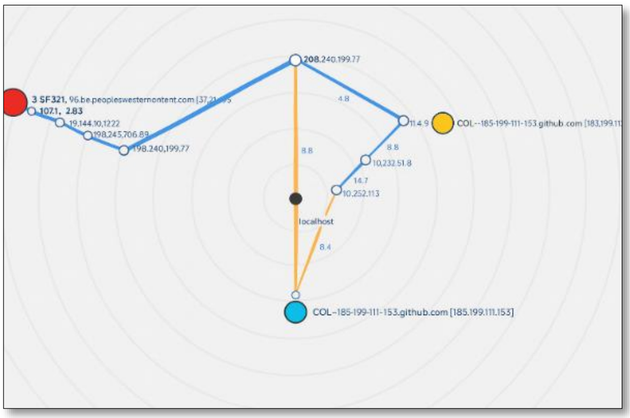
Table 7 shows the QoS test results on the subscriber node. A throughput of 82 bit/s, 2% packet loss, 97ms delay, and 1ms jitter indicate that communication performance remains in the good category. Although packet loss increased slightly, it remained within an acceptable range and did not significantly impact the integrity of the information received by users.

Table 7. QoS Measurement Results for Subscriber (185.199.111.153)

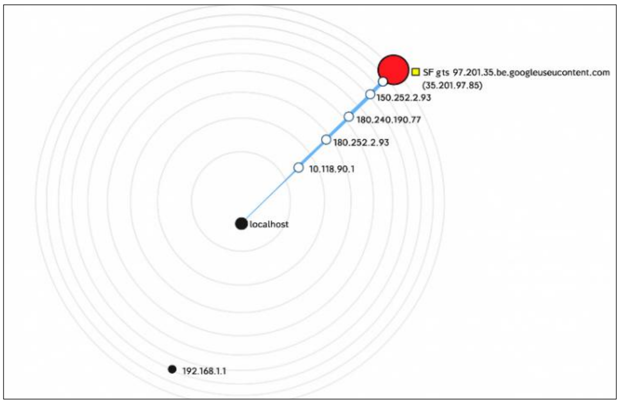
No.	QoS Parameter	Measurement Results	Category
1.	Throughput	82 bit/s	Very good
2.	Packet loss	2%	Very good
3.	Delay	97 ms	Very good
4.	Jitter	1 ms	Very good

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 2 | Pages 618-629 | e-ISSN: 2347-470X

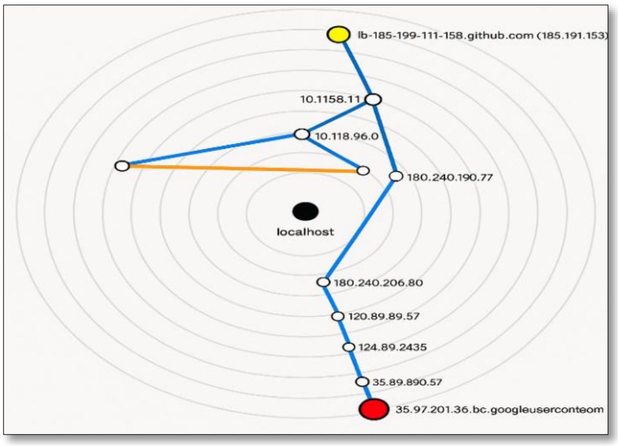
Round Trip Time (RTT) testing is shown in *figure 6(a)* publish host detail (server), *figure(b)* broker host detail, *figure(c)* broker host detail. The throughput value of 82 bit/s, packet loss of 2%, delay of 97ms, and jitter of 1ms indicates that the two-way communication process is running well. The RTT parameter is very important in real-time systems because it describes how quickly data can be sent and responses returned by the system. These results confirm that the MQTT communication mechanism works optimally in the network environment used. The results of the detailed IP Address host topology experiment can be seen in *figure 6*.



(a) Host publish details (server)



(b) Host broker details



(c) Host detail subscribes (client)

Figure 6. Results of the topology host IP Address detail experiment

Quality of Service (QoS) assessments across four representative transmission paths demonstrate that the MQTT protocol remains reliable and efficient in continuously delivering water-level data. Low end-to-end latency, controlled jitter, and minimal packet loss confirm the system's ability to support timely and efficient sensor data distribution under edge network constraints. These findings confirm MQTT as the primary communication protocol for the IoT-based water level monitoring and prediction framework developed in this study and provide a quantitative reference for evaluating MQTT performance in an edge IoT-based hydrological monitoring system. The reported Quality of Service (QoS) outcomes establish a benchmark for future studies evaluating MQTT-based communication performance in edge-oriented Internet of Things (IoT) water-level monitoring systems.

The prediction outcomes in *section 3.2* show that communication latency and packet loss in the MQTT pipeline do not significantly affect short-term prediction accuracy. Packet loss remains below 2%, and end-to-end latency is under one second, which preserves the temporal coherence of input time series for the Random Forest model and maintains feature integrity for accurate inference. System-level measurements of delay and Round-Trip Time (RTT) demonstrate reliable end-to-end delivery of water level information. Network delays are consistently below 250ms across all communication paths, which is well within both the sampling interval and prediction horizon. These results confirm that network-induced delays are not a performance bottleneck and support timely prediction delivery for real-time monitoring and early warning.

3.5. Integration of Water Level Monitoring and Prediction Systems

System-level performance was evaluated by conducting an integrated analysis of the interactions among the sensing, communication, and predictive modules within the proposed water level monitoring and prediction framework. This framework incorporates the HC-SR04 ultrasonic sensor, a NodeMCU-based data acquisition unit, the MQTT messaging protocol, a data storage layer, and the Random Forest regression model. This integration aims to create a system capable of continuous data acquisition, real-time data processing, and producing highly accurate water level estimates. This approach aligns with the IoT–machine learning integration concept, which is widely applied to time series data-based monitoring systems in various fields. Research by Shi et al. [43] shows that an edge cloud flow with intelligent sensors can improve the efficiency and accuracy of time series analysis in large-scale industrial IoT applications, especially when data must be processed quickly to generate real-time decisions.

The integration process begins with the data acquisition stage, where the HC-SR04 sensor measures the distance between it and the water surface at specific time intervals. The data is then sent to the NodeMCU for simple preprocessing, such as rounding values and removing invalid readings. This preprocessing stage is crucial for maintaining signal quality because sensor time series are generally susceptible to noise, as explained by Kumar and colleagues, who studied air quality prediction using IoT sensor data. They emphasized that preprocessing sensor data significantly improves the performance of machine learning-based prediction models. In this study, after processing, the NodeMCU sends the data to an MQTT broker. Based on QoS evaluation results, the publisher-broker and broker-subscriber paths have low latency, stable jitter, and minimal packet loss, ensuring efficient data integration without significant information loss.

The next stage involves integration with a machine learning module. The data received by the broker is forwarded to a server or subscriber application running the model of H. Tyralis and G. Papacharalampous [44], This model demonstrated excellent predictive performance with an R^2 value of 0.995488, indicating that the model is capable of representing patterns of rising and falling water levels with high precision. This finding is consistent with the research of Tyralis and Papacharalampous, who examined the performance of Random Forest for one-step-ahead prediction. They found that Random Forest performed best when using a small number of recent lag variables as predictors, particularly for short-duration time series like those used in this study. This confirms that a simple lag-based input structure can yield high prediction accuracy.

Unlike previous studies that typically separate prediction accuracy and communication performance, this study offers a holistic system-level evaluation by simultaneously assessing machine learning-based prediction fidelity and MQTT communication transmission in an integrated IoT architecture. Unlike Shi et al.'s work, which highlights the efficiency of edge-cloud computing in industrial IoT, this study specifically analyses the impact of communication latency and packet loss on short-term water-level prediction performance. The results show that the proposed system maintains operational consistency across the sensing, transmission, and predictive processing layers, enabling data to flow smoothly from microcontroller-based acquisition to Random Forest inference. The integration of lightweight MQTT messaging with ensemble learning-based prediction results in a responsive and reliable monitoring framework, aligning with the principle of edge-cloud workload distribution and reinforcing the findings of Kumar et al. [45], on the importance of a robust IoT pipeline. Overall, these findings emphasize the importance of integrated design across sensing, communication, and prediction, and demonstrate the potential of an integrated IoT–machine learning architecture to enhance local monitoring and early warning support for short-term water level changes in sensor-based environments. These findings suggest that the co-design of sensing resolution, communication Quality of Service, and machine learning models is critical for ensuring the effective deployment of real-time IoT-based monitoring systems under network-constrained conditions.

While the study demonstrates good system performance under controlled experimental conditions, these findings must be understood in the context of the limitations of the sensing resolution, network configuration, and prediction horizon used. The systemic analysis highlights the importance of integrated design between data acquisition strategies, communication quality of service, and machine learning models. These lessons are general and applicable to other real-time environmental monitoring applications running on resource-constrained edge networks, regardless of specific hardware.

4. DISCUSSION

4.1. Overfitting Risk Analysis and Bias–Variance Trade-off

The very high coefficient of determination ($R^2 \approx 0.995$) may raise concerns about overfitting. However, this result needs to be understood in the context of the experimental design and the prediction horizon used. The dataset was collected in a controlled laboratory setting with smooth water level variations and low noise levels, thus maintaining a stable time series structure. The Random Forest model naturally mitigates the risk of overfitting through ensemble averaging, which reduces model variance compared to a single decision tree. The consistency of model performance across multiple testing sessions and cross-validation demonstrates good prediction stability in this experimental domain. The authors acknowledge that the reported accuracy represents an upper limit of model performance under controlled conditions. In field applications, more abrupt hydrological changes, environmental disturbances, and sensor degradation can increase prediction uncertainty. From a bias–variance perspective, the model configuration focused on low variance for short-term predictions, at the expense of increased computational complexity. This trade-off is considered reasonable because the inference process is performed on servers or edge nodes with more resources than sensor nodes. These results should be interpreted as an evaluation of model suitability under controlled experimental conditions, rather than evidence of algorithmic or theoretical novelty

4.2. Limitations and Scope of Generalization

This study has several limitations that need to be considered. First, the experimental data were obtained from a laboratory environment with controlled water level changes, so the results do not fully represent the complexity of real-world flood conditions involving rainfall, runoff, and watershed dynamics. Second, the prediction scope is limited to short-term data using local sensor data, without considering external hydrometeorological variables. Therefore, the developed system is more suitable as a prototype for local-scale water level monitoring and prediction, rather than as a comprehensive flood forecasting system. However, the proposed experimental framework makes a significant contribution to system design, particularly in the integration of sensing resolution, communication service quality, and machine learning-based prediction performance in limited IoT networks. These findings are general at the system level and can be used as a reference for the development of early warning components in other IoT-based environmental monitoring applications.

5. CONCLUSIONS

This research successfully designed and implemented a water level monitoring and prediction system based on IoT and the Random Forest machine learning algorithm. Based on testing results, the system is capable of performing data acquisition, information transmission, and real-time water level estimation with stable and accurate performance. The HC-SR04 ultrasonic sensor provided consistent distance data with minimal measurement deviation, making it suitable for use as the primary data source in predictive modeling. The distance-time relationship graph also shows that the sensor responds responsively to changes in water level and exhibits no significant anomalies.

The Random Forest algorithm used as a predictive model demonstrated excellent performance with a coefficient of determination (R^2 = 0.995488) and low prediction error, as demonstrated by the MSE, RMSE, and MAE values in *table 2*. This high accuracy indicates the model's good generalization ability in representing water level change patterns based on ultrasonic sensor data. The visualization of predictions and actual data also demonstrates strong agreement, making Random Forest a reliable predictive module in IoT-based early warning systems.

Quality of Service (QoS) testing of the MQTT protocol demonstrated that the communication channel between publisher, broker, and subscriber had stable throughput, minimal packet loss, and low delay and jitter. Thus, the communication system maintained consistent performance throughout the monitoring process. Good QoS parameters indicate that MQTT is capable of supporting real-time data transmission requirements, enabling its integration with sensors and predictive models to function smoothly.

Overall, the integration of sensors, microcontrollers, the MQTT protocol, and the Random Forest model successfully created a reliable, efficient, and responsive water level monitoring and prediction system. This study contributes as an experimental and integrative evaluation of an IoT–machine learning pipeline for short-term water level monitoring under controlled conditions. The success of this research demonstrates that an IoT-based approach combined with machine learning can improve the accuracy and speed of detecting changes in water conditions, making it a relevant technological solution for future experimental studies on local water level monitoring and early warning support systems. The findings provide evaluative insights that may serve as a reference for future experimental studies in similar edge-based monitoring contexts.

Acknowledgements: The authors gratefully acknowledge the financial support provided by the RIIM LPDP–BRIN Research Grant under Grant No. **118/IV/KS/07/2025**. The authors also sincerely thank Universitas Jember for its institutional support and PT. Realworldwide Wholesale Indonesia for their valuable collaboration and technical support throughout this research.

Funding Source: "This research received no external funding"

Conflicts of Interest: "The authors declare no conflict of interest."

REFERENCES

[1] M. N. K. Boutros, A. Meroth, N. Sharaf, and Y. A. Zaghloul, "Development of a MQTT-based server software Agriculture Precision," 2024.

[2] C.-H. Liu, T.-H. Yang, and O. T. Wijaya, "Development of an Artificial Neural Network Algorithm Embedded in an On-Site Sensor for Water Level Forecasting," vol. 22, pp. 1–22, 2022.

[3] H. J. J. Ochoa, R. Peña, Y. L. Mezquita, E. Gonzalez, and S. Camacho-leon, "Comparative Analysis of Power Consumption between MQTT and HTTP Protocols in an IoT Platform Designed and Cold Chain Transport Operations," 2023.

[4] H. Nurwarsito and M. A. S. Alkaf, "Development of LoRa Multipoint Network Integrated with MQTT-SN Protocol for Microclimate Data Logging in UB Forest," vol. 17, no. 1, pp. 1–16, 2025.

[5] U. B. Iyekekpolo, F. E. Idachaba, and S. I. Popoola, "Early Flood Detection and Monitoring System Based on Wireless Sensor Network," vol. 2011, no. October 2010, pp. 1381–1394, 2018.

[6] S. A. S. Bukhari, I. Shafi, J. Ahmad, H. T. Butt, T. Khurshaid, and I. Ashraf, "Enhancing flood monitoring and prevention using machine learning and IoT integration," no. 0123456789, 2024.

[7] E. Shahri, P. Pedreiras, and L. Almeida, "Extending MQTT with Real-Time Communication Services Based on SDN," pp. 1–19, 2022.

[8] M. Mousa, X. Zhang, and C. Claudel, "Flash Flood Detection in Urban Cities Using Ultrasonic and Infrared Sensors Flash Flood Detection in Urban Cities Using Ultrasonic and Infrared Sensors," 2025, doi: 10.1109/JSEN.2016.2592359.

[9] T. Khan et al., "Flash Floods Prediction using Real Time data : An Implementation of ANN-PSO with less False Alarm," pp. 1–6, 2019.

[10] J. G. Natividad and J. M. Mendez, "Flood Monitoring and Early Warning System Using Ultrasonic Sensor Flood Monitoring and Early Warning System Using Ultrasonic Sensor," 2017, doi: 10.1088/1757-899X/325/1/012020.

[11] M. Siddique, T. Ahmed, and M. S. Husain, "EAI Endorsed Transactions Flood Monitoring and Early Warning Systems – An IoT Based Perspective," vol. 9, no. 2, pp. 1–10, 2023.

[12] M. Wajid and A. Raza, "Flood Prediction System Using IOT & Artificial Neural Network," 2024, doi: 10.21015/vtse.v12i1.1603.

[13] A. Hanif and R. Amri, "Implementasi Internet Of Things Pada Protokol MQTT Dan HTTP Dalam Sistem Pendeteksi Banjir," pp. 489–501, 2023.

[14] J. H. Emon, S. M. Hussain, A. Islam, and S. S. Ahmed, "Integration of IoT , Machine Learning , and Sensors for Intelligent Environmental Monitoring and Agricultural Development," vol. 2025, 2025.

[15] S. Kushwaha and R. Pandey, "Intelligent IoT-Based Water Quality Monitoring and Predictive Analysis Using Machine Learning," pp. 8–16, 2025.

[16] V. De Paul, N. Kwizera, Z. Li, V. E. Lumorvie, F. Nambajemariya, and X. Niu, "IoT Based Greenhouse Real-Time Data Acquisition and Visualization through Message Queuing Telemetry Transfer (MQTT) Protocol," pp. 77–93, 2021, doi: 10.4236/ait.2021.112006.

[17] C. Moreno et al., "RiverCore: IoT Device for River Water Level Monitoring over Cellular Communications," vol. 19, no. 127, pp. 1–21, 2019, doi: 10.3390/s19010127.

[18] B. Nemade et al., "IoT-based automated system for water-related disease prediction," pp. 1–30, 2024.

[19] M. E. Rana, "IoT-Based Flood Monitoring System Using Machine Learning Approach," 2023.

[20] C. T. Selvi, R. S. S. Subramanian, M. M. Krishnan, and P. G. Priya, "IoT-Enabled Flood Monitoring System for Enhanced Dam Surveillance and Risk Mitigation," vol. 6, no. 3, pp. 144–153, 2024.

[21] C. Mydlarz et al., "FloodNet: Low-Cost Ultrasonic Sensors for Real-Time Measurement of Hyperlocal, Street-Level Floods in New York City," Adv. Earth Sp. Sci., pp. 1–22, 2024.

[22] V. Uddameri and E. A. Hernandez, "Machine Learning for Flood Resiliency — Current Status and Unexplored Directions," 2025.

[23] T. Khan, M. Alam, and M. S. Mazliham, "Artificial Intelligence Based Multi-modal Sensing for Flash Flood Investigation," pp. 40–47, 2018.

[24] I. R. Widiasari, L. E. Nugroho, and Widyawan, "Deep Learning Multilayer Perceptron (MLP) for Flood Prediction Model Using Wireless Sensor Network based Hydrology Time Series Data Mining," 2023.

[25] E. D. Wardihani, E. U. Sari, A. S. Nugroho, Y. Badruzzaman, and A. Nursyahid, "Monitoring and Controlling of IoT-Based Greenhouse Parameters W ith the MQTT Protocol," vol. 13, no. 1, pp. 38–43, 2024.

[26] J. A. Coelho and A. Glória, "Precise Water Leak Detection Using Machine Learning and Real-Time Sensor Data," pp. 474–493, 2020, doi: 10.3390/iot1020026.

[27] K. Mohamed et al., "Real-time flood forecasting in Amo Chhu using machine learning model and internet of things and internet of things," vol. 1916, 2024, doi: 10.1080/23311916.2024.2370900.

[28] J. L. R. Neypes, M. Sultana, and A. J. R. Miyajima, "Real-time Flood Surveillance and Early Warning System using IoT- AI Fusion for Flood Prediction : Smart Framework to Monitoring Solutions," vol. 6, no. 4, pp. 30–40, 2024.

[29] M. Mavroforakis, H. Georgiou, V. Malisiovas, C. Psychias, and K. Sarioglou, "' Water Underground ' : Real-time , continuous monitoring of the underground water ' s quantity and quality .," no. September, 2021.

[30] J. Song, Z. Shao, Z. Zhan, and L. Chen, "State-of-the-Art Techniques for Real-Time Monitoring of Urban Flooding : A Review," 2024.

[31] M. A. Rahu et al., "Toward Design of Internet of Things and Machine Learning-Enabled Frameworks for Analysis and Prediction of Water Quality," vol. 11, no. September, pp. 101055–101086, 2023.

[32] G. Modalavalasa, "Towards Sustainable Development Based on Machine Learning Models for Accurate and Efficient Flood Prediction," vol. 8, no. 2, pp. 940–944, 2021.

[33] S. Salama, H. Hajjaj, M. Thariq, and H. Sultan, "Utilizing the Internet of Things (IoT) to Develop a Remotely Monitored Autonomous Floodgate for Water Management and Control," water, vol. 12, no. 502, pp. 1–18, 2020.

[34] T. R. Mahesh et al., "Water quality level estimation using IoT sensors and probabilistic machine learning model," vol. 55, no. 7, pp. 775–789, 2024, doi: 10.2166/nh.2024.048.

[35] G. Al-Rawas, M. R. Nikoo, M. Al-Wardy, and T. Etri, "A Critical Review of Emerging Technologies for Flash Flood Prediction: Examining Artificial Intelligence, Machine Learning, Internet of Things, Cloud Computing, and Robotics Techniques," pp. 1–28, 2024.

[36] K. Das, S. Ghosh, and H. S. Dutta, "A Machine Learning and IoT based approach for Flood Monitoring and Prevention," vol. 12, no. 5, pp. 11–19, 2022, doi: 10.9790/9622-1205021119.

[37] L. Mdegela, Y. De Bock, E. Municio, E. Luhanga, J. Leo, and E. Mannens, "A Multi-Modal Wireless Sensor System for River Monitoring : A Case for Kikuletwa River Floods in Tanzania," pp. 1–23, 2023.

[38] M. Mousa, E. Oudat, and C. Claude, "A Novel Dual Traffic / Flash Flood Monitoring System Using Passive Infrared / Ultrasonic Sensors A Novel Dual Traffic / Flash Flood Monitoring System Using Passive Infrared / Ultrasonic Sensors," 2015.

[39] M. Abdurohman, A. J. I. G. Putrada, and M. M. A. T. Deris, "A Robust Internet of Things-Based Aquarium Control System Using Decision Tree Regression Algorithm," pp. 56937–56951, 2022.

[40] Hanan, A. A. N. Gunawan, and M. Sumadiyasa, "Water Level Detection System Based on Ultrasonic Sensors HC-SR04 and ESP8266-12 Modules with Telegram and Buzzer Communication Media," vol. 18, no. 3, pp. 305–309, 2019.

[41] A. Djalilov, E. Sobirov, O. Nazarov, S. Urolov, and I. Gayipov, "Study on automatic water level detection process using ultrasonic sensor," 2022, pp. 1–9. doi: 10.1088/1755-1315/1142/1/012020.

[42] E. Sze, D. Hindarto, I. K. A. Wirayasa, and Haryono, "Performance Comparison of Ultrasonic Sensor Accuracy in Measuring Distance," vol. 6, no. 4, pp. 2556–2562, 2022.

[43] F. Shi, L. Yan, X. Zhao, and R. X. Gao, "Machine Learning-Based Time-Series Data Analysis in Edge-Cloud-Assisted Oil Industrial IoT System," vol. 2022, 2022.

[44] H. Tyralis and G. Papacharalampous, "Variable Selection in Time Series Forecasting Using Random Forests," 2017, doi: 10.3390/a10040114.

[45] R. Kumar, "Time Series Data Prediction using IoT and Machine Learning Technique," vol. 00, no. 2019, 2020.