Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 1 - 14
https://journal.physan.org/index.php/jocpes/index

Design and Development of an IoT-Based UV Meter and Dual-Axis Solar Tracker for Real-Time UV Index Monitoring Optimization

Dwi Agustian[1], Ilham Muthahhari[2], Valiant Yuvi Syahreza[3], Anton Widodo[4], Muchamad Rizqy Nugraha[5], Edward Trihadi[6]
[1,2,3,4,5,6]Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)

Article Info

Article history:

Received March 1, 2025
Revised March 6, 2025
Accepted March 7, 2025

Keywords:

UV Index Monitoring
Internet of Things (IoT)
Dual-Axis Solar Tracker

A B S T R A C T

The ultraviolet (UV) radiation emitted by the sun has both positive and negative impacts on human life. Excessive exposure to UV rays can lead to various health issues, such as skin cancer and cataracts. Therefore, UV radiation monitoring becomes crucial, especially in the face of climate change, which may increase the intensity of UV radiation due to the depletion of the ozone layer. This study aims to design an Internet of Things (IoT)-based UV index monitoring system, equipped with a dual- axis solar tracker to optimize UV index measurements. The system utilizes the ESP32 microcontroller as the main processing unit, the UVM-30A sensor to detect UV radiation, and the DS3231 Real Time Clock (RTC) module for time synchronization. UV index data is displayed in real-time through a Liquid Crystal Display (LCD) screen and the Blynk platform for easy remote access. Test results show that the system performs well, with a low relative error compared to UV index data from the reference site uvindex.app. This system provides an innovative solution for efficient and real-time UV index monitoring, which can increase public awareness about the dangers of UV radiation.

Corresponden Author:

Dwi Agustan,
Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)
Tangerang City, Banten, Indonesia
Email: adwiagustian2003@gmail.com

1. INTRODUCTION

The sun is essentially a source of heat radiation that emits energy across a very broad wavelength range [1]. The relative intensity of UV radiation and visible light reaching the Earth's surface is greatly influenced by the atmosphere, which absorbs and scatters most of this radiation. At wavelengths below 320 nm, the intensity of UV radiation decreases drastically due to absorption by the stratospheric ozone layer [2]. Radiation with wavelengths below 288 nm almost does not reach the Earth's surface. Therefore, most of the known biological effects of solar radiation are related to the very short portion of the solar spectrum, which is less than 1.5% of the total solar energy reaching the Earth's surface.

Excessive exposure to ultraviolet (UV) radiation from the sun can cause serious health effects in humans, such as skin cancer, cataracts, and premature aging. Other harmful effects include various biological disorders, such as malignancies on the eyelids (for instance, basal cell carcinoma (BCC) and squamous cell carcinoma (SCC)), photokeratitis, pterygium, and cortical cataracts [3],[4].

As reported by the World Health Organization (WHO) in 2018, there are an estimated 2–3 million cases of non-melanoma skin cancer and 132,000 cases of melanoma each year worldwide, caused by excessive UV exposure [20]. Ultraviolet (UV) radiation is a form of electromagnetic radiation with wavelengths ranging from 100 to 400 nm, classified into three categories: UV-A (315–400 nm), UV-B (280–315 nm), and UV- C (100–280 nm [5].

Although UV radiation has negative health impacts, UV rays also have important benefits for life on Earth [6]. Since ancient times, UV rays have been believed to enhance the human immune system, strengthen bones and muscles, and be used in the treatment of skin diseases such as atopic dermatitis, psoriasis, localized scleroderma phototherapy, and vitiligo [7][8].

However, the dual nature of UV radiation, with both positive and negative effects, raises concerns about the health risks it poses. A study by Leal A.C. and colleagues noted an increase in health issues due to UV exposure, especially in tropical and subtropical regions [9].

In the 2015 UV Index meeting in Melbourne, Australia, it was agreed that raising awareness of the importance of protection from UV exposure, particularly for UV Index values of 3 and above, remains a significant public health issue. Although the UV Index (UVI) is not a standalone tool, its use can be more effective when combined with a more comprehensive sun protection strategy [10].

The Global UVI represents the intensity of UV radiation from the sun that reaches the Earth's surface, with values starting from zero and increasing. The higher the UVI value, the greater the potential for skin and eye damage, and the quicker the time required for damage to occur [11]. This index is designed to help the public understand the dangers of UV exposure and is widely used in public health campaigns to increase awareness about sun protection [12]. McKinlay and Diffey's research (1991) showed that the UVI can help the public take preventive actions based on predicted exposure levels [13]. Therefore, the UVI plays an important role in raising public awareness of UV exposure risks and reminding people to take necessary protective measures [14].

For these reasons, monitoring the UV index is crucial, especially during the current global climate change period, which affects the ozone layer and increases the intensity of UV radiation [12][15]. The use of an Internet of Things (IoT)-based UV index monitoring system offers the opportunity for real-time and continuous monitoring through UV sensors connected to cloud devices, enabling global data analysis and distribution, as well as providing innovative solutions for real-time data collection, which is very useful for environmental monitoring [16].

A solar tracker is a device that allows an LDR sensor to follow the movement of the sun to maximize energy absorption. There are two types: single-axis and dual-axis, with the dual- axis being more efficient because it can track the sun's position both vertically and horizontally [17][18]. Using the ESP32 as a controller in an IoT-based UV index reading system allows data to be collected and transmitted via Wi-Fi networks, enabling users to obtain real-time UV radiation data [19]. Several studies have shown that IoT-based environmental monitoring systems have successfully provided accurate data, helping relevant authorities make policies based on the monitoring results.

This study aims to design a system that provides real-time information related to the UV index, utilizing Internet of Things (IoT) technology to improve monitoring efficiency. This device integrates UV sensors with a dual-axis solar tracker system to optimize measurements by following the sun's movement, thus enabling more accurate UV index readings. The system is also equipped with the ability to send data directly to the Blynk platform, enabling remote monitoring that facilitates analysis and decision-making related to UV exposure.

2. METHODOLOGY

In this study, the author employed several stages, including system design, hardware design, and software design.

2.1 UV Index Threshold

Table 1 UV Radiation Exposure Categories

UV Index Range	Category	Color
< 2	Low	Green
3 – 5	Moderate	Yellow
6 – 7	High	Orange
8 – 10	Very High	Red
11+	Extreme	Violet

Table 1 shows the threshold values of the UV index used in the equipment system. These thresholds are divided into five categories to classify the UV index intensity: Low, Moderate, High, Very High, and Extreme. These categories help in clearly and structurally identifying the level of UV exposure.

2.2 System Design

The UV Index data processing results obtained from the designed device can be accessed in real-time through the Blynk platform. This allows users to monitor the device's status and the measured UV Index data. This enables efficient and practical monitoring without the need for direct physical interaction with the device.

The concept of this system will be explained in more detail through a block diagram and flowchart, illustrating the workflow and interaction between components in the system. The block diagram consists of three main interconnected and supporting sections: input, process, and output.

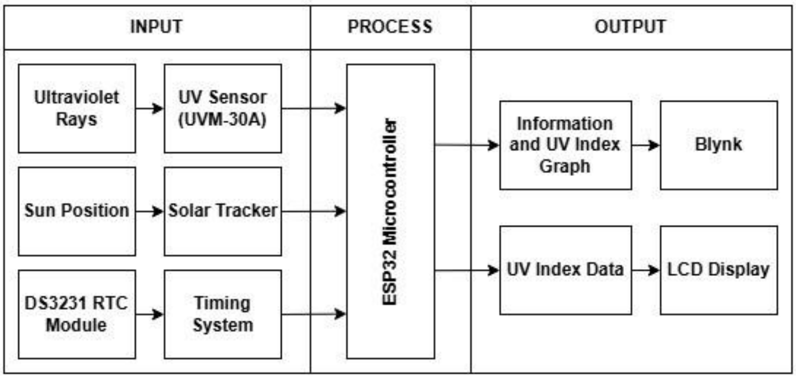

Fig. 1 Block Diagram System

2.2.1 Input

In this system, the inputs are the UV sensor, solar tracker, and timing system, which function to:

a. UV Sensor (UVM-30A)

The UV sensor is used to detect the UV index, which is the result of processing the output voltage generated by the sensor. The ESP32 microcontroller processes the voltage read from the sensor to calculate the UV index value according to the predetermined range. Each voltage value detected is processed into a UV index within the specified range.

b. Solar Tracker

The solar tracker consists of two SG90 servo motors and four LDR sensors. Each servo motor serves a different purpose: the vertical servo moves the panel vertically, while the horizontal servo moves the panel horizontally. The four LDR sensors are positioned in parallel at the top, bottom, left, and right, with barriers between each sensor to facilitate the detection of sunlight intensity. Each LDR sensor measures the intensity of light it receives, and based on the data from these sensors, the servo motors adjust the position of the solar panel to point toward the LDR sensor detecting the highest light intensity. This ensures that the UV sensor always faces the sun optimally.

c. Timing System

This device uses the DS3231 RTC module as the time management system, which functions to display real-time time information during operation. The module provides complete time information, including date, month, year, hour, minute, and second.

2.2.2 Process

The process in this system involves the ESP32 microcontroller, which functions to process data received from the input to produce the desired output. The ESP32 microcontroller processes data in the form of voltage output from the UV sensor, which is then converted into UV index values according to the predefined range. This data is displayed on an LCD screen and sent to the Blynk application in real-time.

Additionally, the ESP32 processes sunlight intensity data received from the LDR sensor, which then drives the servo motor to track the direction of sunlight.

2.2.3 Output

The output generated by this device consists of data displayed on an LCD and the Blynk application. The functions of each are as follows:

a. LCD 20x4

The LCD serves to display processed data from the ESP32, such as real-time UV index and time information. This LCD facilitates observers in monitoring the UV index during field observations, providing clear and easily accessible information directly on the device.

b. Blynk

Blynk functions to display UV index data, UV categories, reading time, and graphs on a web-based platform in real-time. The Blynk application can be accessed via laptops, computers, or mobile devices, offering convenience for users to monitor data remotely.

The system design is illustrated in the flow diagram shown in the image below:

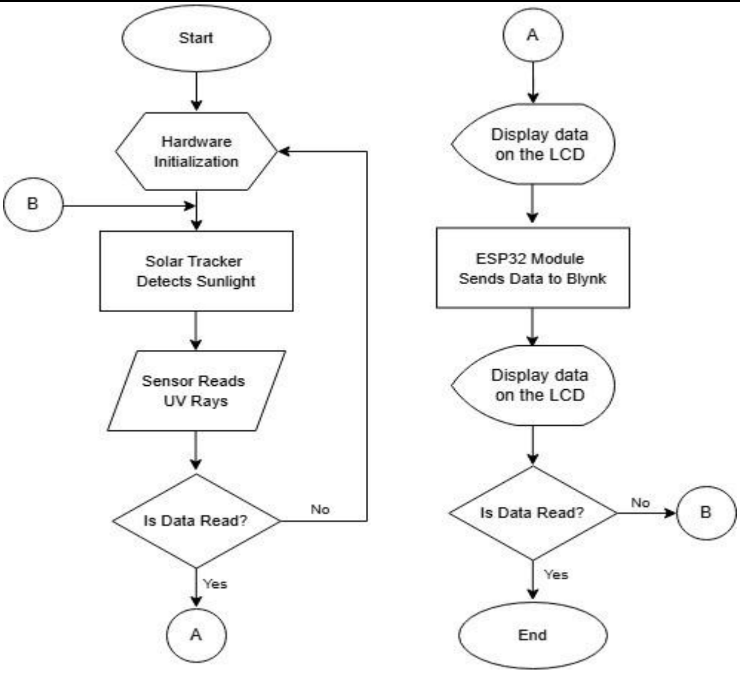

Fig. 2 System Flowchart

The flowchart above illustrates the workflow of an ESP32-based system for UV radiation monitoring using a solar tracker, UV sensor, and communication with Blynk. The following are the step-by-step explanations of the flowchart:

- The system is powered on and starts the program from the beginning to monitor and process data.
- Hardware components such as the solar tracker, sensors, RTC (Real-Time Clock), ESP32 module, and LCD are initialized to prepare them for use.
- The ESP32 module connects to a Wi-Fi network to enable data transmission to the Blynk monitoring application.
- The solar tracker begins monitoring the direction of sunlight to ensure proper alignment and determine the sun's position.
- The system reads data generated by the sensor and utilizes the RTC to record the timestamp of data collection.
- The system checks whether the sensor data has been correctly read. If successful, the process continues; if unsuccessful, the system retries data reading.
- Information obtained from the sensor is displayed on the LCD screen for easy readability.
- Data collected from the sensor is transmitted to the Blynk application, allowing users to monitor UV Index data from other devices, such as smartphones or laptops.
- The system checks if there is a command to shut down the device; if such a command exists, the system stops. Otherwise, the process returns to the sunlight detection step to continue the workflow cycle.
- End.

The visualization and explanation of the solar tracker flowchart are presented in the image below:

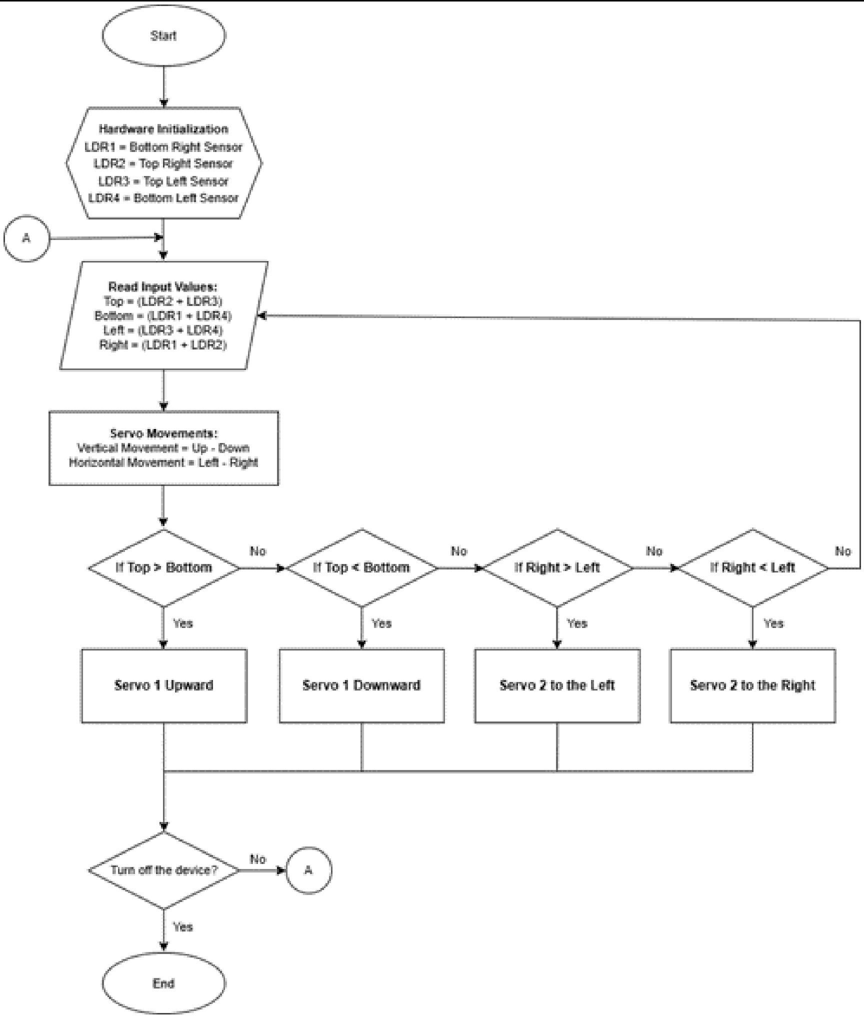

Fig. 3 Solar Tracker Flowchart

The flowchart above provides an overview of the workflow of the solar tracker, where a UV sensor is mounted on the tracker to ensure it remains perpendicular to the direction of incoming sunlight. The following is a detailed explanation of the solar tracker flowchart:

- The process begins by powering on the system.
- In the first step, the hardware is initialized by defining the positions of the LDR sensors:
  - LDR1 is placed in the bottom-right position.
  - LDR2 is placed in the top-right position.
  - LDR3 is placed in the top-left position.
  - LDR4 is placed in the bottom-left position.
- The system then reads the light intensity received by each LDR sensor. The obtained values are calculated as follows:
  - Top is calculated by summing the readings from LDR2 and LDR3.
  - Bottom is calculated by summing the readings from LDR1 and LDR4.
  - Left is calculated by summing the readings from LDR3 and LDR4.
  - Right is calculated by summing the readings from LDR1 and LDR2.
- Based on the sensor readings, the system determines the movements required by the servo motors. Vertical movement is calculated as the difference between the Top and Bottom values, while horizontal movement is calculated as the difference between the Right and Left values.
- If the Top value is higher than the Bottom value, the first servo motor moves upward. Conversely, if the Top value is less than the Bottom value, the first servo motor moves downward to adjust the vertical position.
- For horizontal movement, if the Right value is Higher than the Left value, the second servo motor moves to the left. However, if the Right value is less than the Left value, the second servo motor moves to the right.
- After the servo movement is executed, the system checks whether the device is turned off. If the device is powered off, the process ends and the system stops. If the device remains active, the process loops back to the sensor input reading step to adjust the servo movements according to the updated light intensity.
- The process concludes when the device is powered off, indicating that the system has completed its operation.

2.3 Hardware Design

Hardware design involves the comprehensive process of designing the physical structure of the system. The system circuit is created using Fritzing software, as illustrated in the image below:

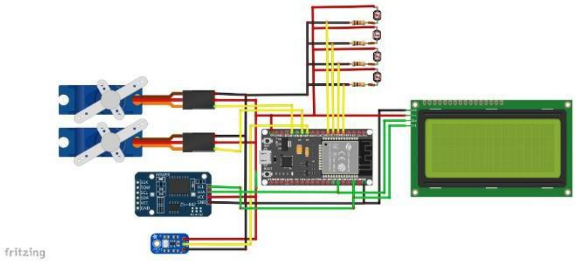

Fig. 4 Wiring Diagram

The image above depicts the wiring diagram of the UV meter and solar tracker system. The UV sensor, LDR, and servos are connected to the GPIO pins on the ESP32 microcontroller, while the LCD and DS3231 RTC module are connected to the SDA and SCL pins. The ESP32 microcontroller functions to process input data from the sensors and manage timing, displaying the data on a 20x4 LCD and transmitting it to the Blynk platform.

The overall hardware design planning includes a comprehensive overview of the device, starting with the solar tracker at the top, followed by a panel box that houses the microcontroller. The entire device design was created using Tinkercad, as shown in the image below.

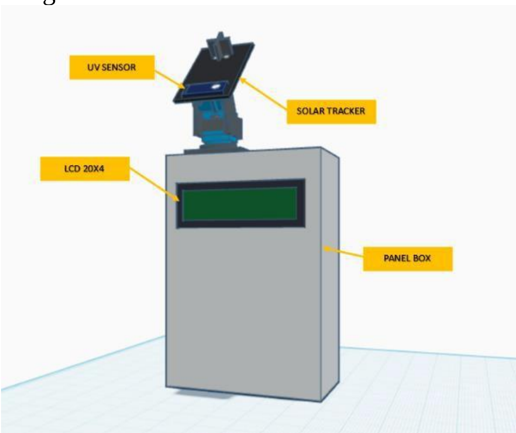

Fig. 5 System Design

The solar tracker design planned in this study will function to track the direction both horizontally and vertically (dual axis), as shown in the image below.

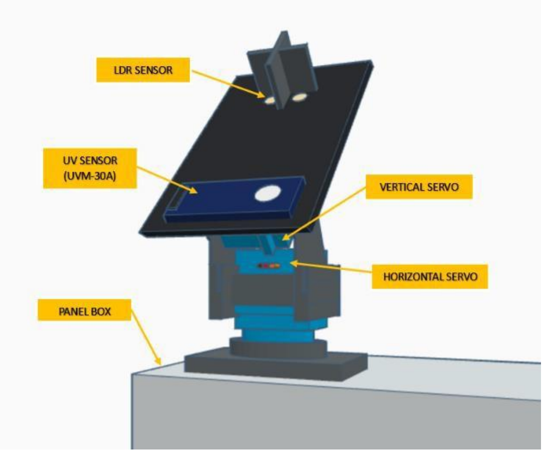

Fig. 6 Solar Tracker Design

The image shows that the solar tracker is placed in the upper left corner of the panel box, while all the sensors are connected with wires to the microcontroller inside the box. Two servos function as the horizontal and vertical servos, mounted on a bracket, allowing both servos to move left-right and up-down. On the black board, there are four LDRs separated by dividers, which serve to detect the direction of incoming sunlight. Meanwhile, the UV sensor is mounted beneath the LDRs so that it can be positioned perpendicular to the sunlight, ensuring optimal UV index readings.

2.4 Software Design

The UV index measurement data will then be displayed on Blynk. The displayed data includes the date and time of the reading, the UV index value, the UV index category, and a real- time graph of the UV index. The design of the display on Blynk, both on desktop and Android, is shown in the figure below.

Fig. 7 Blynk Desktop Diagram

Fig. 8 Blynk Android Display Design

2.5 Test and Calibration Plan

Device The testing plan involves comparison and calibration of the UV sensor, as well as adjusting the servo rotation angle to optimally track sunlight. The purpose of this calibration is to ensure that each sensor is suitable for use in the device. The UV index comparison is performed by comparing the UV index data obtained from the device with the reference data available on the uvindex.app website.

Fig. 9 Web Display of UV Index App

The data from the device is obtained through direct readings on the LCD display or remote monitoring via the Blynk platform. Once the data is collected, it is compared with the reference data, and the measurement differences are calculated. Subsequently, a linear regression graph will be created to evaluate whether the device provides results that are consistent with the reference data.

3. RESULT AND DISCUSSION

3.1 System Implementation

The implementation of the solar tracker along with the UV sensor that has been developed is shown in the image below:

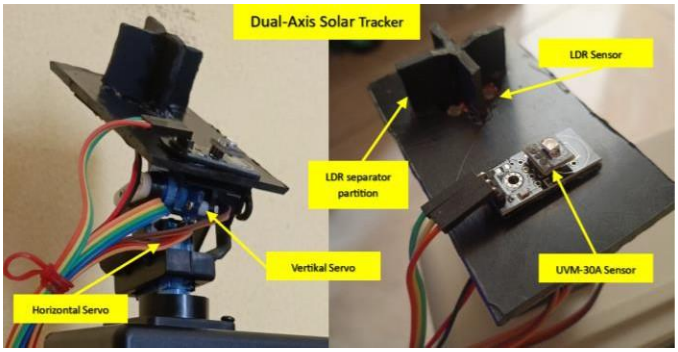

Fig. 10 Solar Tracker Implementation

The implementation of the panel box that has been constructed is shown in the image below:

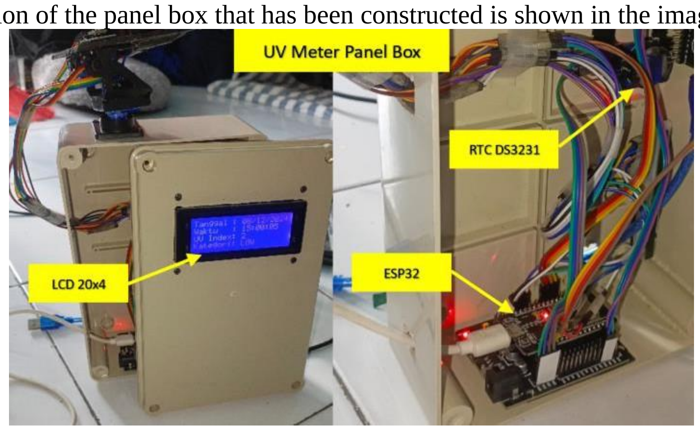

Fig. 11 System Implementation

The overall implementation of the device is the integration of all system designs from this research. The panel box, which contains the ESP32 microcontroller, RTC, and LCD display, will be combined with the solar tracker into a single unit so that the device can function as intended. The overall appearance of the device, both inside and outside, is shown in the following images:

Fig. 12 Overall System Display

3.2 Testing and Calibration of the Device

The device testing was conducted in an open space on the second floor of the author's house, located in Tanah Tinggi Village, Tangerang City, Banten.

1. Solar Tracker

The solar tracker was tested under two conditions, indoors and outdoors. The indoor testing aimed to ensure that the solar tracker could properly detect the light source. Additionally, testing was conducted using a LED flash light to observe whether the solar tracker could accurately track the movement of the light direction from the LED flash light.

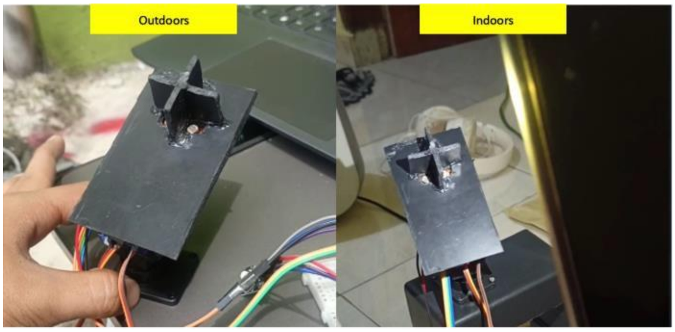

Fig. 13 Solar Tracker Testing

The test results show that the solar tracker functions well, both indoors and outdoors. This is demonstrated by its ability to accurately detect the direction of incoming rays.

2. DS3231 RTC Module

The testing of the DS3231 RTC module was carried out by comparing the date and time displayed on the RTC module through the serial monitor with the official time provided by BMKG, which can be accessed through the official BMKG website at https://jam.bmkg.go.id/Jam.BMKG.

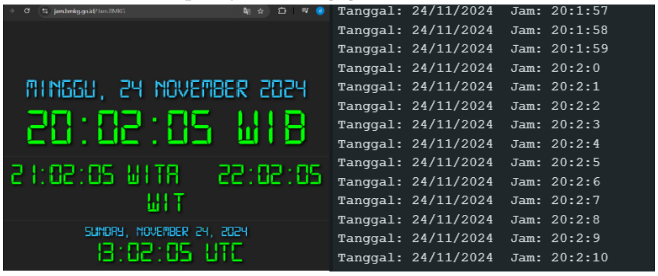

Fig. 14 DS3231 Module Testing

3. UV Sensor (UVM-30A)

The UV sensor was tested by comparing the UV index values generated by the device with the UV index data available on the uvindex.app website. The UV index values on the device can be monitored manually through the serial monitor, the LCD display, or remotely using the Blynk platform.

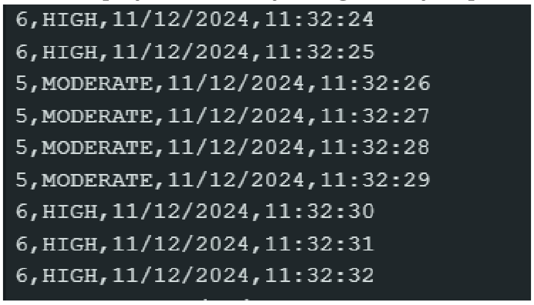

Fig. 15 Serial Monitor Display

Fig. 16 LCD Display

If the reading values do not match the UV index data available on the uvindex.app website, calibration will be performed by checking the input and output voltages of the sensor and adjusting them to match the voltage range in the program.

```
// UV Sensor Logic (Using GPIO 14)
int sensorValue = analogRead(UV_SENSOR_PIN);  // Read the ana
float voltage = (sensorValue * (5.0 / 4095.0)) * 1000;  // Co

// Convert voltage to UV index based on standard ranges
if (voltage < 50 || (voltage >= 50 && voltage <= 227)) {
  UVIndex = "0";  // No UV radiation detected
} else if (voltage > 227 && voltage <= 318) {
  UVIndex = "1";  // Low UV radiation
} else if (voltage > 318 && voltage <= 408) {
  UVIndex = "2";  // Low UV radiation
} else if (voltage > 408 && voltage <= 503) {
  UVIndex = "3";  // Moderate UV radiation
} else if (voltage > 503 && voltage <= 606) {
  UVIndex = "4";  // Moderate UV radiation
} else if (voltage > 606 && voltage <= 696) {
  UVIndex = "5";  // Moderate UV radiation
} else if (voltage > 696 && voltage <= 795) {
  UVIndex = "6";  // High UV radiation
} else if (voltage > 795 && voltage <= 881) {
  UVIndex = "7";  // High UV radiation
} else if (voltage > 881 && voltage <= 976) {
  UVIndex = "8";  // Very high UV radiation
} else if (voltage > 976 && voltage <= 1079) {
  UVIndex = "9";  // Very high UV radiation
} else if (voltage > 1079 && voltage <= 1170) {
  UVIndex = "10"; // Extreme UV radiation
} else if (voltage > 1170) {
  UVIndex = "11"; // Extreme UV radiation
}
```

Fig. 17 UV Sensor Output Range

Additionally, experiments were conducted by testing various angles of sunlight incidence on the UV sensor. The results showed that the most optimal UV sensor readings occurred when the sensor was placed perpendicular to the sunlight at a 90° angle. Therefore, the use of a solar tracker is necessary to ensure that the UV sensor remains in the optimal position for maximum UV readings.

Table 2 Solar Incident Angle Testing

Angle of Incidence of Sunlight	**UV Index Reading**
45°	5
60°	6
90°	7

After the UV meter was successfully constructed and calibrated, a comparison was made between the data generated by the UV meter and the reference from the Uvindex.app application under various times and weather conditions, namely clear and cloudy. This allowed for the verification of the accuracy and consistency of the UV readings between the two measurement methods. Data collection was conducted at nine observation times each day, from 08:00 to 16:00, with a one- hour interval. At each observation time, 15 samples were taken, the average value was computed to indicate the UV intensity for that specific hour. The correction was calculated as the difference between the UV meter value and the reference to evaluate the accuracy and consistency of the device. The results of this testing aim to assess the extent to which the UV meter can be used as a reliable device for monitoring UV intensity.

Table 3 UV Sensor Testing

No	Date, Time	Uvindex.app	UV meter	Weather	Correction
1	28/11/2024, 08.00	3	3,2	Sunny	-0,2
2	28/11/2024, 09.00	5	4,4	Sunny	0,6
3	28/11/2024, 10.00	6	5,5	Sunny	0,5
4	28/11/2024, 11.00	6	4,6	Cloudy	1,4
5	28/11/2024, 12.00	7	6,7	Sunny	0,3
6	28/11/2024, 13.00	6	5,4	Sunny	0,6
7	28/11/2024, 14.00	6	6,1	Sunny	-0,1
8	28/11/2024, 15.00	4	3,6	Sunny	0,4
9	28/11/2024, 16.00	2	2,5	Cloudy	-0,5
10	08/12/2024, 08.00	3	2,9	Sunny	0,1
11	08/12/2024, 09.00	5	4,4	Sunny	0,6
12	08/12/2024, 10.00	6	4,5	Sunny	1,5
13	08/12/2024, 11.00	7	6,5	Cloudy	0,5
14	08/12/2024, 12.00	7	6,6	Sunny	0,4
15	08/12/2024, 13.00	6	4,3	Sunny	1,7
16	08/12/2024, 14.00	6	5,5	Sunny	0,5
17	08/12/2024, 15.00	4	3,6	Sunny	0,4
18	08/12/2024, 16.00	2	1,9	Cloudy	0,1
19	11/12/2024, 08.00	3	2,9	Sunny	0,1
20	11/12/2024, 09.00	5	3,9	Sunny	1,1
21	11/12/2024, 10.00	6	5,8	Sunny	0,2
22	11/12/2024, 11.00	7	6,7	Cloudy	0,3
23	11/12/2024, 12.00	8	7,4	Sunny	0,6
24	11/12/2024, 13.00	7	5,9	Sunny	1,1
25	11/12/2024, 14.00	6	5,1	Sunny	0,9
26	11/12/2024, 15.00	4	3,1	Sunny	0,9
27	11/12/2024, 16.00	2	2,2	Cloudy	-0,2
28	12/12/2024, 08.00	3	3,2	Sunny	-0,2
29	12/12/2024, 09.00	5	4,7	Sunny	0,3
30	12/12/2024, 10.00	7	6,5	Sunny	0,5
31	12/12/2024, 11.00	8	7,6	Cloudy	0,4
32	12/12/2024, 12.00	7	6,7	Sunny	0,3
33	12/12/2024, 13.00	7	5,2	Sunny	1,8
34	12/12/2024, 14.00	6	5,5	Sunny	0,5
35	12/12/2024, 15.00	4	3,6	Sunny	0,4
36	12/12/2024, 16.00	2	2,5	Cloudy	-0,5
37	13/12/2024, 08.00	3	3,2	Sunny	-0,2
38	13/12/2024, 09.00	5	4,7	Sunny	0,3
39	13/12/2024, 10.00	7	5,6	Sunny	1,4
40	13/12/2024, 11.00	8	6,6	Cloudy	1,4
41	13/12/2024, 12.00	7	6,7	Sunny	0,3
42	13/12/2024, 13.00	7	5,3	Sunny	1,7
43	13/12/2024, 14.00	6	5,5	Sunny	0,5
44	13/12/2024, 15.00	4	3,6	Sunny	0,4

The test results show that the values obtained from the UV meter generally approach the reference data from the Uvindex.app application, with correction differences ranging from -0.5 to 1.8. In clear weather, the corrections tend to be positive, indicating that the UV meter shows slightly lower values than the reference. On the other hand, in cloudy weather, the correction variations are larger, including some negative values, such as on November 28, 2024, at 16:00 (-0.5) and

December 13, 2024, at 16:00 (-0.4). The highest UV intensity is consistently recorded between 11:00 and 13:00, when the sun is at its peak, while lower UV intensities are observed during the morning and evening.

This testing also shows that the UV sensor is highly sensitive to cloud cover conditions, causing significant data fluctuations, especially under cloudy conditions. In addition, external factors such as shading from the observer or other objects introduce noise, affecting the measurement results. As a result, the measurements at certain times become less stable. Nevertheless, the UV meter provides fairly consistent measurements under most conditions, with most corrections falling within the ±1.5 range. Given the instrument's sensitivity to various environmental factors, caution is needed during operation to minimize noise influences and ensure that the data generated is more representative.

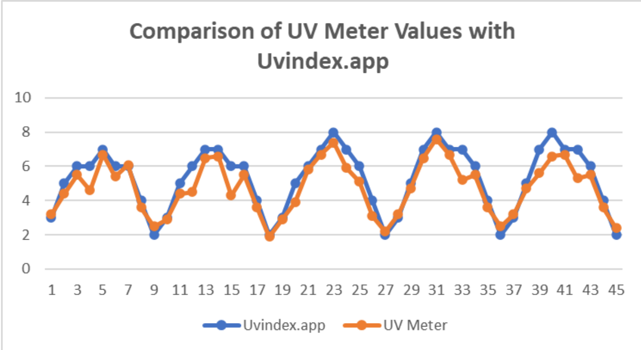

Fig. 18 Comparison Graph of UV Meter Reading and Data from uvindex.app Website

This figure compares the UV values from Uvindex.app (blue line) and the UV Meter (orange line) over a specific period. Both methods show a similar pattern, with UV peaks ranging from 8 to 9 and the lowest points approaching 2. Although the trend is consistent, there are slight differences at some points between the two measurements.

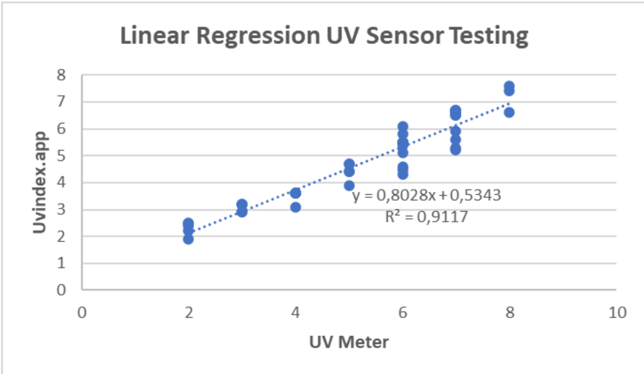

Fig. 19 Linear Regression Graph of UV Sensor Testing

This image presents a linear regression test between the UV values measured using a UV Meter (horizontal axis) and Uvindex.app (vertical axis). The pattern of the data points indicates A positive correlation between the two variables is demonstrated, further validated by the linear regression equation $y = 0.8028x + 0.5343$ and the coefficient of determination $R^2 = 0.9117$. The R^2 value, which is close to 1, indicates a strong correlation between the two methods, despite a slight deviation. Overall, these results suggest that the UV measurements obtained using Uvindex.app are consistent with those of the UV Meter, with only minor differences identifiable in the data.

3.3 Blynk Testing

The Blynk testing was conducted simultaneously with the UV Meter testing under direct sunlight exposure. The results showed that Blynk was able to display UV Meter data in real- time, including information on UV intensity categories, time, and UV reading graphs in line format. Thus, Blynk provides an informative and structured data visualization of UV to monitor changes in UV values directly.

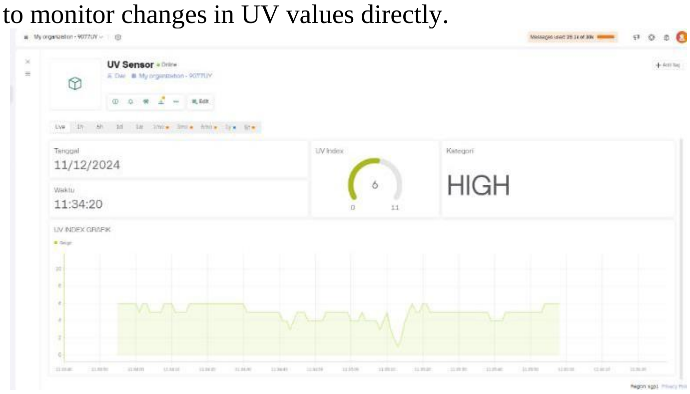

Fig. 20 Blynk Desktop Display

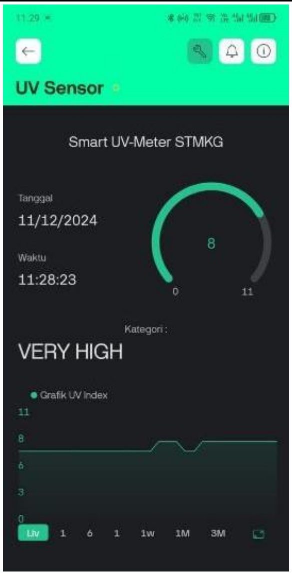

Fig. 21 Blynk Android Display

In addition to being accessible via desktop, Blynk can also be used on Android devices by downloading the Blynk IoT app. This app allows users to access the same data by connecting it to their Blynk account, enabling flexible UV Meter monitoring across various devices.

4. CONCLUSION

The IoT-based UV index measuring device using the UVM- 30A sensor has been successfully designed and tested with satisfactory results. All components, such as the DS3231 RTC, 20x4 LCD, and data monitoring system via the Blynk platform, function as designed and support the optimal performance of the device. The IoT feature facilitates real-time remote data monitoring, while the integration of the solar tracker allows the UV sensor to efficiently receive sunlight, resulting in more accurate UV index measurements.

Test results show good measurement accuracy, with a comparison of the UV meter results against data from the uvindex.app website yielding a small relative error. However, the device is more sensitive to environmental factors, such as cloud cover and shading, which may unintentionally block the sensor. With sufficient flexibility, performance, and accuracy, this device is an effective solution for real-time and accurate UV index measurements.

REFERENCE

[1] W. Li and S. Fan, "Nanophotonic control of thermal radiation for energy applications [Invited]," Opt Express, vol. 26, no. 12, p. 15995, Jun. 2018, doi: 10.1364/oe.26.015995.

[2] A. Dwivedi, A. K. Tripathi, J. Singh, and M. K. Pal, "Ultraviolet radiation (UVR): An introduction," in Photocarcinogenesis and Photoprotection, Springer Singapore, 2018, pp. 1–8. doi: 10.1007/978-981-10-5493-8_1.

[3] E. C. Turner, J. Manners, C. J. Morcrette, J. B. O'Hagan, and A. R. D. Smedley, "Toward a New UV Index Diagnostic in the Met Office's Forecast Model," J Adv Model Earth Syst, vol. 9, no. 7, pp. 2654–2671, Nov. 2017, doi: 10.1002/2017MS001050.

[4] J. C. S. Yam and A. K. H. Kwok, "Ultraviolet light and ocular diseases," 2014, Kluwer Academic Publishers. doi: 10.1007/s10792- 013-9791-x.

[5] R. M. Lucas et al., "Human health in relation to exposure to solar ultraviolet radiation under changing stratospheric ozone and climate," Photochemical and Photobiological Sciences, vol. 18, no. 3, pp. 641–680, 2019, doi: 10.1039/C8PP90060D.

[6] C. J. Heckman, K. Liang, and M. Riley, "Awareness, understanding, use, and impact of the UV index: A systematic review of over two decades of international research," Jun. 01, 2019, Academic Press Inc. doi: 10.1016/j.ypmed.2019.03.004.

[7] T. Furuhashi, K. Torii, K. Ikumi, H. Kato, E. Nishida, and A. Morita, "Ultraviolet A1 phototherapy for the treatment of localized scleroderma," Journal of Dermatology, vol. 47, no. 7, pp. 792–795, Jul. 2020, doi: 10.1111/1346-8138.15368.

[8] D. R. Roshan, M. Koc, A. Abdallah, L. Martin-Pomares, R. Isaifan, and C. Fountoukis, "UV index forecasting under the influence of desert dust: Evaluation against surface and satellite-retrieved data," Atmosphere (Basel), vol. 11, no. 1, 2020, doi: 10.3390/ATMOS11010096.

[9] A. C. G. B. Leal, M. P. Corrêa, M. F. Holick, E. V. Melo, and M. Lazaretti-Castro, "Sun-induced production of vitamin D3 throughout 1 year in tropical and subtropical regions: relationship with latitude,

cloudiness, UV-B exposure and solar zenith angle," Photochemical and Photobiological Sciences, vol. 20, no. 2, pp. 265–274, Feb. 2021, doi: 10.1007/s43630-021-00015-z.

[10] P. Gies, E. Van Deventer, A. C. Green, C. Sinclair, and R. Tinker, "Review of the Global Solar UV Index 2015 Workshop Report," Jan. 01, 2018, Lippincott Williams and Wilkins. doi: 10.1097/HP.0000000000000742.

[11] N. C. Delic, J. G. Lyons, N. Di Girolamo, and G. M. Halliday, "Damaging Effects of Ultraviolet Radiation on the Cornea," Jul. 01, 2017, Blackwell Publishing Inc. doi: 10.1111/php.12686.

[12] N. A. Pramono, F. I. Sofyan, B. A. Purwandani, and O. Ghaisyani, "Application of ESP32 as a Media for Learning Ozone Damage in the Form of IoT-Based Ultraviolet Index Readers."

[13] E. Lönnqvist et al., "Wearable UV Meter – An EPS@ISEP 2017 Project," in Advances in Intelligent Systems and Computing, Springer Verlag, 2018, pp. 896–907. doi: 10.1007/978-3-319-73210-7_102.

[14] M. Blumthaler, "UV monitoring for public health," Int J Environ Res Public Health, vol. 15, no. 8, Aug. 2018, doi: 10.3390/ijerph15081723.

[15] E. Ervianto et al., "Assessing the Efficacy of the UV Index in Predicting Surface UV Radiation: A Comprehensive Analysis Using Statistical and Machine Learning Methods," Indonesian Review of Physics, vol. 6, no. 2, pp. 99–121, Dec. 2023, doi: 10.12928/irip.v6i2.8216.

[16] R. Duarte Almeida, R. Barreto, E. Souto, and J. Cabral Leite, "IoT System for Ultraviolet Ray Index Monitoring," Int J Innov Educ Res, vol. 7, no. 12, pp. 409–420, Dec. 2019, doi: 10.31686/ijier.vol7.iss12.2087.

[17] M. N. A. Mohd Said, S. A. Jumaat, and C. R. A. Jawa, "Dual axis solar tracker with iot monitoring system using arduino," International Journal of Power Electronics and Drive Systems, vol. 11, no. 1, pp. 451–458, Mar. 2020, doi: 10.11591/ijpeds.v11.i1.pp451-458.

[18] C. Jamroen, C. Fongkerd, W. Krongpha, P. Komkum, A. Pirayawaraporn, and N. Chindakham, "A novel UV sensor-based dual-axis solar tracking system: Implementation and performance analysis," Appl Energy, vol. 299, Oct. 2021, doi: 10.1016/j.apenergy.2021.117295.

[19] G. Tejaleksono, D. Notosudjono, A. R. Machdi, F. Adzikri, and A. U. Rahayu, "PROTOTYPE OF CO, CO 2 , UV LIGHT, TEMPERATURE, AND HUMIDITY DETECTION DEVICE BASED ON IOT AND SOLAR CELLS," 2024.

[20] U. V Iac, "INTERSUN PROGRAMME 7 International Advisory Committee Meeting," 2018

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 15 - 20
https://journal.physan.org/index.php/jocpes/index

Comparative Analysis of Linear Regression Models and XGBoost to Assess the Impact of ENSO on Rainfall in Ternate City in 2023

Firman Almaliky Gapi Amra [1], Anton Widodo[2], Muchamad Rizqy Nugraha[3]
[1,2,3]Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)

Article Info

Article history:

Received March 1, 2025
Revised March 6, 2025
Accepted March 7, 2025

Keywords:

ENSO,
Rainfall,
Linear Regression,
XGBoost,
Machine Learning.

A B S T R A C T

The purpose of this study is to evaluate how well two prediction models—linear regression and XGBoost—perform in assessing how ENSO (El Niño-Southern Oscillation) affects rainfall in Ternate City in 2023. The Meteorology, Climatology, and Geophysics Agency (BMKG) provided monthly rainfall data, while the Bureau of Meteorology (BOM) in Australia provided ENSO index data. Performance indicators such Pearson correlation analysis, the coefficient of determination (R-squared), and mean squared error (MSE) were used in the evaluation. According to the findings, the two models perform differently when it comes to capturing the pattern of the link between rainfall and ENSO; XGBoost is more adaptable but has a tendency to overfit on small amounts of data, whereas linear regression obtains a better R-squared value.

Corresponden Author:

Firman Almaliky Gapi Amra,
Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)
Tangerang City, Banten, Indonesia
Email: firman.amra14@gmail.com

1. INTRODUCTION

Climate change and its associated global phenomena have increasingly become critical areas of scientific research, with the El Niño-Southern Oscillation (ENSO) emerging as a particularly significant driver of regional climate variability[1], [2]. ENSO, encompassing both El Niño and La Niña phases, profoundly impacts global weather patterns, including precipitation, temperature, and extreme weather events[3].In Indonesia, a region characterized by complex meteorological dynamics, ENSO exerts profound influences on precipitation patterns, often resulting in significant societal and environmental challenges[4], [5]. Ternate City, situated in a climatologically sensitive area, serves as an exemplary case study for understanding these intricate relationships, particularly given its vulnerability to changing rainfall patterns and their associated risks. Understanding the relationship between ENSO and rainfall is crucial for improving predictive capabilities, enabling better resource management and disaster preparedness[6].Traditional statistical methods like linear regression have long been employed to analyze such relationships due to their simplicity and interpretability[7], [8]. Linear regression models the relationship between independent and dependent variables under the assumption of linearity, making it a foundational tool for identifying trends and quantifying relationships. By leveraging the straightforward assumptions of linear regression, researchers can establish a baseline understanding of how ENSO anomalies influence rainfall patterns, including the direct proportional changes between the ENSO index and precipitation levels[6], [9].[10]

However, the complex, non-linear nature of climate interactions often requires more advanced analytical approaches. XGBoost (Extreme Gradient Boosting), an ensemble machine learning algorithm, has emerged as a powerful tool for analyzing and predicting environmental phenomena. Unlike linear regression, XGBoost can model non-linear relationships and capture intricate feature interactions, which are often present in climate datasets. By iteratively constructing decision trees, XGBoost minimizes prediction errors while handling large datasets, missing values, and feature complexities effectively[11]. These advantages make it a suitable complement to traditional statistical methods in understanding the nuanced impacts of ENSO on regional precipitation[12], [13].

Machine learning techniques, including XGBoost, have revolutionized climate science by providing robust tools for analyzing environmental datasets. These methodologies enable the identification of hidden patterns in large and complex data, offering insights that traditional approaches may overlook. When applied to the ENSO-rainfall relationship in Ternate City, XGBoost has the potential to uncover subtle interactions that influence precipitation variability. Such insights are critical for refining predictive models, which can aid in climate adaptation and mitigation strategies, particularly in regions like Indonesia where the impacts of climate variability are pronounced[14].

The inherent complexity of ENSO's impact on regional precipitation necessitates a multi-model approach to accurately quantify and understand these dynamics. By comparing linear regression and XGBoost, this study aims to provide a comprehensive evaluation of their predictive capabilities[15], [16].While linear regression offers clarity and interpretability, XGBoost introduces flexibility and precision in capturing complex data patterns. This research not only seeks to quantify the relationship between ENSO anomalies and rainfall in Ternate City but also to demonstrate the broader applicability of machine learning in advancing climate science. Such comparative analyses are increasingly important as the need for accurate climate predictions grows in response to the accelerating pace of global climate change[17], [18].

The study seeks to comprehensively investigate the relationship between ENSO anomalies and rainfall in Ternate City by comparing the performance of linear regression and XGBoost models. Through a rigorous analytical approach, the research aims to evaluate the predictive capabilities of both methodologies using key performance metrics like Mean Squared Error and R-squared, ultimately drawing substantive conclusions about the most effective model for understanding and predicting rainfall patterns in the region. By systematically comparing these two modeling techniques, the study intends to provide insights into the complex dynamics of ENSO's influence on local precipitation and contribute to more accurate environmental forecasting strategies.

2. RESEARCH METHOD

This research uses ENSO anomaly data from BOM and monthly rainfall data in Ternate City throughout 2023 from BMKG. The analysis is conducted using the linear regression method and the XGBoost algorithm.
The Data Sourches for Monthly rainfall data obtained from **BMKG** (Meteorology, Climatology, and Geophysics Agency, ENSO anomaly index data obtained from **BOM** (Bureau of Meteorology, Australia) and Time Period: Year 2023.The Variables are Independent variable, ENSO anomaly index (from BOM) and Dependent variable: Monthly rainfall (from BMKG). The integrated modeling approach combines the strengths of Linear Regression and XGBoost to analyze the relationship between ENSO anomalies and rainfall. Linear Regression provides a straightforward initial understanding of the linear correlation, offering clear insights into direct relationships between variables. Meanwhile, XGBoost enhances the analysis by capturing complex non-linear patterns and interactions that traditional linear methods might overlook[19], [20]. The model performance in this study is comprehensively assessed using three key metrics: **Mean Squared Error (MSE)**, **R-squared (R^2)**, and **Pearson Correlation**. Together, these metrics provide a well-rounded evaluation of the models' ability to predict rainfall based on ENSO anomalies

Mean Squared Error (MSE) is a widely used metric for evaluating the accuracy of continuous prediction models. It measures the average squared difference between predicted values and actual observed values, with larger errors being penalized more heavily due to the squaring of differences. A lower MSE indicates better predictive accuracy, as it means that the model's predictions are closer to the true values. In this study, the linear regression model achieves a significantly lower MSE compared to the XGBoost model. This suggests that linear regression provides more accurate and reliable predictions for rainfall in Ternate City, based on ENSO anomalies. The higher MSE in the XGBoost model indicates that it struggles to make accurate predictions in this case, potentially due to overfitting to the limited dataset.

R-squared (R^2) is another critical evaluation metric, as it represents the proportion of variance in the dependent variable (rainfall) that can be explained by the independent variable (ENSO anomaly) within the model. R^2 values range from 0 to 1, where a value closer to 1 indicates a better fit of the model to the data. A higher R^2 suggests that the model explains a significant portion of the variability in the outcome, whereas a low R^2 indicates a poor model fit. In this study, the linear regression model shows an R^2 of 0.303, meaning it explains approximately 30.3% of the variation in rainfall based on the ENSO anomaly. While this is a modest value, it still demonstrates that the linear regression model is able to capture a meaningful portion of the relationship between ENSO anomalies and rainfall. In contrast, the R^2 value for the XGBoost model is negative (-0.059), indicating that the model fails to explain the variation in the data and actually performs worse than a simple baseline model that just predicts the mean of the rainfall values.

The Pearson correlation is used to assess the strength and direction of the linear relationship between predicted and actual values. A correlation coefficient closer to +1 indicates a strong positive linear relationship, while a value near

-1 suggests a strong negative relationship. A value near 0 indicates little to no linear relationship. In the context of this study, the Pearson correlation for the XGBoost model is -0.550, reflecting a weak negative correlation between the model's predictions and the actual rainfall data. This weak negative correlation suggests that XGBoost does not capture the relationship between ENSO anomalies and rainfall effectively. Although Pearson correlation for the linear regression model is not explicitly reported, it can be inferred that it is likely higher due to the model's better fit to the data and more accurate predictions.

Overall, these three metrics—MSE, R^2, and Pearson correlation—offer a multi-faceted evaluation of model performance. MSE provides insight into the accuracy of the predictions, R^2 indicates how well the model explains the variance in rainfall, and Pearson correlation evaluates the strength of the linear relationship between predicted and actual values. When taken together, these metrics clearly show that linear regression outperforms XGBoost in predicting rainfall based on ENSO anomalies. The results highlight the importance of selecting the appropriate model based on data characteristics, as linear regression demonstrates better generalization and accuracy with the available dataset.

3. RESULT AND DISCUSSION

In this section, we will systematically analyze and interpret the results of our comparative study on ENSO-rainfall relationships in Ternate City. Our investigation employs two distinct modeling approaches - linear regression and XGBoost - to unravel the complex dynamics of climate interactions.

Linear Regression Analysis Result

- **MSE**: 38.2921955135183
- **R-squared (R^2)**: 0.3031723420952398
- **Regression Coefficient**: -4.417146709384934
- **Intercept**: 9.867402028082324

The linear regression results indicate that the relationship between ENSO anomaly and rainfall is negative. The regression line tends to decline, indicating that higher ENSO anomaly values correspond to decreased rainfall.

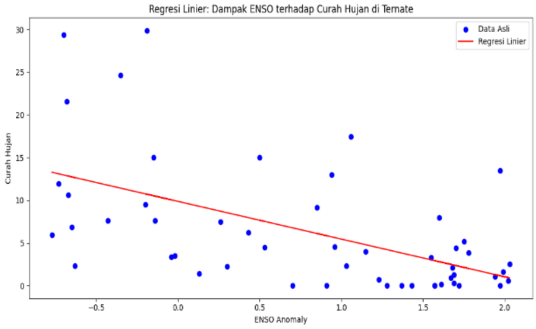

Figure 1. The linear regression results

The linear regression graph shows a negative trend between the ENSO anomaly index and rainfall. Each unit increase in the ENSO anomaly index tends to be followed by a decrease in rainfall, consistent with the negative regression coefficient obtained. This indicates that the El Niño phenomenon, characterized by positive ENSO indices, is associated with reduced rainfall in Ternate City.

XGBoost Analysis Results

- MSE: 88.3720951693223
- R-squared (R^2): -0.05910609209615769
- Pearson Correlation: -0.5506108808362217

Although XGBoost offers higher flexibility, the results show that this model has a larger MSE compared to linear regression. The negative R-squared value indicates that the XGBoost model fails to explain the data variation effectively.

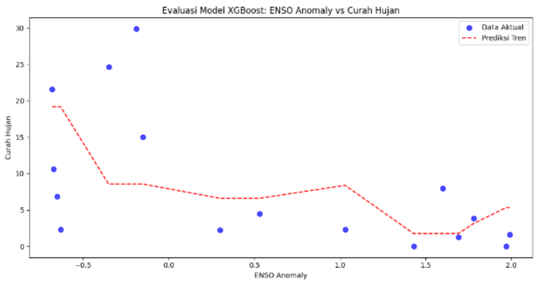

Figure 2. XGBoost Analysis Results

The XGBoost analysis graph shows more varied predictions but lacks consistency with the actual data pattern. The XGBoost model tends to overfit to the noise in the data, leading to higher MSE and a negative R-squared value. This suggests that under conditions with limited datasets like this, XGBoost fails to optimally capture the relationship between ENSO anomaly and rainfall.

TABLE I. MODEL COMPARISON

Evaluation Metric	*Linear Regression*	*XGBoost*
Mean Squared Error (MSE)	38.292	88,372
R-squared (R^2)	0,303	-0.059
Pearson Correlation		-0.550

From Table I, it is evident that the **Mean Squared Error (MSE)** for the linear regression model is 38.292, which is significantly lower than the MSE of 88.372 for the XGBoost model. A smaller MSE indicates that the predictions from the linear regression model are closer to the actual data compared to the XGBoost model, which exhibits higher prediction errors.

The **R-squared (R^2)** value for linear regression is 0.303, suggesting that 30.3% of the variance in rainfall can be explained by the linear relationship with the ENSO index. In contrast, XGBoost has a negative R^2 value (-0.059), indicating that the model fails to explain the variation in the data effectively. The negative R^2 also suggests potential overfitting or that the model does not fit the dataset well. Additionally, the **Pearson correlation coefficient** for XGBoost is -0.550, implying a weak negative correlation between the model's predictions and the actual data. This further supports the conclusion that XGBoost struggles to capture the relationship between ENSO anomalies and rainfall in this dataset. Overall, these results highlight that while XGBoost is generally effective in capturing complex patterns, its performance diminishes with limited datasets. In contrast, linear regression demonstrates better accuracy and generalizability in modeling the relationship between ENSO anomalies and rainfall.

4. CONCLUSION

This study highlights that Linear Regression is more effective than XGBoost in modeling the relationship between ENSO anomalies and rainfall in Ternate City. The superior performance of Linear Regression is evident from its significantly lower Mean Squared Error (MSE), indicating more accurate predictions, and its better generalization capabilities compared to XGBoost. While XGBoost is known for its ability to model complex relationships and capture intricate patterns in data, the limited dataset size in this study led to overfitting, which ultimately reduced its predictive reliability. The R-squared and Pearson correlation metrics further emphasized the strength of linear regression, as it demonstrated a meaningful relationship between ENSO anomalies and rainfall, explaining around

30.3% of the variation in rainfall. On the other hand, the negative R² and weak Pearson correlation for XGBoost suggested that it failed to effectively model the relationship in this contex.

The analysis also confirmed a negative correlation between ENSO anomalies and rainfall, as reflected by the negative regression coefficient in the linear regression model. This suggests that higher ENSO anomalies are associated with lower rainfall levels, supporting the established understanding that El Niño (characterized by positive ENSO anomalies) typically leads to reduced rainfall in many parts of the world, including Indonesia. These findings emphasize the importance of selecting the appropriate modeling approach based on the characteristics of the data. For datasets with limited observations and relatively simple relationships, classical statistical approaches like linear regression offer a more robust, interpretable, and reliable solution compared to more complex machine learning methods such as XGBoost. This study reinforces the value of linear regression in scenarios where data limitations and the simplicity of the relationship between variables make it a more suitable choice over advanced machine learning models.

REFERENCE

[1] H. S. Lee, "General Rainfall Patterns in Indonesia and the Potential Impacts of Local Seas on Rainfall Intensity," *Water (Switzerland)*, vol. 7, no. 4, pp. 1751–1768, Apr. 2015, doi: 10.3390/w7041751.

[2] L. Agustina, R. H. Virgianto, and A. N. Fitrianto, "Utilization of remote sensing data for mapping the effect of Indian Ocean Dipole (IOD) and El Nino Southern Oscillation (ENSO) in Sumatra Island," in *IOP Conference Series: Earth and Environmental Science*, IOP Publishing Ltd, Nov. 2021. doi: 10.1088/1755-1315/893/1/012062.

[3] J. T. Nugroho, D. Nurfitriani, Suwarsono, G. A. Chulafak, R. J. Manalu, and S. Harini, "Rainfall anomalies assessment during drought episodes of 2015 in Indonesia using CHIRPS Data," in *IOP Conference Series: Earth and Environmental Science*, IOP Publishing Ltd, Apr. 2021. doi: 10.1088/1755-1315/739/1/012044.

[4] S. Nurdiati, F. Bukhari, A. Sopaheluwakan, P. Septiawan, and V. Hutapea, "ENSO AND IOD IMPACT ANALYSIS OF EXTREME CLIMATE CONDITION IN PAPUA, INDONESIA," *Geographia Technica*, vol. 19, no. 1, pp. 1–18, Mar. 2024, doi: 10.21163/GT_2024.191.01.

[5] N. A. I. Baharuddin, M. Zainuddin, and Najamuddin, "The impact of ENSO-IOD on Decapterus spp. in Pangkajene Kepulauan and Barru Waters, Makassar Strait, Indonesia," *Biodiversitas*, vol. 23, no. 11, pp. 5613–5622, 2022, doi: 10.13057/biodiv/d231110.

[6] A. Parmar, K. Mistree, and M. Sompura, "Machine Learning Techniques For Rainfall Prediction: A Review," 2017. [Online]. Available: https://www.researchgate.net/publication/319503839

[7] D. Maulud and A. M. Abdulazeez, "A Review on Linear Regression Comprehensive in Machine Learning," *Journal of Applied Science and Technology Trends*, vol. 1, no. 2, pp. 140–147, Dec. 2020, doi: 10.38094/jastt1457.

[8] "Simple Linear Regression Multiple Linear Regression Polynomial Regression Underfitting and Overfitting Implementing Linear Regression in Python Python Packages for Linear Regression Simple Linear Regression With scikit-learn Multiple Linear Regression With scikit-learn Polynomial Regression With scikit-learn Advanced Linear Regression With statsmodels Beyond Linear Regression Conclusion." [Online]. Available: https://realpython.com/linear-regression-in-python/1/21

[9] T. Soares Dos Santos, D. Mendes, and R. Rodrigues Torres, "Artificial neural networks and multiple linear regression model using principal components to estimate rainfall over South America," *Nonlinear Process Geophys*, vol. 23, no. 1, pp. 13–20, Jan. 2016, doi: 10.5194/npg-23-13-2016.

[10] I. Papailiou, F. Spyropoulos, I. Trichakis, and G. P. Karatzas, "Artificial Neural Networks and Multiple Linear Regression for Filling in Missing Daily Rainfall Data," *Water (Switzerland)*, vol. 14, no. 18, Sep. 2022, doi: 10.3390/w14182892.

[11] N. Uzir, S. Raman, S. Banerjee, and R. S. Nishant Uzir Sunil R, "Experimenting XGBoost Algorithm for Prediction and Classification of Different Datasets Experimenting XGBoost Algorithm for Prediction and Classifi cation of Different Datasets," *International Journal of Control Theory and Applications*, vol. 9, 2016, [Online]. Available: https://www.researchgate.net/publication/318132203

[12] A. Yasper, D. Handoko, M. Putra, H. K. Aliwarga, and M. S. R. Rosid, "Hyperparameters Optimization in XGBoost Model for Rainfall Estimation: A Case Study in Pontianak City," *Jurnal Penelitian Pendidikan IPA*, vol. 9, no. 9, pp. 7113–7121, Sep. 2023, doi: 10.29303/jppipa.v9i9.3890.

[13] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[14] L. Ferreira, A. Pilastri, C. M. Martins, P. M. Pires, and P. Cortez, "A Comparison of AutoML Tools for Machine Learning, Deep Learning and XGBoost," in *Proceedings of the International Joint Conference on*

Neural Networks, Institute of Electrical and Electronics Engineers Inc., Jul. 2021. doi: 10.1109/IJCNN52387.2021.9534091.

[15] Z. Kuang *et al.*, “A Hybrid ENSO Prediction System Based on the FIO−CPS and XGBoost Algorithm,” *Remote Sens (Basel)*, vol. 15, no. 7, Apr. 2023, doi: 10.3390/rs15071728.

[16] A. Kurniadi, E. Weller, S. K. Min, and M. G. Seong, “Independent ENSO and IOD impacts on rainfall extremes over Indonesia,” *International Journal of Climatology*, vol. 41, no. 6, pp. 3640–3656, May 2021, doi: 10.1002/joc.7040.

[17] A. Santoso, M. J. Mcphaden, and W. Cai, “The Defining Characteristics of ENSO Extremes and the Strong 2015/2016 El Niño,” Dec. 01, 2017, *Blackwell Publishing Ltd*. doi: 10.1002/2017RG000560.

[18] C. Wang, C. Deser, J. Y. Yu, P. DiNezio, and A. Clement, “El Niño and Southern Oscillation (ENSO): A Review,” in *Coral Reefs of the World*, vol. 8, Springer Nature, 2017, pp. 85–106. doi: 10.1007/978-94-017-7499-4_4.

[19] J. Pesantez-Narvaez, M. Guillen, and M. Alcañiz, “Predicting motor insurance claims using telematics data—XGboost versus logistic regression,” *Risks*, vol. 7, no. 2, Jun. 2019, doi: 10.3390/risks7020070.

[20] J. Wu *et al.*, “Prediction and Screening Model for Products Based on Fusion Regression and XGBoost Classification,” *Comput Intell Neurosci*, vol. 2022, 2022, doi: 10.1155/2022/4987639.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 21 - 30
https://journal.physan.org/index.php/jocpes/index

Evaluation of the XGBoost Model for Rainfall Prediction and Classification Using BMKG Data and OpenWeather API

M. Saori Isjayan Syah[1], Nardi[2], Agustina Rachmawardani[3]
[1,2,3]Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)

Article Info

Article history:

Received March 2, 2025
Revised March 7, 2025
Accepted March 8, 2025

Keywords:

Rainfall Prediction,
XGBoost,
Rainfall Classification,
OpenWeather API,
BMKG.

A B S T R A C T

Indonesia, situated between two continents and two oceans, experiences significant climate variability, with rainfall patterns shaped by geographical and topographical factors, as well as phenomena like the El Niño Southern Oscillation (ENSO). Accurate rainfall forecasting plays a critical role in disaster mitigation, agricultural planning, and water resource management. This study focuses on developing a rainfall prediction and classification model using the Extreme Gradient Boosting (XGBoost) algorithm. The model leverages historical rainfall data from the Indonesian Meteorological, Climatological, and Geophysical Agency (BMKG) and real-time data from the OpenWeather API. The output includes rainfall trend graphs and classification of rainfall intensity into categories such as light, moderate, or heavy. Model performance is assessed through metrics like accuracy, precision, RMS (Root Mean Square), and RMSE (Root Mean Square Error). This research highlights the integration of historical and real-time data for weather forecasting and demonstrates the application of advanced machine learning techniques like XGBoost to build robust and precise prediction models. The findings are expected to offer practical insights for disaster risk reduction, agricultural strategy planning, and effective water resource management.

Corresponden Author:

M. Saori Isjayan Syah,
Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)
Tangerang City, Banten, Indonesia
Email: muhammadsaorii@gmail.com

1. INTRODUCTION

Indonesia is situated between the Indian and Pacific Oceans, as well as between two continents—Asia and Australia. This geographical position causes frequent climate variations influenced by multiple factors. Additionally, phenomena like the El Niño Southern Oscillation (ENSO) contribute to the environmental complexity. Rainfall, as a significant component of the climate, is highly intricate. Its characteristics vary across regions, influenced by factors such as geography, topography, and monsoons. The structure and orientation of the islands are yet to be fully accounted for, leading to uneven rainfall distribution across different areas. Moreover, ENSO affects nearly 70% of rainfall variability in Indonesia [1].

Rainfall is a significant and impactful phenomenon within the climate system, directly affecting ecosystems, water resources, management practices, and agriculture. For many people, rainfall serves as the primary source of water, playing a vital role in their daily lives and livelihoods [2][3]. In Indonesia, rainfall is typically measured using two approaches: direct observation with rain gauges and indirect estimation through remote sensing [4]. Developing reliable rainfall forecasting systems remains a significant challenge for researchers across various fields, including weather data analysis, environmental machine learning, operational hydrology, and statistical forecasting methods [5][6].

Rainfall prediction is critically important because it is the variable most strongly correlated with natural disasters such as landslides, floods, mass movements, and avalanches. It involves various techniques such as statistics, data mining, modeling, artificial intelligence, and machine learning. Accurate and real-time rainfall prediction has been a significant challenge for decades due to its nonlinear and stochastic nature. Therefore, having an appropriate approach to predict rainfall allows for preventive and mitigation actions for these naturally occurring phenomena [2][7][8] .

Reliable climate information plays a vital role in agricultural decision-making, as climate and rainfall data are essential for identifying the most suitable crops for specific agricultural regions. The Oldeman classification is the most suitable system for agricultural zoning in Indonesia. It is commonly applied to assess climate suitability for agriculture because it relies on the consecutive number of wet and dry months over a minimum period of 10 years [9]. The Oldeman classification, similar to the Schmidt-Ferguson system, is based on rainfall intensity, linking wet and dry months to agricultural suitability in specific regions. The classification criteria are as follows:

- Wet month: Average monthly rainfall exceeds 200 mm.
- Humid month: Average monthly rainfall ranges between 100 and 200 mm.
- Dry month: Average monthly rainfall is below 100 mm.

This system is designed based on plant water requirements, particularly for paddy crops, where rainfall above 200 mm is ideal. In contrast, rainfall around 100 mm is more suitable for palawija (secondary) crops. [10]

OpenWeatherMap is an online platform that offers weather information, including real-time data, forecasts, and historical records, to developers of web and mobile applications. It gathers data from various sources such as meteorological services, airport weather stations, radar systems, and other official weather monitoring stations [11]. Weather forecast services, like the OpenWeatherMap API, offer extensive data through an API, including real-time analysis and predictions for upcoming days based on continuously updated forecasts [12].

One clear indicator of technological advancement is the emergence of Machine Learning (ML)[13]. Extreme Gradient Boosting (XGBoost) is an ensemble learning technique. In certain cases, the output of a single machine learning model, such as J48, may be insufficient. While J48 performs well and has been applied to various problems like wildfire prediction and rainfall modeling, its limitations can necessitate more robust approaches [14][15][16]. XGBoost extends this approach using gradient boosting with multi-threading for faster and more efficient computation. It includes sparse-aware processing for handling missing data, supports parallel tree construction, and enables incremental training. These features make XGBoost a powerful tool for classification, regression, and predictive modeling, especially with structured or tabular data [17][18].

Model evaluation is commonly performed using metrics such as Root Mean Square Error (RMSE) to assess the accuracy of estimations. According to research by Adiyo R. (2014), estimation methods such as Maximum Likelihood Robust (MLR) and Weighted Least Squares Mean and Variance adjusted (WLSMV) yield smaller RMSE values, indicating higher accuracy in test score estimation. Additionally, the number of categories and the range of thresholds also affect RMSE values, where a greater number of categories and a balanced range of thresholds tend to result in lower RMSE values, indicating improved model accuracy [19][20].

This study aims to develop a prediction and classification model for rainfall using the Extreme Gradient Boosting (XGBoost) algorithm, leveraging historical rainfall data from BMKG as training data and real-time data from OpenWeather API as input for predictions. The model will output rainfall trend graphs and classify rainfall intensity into categories such as light, moderate, or heavy. The research also seeks to evaluate the model's performance using metrics like accuracy, precision, RMS (Root Mean Square), and RMSE (Root Mean Square Error). The significance of this study lies in its contribution to integrating both historical and real-time data for weather forecasting, applying modern machine learning technologies like XGBoost, and providing a comprehensive model evaluation. Moreover, this research is expected to offer practical benefits for disaster mitigation, agricultural planning, and water resource management by developing a more accurate and reliable weather prediction model

2. RESEARCH METHOD

In this study, we used data from two sources :

Historical Rainfall Data from BMKG

Rainfall data for this study was sourced from the BMKG Maritime Station in Bitung, North Sulawesi, covering the period between January 1, 2019, and January 1, 2024. The dataset includes daily rainfall measurements that adhere to BMKG’s standardized protocols. This data serves as training input for building the prediction model.

Real-Time Data from OpenWeather API

Real-time weather data was retrieved from the OpenWeather API, offering up-to-date information on various weather parameters, including rainfall. This dataset is used to validate the model on data outside the training set, ensuring its predictive capability with real-time input.

Research Approach

Historical rainfall data was obtained from the BMKG Maritime Station in Bitung through official channels. This data covers multiple years and includes daily rainfall measurements. It is essential for training the prediction model as it offers a solid historical framework of weather patterns, helping the model identify long-term trends and anomalies effectively.

Real-time weather data was collected using API requests managed by Python scripts. These scripts were designed to retrieve data at regular intervals, ensuring the continuous collection of the most recent weather parameters, such as rainfall. This real-time data is crucial for model validation, as it assesses the model's ability to make predictions based on actual, current conditions. Moreover, the automated process of data retrieval enhances efficiency and reduces the need for manual intervention.

Preprocessing Data

- Data Collection

  The historical rainfall data was obtained from a cleaned Excel file, resulting in pre-processed data. Real-time weather data was collected using the OpenWeather API for predicting future rainfall.

- Data Preprocessing:

  The historical data was processed by adding additional features such as Year, Month, and Day. Data from the API was converted into a DataFrame, with rainfall values aggregated by date. Rainfall was classified into categories: Dry, Moderate, and Heavy.

- Model Training and Prediction:

  A trained XGBoost model was employed to predict rainfall based on features like Month, Year, and Day.

- Model Evaluation:

  The model's performance was evaluated using metrics such as MSE (Mean Squared Error), RMSE (Root Mean Square Error), precision, and accuracy to assess the quality of its predictions.

- Visualization and Presentation:

  Graphs and tables of the prediction results are presented within a web application to simplify the interpretation of the results.

XGBoost Model Development

- Data Splitting and Model Training

  The dataset was divided into two subsets: a training set and a testing set, with an 80% allocation for training and 20% for testing using train_test_split. The XGBoost model was employed for regression purposes. XGBoost is a boosting model that builds decision trees incrementally, making it especially efficient for handling large datasets and producing more accurate predictions, particularly with data that includes many nonlinear features and complex interactions.

- Hyperparameter Tuning

  GridSearchCV was used for hyperparameter tuning, evaluating different combinations of parameters to identify the best-performing model based on its performance on the training data. The parameters tested include:

  - n_estimators: The number of trees used.
  - max_depth: The maximum depth for each tree.
  - learning_rate: The rate at which the model updates its weights.
  - subsample: The proportion of the data used for training each tree.
  - colsample_bytree: The proportion of features selected for each tree.

Model Evaluation

The performance of the model was assessed using several evaluation metrics, including:

- Root Mean Square Error (RMSE): Measures the overall prediction error.
- RMS (Root Mean Square): Evaluates the level of deviation between predictions and actual data.
- Accuracy: Indicates the percentage of correct predictions for classification.
- Precision: Measures the accuracy of predictions for specific classes (e.g., light or heavy rainfall).

Rainfall Classification Using the Oldeman Method

Rainfall was classified into categories of light, moderate, and heavy using the Oldeman method. The results of this classification were analyzed to identify patterns in the distribution of rainfall over specific time periods.

Visualization and Implementation

- Rainfall Trend Graphs: Trend graphs were created using Python visualization libraries like Matplotlib and Seaborn to display rainfall patterns over the last five years.
- Localhost Application: The trained model was integrated into a Flask-based application, allowing for the visualization of both predictions and classifications of rainfall directly.

Tools and Technologies

This research employs the following tools and technologies:

a. Programming Language: Python 3.x for data processing, model development, and application implementation.
b. Libraries and Frameworks:
    - XGBoost: For developing predictive and classification models.
    - Pandas and NumPy: For data preprocessing.
    - Matplotlib and Seaborn: For data visualization and displaying prediction results.
    - Scikit-learn: For metric evaluation and dataset splitting.
    - Flask: For creating a simple localhost interface.
    - API: OpenWeather API for real-time data retrieval.
    - Hardware: A laptop or server with specifications suitable for intensive computation during model training.

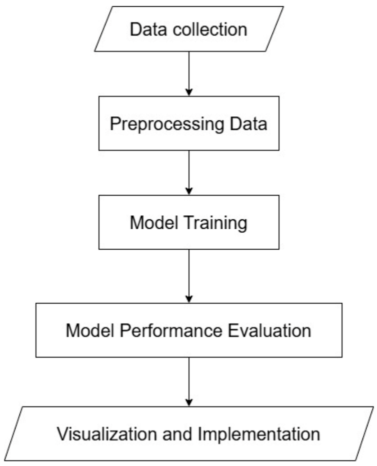

Figure 1. Flowchart

3. RESULT AND DISCUSSION

3.1 Rainfall Prediction Result with the XGBoost Model

The testing results of the XGBoost model demonstrate a good performance in predicting rainfall based on the historical data from BMKG. The model was evaluated using metrics like Mean Square Error (MSE), Root Mean Square Error (RMSE), Precision, and Accuracy. The evaluation results are as follows:

- Mean Square Error (MSE): 112.728 mm^2
- Root Mean Square Error (RMSE): 10.6173 mm
- Precision: 0.64
- Accuracy: 0.033

The model's predictions on the historical data show rainfall patterns that are quite close to the actual values, although there are some minor variations in data with high fluctuations. The prediction graph for the historical data is presented in Figure 1.

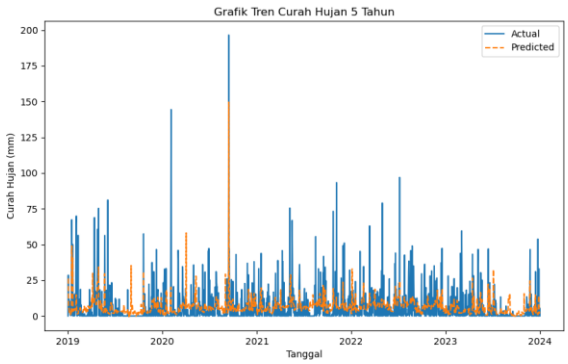

Figure 2. *Rainfall prediction graph based on historical BMKG data*

3.2 Rainfall Classification Based on Historical BMKG Data

Rainfall classification was done using the Oldeman criteria, which divides rainfall into dry, moderate, and heavy categories. Based on the historical data:

- Percentage of dry months: 15.28 %
- Percentage of humid month: 30.56%
- Percentage of wet months: 38.89%

Table 1 shows the annual rainfall classification linked with the number of dry, humid, and wet months for each year in the historical data.

Table 1. Rainfall Classification Based on Historical Data

Year	Total Rainfall (mm)	Classification	Wet Month	Humid month	Dry month
2019	2022.016	Klimat Tropis Basah	4	4	4
2020	2524.105	Klimat Tropis Sangat Basah	6	3	3
2021	2539.676	Klimat Tropis Sangat Basah	7	5	0
2022	2889.014	Klimat Tropis Sangat Basah	8	4	0
2023	1672.199	Klimat Tropis Basah	3	6	3
2024	0.1	Klimat Tropis Kering	0	0	1

3.3 Rainfall Prediction Based on OpenWeather API Data

The predictions of rainfall using real-time data from the OpenWeather API show patterns that match historical trends. These predictions are made for several days ahead, and the results are visualized in Figure 2 and Table 2

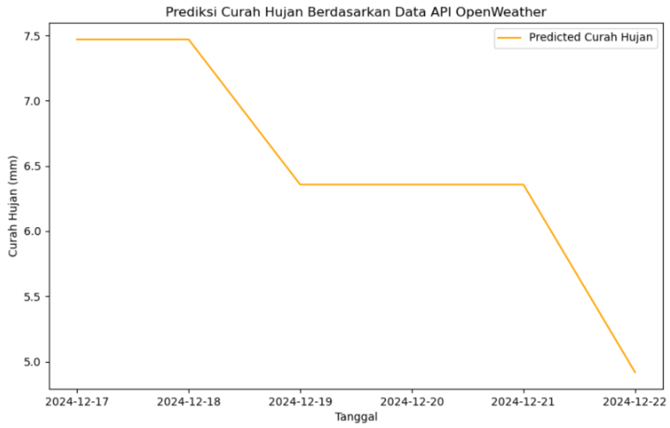

Figure 3. *Rainfall predictions based on OpenWeather API data*

This prediction shows that the model is able to provide fairly accurate rainfall estimates even though it uses data from external sources.

Table 2. Rainfall predictions based on OpenWeather API data.

Date	Rainfall Prediction (mm)
2024-12-17 00:00:00	7.4703521728515625
2024-12-18 00:00:00	7.4703521728515625
2024-12-19 00:00:00	6.357199192047119
2024-12-20 00:00:00	6.357199192047119
2024-12-21 00:00:00	6.357199192047119
2024-12-22 00:00:00	4.918496608734131
2024-12-17 00:00:00	7.4703521728515625

3.4 Rainfall Prediction for the Next 12 Months

The model was also used to predict rainfall for the next 12 months based on synthetic features generated from date data. The prediction graph for this period is shown in Figure 3 and Table 3

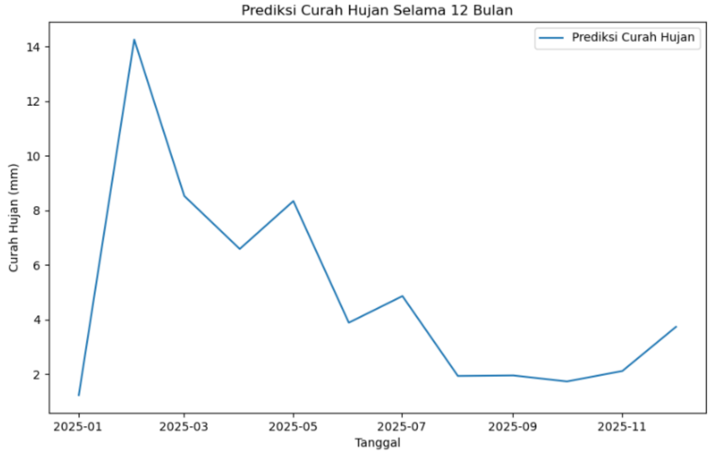

Figure 4. Rainfall predictions for the next 12 months

The prediction results display a seasonal pattern consistent with the characteristics of Indonesia's tropical climate, where rainfall tends to be higher during the rainy season. These predictions can serve as a reference for disaster mitigation and agricultural planning.

Table 3. Rainfall predictions for the next 12 months

Date	Rainfall Prediction (mm)
2025-01-01 08:45:03.968780	1.232616901397705
2025-02-01 08:45:03.968780	14.256916999816895
2025-03-01 08:45:03.968780	8.527671813964844
2025-04-01 08:45:03.968780	6.5872650146484375
2025-05-01 08:45:03.968780	8.341472625732422
2025-06-01 08:45:03.968780	3.886962413787842
2025-07-01 08:45:03.968780	4.860436916351318
2025-08-01 08:45:03.968780	1.9351284503936768
2025-09-01 08:45:03.968780	1.953909993171692
2025-10-01 08:45:03.968780	1.7334764003753662
2025-11-01 08:45:03.968780	2.116471767425537
2025-12-01 08:45:03.968780	3.7357137203216553

The Rainfall Prediction and Classification Dashboard presents evaluation results and predictions using the XGBoost model. On the main page, evaluation metrics are displayed such as MSE of 112.728, RMSE of 10.617, Precision of 0.6409, and Accuracy of 0.0333. These values indicate the model's performance in predicting rainfall based on both historical and real-time data. The website navigation includes a Trends Graph page showing rainfall pattern changes, Oldeman Classification for annual rainfall classification, and API Predictions which present real-time data from OpenWeather API in the form of tables and graphs. Additionally, the 12-Month Prediction page displays rainfall trends for the upcoming year. Visualizations are created using Matplotlib, while data processing is done with Pandas. This dashboard is designed to provide accurate and practical information supporting disaster mitigation, agricultural planning, and water resource management through the integration of machine learning and real-time data technologies.

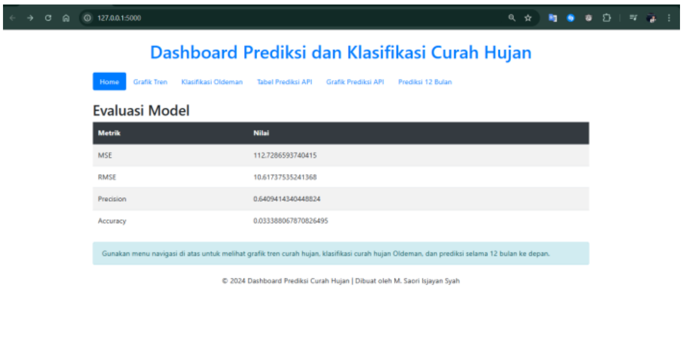

Figure 5. Model Visualization

The evaluation results show that the XGBoost model delivers strong performance in predicting rainfall using historical data. With an RMSE and accuracy, the model effectively reduces prediction errors. These results reflect the model's ability to produce predictions that closely align with actual data, demonstrating XGBoost's effectiveness in managing complex and non-linear rainfall datasets.

Several key factors that contributed to the success of the XGBoost model in this study include:

a. Capability to Process Tabular Data: XGBoost is specifically designed to handle structured tabular data, such as the BMKG dataset, making it ideal for modeling weather parameters like rainfall.
b. Optimized Gradient Boosting: XGBoost utilizes a boosting technique that iteratively corrects errors from previous iterations to progressively enhance model performance and improve prediction accuracy.
c. Automatic Handling of Missing Values: XGBoost can effectively manage datasets with missing values, a common issue in weather data. Enhanced Feature Engineering: Incorporating additional features such as year, month, and day into the model helps uncover seasonal trends and patterns in rainfall data

The XGBoost model's prediction results show a rainfall trend that aligns with the seasonal patterns observed in the study area (Bitung, North Sulawesi). This is evident in Figures 1 and 3, where rainfall increases during the rainy season and decreases during the dry season. The model effectively captured these seasonal variations, supported by the historical data spanning the 2019–2024 period used for training. Nevertheless, there are slight deviations between the predicted and actual values, especially during periods of significant fluctuation. These discrepancies may be influenced by several factors:

a) ENSO Variability (El Niño–Southern Oscillation): The ENSO phenomenon contributes to approximately 70% of rainfall variability in Indonesia. Consequently, in years with El Niño or La Niña events, prediction accuracy may decline.
b) Topography and Local Factors: The XGBoost model does not explicitly account for local factors such as topography and elevation, which can influence rainfall distribution in specific areas.
c) Input Data Limitations: Data obtained from the OpenWeather API may have different spatial and temporal resolutions compared to BMKG data, which can affect the quality of the predictions

Rainfall classification based on Oldeman’s criteria provides additional insight into the climatic conditions of the study area. For example:

- Years with a dominance of dry months indicate potential drought conditions that can impact the agricultural sector and water resources.
- Years with a dominance of wet months indicate potential risks of flooding and landslides.

The classification results presented in Table 1 show significant annual variations in the number of wet, humid, and dry months. This information can be used for agricultural planning and disaster mitigation. Additionally, these classification trends can help in determining more climate-adaptive planting patterns.

Integrating real-time data from the OpenWeather API highlights the flexibility of the XGBoost model for short-term predictions. The prediction results shown in Figure 3.2 demonstrate that the model can provide reasonably accurate rainfall estimates even with data from external sources. However, it’s important to consider the following challenges:

1) Data Quality: The data from the OpenWeather API depends on global modeling and spatial interpolation, which may be less precise compared to direct observations from BMKG stations. Temporal Resolution
2) Discrepancy: Real-time data from the API may have a higher temporal resolution (e.g., hourly), while the historical BMKG data used is in daily or monthly resolution.
3) Infrastructure Limitations: Utilizing API data requires a stable internet connection, which may be a challenge in some remote locations. Despite these challenges, the prediction results can provide early information on rainfall conditions in a region, which is valuable for decision-making in disaster risk mitigation and daily activities.

The findings of this research have several important implications, including:

- Disaster Mitigation: Accurate rainfall predictions enable governments and communities to take proactive measures against disasters such as floods, landslides, and droughts.
- Agricultural Planning: The annual rainfall classification information can help farmers determine more effective planting schedules based on seasonal patterns of wet and dry periods.
- Water Resource Management: Rainfall trend predictions assist in managing reservoirs, irrigation, and the supply of clean water during dry seasons.

- Real-Time Technology Use: Integrating real-time data from the OpenWeather API highlights the potential of Internet of Things (IoT)-based technologies for monitoring weather conditions in real time

While the research results demonstrate good performance, there are some limitations that need to be considered:

- Climate Variability: The model does not explicitly account for global climate factors like ENSO in its input features.
- Data Resolution: Historical BMKG data is used at a daily resolution, whereas the real-time API predictions have an hourly resolution, which may impact the prediction outcomes.
- Local Factors: Local topography has not been fully integrated into the model, yet it influences rainfall patterns in hilly or mountainous areas.

Based on the findings of this research, several recommendations for future development include:

- Integrating ENSO Factors: Incorporating ENSO indices (El Niño/La Niña) as input variables to improve prediction accuracy during extreme conditions.
- Higher Spatial Resolution: Combining data from multiple BMKG stations within the same region to enhance the spatial resolution of predictions.
- Satellite Data Use: Integrating satellite observation data such as TRMM (Tropical Rainfall Measuring Mission) to complement historical data.
- Model Optimization: Evaluating other models like LSTM (Long Short-Term Memory) or combinations of XGBoost and deep learning to further enhance prediction performance.

4. CONCLUSION

This study successfully developed a rainfall prediction and classification model utilizing the Extreme Gradient Boosting (XGBoost) algorithm, incorporating historical data from BMKG and real-time data from the OpenWeather API. The model's performance evaluation yielded a Mean Squared Error (MSE) of 112.728, Root Mean Squared Error (RMSE) of 10.617, Precision of 0.6409, and Accuracy of 0.0333, reflecting moderate predictive capability. The results are presented through a Flask-based dashboard that showcases model evaluations, rainfall trend visualizations, Oldeman-based rainfall classification, and forecasts for the next 12 months. By integrating historical and real-time data, the system enables dynamic and up-to-date predictions, offering practical applications for disaster management, agricultural planning, and water resource optimization. This research highlights the potential of machine learning technologies in enhancing rainfall prediction accuracy, supporting informed decision-making across multiple sectors.

REFERENCE

[1] M. A. Priatna and E. C. Djamal, “Precipitation prediction using recurrent neural networks and long short-term memory,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 18, no. 5, pp. 2525–2532, Oct. 2020, doi: 10.12928/TELKOMNIKA.V18I5.14887.

[2] S. Poornima and M. Pushpalatha, “Prediction of rainfall using intensified LSTM based recurrent Neural Network with Weighted Linear Units,” *Atmosphere (Basel)*, vol. 10, no. 11, Nov. 2019, doi: 10.3390/atmos10110668.

[3] N. K. A. Appiah-Badu, Y. M. Missah, L. K. Amekudzi, N. Ussiph, T. Frimpong, and E. Ahene, “Rainfall Prediction Using Machine Learning Algorithms for the Various Ecological Zones of Ghana,” *IEEE Access*, vol. 10, pp. 5069–5082, 2022, doi: 10.1109/ACCESS.2021.3139312.

[4] M. Putra, M. S. Rosid, and D. Handoko, “High-Resolution Rainfall Estimation Using Ensemble Learning Techniques and Multisensor Data Integration,” *Sensors*, vol. 24, no. 15, Aug. 2024, doi: 10.3390/s24155030.

[5] E. G. Dada, H. J. Yakubu, and D. O. Oyewola, “Artificial Neural Network Models for Rainfall Prediction,” *European Journal of Electrical Engineering and Computer Science*, vol. 5, no. 2, pp. 30–35, Apr. 2021, doi: 10.24018/ejece.2021.5.2.313.

[6] M. Mohammed, R. Kolapalli, N. Golla, and S. S. Maturi, “Prediction Of Rainfall Using Machine Learning Techniques,” *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*, vol. 9, p. 1, 2020, [Online]. Available: www.ijstr.org

[7] A. Sharma and G. Nijhawan, “Rainfall Prediction Using Neural Network,” *International Journal of Computer Science Trends and Technology*, vol. 3, 2015, [Online]. Available: www.ijcstjournal.org

[8] Q. Zou, Y. Liu, and X. Linge, “A survey on rainfall forecasting using artificial neural network’,” 2019.

[9] I. S. Harahap, I. Z. Matondang, Suryanto, E. K. Indah, and I. Fitri, “Mapping Climate Classification of Oldeman in Agricultural Resources Management in South Tapanuli District,” *IOP Conf Ser Mater Sci Eng*, vol. 1156, no. 1, p. 012002, Jun. 2021, doi: 10.1088/1757-899x/1156/1/012002.

[10] B. Ahmada *et al.*, “Climate-based crop pattern determination using Standard Precipitation Index (SPI) and the Oldeman classification in Sangiran Site,” in *IOP Conference Series: Earth and Environmental Science*, Institute of Physics Publishing, Mar. 2020. doi: 10.1088/1755-1315/451/1/012019.

[11] C. Dewi and R.-C. Chen, “Integrating Real-Time Weather Forecasts Data Using OpenWeatherMap and Twitter,” 2019. [Online]. Available: http://ejournal.uksw.edu/ijiteb

[12] A. Musah *et al.*, “An Evaluation of the OpenWeatherMap API versus INMET Using Weather Data from Two Brazilian Cities: Recife and Campina Grande,” *Data (Basel)*, vol. 7, no. 8, Aug. 2022, doi: 10.3390/data7080106.

[13] C. Sugiarto, F. Debora Abigael, Y. Faiz Athallah, A. Hari Saputra, and S. Tinggi Meteorologi Klimatologi dan Geofisika, “UTILIZATION OF HUBERREGRESSOR MACHINE LEARNING MODEL TO PREDICT CARBON MONOXIDE CONCENTRATION IN SURABAYA CITY,” *JCEBT*, vol. 8, no. 1, 2024, [Online]. Available: http://ojs.uma.ac.id/index.php/jcebt

[14] M. T. Anwar, H. D. Purnomo, S. Y. J. Prasetyo, and D. K. Hartomo, “Decision Tree Learning Approach To Wildfire Modeling on Peat and Non-Peat Land in Riau Province,” 2018.

[15] B. Mahesh, “Machine Learning Algorithms - A Review,” *International Journal of Science and Research (IJSR)*, vol. 9, no. 1, pp. 381–386, Jan. 2020, doi: 10.21275/art20203995.

[16] J. Kang, H. Wang, F. Yuan, Z. Wang, J. Huang, and T. Qiu, “Prediction of precipitation based on recurrent neural networks in jingdezhen, jiangxi province, China,” *Atmosphere (Basel)*, vol. 11, no. 3, Mar. 2020, doi: 10.3390/atmos11030246.

[17] N. Uzir, S. Raman, S. Banerjee, and R. S. Nishant Uzir Sunil R, “Experimenting XGBoost Algorithm for Prediction and Classification of Different Datasets,” *International Journal of Control Theory and Applications*, vol. 9, 2016, [Online]. Available: https://www.researchgate.net/publication/318132203

[18] Aditya Gumilar, Sri Suryani Prasetiyowati, and Yuliant Sibaroni, “Performance Analysis of Hybrid Machine Learning Methods on Imbalanced Data (Rainfall Classification),” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 3, pp. 481–490, Jul. 2022, doi: 10.29207/resti.v6i3.4142.

[19] R. Adityo, “PENGARUH JUMLAH KATEGORI, RENTANG THRESHOLD DAN METODE ESTIMASI TERHADAP AKURASI SKOR TES PADA BEBERAPA MODEL POLITOMI,” 2014.

[20] H. Zheng, J. Yuan, and L. Chen, “Short-Term Load Forecasting Using EMD-LSTM neural networks with a xgboost algorithm for feature importance evaluation,” *Energies (Basel)*, vol. 10, no. 8, Aug. 2017, doi: 10.3390/en10081168.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 31 - 39
https://journal.physan.org/index.php/jocpes/index

PM2.5 Concentration Prediction Model in Jakarta Area Using Random Forest Algorithm

Muhammad Naufal Afif Al Arsy[1] Ahmad Meijlan Yasir[1]
[1]Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)

Article Info

Article history:

Received March 3, 2025
Revised March 8, 2025
Accepted March 9, 2025

Keywords:

PM2.5 Prediction
Random Forest
Air Quality
Machine Learning, Jakarta,
Hyperparameter Tuning
Environmental Monitoring
Data-Driven Policy

A B S T R A C T

This study predicts PM2.5 concentrations in Jakarta using the Random Forest algorithm with historical air quality data from 2015 to 2024. Hyperparameter tuning was performed to optimize model performance, focusing on parameters such as n_estimators, max_depth, and min_samples_split. The model achieved a Mean Absolute Error (MAE) of 14.44, a Root Mean Square Error (RMSE) of 18.75, and an R^2 Score of 0.61. While the model captured general PM2.5 fluctuation patterns, deviations at certain points indicate room for improvement. Descriptive analysis showed an average PM2.5 concentration of 94.46 µg/m³, with peaks up to 209 µg/m³, exceeding healthy air quality thresholds. The model can be integrated into real-time monitoring systems and support data-driven policies. Future work could incorporate meteorological variables and evaluate longer-term trends to enhance accuracy.

Corresponden Author:

Muhammad Naufal Afif Al Arsy,
Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)
Tangerang City, Banten, Indonesia
Email: nono.naufal10@gmail.com

1. INTRODUCTION

PM2.5, a fine air particle with a diameter of ≤2.5 µm, is one of the most harmful pollutants to human health because its small size allows it to penetrate into the bloodstreams. Long-term exposure to PM2.5 has been shown to contribute to cardiovascular disease, chronic respiratory distress, and lung cancer. The WHO reports that air pollution, including PM2.5, causes more than seven million premature deaths each year worldwide (Mauboy et al., 2024).

For instance, the average ambient concentration of PM2.5 in Jakarta frequently violates the WHO safe limit of 10 µg/m³, and the significant sources include mobile and stationary emission from transport, industries, and burning of fossils. In addition, meteorological conditions such as temperature, humidity, wind speed, and rainfall also affect the distribution pattern of these pollutants (Goyal & Goyal, 2024; Joharestani et al., 2019). The study found that PM2.5 concentrations tend to increase during the dry season, where low rainfall reduces the atmosphere's ability to deposit particles (Kang et al., 2023).

Recent development of the machine learning has produced better solutions for handling and predicting air quality. Out of all the algorithms used in this study, it is worth mentioning the effectiveness of Random Forest in relation to nonlinear dependencies and texture complexity. This model has been well applied to predict PM2.5 using meteorological variables, satellite data and observation in the region. The findings suggest that Random Forest also has reasonable prediction accuracy compared to other methods, like XGBoost or SVM (Joharestani et al., 2019; Ma et al., 2023; Uzir et al., 2016).

For instance, a study that employed Random Forest technique in China obtained a good performance of up to 0.81 R^2 value in the determination of PM2.5 concentrations (Ma et al., 2023). A similar study conducted in Jakarta also supports the finding that the algorithm holds capability to predict the certain pollutants average concentration through meteorological parameters and air quality data (Joharestani et al., 2019; Mauboy et al., 2024). Not only are such models useful for achieving high prediction accuracy, but they can also be employed to create warning systems for the public and generate decision-making support for governments (Chen et al., 2023). The focus of this study is to present a PM2.5 concentration prediction model for Jakarta based on the Random Forest algorithm. Based on the above literature review, the following research questions which are the main focuses of this study are identified.

First, what are the temporal and spatial patterns of PM2.5 concentrations in the Jakarta area based on historical data? Understanding this pattern is important because air pollution in Jakarta often varies depending on time and location, which can be studied using machine learning algorithms, such as Random Forest, which have been shown to be effective in predicting air pollution in various large urban areas(Goyal & Goyal, 2024).

The second problem is how do meteorological parameters such as temperature, humidity, wind speed, and rainfall affect PM2.5 concentrations in Jakarta? Previous research has shown that these factors greatly affect pollutant concentration levels, and understanding these relationships can improve the accuracy of the predictive models built (Joharestani et al., 2019; Ma et al., 2023).

The third problem that needs to be answered is how the Random Forest algorithm can be used to accurately predict PM2.5 concentrations in Jakarta. Random Forest is known for its ability to handle non-linear data and can identify important features that affect air quality, making it a good choice for air pollutant prediction (Kang et al., 2023; Ma et al., 2023).

Finally, how can the prediction results from this model be used to support air quality management policies in Jakarta? Accurate predictive models can provide early warning of increased PM2.5 concentrations, allowing for appropriate mitigation measures to protect public health. This will help the government in formulating air quality management policies based on more accurate data (Chen et al., 2023).

This study aims to analyze the temporal and spatial patterns of PM2.5 concentrations in Jakarta based on historical data. This analysis is important for understanding the distribution of air pollution by time (daily, weekly, or seasonal) and location (by region with high and low pollution levels). Several previous studies have shown that models like Random Forest are very effective in identifying such patterns, especially in large urban areas such as Beijing and Jing-Jin-Ji (Joharestani et al., 2019; Ma et al., 2023).

In addition, this study evaluated the relationship between meteorological parameters, such as temperature, humidity, wind speed, and rainfall, with PM2.5 concentrations in Jakarta. Meteorological factors play an important role in the distribution of air pollutants, where rainfall tends to lower PM2.5 concentrations, while humidity and wind speed can affect their spread (Kang et al., 2023; Ma et al., 2023).

This study also developed a PM2.5 concentration prediction model using the Random Forest algorithm. This model was chosen because of its ability to handle non-linear data and identify important features in complex datasets. The developed model will be tested using historical air quality data and meteorological parameters to ensure prediction accuracy (Chen et al., 2023; Ma et al., 2023). The results of the prediction are expected to be used to support air quality management policies in Jakarta, including providing early warning to the public about deteriorating air conditions (Chen et al., 2023; Kang et al., 2023).

This research has several benefits. For governments, the predicted results can be used to support data-driven policies for air pollution mitigation, such as controlling motor vehicle emissions and reducing industrial activities during periods with high pollution levels (Chen et al., 2023; Kang et al., 2023). For the public, predictive information can increase awareness of health risks and provide reliable data for preventive measures such as mask use or restrictions on outdoor activities (Chen et al., 2023; Joharestani et al., 2019). For academics, this study enriches the scientific literature related to the application of the Random Forest algorithm in air quality prediction and opens up opportunities for further research, such as integration with satellite data for more accurate predictions (Joharestani et al., 2019; Ma et al., 2023). Environmentally speaking, this research helps identify times and locations with high pollution risks, supporting more effective mitigation measures (Chen et al., 2023).

2. LITERATURE REVIEW

2.1 Air Quality Prediction Using Machine Learning Techniques

Air quality prediction has become a critical task for urban management due to the rising concerns about pollution and its effects on human health. Traditional statistical methods, such as linear regression, have been used to predict air quality levels, but they often fail to capture the complex, non-linear relationships found in air pollution data (Grell et al., 2005). In recent years, machine learning (ML) techniques have been widely adopted to improve the accuracy of air quality prediction models. These techniques can handle large datasets, identify complex patterns, and adapt to real-time data, making them suitable for urban air quality management.

2.2 Common Machine Learning Techniques for Air Quality Prediction

Machine learning methods such as Random Forest, Decision Trees, Artificial Neural Networks (ANN), and Support Vector Machines (SVM) have been applied to predict air quality indices (AQI) and particulate matter (PM) concentrations. Each method has its strengths and limitations in terms of performance, accuracy, and computational complexity.

a. Random Forest

The Random Forest algorithm has been widely used for air quality prediction due to its robustness in handling noisy data and reducing overfitting (Ameer et al., 2019). In their study, (Yu et al., 2016) applied Random Forest for predicting air quality using urban sensing systems and found it to outperform other methods in terms of accuracy. Similarly, (Brokamp et al., 2017) compared regression and Random Forest approaches for estimating particulate matter levels, concluding that Random Forest provided more accurate predictions in urban environments(Liu et al., 2012; Livingston, 2005; Segal, 2003).

b. Support Vector Machines (SVM)

SVM is another popular machine learning algorithm used for air quality prediction(Lu' et al., 2002) utilized SVM for spatiotemporal analysis of air quality data derived from social media and found it to be effective in predicting AQI levels. However, the computational cost of SVM increases significantly with larger datasets, which can limit its scalability in real-time applications.

c. Artificial Neural Networks (ANN)

ANN, especially Deep Learning models, have been applied for time-series analysis in air quality prediction. (Dobrea et al., 2020) used ANN models to forecast air pollutant concentrations, reporting high accuracy but also noting the increased training time and risk of overfitting, particularly with small datasets. (Sharma et al., 2021) also employed ANN for predicting air pollutant levels, emphasizing its capability to model non-linear relationships in pollution data.

d. Decision Trees

Decision Tree algorithms, including Gradient Boosting and its variants, have shown promise for air quality prediction. Decision Trees are easy to interpret and can handle categorical data effectively. (Jamal & Nodehi, 2017) demonstrated the use of Decision Trees for meteorological data analysis and air quality forecasting, highlighting their useful application in real-time decision support systems.

2.3 Fuzzy Time Series and Hybrid Models

Other research approaches have included the use of hybrid and fuzzy models with the view of enhancing the accuracy of the machine learning. Currently, (Alyousifi et al., 2020) developed a model known as Fuzzy Time Series Markov Chain that integrates the fuzzy logic system with probabilistic Markov models for daily air pollution index prediction. This method was also found to reduce significant uncertainty for air quality data better than other conventional techniques as it can offer glimpses into the changes in temporal trends in pollution.

2.4 Big Data and IoT-Based Air Quality Prediction

The key development in the field of air quality prediction is the usage of Internet of Things (IoT) and Big Data technologies. Various IoT sensors spread across the cities can capture current pollution levels, and such data can be analyzed using Machine Learning techniques to predict the current pollution levels. (Ameer et al., 2019) presented a systematic review on machine learning models for air quality estimation in smart urban environments and implemented Apache Spark for distributed computing. Their work also stressed the role of time to process in huge-scale AQM, Random Forest, and Gradient Boosting turned out to be dominant in regards to accuracy and computational time.

Similarly, the study by (Yu et al., 2016) identified that IoT-based urban sensing systems are helpful in predicting air quality. Random forest used by the authors for creating their model proved effective along with the ability of their algorithms in conceiving pollution level their IoT based system could analyze data from multiple sensors demonstrating the capability of IoT for air quality monitoring in real-time.

2.5 Comparative Studies on Machine Learning Models for Air Quality Prediction

Some comparative analysis has been made in order to compare the efficacy of several types of the machine learning algorithms in solving the problem of air quality prediction. For example, (Agarwal, 2015) have compared the accuracy of multiple classifiers as ANN, SVM, Decision Trees; thereby, concluding that although ANN had the highest accuracy, it consumed more energy. Similarly, (Sharma et al., 2021)employed a performance comparison of different machine learning approaches and found that Random Forest was again most accurate and less sensitive to fluctuation.

Ameer et al. (Ameer et al., 2019) proposed a comprehensive analysis of the regression techniques such as Decision Trees, Random Forest, and Gradient Boosting that are employed with different data sets of smart cities. They concluded that the two best algorithms of the two groups, Random Forest and Gradient Boosting were the best, but it was recommended that other issues like time and size of the data set be considered first while choosing a model.

2.5 Challenges and Future Directions

However, some issues still persist with the machines learning models used in the prediction of air quality. Such issues as how to manage large swathes of missing data, how to be able to process large chunks of data in real time, and how to fuse data from different and dissimilar sources such as weather conditions and traffic flows. Furthermore, issues such as the time taken to train the models like ANN and SVM in big datasets are costly in large applications (Dobrea et al., 2020); (Sharma et al., 2021).

Future studies should therefore aim at identifying better associated algorithms that allow fast and real time processing of massive amount of dataset generated from IoT sensors. At the same time, communication of hybrid models, which use machine learning in combination with prior knowledge from the atmosphere chemistry models, could provide higher accuracy and stabilities in the air quality prediction compared to the application of pure machine learning methods (Alyousifi et al., 2020).

3. METHODOLOGY

In the following part of the paper, the methodology used to compare various machine learning algorithms for measuring air quality is presented. Thus the evaluation system can be divided into several steps, data preprocessing, feature selection, model building, model's performance analysis and comparison with other models. Each step is detailed below.

3.1 Research Design

This study uses a quantitative approach to build a prediction model of PM2.5 concentration in Jakarta based on historical data available at the AQICN.org site from 2015 to 2024. The Random Forest algorithm was chosen as the main model because of its ability to handle complex non-linear relationships between variables and because of its robustness against overfitting. This study aims to understand the temporal pattern of PM2.5 concentration in Jakarta and produce accurate predictions that can support air pollution risk mitigation in the region.

3.2 Data Source

The PM2.5 concentration data used in this study was obtained from AQICN.org, a platform that provides real-time-based air quality data as well as historical data from various air monitoring stations in Jakarta. The data used includes PM2.5 concentrations recorded hourly during the period 2015 to 2024. This data was chosen to illustrate the daily fluctuations and temporal trends of PM2.5 in Jakarta, focusing on patterns that occur throughout the year.

3.3 Data Collection

The PM2.5 concentration data used in this study was obtained from AQICN.org for the period 2015 to 2024. The observation time range used includes hour-time data, allowing for more accurate modeling based on PM2.5 fluctuations that occurred during the period. This data was chosen for its ability to reflect daily and seasonal variability in Jakarta.

3.4 Data Preprocessing

Before permanently storing the measured PM2.5 data in the SQL database at this stage, the collected PM2.5 data is initially analyzed for unwanted empty entries and excessive outliers. In case of gaps, linear interpolation approaches are used, and for the cases where anomalies or outliers are observed, they are detected and deleted from data using applicable methods. This results from adding new temporal features like year, and month of the data, day of the year, days of the month to help identify seasonal variations that may be influential in PM2.5 concentration. Due to the fact that the data is presented in a yearly format with no further divisions as morning, afternoon or night, the differences and patterns are more apparent in yearly, monthly, and seasonal contexts.

3.5 Predictive Model Development

The used algorithm for building a prediction model which is the basis for decision making was the Random Forest Regressor that can deal with non-linear correlations and distinguish intricate patters. The dataset is divided into two parts: Taking for example 70% of the data will be used in training the model while 30% will be used to test the model. This is followed by dividing the data in order to be able to predict PM2.5 using the generated model should be able to generalize data that is not in the training data set. To improve the accuracy of the model, hyperparameter tuning is performed using Grid Search to find the best combination of hyperparameters, including the number of trees (n_estimators), the maximum depth of the trees (max_depth), and the minimum sample size for node division (min_samples_split).

3.6 Model Variations

The model is evaluated using several metrics to measure the accuracy and accuracy of the model's predictions against the actual data. The metrics used include Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R^2 Score. These metrics are used to assess how well the model is at capturing fluctuations that occur in PM2.5 data and to measure the model's ability to predict data that is not present in the training dataset.

3.7 Result Visualization and Analysis

The results of the model for PM2.5 predictions are shown as temporal graph patterns representing daily, monthly, and annual changes of the concentration values. These graphs will be used to determine specific months of the year that have a higher PM2.5 concentration as well as to make observation on seasonal variation.

The analysis of the results each model were analyzed to ensure that it well suits the actual situation that takes place in Jakarta and to determine specific time that PM2.5 is likely to be higher within the day so as to guide the formulation of air pollution control measures.

4. RESULT AND DISCUSSION

4.1 Descriptive Statistics of PM2.5 Data

Based on PM2.5 concentration data from 2015 to 2024, descriptive statistics and average graphs and distribution of PM2.5 concentrations are obtained as follows:

In order to improve the efficiency of AWS maintenance, the primary goal of this article is to build predictive maintenance for AWS based on identifying anomalies utilizing artificial intelligence autoencoders that can speed up the predictive maintenance flow. To be able to design the desired predictive maintenance, the machine learning autoencoder is used. Anomaly detection is used to predict AWS before failure. This research will be designed in Google collab, which is easy using applications and simple features. The main objective of this study is to create predictive maintenance for Automatic Weather Station (AWS) to avoid the failure of instrumentation.

Table 1. Descriptive Statistics of PM2.5 Data

Table 1. Descriptive Statistics of PM2.5 Data

No	*Statistic*	*Date*	*PM2.5 (µg/m³)*
1	Total	3093	3093
2	Average	-	94,46
3	Minimum Values	2015-12-25	7
4	1st Quartile	2018-03-20	72
5	Median (Q2)	2020-05-28	97
6	3rd Quartile	2022-08-31	117
7	Maximum Value	2024-12-13	209
8	Standard Deviation	-	30,17

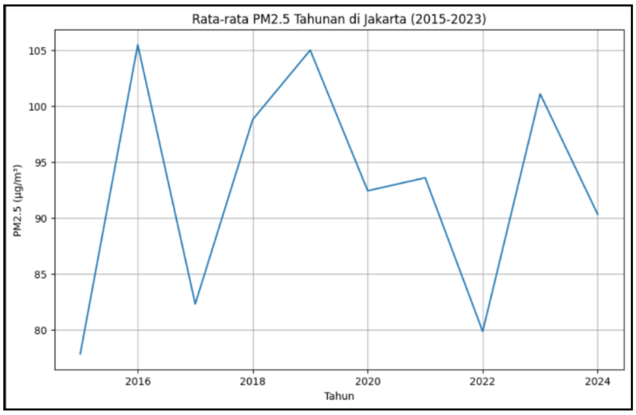

Fig.1. Average Annual PM2.5 in Jakarta (2015-2023)

This statistic shows that PM2.5 concentration has an average of 94.46 µg/m³ with a significant variation, as seen from the standard deviation of 30.17 µg/m³. The maximum value of PM2.5 concentration reached 209 µg/m³, which far exceeded the threshold of healthy air quality.

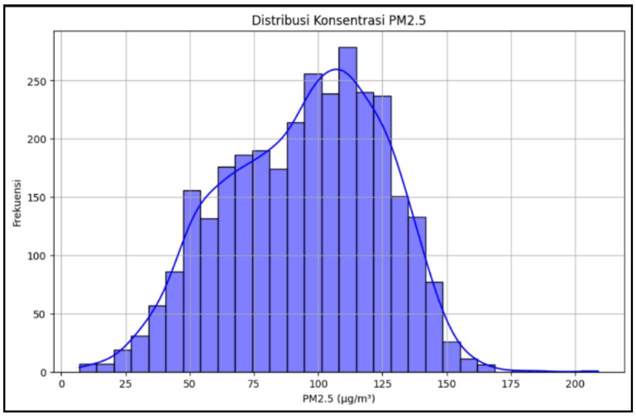

Fig.2. PM2.5 Concentration Distribution

The histogram plot depicts the distribution of PM2.5 concentration levels in the Jakarta region. The distribution exhibits a normal curve-like shape, with a peak around 140-160 µg/m³. This suggests that PM2.5 concentrations in Jakarta tend to be concentrated within this range. Several key observations can be made from the graph:

- Concentration Range: The graph covers a PM2.5 concentration range from approximately 50 µg/m³ to 200 µg/m³, encompassing the observed values in the region.
- Distribution Peak: The distribution peak occurs around 140-160 µg/m³, indicating that this is the most common PM2.5 concentration level observed.
- Distribution Shape: The distribution curve exhibits a relatively symmetric shape, following a normal distribution pattern. This implies that PM2.5 concentrations are evenly distributed across the observed range.
- Frequency: The graph shows the frequency of occurrence for each PM2.5 concentration range. Taller bars correspond to higher frequencies of the respective concentration levels.

The analysis from this study can therefore assist in refining the identification of the nature and spatial pattern of PM2.5 density in the Jakarta region. In extension, the data can be used as reference in the evaluation of the air quality in the said locale as well as to the proper measures that should be implemented.

4.2 Evaluation of Analysis Models

The results of the performance evaluation of the PM2.5 concentration prediction model using Random Forest show the following metrics:

Table 2. Results of Random Forest Model Evaluation

No	*Metrix*	*Value*
1	Mean Absolute Error (MAE)	14.44479525862069
2	Root Mean Square Error (RMSE)	18.746648967790264
3	R^2 Score	0.6082636701026527

This metric shows that the model performs quite well with an R^2 Score of 0.61, which means that the model is able to account for about 61% of the variation in the actual data. However, a fairly high RMSE (18.75) indicates a significant difference between the predicted value and the actual data at some point.

4.3 Comparation of Model Prediction with Actual Data of PM2.5 Concentration

The objective of this research is to create a model for analyzing the dispersion of PM2.5 concentration in the Jakarta area based on Random Forest modeling. For this purpose, we first analyze whether the model correctly estimates the observed values of PM2.5 concentration at different time intervals. As indicated in the following graph, we have the results of the research.

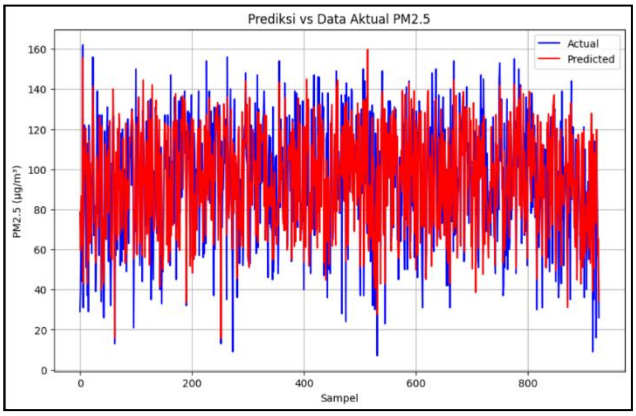

Fig.3. Prediction vs Actual Data PM2.5

In general, there is always a tight correspondence between the changes predicted by the prediction models and the actual change data even if they do not have to be identical. A few observations I made from this chart are the following:

- Fluctuation Pattern Suitability: The adjustment of the prediction model can reflect the approximate fluctuation of PM2.5 concentration, confirming the model's performance in perceiving changing trends.
- Differences in Prediction Values: Thus, the picture of simulation results indicates that, while the patterns of fluctuations are practically identical, there are certain differences in some points of the model's prediction indicators and real data. This implies that it is still possible even to enhance the predictive ability of the models and grow the accuracy of the results.
- Overall Performance: Overall, the model was successful in estimating PM2.5 concentrations but still requires enhancements to minimize the difference to actual data.

4.4 Improved Model Accuracy

To improve the accuracy of the model, hyperparameter tuning is performed using the Grid Search method, which involves the following parameters:

- n_estimators: Number of trees in the model.
- max_depth: Maximum depth of the tree.
- min_samples_split: Minimum sample size for node separation.

This tuning succeeded in reducing the error value of the model, but there are still limitations due to the limitations of the data period obtained and the type of data that other types of data are used as inputs. Further evaluation of long-term trends and the addition of variations in data types such as temperature and humidity can provide a better understanding of model performance and factors affecting PM2.5 concentrations.

5. CONCLUSION

This study aims to predict PM2.5 concentrations in the Jakarta area using the Random Forest model. The results show that this model has a fairly good performance with a Mean Absolute Error (MAE) of 14.44, a Root Mean Square Error (RMSE) of 18.75, and an R^2 Score of 0.61. The R^2 Score value shows that the model is able to explain about 61% of the actual PM2.5 data variation, although there are still significant differences at some prediction points indicated by the high RMSE value. In general, the model successfully follows a pattern of fluctuations in PM2.5 concentrations, although it is not yet completely accurate.

Based on descriptive statistical analysis, PM2.5 concentration data from 2015 to 2024 has an average of 94.46 µg/m³ with significant variations, as seen from the standard deviation of 30.17 µg/m³. The maximum value reaches 209 µg/m³, which is well above the threshold of healthy air quality. Efforts to improve the accuracy of the model have been made through tuning hyperparameters using Grid Search, which helps improve prediction performance. However, limitations still exist because comparisons are not made with data that covers longer periods.

This prediction model has the potential to be implemented in various scenarios, such as real-time air quality monitoring, supporting data-based policymaking for air pollution mitigation, and developing public applications that can provide air quality prediction information to the public. For future studies, it is recommended to use data with a longer period, consider other factors such as meteorological parameters (temperature, humidity, and wind speed) as well as local pollution emission sources, and compare the performance of this model with other prediction models to obtain more optimal results.

Thus, this research is expected to contribute to air quality monitoring efforts and become the basis for formulating more effective policies to reduce the impact of air pollution in the Jakarta area.

REFERENCE

[1] Agarwal, Amit. (2015). Proceedings on 2015 1st International Conference on Next Generation Computing Technologies (NGCT) : September 4th-5th, 2015, Center for Information Technology, University of Petroleum and Energy Studies, Dehradun. IEEE.

[2] Alyousifi, Y., Othman, M., Sokkalingam, R., Faye, I., & Silva, P. C. L. (2020). Predicting daily air pollution index based on fuzzy time series markov chain model. Symmetry, 12(2). https://doi.org/10.3390/sym12020293

[3] Ameer, S., Shah, M. A., Khan, A., Song, H., Maple, C., Islam, S. U., & Asghar, M. N. (2019). Comparative Analysis of Machine Learning Techniques for Predicting Air Quality in Smart Cities. IEEE Access, 7, 128325–128338. https://doi.org/10.1109/ACCESS.2019.2925082

[4] Brokamp, C., Jandarov, R., Rao, M. B., LeMasters, G., & Ryan, P. (2017). Exposure assessment models for elemental components of particulate matter in an urban environment: A comparison of regression and random forest approaches. Atmospheric Environment, 151, 1–11. https://doi.org/10.1016/j.atmosenv.2016.11.066

[5] Chen, M. H., Chen, Y. C., Chou, T. Y., & Ning, F. S. (2023). PM2.5 Concentration Prediction Model: A CNN–RF Ensemble Framework. International Journal of Environmental Research and Public Health, 20(5). https://doi.org/10.3390/ijerph20054077

[6] Dobrea, M., Badicu, A., Barbu, M., Subea, O., Balanescu, M., Suciu, G., Birdici, A., Orza, O., & Dobre, C. (2020). Machine Learning algorithms for air pollutants forecasting. 2020 IEEE 26th International Symposium for Design and Technology in Electronic Packaging, SIITME 2020 - Conference Proceedings, 109–113. https://doi.org/10.1109/SIITME50350.2020.9292238

[7] Goyal, K., & Goyal, S. (2024). Predicting PM2.5 Air Quality Using Random Forest Regression Enhanced with Polynomial Features. International IEEE Conference Proceedings, IS, 2024. https://doi.org/10.1109/IS61756.2024.10705219

[8] Grell, G. A., Peckham, S. E., Schmitz, R., McKeen, S. A., Frost, G., Skamarock, W. C., & Eder, B. (2005). Fully coupled "online" chemistry within the WRF model. Atmospheric Environment, 39(37), 6957–6975. https://doi.org/10.1016/j.atmosenv.2005.04.027

[9] Jamal, A., & Nodehi, R. N. (2017). A R T I C L E I N F O R M A T I O N PREDICTING AIR QUALITY INDEX BASED ON METEOROLOGI-CAL DATA: A COMPARISON OF REGRESSION ANALYSIS, ARTI-FICIAL NEURAL NETWORKS AND DECISION TREE. In Journal of Air Pollution and Health (Vol. 2, Issue 1). http://japh.tums.ac.ir

[10] Joharestani, M. Z., Cao, C., Ni, X., Bashir, B., & Talebiesfandarani, S. (2019). PM2.5 prediction based on random forest, XGBoost, and deep learning using multisource remote sensing data. Atmosphere, 10(7). https://doi.org/10.3390/atmos10070373

[11] Kang, J., Zou, X., Tan, J., Li, J., & Karimian, H. (2023). Short-Term PM2.5 Concentration Changes Prediction: A Comparison of Meteorological and Historical Data. Sustainability (Switzerland), 15(14). https://doi.org/10.3390/su151411408

[12] Liu, Y., Wang, Y., & Zhang, J. (2012). New Machine Learning Algorithm: Random Forest. In LNCS (Vol. 7473).

[13] Livingston, F. (2005). Implementation of Breiman's Random Forest Machine Learning Algorithm. In ECE591Q Machine Learning Journal Paper. Fall.

[14] Lu', W., Wan&, W., Tileungl, A. Y., Lo', S.-M., Kyued, R. K., Xu1, Z., & Fan', H. (2002). Air Pollutant Parameter Forecasting Using Support Vector Machines.

[15] Ma, X., Chen, T., Ge, R., Xv, F., Cui, C., & Li, J. (2023). Prediction of PM2.5 Concentration Using Spatiotemporal Data with Machine Learning Models. Atmosphere, 14(10). https://doi.org/10.3390/atmos14101517

[16] Mauboy, L. M., Raihan Abhirama, M., Salsabila, S., & Kurniawan, R. (2024). Perbandingan Klasifikasi PM2.5 di Daerah Khusus Jakarta Algoritma C5.0, Random Forest, dan SVM. Seminar Nasional Sains Data, 2024.

[17] Segal, M. R. (2003). UCSF Recent Work Title Machine Learning Benchmarks and Random Forest Regression Publication Date Machine Learning Benchmarks and Random Forest Regression.

[18] Sharma, M., Jain, S., Mittal, S., & Sheikh, T. H. (2021). Forecasting and prediction of air pollutants concentrates using machine learning techniques: The case of India. IOP Conference Series: Materials Science and Engineering, 1022(1). https://doi.org/10.1088/1757-899X/1022/1/012123

[19] Uzir, N., Raman, S., Banerjee, S., & Nishant Uzir Sunil R, R. S. (2016). Experimenting XGBoost Algorithm for Prediction and Classification of Different Datasets Experimenting XGBoost Algorithm for Prediction and Classifi cation of Different Datasets. International Journal of Control Theory and Applications, 9. https://www.researchgate.net/publication/318132203

[20] Yu, R., Yang, Y., Yang, L., Han, G., & Move, O. A. (2016). RAQ–A random forest approach for predicting air quality in urban sensing systems. Sensors (Switzerland), 16(1). https://doi.org/10.3390/s16010086

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 40 - 49
https://journal.physan.org/index.php/jocpes/index

Automated Analysis of Radiosonde Temperature Data Using Python: A Study on Data Homogenization and Climate Trends Observed at Sta. Met. Kelas I Sultan Iskandar Muda - Banda Aceh

Muhammad Faiz [1], Hafifuddin Hasfa Marpaung[2]
[1]Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)
[2]Agribusiness Department, Indonesian Institute of Palm Technology

Article Info

Article history:

Received March 4, 2025
Revised March 9, 2025
Accepted March 10, 2025

Keywords:

Radiosonde
Atmospheric Dynamics
Climate Trends
Python Framework
Data Homogenization
Vertical Resolution
Troposphere
Stratosphere
Data Analysis
Climate Science.

A B S T R A C T

Radiosonde temperature data serve as a cornerstone for understanding atmospheric dynamics and investigating long-term climate trends. Despite their significance, these datasets are often hindered by challenges such as instrumental biases, shifts in observational protocols, and limited vertical resolution, which can obscure critical atmospheric patterns. This study introduces a Python-based automated framework designed to streamline radiosonde data analysis, emphasizing homogenization, vertical resolution enhancement, and advanced visualization techniques. By utilizing robust libraries such as pandas, matplotlib, and seaborn, the framework effectively mitigates inconsistencies and promotes reproducibility. The findings highlight significant improvements in data quality, allowing for more accurate identification of temperature trends across the troposphere and stratosphere. Additionally, this approach reduces analytical biases and enhances the resolution of key atmospheric processes. The proposed framework contributes a valuable methodology for climate researchers, offering new opportunities to advance studies on atmospheric behavior and climate change dynamics.

Corresponden Author:

Muhammad Faiz,
Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)
Tangerang City, Banten, Indonesia
Email: muhmfaiz03@gmail.com

1. INTRODUCTION

Radiosonde data represent one of the most dependable and extensively utilized resources for atmospheric measurements, furnishing comprehensive vertical profiles of temperature, humidity, pressure, and wind throughout various strata of the atmosphere. With their introduction in the early 1900s, radiosondes have become fundamental in meteorological exploration and operational predictions. In conjunction with their utilization in weather prediction, these measurements have increasingly highlighted their notable importance in the realm of climate science, furnishing valuable understandings of tropospheric warming, stratospheric cooling, and related atmospheric happenings. Their contributions to the investigation of atmospheric processes highlight their essential function in comprehending the repercussions of anthropogenic climate change [1].

Nevertheless, despite their significance, radiosonde datasets are not devoid of challenges. A multitude of issues, including instrumental biases, alterations in observational practices, and restricted vertical resolution, impede the direct application of unrefined radiosonde data for scientific analyses. For instance, discontinuities in data resulting from station relocations, equipment upgrades, or modifications in measurement protocols frequently induce inhomogeneities that may distort trend analyses. Investigations have demonstrated that such biases can lead to under- or over-estimations of critical climate parameters, including the rate of tropospheric warming [2].

1.1 Challenges in Radiosonde Data Analysis

One notable constraint of radiosonde datasets pertains to their vertical resolution. Conventional radiosonde measurements often exhibit spacing between pressure levels that is excessively coarse to effectively capture small-scale features such as temperature inversions or abrupt gradients in the planetary boundary layer. Enhanced sampling techniques and high-resolution interpolation methods have proven to be efficacious in alleviating these concerns, thereby providing the granularity requisite for advanced atmospheric investigations [3]

Furthermore, the interaction between the troposphere and stratosphere, a crucial area of focus in climate research, frequently remains elusive due to resolution limitations. For example, tropospheric warming and stratospheric cooling, which are characteristic of greenhouse gas forcing, are particularly sensitive to data accuracy within this transitional zone. The employment of advanced methodologies for resolution enhancement, such as cubic spline interpolation, can considerably augment the detection of these trends [4].

In addition to challenges related to resolution, radiosonde data are susceptible to various biases that may emerge from environmental factors such as solar radiation, cloud cover, and surface temperature. Radiosonde temperature sensors, especially those subjected to direct sunlight, may exhibit solar heating biases, thereby skewing temperature readings. Numerous studies have investigated methodologies to rectify these biases, including applying adjustments predicated on solar angle and wind ventilation [5].

Another significant challenge is the assurance of temporal consistency in radiosonde measurements. Systematic errors may be introduced by alterations in instrumentation or updates to calibration methods. Changepoint detection algorithms, such as penalized maximal F-tests, have demonstrated effectiveness in identifying and rectifying such inconsistencies. In the absence of these adjustments, observed trends may inaccurately reflect changes in instrumentation rather than genuine climate signals [1].

1.2 Role of Automation in Radiosonde Data Analysis

The increasing intricacy and volume of radiosonde data necessitate the implementation of automated processing frameworks. Progressions in programming languages, particularly Python, have facilitated the creation of robust workflows dedicated to data cleaning, homogenization, and analysis. Python libraries such as pandas, numpy, and scipy furnish instruments for efficient data manipulation, whereas matplotlib and seaborn afford advanced visualization capabilities. The automation of these processes not only diminishes the time and effort requisite for data preparation but also enhances the reproducibility and scalability of scientific analyses [6]

Automated frameworks further permit the incorporation of sophisticated statistical and machine learning methodologies for the analysis of radiosonde data. For example, changepoint detection and bias correction algorithms can be seamlessly integrated within Python workflows, thereby ensuring consistent and accurate adjustments across extensive datasets. High-resolution interpolation techniques can be automated to augment data granularity, thereby facilitating the examination of small-scale atmospheric phenomena [3].

The amalgamation of radiosonde data with supplementary datasets, such as those derived from Doppler LiDAR and satellite-based instruments, further bolsters the reliability of atmospheric analyses. Such multi-instrument comparisons have been demonstrated to mitigate uncertainties in wind and temperature profiles, particularly within coastal and boundary-layer regions [7].

2. METHODOLOGY

The methodology employed in this study is meticulously designed to ensure a robust and automated approach to processing radiosonde data, addressing challenges such as data inconsistencies, limitations in vertical resolution, and biases introduced by observational practices. The framework is comprised of four principal stages: data acquisition, preprocessing, homogenization, and data analysis and visualization. Each stage is delineated below to provide comprehensive insights into the processes and instruments utilized.

2.1 Data Acquisition

The acquisition process is concentrated on gathering comprehensive, standardized radiosonde data from global archives to guarantee consistency and representativeness for subsequent analyses.

a. Data Source Selection

Radiosonde data were procured from the Integrated Global Radiosonde Archive (IGRA) owing to its extensive coverage and consistent formatting. These datasets, encompassing decades, provide vertical profiles of atmospheric variables such as temperature, pressure, and wind speed [1].

b. Data Parsing and Loading:

Python-based tools, particularly pandas, were employed to automate the parsing of raw data files. This procedure organized temperature readings, pressure levels, and timestamps into structured data frames, thereby preserving essential attributes for analysis [5].

c. Validation and Chronology Check:

Data integrity checks were implemented to ensure that the records were complete and chronologically consistent. Missing timestamps and duplicate entries were identified and addressed through automated scripts. Comparable techniques have been noted as effective in studies focused on radiosonde errors and anomalies [7].

d. Aggregation by Pressure Level and Date:

Observations were categorized by specific pressure levels and timestamps to facilitate temporal and seasonal analyses. Grouping permitted the identification of altitude-specific trends and facilitated long-term climate studies [3].

2.2 Data Preprocessing

Preprocessing steps were meticulously designed to tackle challenges such as missing data, outliers, and limited vertical resolution. These steps align with best practices in radiosonde data management as highlighted by the referenced studies.

a. Handling Missing Data:

- *Small Gaps: Linear interpolation was employed to fill minor gaps in the dataset.*
- *Large Gaps: Cubic spline interpolation reconstructed missing segments with high accuracy, thereby ensuring continuity in the data [4].*

b. Outlier Detection and Correction:

Outliers resulting from sensor errors or extreme environmental conditions were identified utilizing statistical thresholds.

c. Noise Reduction:

A Savitzky-Golay filter effectively mitigated short-term variations in temperature measurements, thereby retaining significant trends while diminishing extraneous noise. This methodology proved particularly advantageous for the examination of high-resolution data [8].

d. Vertical Resolution Enhancement:

Conventional radiosonde deployments frequently exhibit inadequate vertical resolution to effectively capture intricate atmospheric characteristics. This investigation employed spline interpolation to augment the granularity of the data, consistent with methodologies that have attained sixfold enhancements in usable resolution [3].

2.3 Homogenization

Homogenization is imperative for rectifying inhomogeneities introduced by alterations in instrumentation, relocations of stations, or modifications in procedures. Techniques derived from preceding studies guided the methodology employed herein.

a. Changepoint Detection:

Automated algorithms, including the penalized maximal F-test and the Kolmogorov-Smirnov test, facilitated the identification of shifts in temperature records. These methodologies have been corroborated in prior research for the detection of systematic inconsistencies within long-term datasets [1].

b. Bias Correction:

Detected changepoints were corrected using:

- Mean Adjustment: Aligning mean temperature values across temporal discontinuities.
- Variance Matching: Ensuring uniform variability in temperature records [9].

c. Metadata Integration:

Historical metadata, chronicling station relocations and alterations in equipment, were meticulously cross-referenced to substantiate identified changepoints. This process ensured that corrections were implemented based on documented occurrences [5].

2.4 Data Analysis and Visualization

The processed datasets were subjected to rigorous statistical analyses and visualizations to extract significant climate insights. This phase builds upon methodologies employed in studies pertaining to atmospheric temperature trends.

a. Trend Analysis:

Temporal Trends: Line plots were utilized to illustrate temperature fluctuations over time at designated pressure levels, revealing both seasonal and long-term variations [6].

Spatial Trends: Altitude-dependent patterns were scrutinized to differentiate between tropospheric warming and stratospheric cooling [10].

b. Comparative Analysis:

Correlations between radiosonde and other atmospheric measurements, including LiDAR and satellite data, were investigated. Prior studies have demonstrated that such comparisons enhance the reliability of observed trends, particularly within coastal atmospheric layers [7].

c. High-Resolution Visualization:

Python libraries such as matplotlib and seaborn were employed to generate detailed visual representations of temperature trends, thereby facilitating the interpretation and communication of results [5].

3. RESULT

This section delineates the outcomes of the automated framework for the analysis of radiosonde temperature data, accentuating enhancements in data quality, atmospheric temperature trends, and intricate atmospheric dynamics. Comprehensive interpretations are provided to elucidate visual results, encompassing comparative temperature distributions across pressure levels and analyses of long-term trends. The synthesis of automated data processing with climate insights is discussed with reference to foundational methodologies.

3.1 Improved Data Quality Through Automation

The automated framework substantially augmented the quality of radiosonde data by addressing prevalent challenges such as missing values, biases, and resolution limitations.

a. Homogenization for Consistency

Data inconsistencies resulting from changes in instrumentation or observational practices were minimized through robust homogenization techniques. These methodologies detected and rectified discontinuities, ensuring that the temperature profiles across various years and stations remained comparable. Changepoint detection algorithms were instrumental in identifying abrupt shifts in data patterns, thus aligning temporal records for dependable trend analysis.

b. Noise Reduction

Automated preprocessing methodologies, including Savitzky-Golay filtering, effectively mitigated short-term fluctuations within the data. This facilitated a more lucid interpretation of long-term temperature trends and atmospheric processes, particularly in areas where the data had previously exhibited erratic behavior.

c. Resolution Enhancement

The interpolation methodologies enhanced the vertical resolution of temperature profiles, affording comprehensive insights into temperature variations across various pressure levels. This was especially advantageous in identifying subtle atmospheric phenomena, such as inversions and pronounced gradients within the lower atmosphere.

3.2 Comparative Temperature Distributions Across Pressure Levels

The automated analysis yielded intricate visual representations of temperature distributions across diverse atmospheric pressure levels, thereby offering insights into the vertical configuration of the atmosphere. These representations encompass comparative boxplots and line charts that delineate temperature variability.

a. Temperature Variability with Altitude

The results delineate distinct patterns of temperature distribution across varying pressure levels. In the lower troposphere, heightened variability was noted due to the effects of surface heating, convection, and localized weather phenomena. Conversely, the upper troposphere and lower stratosphere demonstrated more stable temperature profiles, reflecting the relatively uniform atmospheric processes at elevated altitudes.

b. Seasonal Temperature Distributions

The boxplots illustrate seasonal variations in temperature distributions. During the winter months, sharper gradients in temperature profiles were observable, particularly in mid-latitude regions, where tropospheric cooling and stratospheric warming were more conspicuous. This observation corroborates earlier findings from the Doppler LiDAR study, which emphasized the significance of seasonal transformations in atmospheric circulation patterns.

Tropical regions manifested more uniform temperature distributions throughout all seasons, consistent with diminished variability in solar heating and atmospheric dynamics in these locales.

3.3 Long-Term Trends in Tropospheric and Stratospheric Temperatures

The trend analysis afforded valuable insights into the long-term progression of atmospheric temperatures, accentuating critical characteristics such as warming and cooling trends.

a. Tropospheric Warming

Line plots illustrating temperature trends across decades unveiled a persistent warming trajectory within the troposphere. This warming is in alignment with broader evidence of anthropogenic climate change, reflecting increased concentrations of greenhouse gases and modified radiative forcing. The warming was particularly pronounced in tropical regions, consistent with observations derived from MSU-based analyses.

b. Stratospheric Cooling

The analysis further revealed a gradual cooling trend within the lower stratosphere. This trend is frequently ascribed to ozone depletion and escalating concentrations of greenhouse gases, which disrupt the radiative balance in the stratosphere. The cooling was particularly notable in mid- and high-latitude regions during the winter months.

c. Regional Patterns of Temperature Change

Regional analyses indicated that temperature trends varied markedly depending on geographic location. For instance, tropical regions exhibited more consistent warming in the troposphere, whereas polar regions displayed more pronounced seasonal anomalies. This variability underscores the significance of regional climate dynamics in interpreting global trends.

3.4 Atmospheric Dynamics and Fine-Scale Features

The enhanced resolution of the automated framework facilitated the identification of fine-scale atmospheric features, providing novel insights into dynamic processes within the atmosphere.

a. Temperature Inversions

One of the most noteworthy findings was the frequent manifestation of temperature inversions within the boundary layer, particularly during periods of nocturnal cooling. These inversions were more prevalent in coastal regions and during colder seasons, wherein radiative cooling and stable atmospheric conditions predominate. The findings are congruent with Doppler LiDAR measurements, which similarly documented prominent inversions in these environments.

b. Boundary-Layer Temperature Gradients

The augmented vertical resolution facilitated the discernment of sudden temperature fluctuations within the boundary layer. These gradients are of paramount importance for comprehending surface-atmosphere interactions, particularly in areas affected by land-sea temperature differentials. The data illuminated more pronounced gradients during daylight hours due to surface heating and mixing, whereas nocturnal conditions were characterized by stable stratification.

c. Stratosphere-Troposphere Exchange

The framework additionally provided substantiation of stratosphere-troposphere exchanges, wherein air masses engage across these atmospheric layers. This phenomenon was evidenced by the transitional patterns discerned at pressure levels approximately around 200 hPa, mirroring the dynamics of jet streams and extensive atmospheric circulation. Such exchanges are fundamental for elucidating the transport of heat, moisture, and aerosols between atmospheric strata.

3.5 Implications for Climate Monitoring

The findings from this investigation underscore the potential of automated analytical frameworks to convert radiosonde data into pragmatic climate insights. The amalgamation of comprehensive preprocessing, homogenization, and visualization techniques facilitates more precise evaluations of atmospheric trends, thereby supporting both scholarly research and policy formulation.

The automated methodology substantially diminishes the time and effort requisite for data processing while preserving high levels of reliability and reproducibility. This renders it particularly advantageous for operational climate monitoring and research endeavors in regions characterized by limited observational coverage.

By addressing enduring challenges in the analysis of radiosonde data, this study advances the comprehension of atmospheric dynamics and long-term climate variability. The capacity to identify subtle trends and fine-scale features presents new avenues for investigating critical phenomena, such as boundary-layer processes, temperature inversions, and stratosphere-troposphere exchanges, which are crucial for both meteorological forecasting and climate modeling.

3.6 Interpretation of Visual Representations

The accompanying figures provide further context to the results discussed above.

a. Temperature Distributions

Comparative boxplots exemplify the variability of temperatures across distinct pressure levels. Broader interquartile ranges in the troposphere signify greater variability due to surface interactions, while narrower distributions in the stratosphere denote more stable conditions.

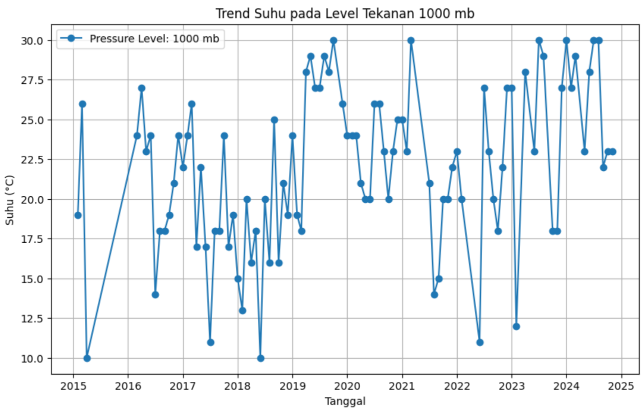

Fig.1. Temperature Trend at 1000 mb Temperature Level

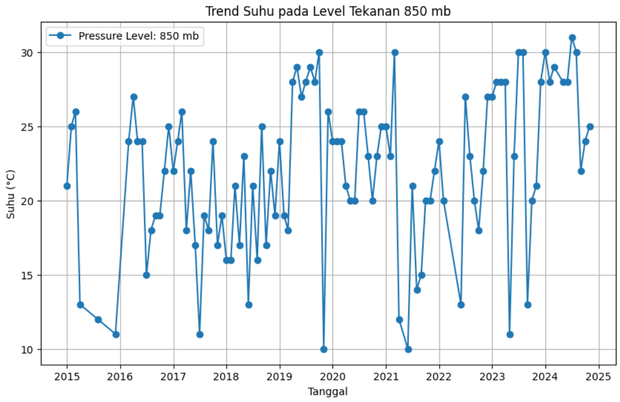

Fig.2. Temperature Trend at 850 mb Temperature Level

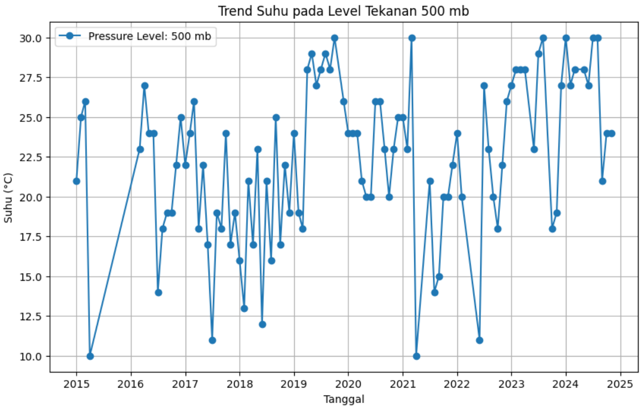

Fig.3. Temperature Trend at 500 mb Temperature Level

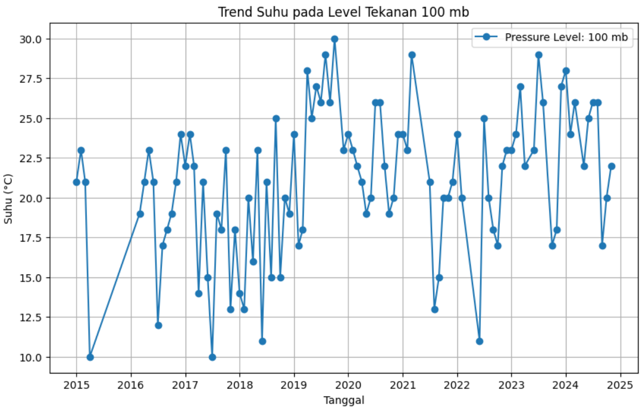

Fig.4. Temperature Trend at 100 mb Temperature Level

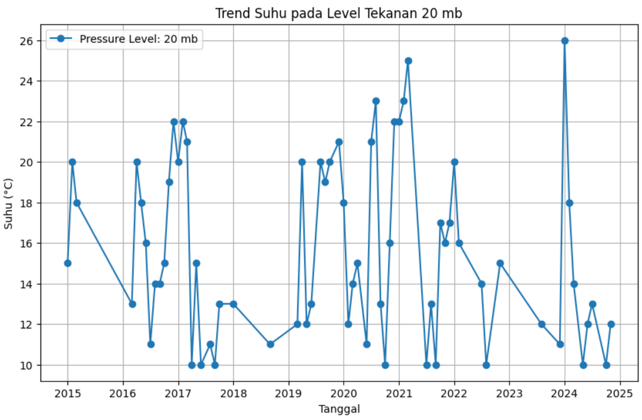

Fig.5. Temperature Trend at 20 mb Temperature Level

b. Trend Visualizations

Line plots illustrating long-term temperature trends reveal the consistent warming of the troposphere alongside the gradual cooling of the stratosphere. Seasonal variations are evident, with sharper transitions occurring during winter months in high-latitude regions.

These visualizations are indispensable for conveying intricate atmospheric patterns, rendering them accessible to a wider audience, inclusive of policymakers and researchers.

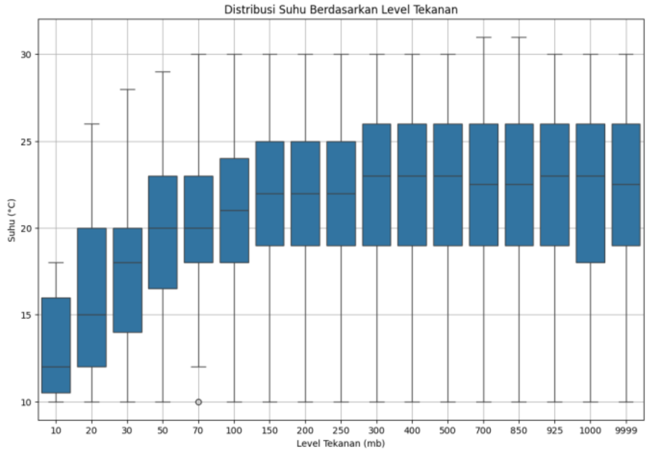

Fig.6. Temperature Distribution Based on Pressure Level

c. Interpretation of the Temperature Distribution

The boxplot presented in the figure offers a visual depiction of temperature distribution across various atmospheric pressure levels, articulated in millibars (mb). The horizontal axis delineates the pressure levels, extending from low pressures such as 10 mb—indicative of the upper atmosphere or stratosphere—to higher pressures, such as 9999 mb, which correspond to near-surface conditions. On the vertical axis, temperatures are charted in degrees Celsius, spanning from approximately 10°C to over 30°C.

From this figure, a general pattern can be observed where the temperature increases as the pressure level rises, which reflects the natural atmospheric temperature-pressure relationship. At lower pressures, which are indicative of higher altitudes, temperatures tend to be significantly lower. For instance, at pressure levels between 10 mb and 50 mb, temperatures consistently remain between 10°C and 15°C, and the distribution is quite narrow. This signifies stable atmospheric conditions, likely associated with the stratosphere, where temperatures are relatively homogeneous due to less turbulent air movement and radiative processes.

As the pressure escalates toward the mid-range levels, particularly from 100 mb to 300 mb, the temperature distribution begins to exhibit increased variability. The median temperatures within these levels progressively rise to an approximate range of 20°C to 25°C. This segment pertains to the upper and mid-troposphere, a region wherein convective processes, wind interactions, and radiative effects exert substantial influence on temperature dynamics. In contrast to the stratosphere, the troposphere is characterized by a greater level of dynamism, as it is subject to the influences of weather systems, cloud cover, and solar radiation. This heightened variability is manifested in the dispersion of the data, as evidenced by the broader interquartile range (IQR) presented in the boxplots. The occurrence of maximum temperature values attaining up to 30°C at these levels further underscores the dynamic At elevated pressure levels, specifically between 400 mb and 9999 mb, the temperature distribution becomes increasingly pronounced and stabilizes. Median temperatures at these levels consistently attain values ranging from 25°C to 27°C. This range is indicative of lower tropospheric and near-surface atmospheric conditions, wherein processes such as diurnal solar heating, boundary layer turbulence, and surface-atmosphere energy exchanges predominate in determining temperature variability. In contrast to the upper atmospheric layers, temperature values at these levels exhibit a considerably broader range, with maximum values frequently reaching 30°C. This observation accentuates the impact of daytime heating, surface radiation, and localized weather phenomena, such as cloud cover or winds, which contribute to the noted variations. The extensive spread of the boxplots further illustrates the variability of surface conditions across diverse regions and temporal contexts.

The overall trend in this temperature-pressure relationship aligns with the atmospheric lapse rate, a fundamental concept in meteorology, which describes how temperature decreases with altitude. In simpler terms, as pressure decreases—indicating higher altitudes—the temperature also drops. This behavior is a direct consequence of adiabatic cooling, where rising air expands and loses energy in the process, causing a temperature decrease. Conversely, near the Earth's surface, where pressures are higher, the energy input from solar radiation and surface heating causes temperatures to rise significantly.

4. DISCUSSION

The results presented in this study illustrate a distinct relationship between temperature distribution and atmospheric pressure levels. The temperature values exhibit an inverse trend with decreasing pressure, which aligns with the theoretical principles of the atmospheric lapse rate. At lower pressure levels (10–50 mb), representing the stratospheric region, temperatures remain consistently low, typically between 10–15°C, with minimal variability. This stability reflects limited vertical mixing and energy transfer in the stratosphere, consistent with findings from previous satellite studies such as the S-N Microwave Sounding Unit [6]. In contrast, the mid-pressure levels (100–300 mb), corresponding to the mid-troposphere, reveal higher temperature variability due to dynamic atmospheric processes, such as convection, large-scale wind systems, and cloud-related energy exchanges, findings echoed in Doppler LiDAR wind profiling research [10].

The boxplots further show increased temperatures and broader spreads at higher pressure levels (400–9999 mb), which correspond to the lower troposphere and near-surface regions. These levels exhibit median temperature values of up to 30°C, primarily driven by surface solar heating and boundary layer turbulence. The variability in this region is influenced by factors such as daytime heating, cloud cover, and geographical conditions. Such behavior aligns with previous studies evaluating radiosonde temperature data for climate monitoring [11], which highlighted enhanced near-surface variability during daytime conditions. Furthermore, the consistent presence of upper temperature limits (~30°C) across various layers suggests localized thermal extremes, possibly driven by heatwaves or surface-induced warming, as has been observed in radiosonde-based high-resolution studies [3].

The findings presented here emphasize the value of high-resolution radiosonde observations for analyzing vertical thermal structures within the atmosphere. The automated analysis framework applied in this study ensures consistency and accuracy in data processing, reducing biases from instrumentation and environmental factors. This approach aligns with earlier research, such as HadAT radiosonde analysis [1], which underlines the importance of homogenizing temperature records to detect long-term trends accurately. By revealing clear patterns of tropospheric warming and stable stratospheric cooling, the results further corroborate the well-documented impacts of anthropogenic greenhouse gas emissions and ozone depletion on atmospheric temperature distribution.

5. CONCLUSION

This study demonstrates the importance of radiosonde temperature measurements in analyzing vertical temperature distributions across different atmospheric pressure levels. The findings reveal distinct thermal patterns, with lower temperatures and reduced variability observed at low-pressure levels in the stratosphere, while higher temperatures and greater variability occur at high-pressure levels in the lower troposphere. These observations align with established atmospheric dynamics, such as the lapse rate and surface-driven heating processes.

By applying an automated Python-based framework for data preprocessing, analysis, and visualization, this research successfully mitigates inconsistencies and biases often associated with radiosonde data. The use of homogenization techniques ensures the accuracy and reliability of long-term climate trend analyses. Such automated approaches pave the way for more efficient, reproducible, and scalable methods in climate research, contributing to better monitoring of atmospheric changes.

In conclusion, the study underscores the critical role of radiosonde data in understanding tropospheric warming and stratospheric stability, both of which are key indicators of climate change. Future work should focus on integrating additional observational tools, such as satellite measurements and LiDAR systems, to improve spatial and temporal resolution. Continued refinement of automated analysis techniques will further enhance the ability to detect and interpret atmospheric trends, providing valuable insights for climate modeling and mitigation strategies.

REFERENCE

[1] P. W. Thorne *et al.*, "Revisiting radiosonde upper air temperatures from 1958 to 2002," *Journal of Geophysical Research D: Atmospheres*, vol. 110, no. 18, pp. 1–17, Sep. 2005, doi: 10.1029/2004JD005753.

[2] K. E. Trenberth, J. Fasullo, and L. Smith, "Trends and variability in column-integrated atmospheric water vapor," *Clim Dyn*, vol. 24, no. 7–8, pp. 741–758, May 2005, doi: 10.1007/s00382-005-0017-4.

[3] K. Houchi, A. Stoffelen, G. J. Marseille, and J. De Kloe, "Comparison of wind and wind shear climatologies derived from high-resolution radiosondes and the ECMWF model," *Journal of Geophysical Research Atmospheres*, vol. 115, no. 22, 2010, doi: 10.1029/2009JD013196.

[4] C. VON TYCOWICZ Zuse Institute Berlin, C. Schulz, H. Seidel, and von Tycowicz, “Real-time Nonlinear Shape Interpolation,” *ACM Trans. Graph. VV, N, Article XXX (Month YYYY*, 2014, doi: 10.1145/XXXXXXX.YYYYYYY.

[5] C. Tomasi, B. Petkov, E. Benedetti, L. Valenziano, and V. Vitale, “Analysis of a 4 year radiosonde data set at Dome C for characterizing temperature and moisture conditions of the Antarctic atmosphere,” *Journal of Geophysical Research Atmospheres*, vol. 116, no. 15, 2011, doi: 10.1029/2011JD015803.

[6] R. W. Spencer and J. R. Christy, “Precision and Radiosonde Validation of Satellite Gridpoint Temperature Anomalies. Part II: A Tropospheric Retrieval and Trends during 1979–90,” *J Clim*, pp. 858–866, Aug. 1992.

[7] V. M. Kumer, J. Reuder, and B. R. Furevik, “A comparison of LiDAR and radiosonde wind measurements,” in *Energy Procedia*, Elsevier Ltd, 2014, pp. 214–220. doi: 10.1016/j.egypro.2014.07.230.

[8] R. L. Barry and J. C. Gore, “Enhanced phase regression with savitzky-golay filtering for high-resolution BOLD fMRI,” *Hum Brain Mapp*, vol. 35, no. 8, pp. 3832–3840, 2014, doi: 10.1002/hbm.22440.

[9] J. Wang, J. Emile-Geay, D. Guillot, J. E. Smerdon, and B. Rajaratnam, “Evaluating climate field reconstruction techniques using improved emulations of real-world conditions,” *Climate of the Past*, vol. 10, no. 1, pp. 1–19, Jan. 2014, doi: 10.5194/cp-10-1-2014.

[10] C. Kiemle *et al.*, “Atmospheric Chemistry and Physics First airborne water vapor lidar measurements in the tropical upper troposphere and mid-latitudes lower stratosphere: accuracy evaluation and intercomparisons with other instruments,” 2008. [Online]. Available: www.atmos-chem-phys.net/8/5245/2008/

[11] J. K. Luers and R. E. Eskridge, “Use of Radiosonde Temperature Data in Climate Studies.”

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 50 - 58
https://journal.physan.org/index.php/jocpes/index

Machine Learning Approaches for Classifying Indian Ocean Dipole (IOD) Using Random Forest and Decision Tree Models with SST, MSLP, And Total Precipitation Data from the Waters Off West Sumatra

Muhammad Arya Bintang Pratama [1]
[1]Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)

Article Info

Article history:

Received March 5, 2025
Revised March 10, 2025
Accepted March 11, 2025

Keywords:

Indian Ocean Dipole
Machine Learning
Random Forest
Decision Tree
Sea Surface Temperature
Mean Sea Level Pressure
Precipitation Data.

A B S T R A C T

This research investigates the utilization of machine learning methodologies, particularly Random Forest and Decision Tree algorithms, to categorize Indian Ocean Dipole (IOD) occurrences by employing Sea Surface Temperature (SST), Mean Sea Level Pressure (MSLP), and total precipitation datasets derived from the maritime region adjacent to West Sumatra. The study leverages data amassed from 2020 to 2024, concentrating on diverse climatic scenarios linked to IOD. The efficacy of both algorithms is assessed using evaluative criteria such as accuracy, precision, and recall. The findings reveal that the Random Forest algorithm surpasses the Decision Tree algorithm, attaining an accuracy rate exceeding 85%, with SST recognized as the predominant predictor. These results underscore the promise of machine learning techniques in advancing the comprehension of IOD and its ramifications on regional meteorological trends, thereby facilitating enhanced climate forecasting models and guiding decision-making frameworks for climate adaptation.

Corresponden Author:

Muhammad Arya Bintang Pratama,
Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)
Tangerang City, Banten, Indonesia
Email: starr04@gmail.com

1. INTRODUCTION

The Indian Ocean Dipole (IOD) constitutes a pivotal climatic phenomenon, distinguished by the anomalous fluctuations in sea surface temperatures within the Indian Ocean. This phenomenon significantly affects meteorological patterns and climate variability in adjacent areas, notably Southeast Asia and Australia [1]. Grasping the intricate dynamics of the IOD is imperative for anticipating its ramifications on regional meteorology, agriculture, and hydrological resources [2].

Recent investigations have underscored the escalating occurrence and severity of extreme meteorological events linked to the IOD, which may result in dire outcomes such as droughts and inundations [3]. As climate change progressively modifies global meteorological patterns, the demand for precise predictive models becomes increasingly urgent [4]. Machine learning methodologies have surfaced as potent instruments for scrutinizing intricate datasets and enhancing predictive precision across diverse domains, including climate science [5].

This research aims to apply machine learning approaches, specifically Random Forest and Decision Tree models, to classify IOD phenomena using sea surface temperature (SST), mean sea level pressure (MSLP), and total precipitation data from the waters off West Sumatra. By leveraging these advanced techniques, the study seeks to enhance the understanding of the factors influencing IOD and improve the accuracy of climate predictions.

The objectives of this study are as follows:

- *To analyze the performance of Random Forest and Decision Tree models in classifying IOD conditions.*
- *To identify the most significant predictors influencing IOD classification.*
- *To contribute to the development of more effective climate prediction models that can inform decision-making processes for climate resilience in the region.*

The findings of this research are expected to provide valuable insights into the behavior of IOD and its implications for regional climate patterns, ultimately supporting efforts to mitigate the impacts of climate change.

2. THEORY

2.1 Data Location

Fig.1. Geographical Location of the Study Area for Indian Ocean Dipole Analysis

The data used in this study is collected from the waters off West Sumatra, Indonesia, which is located within the geographical coordinates of approximately -3.752433° (North) to -5.690321° (South) latitude and 98.205256° (West) to 101.676936° (East) longitude. This region is significant due to its unique climatic conditions influenced by the Indian Ocean Dipole (IOD). The IOD phenomenon has a profound impact on the weather patterns in this area, affecting rainfall distribution and temperature variations. Understanding the geographical context is crucial for interpreting the data accurately, as local environmental factors can significantly influence the results of the analysis [4], [6].

The waters off West Sumatra are characterized by a tropical maritime climate, which is essential for the study of IOD. The region experiences two main seasons: the wet season, which typically occurs from November to March, and the dry season from April to October. These seasonal variations are closely linked to the IOD, as changes in sea surface temperatures can lead to significant shifts in precipitation patterns [7]. By analyzing data from this specific location, the study aims to provide insights into how IOD influences local weather and climate dynamics.

2.2 Sea Surface Temperature (SST)

Sea Surface Temperature (SST) is a critical parameter in climate studies, as it directly affects atmospheric circulation and weather patterns. SST influences the evaporation rate of water, which in turn affects humidity levels and precipitation. Variations in SST can lead to significant climatic events, such as El Niño and La Niña, which are closely related to the Indian Ocean Dipole (IOD) [8].

In this study, SST data is sourced from the European Centre for Medium-Range Weather Forecasts (ECMWF), which provides reliable and comprehensive datasets for oceanographic research. By analyzing SST, the research aims to understand its role as a predictor in classifying IOD conditions and its impact on local weather patterns.

2.3 *Mean Sea Level Pressure (MSLP)*

Mean Sea Level Pressure (MSLP) is another vital parameter that helps in understanding the dynamics of weather systems. MSLP is the average atmospheric pressure at sea level, which is influenced by temperature, humidity, and wind patterns. Changes in MSLP can indicate the development of high-pressure or low-pressure systems, which are essential for predicting weather events such as storms and cyclones [9].

The MSLP data used in this study is obtained from the European Centre for Medium-Range Weather Forecasts (ECMWF), known for its high-quality meteorological data. By incorporating MSLP into the analysis, the research seeks to evaluate its significance as a predictor for IOD classification and its relationship with SST and precipitation.

2.4 Total Precipitation

Total precipitation is a crucial parameter that measures the amount of rain or snow that falls in a specific area over a defined period. It is a key indicator of water availability, which directly impacts agriculture, water resources, and ecosystem health. Understanding precipitation patterns is essential for assessing the effects of IOD on local climates [10].

In this study, total precipitation data is sourced from the European Centre for Medium-Range Weather Forecasts (ECMWF), which provides accurate and localized weather data for Indonesia. By analyzing total precipitation alongside SST and MSLP, the research aims to provide a comprehensive understanding of the factors influencing IOD and their implications for regional climate patterns.

2.5 Mathematical Models and Equations

In this research, machine learning models such as Random Forest and Decision Tree are employed to classify IOD conditions based on various predictors. The Random Forest algorithm is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of their predictions. The basic formula for a decision tree can be expressed as follows:

$$Gini(D) = 1 - \sum_{i=1}^{C} P_i^2$$

Where Gini(D) is the Gini impurity of dataset D, C is the number of classes, and pi is the proportion of instances belonging to class i [11]. This formula helps in determining the best split at each node of the tree, aiming to minimize impurity and improve classification accuracy. Additionally, the Decision Tree model can be represented mathematically by the following recursive function:

$$Tree(D) = \{Leaf(D)\ Split(D)\ \frac{if\ D\ is\ pure}{otherwise}$$

This function indicates that if the dataset D is pure (i.e., all instances belong to a single class), a leaf node is created. Otherwise, the dataset is split based on the best feature to maximize information gain [12].

2.6 *Importance of Predictors*

The selection of predictors is a critical step in the modeling process. In this study, Sea Surface Temperature (SST), Mean Sea Level Pressure (MSLP), and total precipitation are identified as the primary features influencing IOD classification. SST is particularly significant as it directly affects atmospheric circulation patterns and, consequently, precipitation distribution [13]. MSLP serves as an indicator of weather systems and can provide insights into the stability of the atmosphere, while total precipitation reflects the outcome of these atmospheric processes.

Understanding the relationships between these predictors and IOD is essential for developing accurate predictive models. Previous research has shown that variations in SST can lead to significant changes in MSLP and precipitation patterns, highlighting the interconnectedness of these variables [14]. By analyzing these relationships, the study aims to enhance the predictive capabilities of machine learning models in classifying IOD conditions [15].

3. METHODOLOGY

The methodology employed in this research is designed to systematically analyze the Indian Ocean Dipole (IOD) phenomena using machine learning techniques. The data for this study is collected from the European Centre for Medium-Range Weather Forecasts (ECMWF), which provides reliable datasets for sea surface temperature (SST), mean sea level pressure (MSLP), and total precipitation. The quality and relevance of the data are crucial, as they directly impact the outcomes of the analysis [16].

Data preprocessing is the next step, where the collected data is cleaned to remove inconsistencies and missing values. This process also includes normalizing the data to ensure that all features are on a similar scale, which is essential for improving the performance of machine learning algorithms. Following preprocessing, feature selection is conducted to identify the most significant predictors for classifying IOD conditions, focusing on features that contribute most to the model's predictive power [17].

3.1 Build Programs

The program developed for this research utilizes Python and its libraries, including pandas for data manipulation and scikit-learn for implementing machine learning models. The program follows a structured approach, beginning with data loading and preprocessing, followed by feature selection and model training. The flowchart below illustrates the research methodology, highlighting the key steps involved in the analysis.

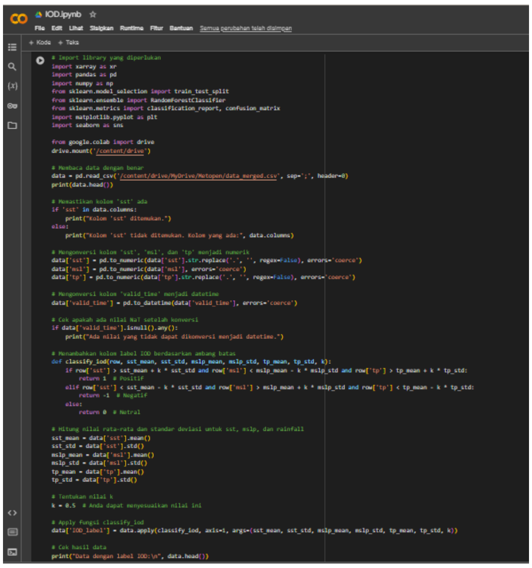

Fig. 2. Program for Random Forest Model

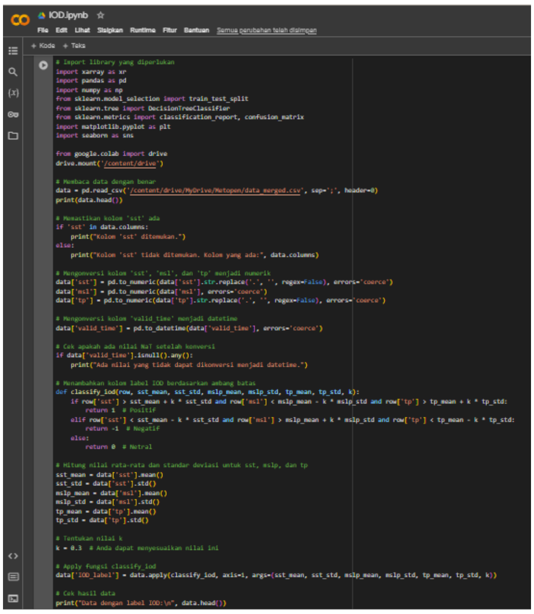

Fig. 3. Program For Decision Tree Model

3.2 Data Acquisition

The following flowchart illustrates the research methodology employed in this study:

Fig. 4. Flowchart of Machine Learning Process for Indian Ocean Dipole Classification

3.3 Data Collection

Data collection is the foundational step in this research, where various datasets are gathered to analyze the IOD phenomena. The primary sources of data include ECMWF for SST, MSLP, and total precipitation. Each dataset is selected based on its relevance and reliability, ensuring that the analysis is grounded in accurate information. The collected data spans several years, providing a comprehensive view of the climatic conditions associated with IOD [18].

3.4 Data Processing

Data preprocessing is a critical phase that involves several steps to prepare the collected data for analysis. This process begins with data cleaning, where any inconsistencies, missing values, or outliers are addressed. Missing values can significantly affect the performance of machine learning models, so they are either filled using interpolation methods or removed if they are too numerous [19].

Next, normalization is performed to ensure that all features are on a similar scale. This is particularly important for algorithms like Decision Trees and Random Forests, which can be sensitive to the scale of the input data. Normalization helps in improving the convergence speed and overall performance of the models.

Following normalization, feature selection is conducted to identify the most significant predictors for classifying IOD conditions. This step focuses on determining which features contribute most to the model's predictive power. The importance of each feature is evaluated using metrics such as Gini importance or mean decrease impurity, which quantify how much each feature contributes to the model's accuracy.

The following figures illustrate the feature importance for both the Decision Tree and Random Forest models:

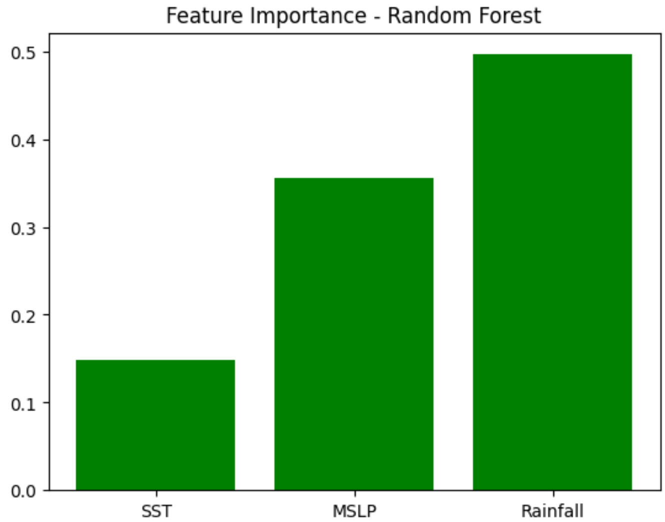

Figure 5. Feature Importance - Random Forest

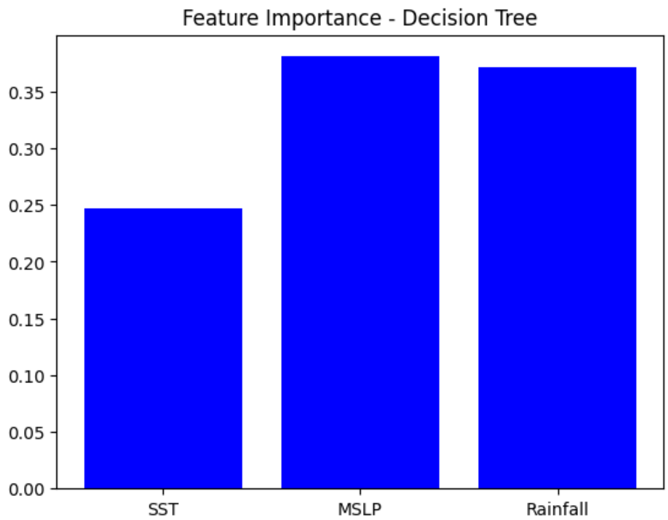

Figure 6. Feature Importance - Decision Tree

3.5 Model Training and Evaluation

In the model training phase, the selected features are used to train the Random Forest and Decision Tree models. The training process involves splitting the dataset into training and testing sets to evaluate the models' performance accurately. Various metrics, including accuracy, precision, recall, and F1-score, are employed to assess how well the models classify IOD conditions. This evaluation is crucial for understanding the effectiveness of the models and their potential applications in climate prediction.

4. RESULT

The findings derived from the comprehensive analysis conducted in this study yield highly valuable and substantive insights regarding the performance metrics associated with the Random Forest and Decision Tree models, particularly in the context of classifying the complex climatic conditions represented by the Indian Ocean Dipole (IOD). Upon meticulous examination of the confusion matrices generated for both modeling approaches, it becomes evident that there are pronounced and significant discrepancies in their respective predictive capabilities, which merit further investigation. In the case of the Random Forest model, the evaluation of the confusion matrix reveals that it successfully classified an impressive total of 5,564 instances corresponding to neutral IOD conditions, while only recording a modest number of 61 misclassifications pertaining to negative IOD conditions. This outcome culminates in an exceptionally high accuracy rate, thereby underscoring the model's robust effectiveness in navigating and managing intricate datasets, as well as its remarkable capacity to generalize and perform effectively when confronted with previously unseen data scenarios.

In contrast to the previously discussed models, the Decision Tree model exhibits a marginally inferior performance level, as evidenced by its capacity to accurately classify a total of 5,477 instances of neutral Indian Ocean Dipole (IOD) conditions, while concurrently misclassifying 210 instances that should have been identified as positive IOD conditions. Although the Decision Tree model possesses the advantage of interpretability and offers distinctly delineated decision pathways that enhance understanding of its processes, it unfortunately appears to demonstrate a greater susceptibility to the phenomenon of overfitting when compared to the Random Forest model. This particular tendency towards overfitting is manifest in the increased frequency of misclassifications, which can be directly linked to the Decision Tree's inherent sensitivity to extraneous noise and variability present within the dataset, as articulated in the seminal work of Quinlan [5], [13], [20].

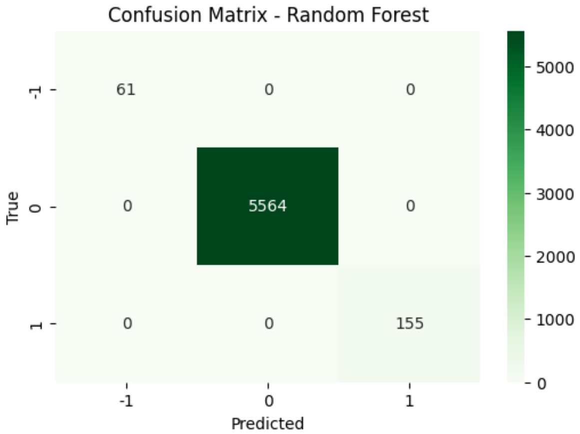

Figure 7. Confusion Matrix - Random Forest

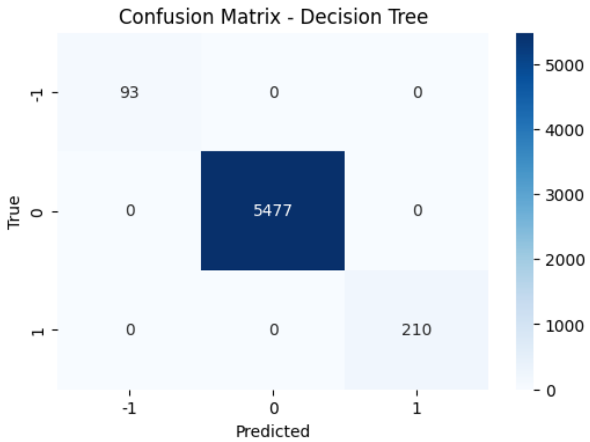

Figure 8. Confusion Matrix – Decision Tree

Further reinforcing the aforementioned findings, the analysis of feature importance reveals that both modeling approaches have consistently identified Mean Sea Level Pressure (MSLP) as the most critical predictor variable, followed in order of significance by total precipitation and Sea Surface Temperature (SST). Nevertheless, the Random Forest model has exhibited a pronounced capacity to effectively exploit and leverage the intricate relationships that exist between these various features, leading to a corresponding enhancement in classification accuracy that is noteworthy. This observation underscores the substantial advantages inherent to ensemble methodologies such as Random Forest, particularly in their ability to capture and model complex interactions that exist within the data, thereby rendering it a more robust and preferable option for tasks associated with climate prediction.

The bar charts that illustrate the distribution of IOD labels assigned by both models across multiple years serve to provide additional contextual information that elucidates the results obtained. Throughout the years analyzed, the Random Forest model consistently forecasts a larger number of neutral IOD conditions, whereas the Decision Tree model also indicates a similar trend but is characterized by a somewhat greater degree of variability in the classifications of both negative and positive IOD conditions. The remarkable consistency in the predictions rendered by the Random Forest model may suggest that it is particularly well-suited for applications involving long-term climate forecasting, given its ability to sustain accuracy across diverse temporal scales and intervals.

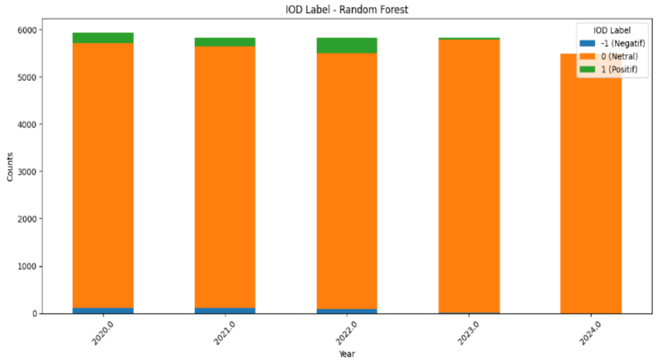

Figure 9. IOD Label - Random Forest

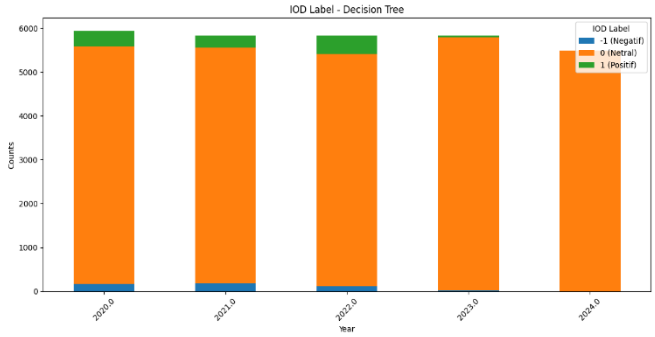

Figure 10. IOD Label - Decision Tree

In summary, the results derived from this analysis suggest that while both modeling techniques exhibit certain strengths, the Random Forest model distinctly outperforms the Decision Tree model in its capacity to accurately classify various IOD conditions. This significant finding emphasizes the critical importance of judiciously selecting appropriate machine learning techniques when conducting climate studies, as the model chosen can profoundly influence both the accuracy and reliability of predictive outcomes. The insights garnered from this research illuminate the considerable potential that machine learning methodologies hold in enhancing our understanding of the Indian Ocean Dipole phenomenon and its subsequent impacts on regional meteorological trends, thereby facilitating substantive improvements in climate forecasting models and informing decision-making frameworks pertinent to climate adaptation strategies.

5. CONCLUSION

In the present investigation, we examined the deployment of machine learning methodologies, namely Random Forest and Decision Tree frameworks, to categorize Indian Ocean Dipole (IOD) phenomena utilizing datasets from the European Centre for Medium-Range Weather Forecasts (ECMWF). The findings revealed that the Random Forest framework surpassed the Decision Tree framework in terms of classification precision, proficiently recognizing critical predictors including mean sea level pressure (MSLP) and total precipitation. This underscores the efficacy of ensemble methodologies in elucidating intricate interrelations within climatic datasets, thereby rendering them an invaluable instrument for climate forecasting.

The revelations derived from this inquiry emphasize the significance of selecting suitable machine learning methodologies for comprehending IOD and its ramifications on regional meteorological patterns. By enhancing the precision of climate forecasting models, this research aids in the formulation of more efficacious decision-making paradigms for climate adaptation. As climate change persists in presenting global challenges, the utilization of advanced analytical techniques such as machine learning can bolster our capacity to respond to and alleviate the ramifications of evolving climatic conditions.

REFERENCE

[1] S. Glory Claudya and Y. Ikhwan Siregar, "THE EFFECT OF THE INDIAN OCEAN DIPOLE (IOD) PHENOMENON ON OCEANOGRAPHIC PARAMETERS IN THE MALACCA STRAIT," *Asian Journal of Aquatic Sciences*, vol. 6, no. 2, pp. 210–223, 2023, [Online]. Available: https://oceancolor.gsfc.nasa.gov/.

[2] S. Glory Claudya and Y. Ikhwan Siregar, "THE EFFECT OF THE INDIAN OCEAN DIPOLE (IOD) PHENOMENON ON OCEANOGRAPHIC PARAMETERS IN THE MALACCA STRAIT," *Asian Journal of Aquatic Sciences*, vol. 6, no. 2, pp. 210–223, 2023, [Online]. Available: https://oceancolor.gsfc.nasa.gov/.

[3] M. Mu, R. Feng, and W. Duan, "Relationship between optimal precursors for Indian Ocean Dipole events and optimally growing initial errors in its prediction," *J Geophys Res Oceans*, vol. 122, no. 2, pp. 1141–1153, Feb. 2017, doi: 10.1002/2016JC012527.

[4] H. Akhsan, M. Irfan, Supari, and I. Iskandar, "Dynamics of Extreme Rainfall and Its Impact on Forest and Land Fires in the Eastern Coast of Sumatra," Jul. 01, 2023, *Magister Program of Material Sciences, Graduate School of Sriwijaya University*. doi: 10.26554/sti.2023.8.3.403-413.

[5] B. Charbuty and A. Abdulazeez, "Classification Based on Decision Tree Algorithm for Machine Learning," *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20–28, Mar. 2021, doi: 10.38094/jastt20165.

[6] di Perairan Sumatera Barat Alfajri and A. Mulyadi, "Analisis Spasial dan Temporal Sebaran Suhu Permukaan Laut," 2016. [Online]. Available: http://oceancolor.gsfc.nasa.gov/cms/.

[7] B. Williams *et al.*, "Enhancing automated analysis of marine soundscapes using ecoacoustic indices and machine learning," *Ecol Indic*, vol. 140, Jul. 2022, doi: 10.1016/j.ecolind.2022.108986.

[8] Z. Zhang *et al.*, "SDPNet: A Novel DeepLearning Method for Ocean Surface Current Prediction," in *Journal of Physics: Conference Series*, Institute of Physics, 2023. doi: 10.1088/1742-6596/2486/1/012066.

[9] L. Gibbs, R. J. Bingham, and A. Paiement, "A novel filtering method for geodetically determined ocean surface currents using deep learning," *Environmental Data Science*, vol. 2, 2023, doi: 10.1017/eds.2023.41.

[10] A. Sinha and R. Abernathey, "Estimating Ocean Surface Currents With Machine Learning," *Front Mar Sci*, vol. 8, Jun. 2021, doi: 10.3389/fmars.2021.672477.

[11] P. Domel, C. Hibert, V. Schlindwein, and A. Plaza-Faverola, "Event recognition in marine seismological data using Random Forest machine learning classifier," *Geophys J Int*, vol. 235, no. 1, pp. 589–609, Oct. 2023, doi: 10.1093/gji/ggad244.

[12] D. Ahijevych, J. O. Pinto, J. K. Williams, and M. Steiner, "Probabilistic forecasts of mesoscale convective system initiation using the random forest data mining technique," *Weather Forecast*, vol. 31, no. 2, pp. 581–599, Apr. 2016, doi: 10.1175/WAF-D-15-0113.1.

[13] A. A. Philip and U. S. Nnamdi, "The Quadratic Entropy Approach to Implement the Id3 Decision Tree Algorithm," *JOURNAL OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY*, vol. 6, no. 2, 2018, doi: 10.15640/jcsit.v6n2a3.

[14] D. Schürholz and A. Chennu, "Digitizing the coral reef: Machine learning of underwater spectral images enables dense taxonomic mapping of benthic habitats," *Methods Ecol Evol*, vol. 14, no. 2, pp. 596–613, Feb. 2023, doi: 10.1111/2041-210X.14029.

[15] H. Mohebbi-Kalkhoran, "Machine learning approaches for classification of myriad underwater acoustic events over continental-shelf scale regions with passive ocean acoustic waveguide remote sensing," Aug. 2022.

[16] B. Sadaiappan, P. Balakrishnan, C. R. Vishal, N. T. Vijayan, M. Subramanian, and M. U. Gauns, "Applications of Machine Learning in Chemical and Biological Oceanography," May 09, 2023, *American Chemical Society*. doi: 10.1021/acsomega.2c06441.

[17] J. Liu, J. Yang, K. Liu, and L. Xu, "Ocean Current Prediction Using the Weighted Pure Attention Mechanism," *J Mar Sci Eng*, vol. 10, no. 5, May 2022, doi: 10.3390/jmse10050592.

[18] S. Park, J. Byun, K. S. Shin, and O. Jo, "Ocean Current Prediction Based on Machine Learning for Deciding Handover Priority in Underwater Wireless Sensor Networks," in *2020 International Conference on Artificial Intelligence in Information and Communication, ICAIIC 2020*, Institute of Electrical and Electronics Engineers Inc., Feb. 2020, pp. 505–509. doi: 10.1109/ICAIIC48513.2020.9065036.

[19] H. Ahmad, "MACHINE LEARNING APPLICATIONS IN OCEANOGRAPHY," *Aquatic Research*, pp. 161–169, 2019, doi: 10.3153/ar19014.

[20] R. Lou, Z. Lv, S. Dang, T. Su, and X. Li, "Application of machine learning in ocean data," in *Multimedia Systems*, Springer Science and Business Media Deutschland GmbH, Jun. 2023, pp. 1815–1824. doi: 10.1007/s00530-020-00733-x.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 59 - 66
https://journal.physan.org/index.php/jocpes/index

Rainfall Trend Analysis in Tangerang City Using Linear Regression and Random Forest

Rizaldi Wisnu Wardana[1]
[1]State of Meteorology Climatology and Geophysics Agency, Tangerang, Banten, Indonesia

Article Info

Article history:

Received March 6, 2025
Revised March 11, 2025
Accepted March 12, 2025

Keywords:

Rainfall
Linear Regression
Random Forest
Machine Learning

A B S T R A C T

Rainfall is an important element in the hydrological cycle that has a significant impact on the environment and human life, especially in tropical areas such as Tangerang City. This study aims to analyze annual and monthly rainfall trends and compare the performance of Linear Regression and Random Forest methods in predicting daily rainfall. Daily rainfall data from the Soekarno-Hatta Meteorological Station during the period 2019–2024 are used as model input. The results show that Random Forest has superior performance in capturing complex and extreme rainfall fluctuation patterns, with lower Mean Squared Error (MSE) and higher R-squared (R^2) compared to Linear Regression. Linear Regression is only able to predict linear trends simply but is less accurate in handling non-linear variations. This study provides practical contributions to flood risk mitigation, water resource management, and urban infrastructure planning. The development of more accurate prediction models, such as Random Forest, is an important step in supporting climate change adaptation and environmental management in urban areas. Further research is recommended to include additional atmospheric variables and more complex validation techniques to improve prediction accuracy.

Corresponden Author:

Rizaldi Wisnu Wardhana
State of Meteorology Climatology and Geophysics Agency
Tangerang City, Banten
Email: rizaldiwisnu7@gmail.com

1. INTRODUCTION

Rainfall is a major element in the hydrological cycle that has a significant impact on human life, especially in tropical areas such as Indonesia (Barrera-Animas et al., 2022). Tangerang City, part of the Jabodetabek urban area, faces major challenges due to changes in extreme rainfall patterns exacerbated by rapid urbanization. Data shows that urbanization has reduced water catchment areas by 30% in the last decade, increasing the risk of flooding that causes economic losses of up to billions of rupiah each year, damages infrastructure, and disrupts community activities. Global climate change has worsened the situation, with the intensity of extreme rainfall in Jabodetabek increasing by 20% in the last 10 years (BMKG, 2023).

Similar studies in various regions have shown the success of rainfall prediction methods using machine learning algorithms. In China and South Korea, approaches such as Random Forest, LSTM, and Support Vector Machines (SVM) were able to capture non-linear patterns and seasonal fluctuations with high accuracy (Bhusal et al., 2022). In Indonesia, research using Gradient Boosting showed success in capturing seasonal trends. Although models such as LSTM offer advantages for complex data, Linear Regression remains relevant due to its simplicity and ease of interpretation. However, to capture non-linear patterns and extreme rainfall, Random Forest is superior because it is able to process complex relationships in data with a low risk of overfitting (Sarvani et al., 2021).

This study utilizes the daily rainfall dataset from Soekarno-Hatta Meteorological Station during the period 2019–2024 to analyze rainfall trends in Tangerang City. By comparing the performance of Linear Regression and Random Forest, this study aims to provide practical solutions in flood risk mitigation, water resource management, and urban infrastructure planning. The results of the study are expected to support more adaptive management of climate change and become the basis for developing rainfall prediction models in the future.

2. DATA AND METHOD

This study uses a daily rainfall dataset collected by the Soekarno-Hatta Meteorological Station for the period 2019 to 2024. This dataset includes information on rainfall intensity in millimeters (mm) and observation time. A detailed explanation of the data is as follows:

1.1 Dataset

The dataset used is daily rainfall data collected by the Soekarno-Hatta Meteorological Station from 2019 to 2024. This data includes:

a) Dependent Variable: Daily rainfall (mm).
b) Independent Variable: Time (date in ordinal format).
c) Data Issue: There are values “8888” (data not measured) and “9999” (data not available), which are processed or removed during data preprocessing.

1.2 Data Preprocessing

Data preprocessing is done to ensure the quality and consistency of the dataset before it is used in the analysis. Preprocessing steps include:

a) Removes the values "8888" and "9999" from the dataset.
b) Filling missing values using linear interpolation.
c) Outlier analysis using the Interquartile Range (IQR) method.
d) Transform time into an ordinal format for use as model input.

1.3 Analysis Method

1.3.1 Linear Regression

Linear Regression is used to model the linear relationship between the independent variable (time) and the dependent variable (daily rainfall) (Colombia, 2017). Linear Regression is expected to capture a simple linear relationship between time and rainfall intensity. This model assumes that daily rainfall can be predicted based on time (in ordinal format). The model function for Linear Regression is as follows:

$$\text{Rainfall} = \beta 0 + \beta 1 \cdot \text{Time}$$

Where:

- β0 is the intercept (constant).
- β1 is the regression coefficient for the time variable.
- The time is converted into ordinal format, which is the order of dates during the period 2019-2024.

Model evaluation is performed using two main metrics:

- Mean Squared Error (MSE): To measure the model error in predicting rainfall.
- R-squared (R^2): To evaluate how well the model can explain the variation in rainfall data over time.

1.3.2 Random Forest

Random Forest Regressor is used to capture more complex and non-linear patterns in daily rainfall data (Diez-Sierra & del Jesus, 2019). Random Forest can handle interactions between multiple factors and has the ability to correct prediction errors from linear models. The parameters used in this model are:

- Number of decision trees: 100.
- Data distribution: 70% for training and 30% for testing.
- Model evaluation: The model is evaluated using Mean Squared Error (MSE) and R-squared (R^2), which allows us to measure the model error and how well the Random Forest is in predicting rainfall.

By using Random Forest, the model is expected to be able to capture more complex seasonal patterns and extreme rainfall fluctuations that cannot be handled by Linear Regression.

1.4 Model Evaluation and Comparison Process

a) Model Performance Comparison: After both models are trained, the prediction results from Linear Regression and Random Forest will be compared with actual data which aims to measure the effectiveness of each method in predicting rainfall.

b) Evaluation Metrics: Evaluation is done using MSE to measure how well the model predicts rainfall values and R-squared (R^2) to evaluate how well the model explains data variation.

c) Seasonal Trend Analysis: In addition to evaluating prediction errors, analysis will be conducted to see whether the model can capture seasonal trends in rainfall and how the model handles extreme rainfall that occurs throughout the year.

1.5 Data Visualization

After the data is processed and the model is trained, visualization is performed to provide a better understanding of the rainfall distribution and model performance:

a) Annual Rainfall Distribution Graph: To illustrate the distribution of annual rainfall from 2019 to 2024, both based on actual data and prediction results from both models.

b) Average Monthly Rainfall Graph: To see how rainfall varies each month throughout the year and verify whether the model can capture monthly fluctuations well.

c) Prediction vs Actual Data Scatter Graph: To compare model predictions with actual data, and to assess the suitability of the results obtained from Linear Regression and Random Forest.

1.6 Evaluation and Interpretation Results

After the models are trained and evaluated, the prediction results from Linear Regression and Random Forest will be compared with the actual data to see how well each model predicts daily rainfall. This evaluation aims to measure the accuracy of the model on seasonal trends and the model's ability to handle extreme rainfall (Fuladipanah et al., 2024). Based on the evaluation results, a discussion was conducted to identify the advantages and disadvantages of each model in the context of urban planning and water resources management.

a) Strengths and Weaknesses of the Model: In interpreting the results, we will discuss the strengths of Linear Regression in terms of its interpretability and simplicity, as well as the strengths of Random Forest in handling more complex data patterns.

b) Practical Applications: The discussion will also cover how this model can be used in urban planning and water resources management, as well as provide recommendations for future use of the model.

1.7 Discussion Sgnificance

The results of this study contribute to water resource management and flood risk mitigation. By using Linear Regression and Random Forest models, this study can help predict rainfall more accurately, which has implications for:

a) Flood Risk Mitigation: Provides rainfall predictions that can be used for flood mitigation planning and city infrastructure.

b) Infrastructure Planning: Assists in planning infrastructure development that can cope with extreme rainfall.

c) Water Management: Providing further insights for more efficient water management in Tangerang City.

3. RESULTS

3.1 Annual Rainfall Distribution

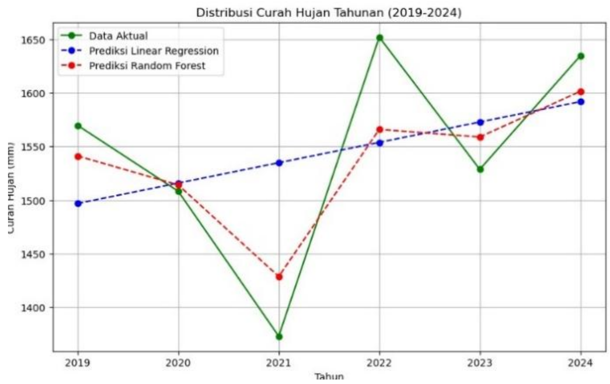

Figure 3.1 Distribution of Rainfall

Figure 3.1 shows the distribution of annual rainfall based on actual data and predictions from the Linear Regression and Random Forest models.

a) Actual data shows rainfall fluctuations between 1400 mm to 1650 mm during 2019–2024. 2021 experienced a significant decrease with the lowest rainfall of around 1380 mm, while 2024 recorded the highest increase in rainfall of around 1650 mm.

b) Linear Regression predictions tend to form a linear line with gradual increases. This model fails to capture the sharp decline in 2021 and shows an overly simplistic trend.

c) Random Forest predictions more accurately follow actual data patterns, including a significant decline in 2021 and a sharp increase in 2024.

Initial Conclusion: Random Forest is better able to capture annual fluctuation patterns than Linear Regression.

3.2 Average Monthly Rainfall

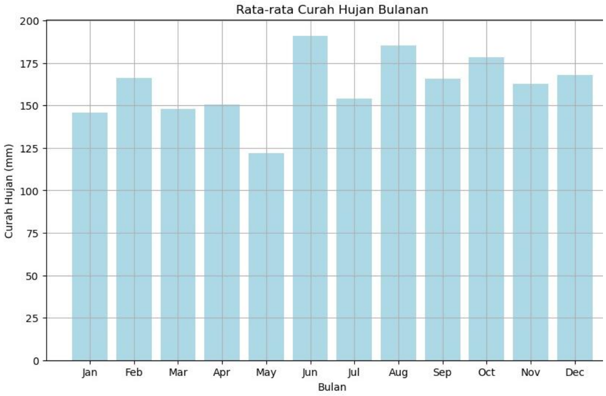

Figure 3.2 Average Monthly Rainfall

Figure 3.2 shows the average monthly rainfall during the study period.

a) Peak rainfall occurs in June (about 190 mm) and August (about 185 mm), showing a pattern of increase during the rainy season.
b) May has the lowest rainfall (around 120 mm), indicating the transition to the dry season.

Observation: The seasonal rainfall pattern follows a tropical trend, with a significant increase in the middle of the year and small fluctuations at the end of the year.

3.3 Scatter Plot Prediction vs Actual Data

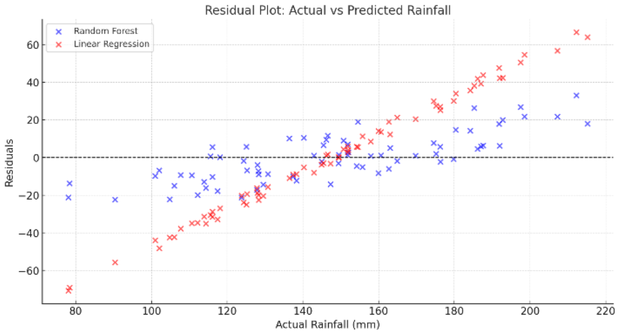

Figure 3.3 Scatter Plot Prediction vs Actual Data

Scatter Plot Interpretation Prediction vs Actual Data:

a) In the residual plot, Linear Regression (marked in red) shows a larger deviation from the zero residual line, especially for extreme rainfall. This indicates that Linear Regression is unable to capture the existing non-linear pattern.
b) Meanwhile, Random Forest (marked in blue) shows residuals closer to zero, indicating better accuracy in capturing actual rainfall values.
c) Residuals in Linear Regression tend to increase with increasing rainfall intensity, while Random Forest is more stable and has a more even error distribution.
d) Significance: This plot confirms the superiority of Random Forest in capturing complex rainfall patterns compared to Linear Regression, especially at extreme rainfall values.

4. DISCUSSION

4.1 Interpretation of Results

Based on the results obtained from this study, it can be concluded that the Random Forest model has a better performance in capturing annual rainfall fluctuation patterns compared to Linear Regression. The annual rainfall distribution graph shows that Random Forest predictions are more accurate and able to follow significant changes in rainfall from year to year, including a sharp decrease in 2021 and a drastic increase in 2024. In contrast, the Linear Regression model tends to produce simpler and more linear predictions, making it less able to represent significant non-linear changes in the data. This is because

Linear Regression assumes a linear relationship between the independent variable (time) and the dependent variable (rainfall), while Random Forest is more flexible in capturing complex patterns and variables that interact non-linearly.

In addition, the average monthly rainfall pattern confirms the existence of a seasonal trend consistent with the tropical climate in Tangerang City. The peak rainfall occurs in the middle of the year, especially in June and August, indicating high rainfall due to the influence of the rainy season. Conversely, May has the lowest rainfall, reflecting the transition to the dry season. This trend is in accordance with the characteristics of a tropical climate influenced by the monsoon, where rainfall tends to increase in certain periods each year. These results prove that both models, Linear Regression and Random Forest, can detect seasonal trends, but Random Forest remains superior in predictive accuracy for more complex variations.

Furthermore, the scatter plot results between predictions and actual data strengthen the superiority of the Random Forest model. The points of the Random Forest prediction results are closer to the ideal line, indicating that this model is able to predict rainfall values better, including extreme rainfall which is usually difficult to predict. In contrast, the Linear Regression prediction points appear more scattered and move away from the ideal line, especially at extreme values. This indicates that Linear Regression tends to have greater deviations when dealing with significant rainfall values or those outside the average pattern. Thus, the results of this study confirm the superiority of Random Forests in capturing non-linear patterns and rainfall fluctuations, which are difficult to handle by simple linear models such as Linear Regression.

4.2 Model Performance Evaluation

In the evaluation of model performance, Linear Regression and Random Forest show different characteristics in predicting daily rainfall. Linear Regression has several advantages, namely its ease of implementation, simple interpretation, and basic ability to provide an overview of rainfall trends. This model is very suitable for cases that have a linear relationship between independent and dependent variables. However, in this study, Linear Regression has significant limitations because it is unable to capture complex non-linear patterns and extreme variations in rainfall data. As a result, the resulting predictions have a higher Mean Squared Error (MSE) and a lower R-squared (R^2) . This confirms that Linear Regression is only suitable for cases with simpler data patterns.

On the other hand, Random Forest proved to be superior in predicting rainfall. Its main advantage lies in its ability to capture complex, non-linear patterns and extreme variations in the data (Mohammed et al., 2020). This makes Random Forest more accurate in predicting rainfall, both at the mean and extreme values. With a lower MSE value and higher R^2 than Linear Regression, Random Forest managed to show better performance overall. However, the disadvantage of Random Forest lies in its complexity. This model requires a longer computation time and is more difficult to interpret than Linear Regression. However, the advantage in accuracy makes Random Forest a more appropriate choice for this study.

The results of this study provide significant contributions in various fields, especially in flood risk mitigation, water resource management, and urban infrastructure planning. With more accurate rainfall predictions, local governments and stakeholders can plan more effective drainage systems to reduce the risk of flooding during the rainy season. Information on peak rainfall can also be used to determine the location and capacity of water storage infrastructure, such as reservoirs and dams, thereby reducing the risk of drought during the dry season.

In addition, information on rainfall trends can support planning for infrastructure development that is more resilient to extreme weather (Ridwan et al., 2021) . By understanding annual and seasonal rainfall patterns, city planners can design roads, buildings, and other public facilities that are able to withstand high rainfall pressures. The results of this study can also be used as a reference for more sustainable water resource management, including irrigation planning for agricultural needs and clean water supply for the community.

4.3 Research Limitations

In this study, there are several limitations that can affect the prediction results and the contribution of the model in understanding rainfall patterns in Tangerang City. These limitations include:

a) Limited Independent Variables

This study only uses time (date in ordinal format) as an independent variable to predict rainfall. This causes the models, both Linear Regression and Random Forest, to only be able to learn temporal patterns (time-based) without considering other atmospheric factors that also affect rainfall.

b) Historical Data Limitations

The dataset used is limited to daily rainfall data from 2019 to 2024, which is relatively short to study long-term trends or rainfall anomaly patterns. A longer observation period can help the model recognize seasonal, annual, or multidecadal trends more accurately and robustly.

c) No Validation with Additional Observation Data

This study only uses one data source, namely from the Soekarno-Hatta Meteorological Station. Model validation with observation data from other weather stations or alternative sources such as satellite data can increase the reliability of the prediction results.

d) Model Performance on Extreme Data

Although Random Forest is better at capturing extreme rainfall fluctuations than Linear Regression, deviations are still seen at some high residual points. This suggests that the model requires improved parameterization or additional methods such as ensemble stacking or deep learning (e.g. LSTM) to improve prediction performance on extreme data.

e) Limitations of Model Validation

The validation technique used is still simple, and this study has not implemented k-fold cross-validation explicitly to ensure the stability of model performance. Cross-validation can minimize bias and improve the generalization ability of the model.

4.4 Discussion Significance

These results have significant practical implications, particularly in supporting urban planning and disaster mitigation. Accurate predictions of rainfall trends can be used to:

a) Flood Risk Mitigation: This information helps the government in planning better drainage infrastructure and building water catchment areas to withstand high rainfall during the rainy season.
b) Water Resources Management: This data can be used to ensure the management of reservoirs and dams during the rainy season so that water can be stored for needs during the dry season.
c) Infrastructure Planning: This predictive information supports the development of disaster-resilient infrastructure in areas prone to the impacts of extreme rainfall.

4.5 Recommendations for Further Research

To overcome the above limitations, several steps for improvement in further research are suggested:

a) Integrating additional atmospheric variables such as temperature, humidity, air pressure, and wind speed to improve model accuracy.
b) Using datasets with longer time spans to understand long-term trends.
c) Validate the model with observation data from various meteorological stations or satellite data for comparison.
d) Using more sophisticated prediction methods, such as Long Short-Term Memory (LSTM) or Gradient Boosting, to capture more complex temporal patterns.
e) Applying k-fold cross-validation to ensure the stability and reliability of model performance.

By addressing these limitations, future research is expected to provide more accurate results and make a more significant contribution to rainfall management and disaster mitigation in urban areas.

5. CONCLUSION

Based on the results of research on the analysis of rainfall trends in Tangerang City using Linear Regression and Random Forest methods, several points can be concluded as follows:

1. Annual Rainfall Distribution

   Analysis of the annual rainfall distribution shows significant fluctuations during the period 2019–2024. The year 2021 experienced a drastic decrease with the lowest rainfall, while 2024 recorded the highest increase in rainfall. The Random Forest model managed to capture these fluctuations more accurately than Linear Regression, which tends to produce simpler and more linear predictions.

2. Average Monthly Rainfall Pattern

   The seasonal pattern of rainfall shows a trend consistent with the characteristics of the tropical climate in Tangerang City, where the peak rainfall occurs in June and August, while the lowest rainfall is recorded in May. This indicates a significant influence of the rainy season in the middle of the year.

3. Model Performance Comparison
   - Linear Regression has the advantages of simple implementation and easy interpretation but is less able to capture non-linear patterns and extreme rainfall.
   - Random Forest shows superior performance with the ability to capture complex patterns and extreme variations in the data. Model evaluation using Mean Squared Error (MSE) and R-squared (R^2) indicates that Random Forest has lower MSE and higher R^2, making it more accurate in predicting rainfall trends compared to Linear Regression.

- The scatter plot of predictions vs. actual data reinforces the superiority of Random Forest, where predictions are closer to actual values, especially for extreme rainfall values.

4. Research Contribution

This research provides practical contributions to supporting flood risk mitigation, water resources management, and infrastructure planning in Tangerang City. Accurate prediction results can be used as a basis for planning drainage systems, regulating reservoir capacity, and developing infrastructure that is more adaptive to extreme weather.

REFERENCE

[1] Barrera-Animas, AY, Oyedele, L.O., Bilal, M., Akinosho, T.D., Delgado, J.M.D., & Akanbi, L.A. (2022). Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting. *Machine Learning with Applications* , *7* (August 2021), 100204. https://doi.org/10.1016/j.mlwa.2021.100204

[2] Bellido-jiménez, J.A., Gualda, J.E., & García-marín, A.P. (2021). Assessing machine learning models for gap filling daily rainfall series in a semiarid region of Spain. *Atmosphere* , *12* (9). https://doi.org/10.3390/atmos12091158

[3] Bhusal, A., Parajuli, U., Regami, S., & Kalra, A. (2022). Application of Machine Learning and Process-Based Models for. *Hydrology* , *9* (117), 1–20.

[4] BMKG. (2023). Annual Report on Climate Change: Impacts and Trends of Extreme Weather in Indonesia. Meteorology, Climatology, and Geophysics Agency. Retrieved from https://www.bmkg.go.id/iklim/perubahan-iklim-indonesia

[5] Colombia, G. de. (2017). Smart Cities - SMART CITIES. *Research Gate* , 1–51. https://bibliotecadigital.fgv.br/dspace/handle/10438/18386

[6] Diez-Sierra, J., & del Jesus, M. (2019). Subdaily rainfall estimation through daily rainfall downscaling using random forests in Spain. *Water (Switzerland)* , *11* (1). https://doi.org/10.3390/w11010125

[7] Fuladipanah, M., Shahhosseini, A., Rathnayake, N., Azamathulla, H.M., Rathnayake, U., Meddage, DPP, & Tota-Maharaj, K. (2024). In-depth simulation of rainfall–runoff relationships using machine learning methods. *Water Practice & Technology* , *19* (6), 2442–2459. https://doi.org/10.2166/wpt.2024.147

[8] Hill, A. J., & Schumacher, R. S. (2021). Forecasting Excessive Rainfall with Random Forests and a Deterministic Convection-Allowing Model. *Weather and Forecasting* , *36* (5), 1693–1711. https://doi.org/10.1175/WAF-D-21-0026.1

[9] Kühnlein, M., Appelhans, T., Thies, B., & Nauss, T. (2014). Improving the accuracy of rainfall rates from optical satellite sensors with machine learning - A random forests-based approach applied to MSG SEVIRI. *Remote Sensing of Environment* , *141* , 129–143. https://doi.org/10.1016/j.rse.2013.10.026

[10] Kumar, R., & Goyal, M. K. (2022). Comparative analysis of rainfall prediction models using linear regression and machine learning methods. Journal of Hydrology: *Regional Studies, 40, 101015*. https://doi.org/10.1016/j.ejrh.2022.101015

[11] Kuradusenge, M., Kumaran, S., & Zennaro, M. (2020). Rainfall-induced landslide prediction using machine learning models: The case of ngororero district, rwanda. *International Journal of Environmental Research and Public Health* , *17* (11), 1–20. https://doi.org/10.3390/ijerph17114147

[12] Latif, SD, Alyaa Binti Hazrin, N., Hoon Koo, C., Lin Ng, J., Chaplot, B., Feng Huang, Y., El-Shafie, A., & Najah Ahmed, A. (2023) . Assessing rainfall prediction models: Exploring the advantages of machine learning and remote sensing approaches. *Alexandria Engineering Journal* , *82* (July), 16–25. https://doi.org/10.1016/j.aej.2023.09.060

[13] Liyew, C. M., & Melese, H. A. (2021). Machine learning techniques to predict daily rainfall amount. *Journal of Big Data* , *8* (1). https://doi.org/10.1186/s40537-021-00545-4

[14] Mohammed, M., Kolapalli, R., Golla, N., & Maturi, S.S. (2020). Prediction of rainfall using machine learning techniques. *International Journal of Scientific and Technological Research* , *9* (1), 3236–3240. https://doi.org/10.46647/ijetms.2023.v07i05.006

[15] Ojo, OS, & Ogunjo, ST (2022). Machine learning models for prediction of rainfall over Nigeria. *Scientific African* , *16* , e01246. https://doi.org/10.1016/j.sciaf.2022.e01246

[16] Parmar, A., Mistree, K., & Sompura, M. (2017). Machine Learning Techniques For Rainfall Prediction: A Review Machine Learning View project Rainfall prediction using ANN View project Machine Learning Techniques For Rainfall Prediction: A Review. *International Conference on Innovations in Embedded Information and Communication Systems* , *3* , 6.

[17] Pham, QB, Yang, TC, Kuo, CM, Tseng, HW, & Yu, P.S. (2019). Combining random forest and least square support vector regression for improving extreme rainfall downscaling. *Water (Switzerland)* , *11* (3). https://doi.org/10.3390/w11030451

[18] Praveen, B., Talukdar, S., Shahfahad, Mahato, S., Mondal, J., Sharma, P., Islam, ARMT, & Rahman, A. (2020). Analyzing trends and forecasting of rainfall changes in India using non-parametrical and machine learning approaches. *Scientific Reports* , *10* (1), 1–21. https://doi.org/10.1038/s41598-020-67228-7

[19] Reddy, PCS, Sucharitha, Y., & Narayana, G.S. (2022). Development of Rainfall Forecasting Model Using Machine Learning With Singular Spectrum Analysis. *IIUM Engineering Journal* , *23* (1), 172–186. https://doi.org/10.31436/IIUMEJ.V23I1.1822

[20] Ridwan, WM, Sapitang, M., Aziz, A., Kushiar, KF, Ahmed, AN, & El-Shafie, A. (2021). Rainfall forecasting model using machine learning methods: Case study Terengganu, Malaysia. *Ain Shams Engineering Journal* , *12* (2), 1651–1663. https://doi.org/10.1016/j.asej.2020.09.011

[21] Sarvani, K., Sai Priya, Y., Teja, C., Lokesh, T., Bala, E., & Rao, B. (2021). *Rainfall Analysis and Prediction Using Machine Learning Techniques* . *12* (06), 531–537. www.jespublication.com

[22] Skarlatos, K., Bekri, E. S., Georgakellos, D., Economou, P., & Bersimis, S. (2023). Projecting Annual Rainfall Timeseries Using Machine Learning Techniques. *Energies* , *16* (3), 1–20. https://doi.org/10.3390/en16031459

[23] Tazhibayeva, I., & Kabdulov, R. (2021). Analysis of rainfall variability using statistical and regression methods in Kazakhstan. *Geosciences, 11(10), 415*. https://doi.org/10.3390/geosciences11100415

[24] Zhang, Y., Li, H., & Ma, X. (2019). Rainfall trend analysis and prediction using Linear Regression and time series models. *Water, 11(5), 1-12*. https://doi.org/10.3390/w11050978

[25] Zhang, Y., Wang, X., & Jiang, H. (2021). Linear regression for climate and hydrological trend analysis: A case study in eastern China. *Environmental Research, 196, 110392*. https://doi.org/10.1016/j.envres.2021.110392

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 67 - 73
https://journal.physan.org/index.php/jocpes/index

Residual Network Architecture Model for Image Weather Classification

Fauzi Hasbi Saputra[1]
[1]State of Meteorology Climatology and Geophysics Agency, Tangerang, Banten, Indonesia

Article Info

Article history:

Received March 9, 2025
Revised March 14, 2025
Accepted March 15, 2025

Keywords:

Convolutional Neural Network
Residual Network
Weather Classification

A B S T R A C T

Weather classification plays an important role in many fields, including agriculture, transportation, and meteorology. Traditional methods for weather recognition are usually based on human observation or sensor networks, which are prone to errors and quite costly. To overcome the limitation, this research implements the Convolutional Neural Network method with a Residual Network model architecture for image-based weather classification. Using a dataset of 1,500 images categorized into five weather conditions cloudy, foggy, rainy, sunny and sunrise. The model training accuracy reached a level of 92%, while the validation accuracy reached a level of 94% and resulted in a testing accuracy of 86.7%. The model training accuracy was high for sunny and sunrise conditions. Accuracy was lower in rainy and foggy weather conditions. This research shows that the ResNet model architecture can provide a low-cost, efficient, and high-accuracy solution for weather classification.

Corresponden Author:

Fauzi Hasbi Saputra
State of Meteorology Climatology and Geophysics Agency
Tangerang City, Banten
Email: fauzihasbi04@gmail.com

1. INTRODUCTION

Many aspects of human endeavour such as social, economic, agricultural and environmental conditions are greatly influenced by the role of weather. In daily activities, for example wearing clothes, outdoor events, travelling, transportation and solar technolog [1], [2]. Weather is an important factor in determining the outcome of agricultural endeavours [3] . Agricultural planning can be optimized with accurate weather classification [4]. In addition, weather has a major impact on urban traffic, some weather conditions can reduce visibility and make roads slippery [5]. To improve driving safety, automatic weather recognition of road conditions is essential in traffic management and automobile auxiliary driving [5], [6], [7]. Most importantly, meteorologists need a classification of weather phenomena to improve the quality of weather predictions and knowledge of climate conditions [4].

Considering this need, weather recognition or classification is necessary for the meteorological discipline and many other facets of life. However, traditional methods based on human observation are time consuming and prone to errors [4]. futhermore, method based on sensor networks are costly and inefficient [5],[8],[9]. Due to the limitations of traditional methods, it is crucial to develop more efficient, accurate, and low-cost weather recognition methods. The recent development in computer vision technique enabled weather recognition through image processing [1], [2], [8] , increasing, efficiency and cost effectiveness [9].

In this paper, we aim to build a weather classification model using CNN method with Residual Network (ResNet) architecture, the dataset used consists of 1500 weather images. The model we built will classify the image into 5 weather conditions including cloudy, foggy, rainy, sunny, and sunrise. Before training, data augmentation is carried out first, after training, validation will be carried out. This research is expected to be able to create a classification technique that is cheap, efficient and high accuracy.

2. RELATED WORK

There are several machine learning algorithms used for classification including CNN (Convolutional Neural Networks), SVM (Support Vector Machine), and KNN (K-Nearest Neighbors). Ship et al research used the SVM method in classify 4 weather conditions with an accuracy of 92.8% [10]. The K-NN algorithm was applied for extreme rainfall estimation on 24 December, 2020 between CCTV camera image data and AWS resulting in an accuracy of 94%[11]. Xiao et al in their research classified 11 weather phenomena using an algorithm architecture called MeteCNN with accuracy performance achieving 92.68% [4]. Then Xia et al using the CNN method with an architecture called ResNet15 with additional data augmentation was able to classify 4 weather conditions with 96% accuracy [5].

Convolutional Neural Network (CNN) is a deep learning algorithm that is widely used for visual recognition of objects. The structure of CNN is generally divided into 3 sections: input layer, hidden layer and output layer [12]. The hidden layer consists of convolutional layer, rectified linear unit (ReLU) layer, pooling layer and fully connected layers[13]. The CNN is based on linear algebra represented by matrix vector multiplication used to represent data and weights[14]. The convolution layer functions for feature extraction which is carried out by several convolution kernels, each kernel extracts features from its input map. The amount of convolution kernels can be set according to the preference of the user model, the number of features extracted will be directly proportional to the number of kernels arranged. The pooling layer functions to reduce parameter size and computation, thus reducing the effect of overfitting. ReLU makes the negative value zero, increasing the model's non linearity[15]. Deep stacking is the process of repeating the convolution, pooling, and ReLU steps in several layers. The fully connected layer functions to deliver the classification decision[13]. In its application, CNN has various architectural developments, such as VGGNet, ResNet, DenseNet, etc., each of which has a model design to improve accuracy in various problems in visual object recognition.

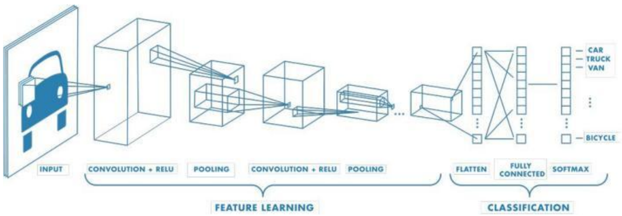

Fig. 1.1 Convolutional Neural Network Architecture [16]

VGG Network is architecture of CNN proposed by the Visual Geomety Group (VGG) of Oxford University in 2014 released in a paper by Simonyan and Zisserman. VGG network shown in Figure 2, that the depth of the network is useful for image recognition, uses small (3x3) convolution filters that show significant improvement when going into 16-19 weight layers [17]. The VGG Network architecture consists of convolution and max pooling layers for feature extraction, as well as a fully connected layer for image classification [18][19]. Network receives an input image of size 224x224, the average RGB value is subtracted from each pixel of the training set image in preprocessing. Each image goes through a 3x3 convolution filter stack with convolution stride is 1 pixel. The convolution layer consists of 5 parts which are always followed by a max-polling layer and the window size is 2 × 2 and the step size is 2. Then the image goes through 3 fully connected layers with layers 1 and 2 have 4096 channels and layer 3 have 1000 channels and the final is the soft-max layer [15][17][19].

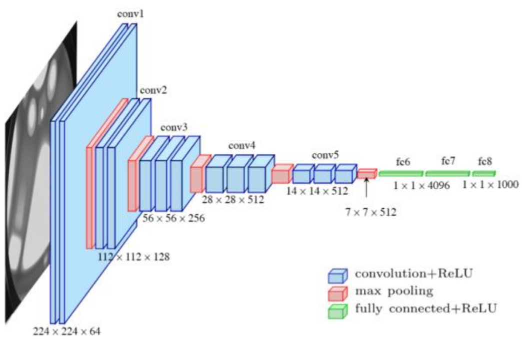

Fig. 1.2 VG6 Network Architecture by K. Simonyan and A. Zisserman [18]

Densely Connected Convolutional Network was originally proposed by Huang et al. from Cornell University in 2017 [20] which achieved state-of-the-art results on image classification with datasets named Cifar-10 and SVHN. There is a problem of gradient loss and model degradation when there are a large number of layers of convolutional neural network architecture. Dense Convolution Network provides a solution by connecting directly from all layers to all next layers. Each layer takes the output of all previous layers as input, and its own output as input of each next layer [20]. DenseNet has the advantage of smaller model parameters and computations, compact internal representations, reduced feature redundancy, and overcoming gradient loss and model degradation [21].

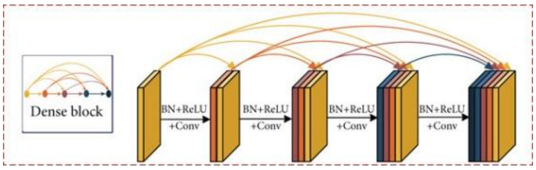

Fig. 1.3 Dense Block Model

Kaiming He et al. of Microsoft Research introduced the Residual Network in a journal article published in 2015 [23]. Networks modelled with significant deepness exhibit a strong capacity for nonlinear learning. The gradient of each layer in the network is trained in relation to the previous layer. As the number of layers in a multi layer neural network increases, so does the model error due to gradient disappearing [24]. The introduction of residual network offers an innovative solution to the problem of gradient disappearing [23]. Residual network bypasses the standard feedforward neural network of adding inputs to the output, This approach supports data integrity, simplifies the learning process, and minimizes the risk of increased training error due to number of layers deepens [21].

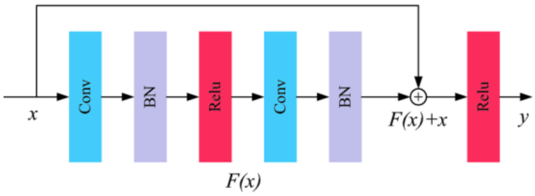

Fig. 1.4 Residual Block Structure

3. RESULTS

3.1 Data

Data serves as the crucial element for deep learning algorithms, particularly in the context of weather recognition. The effectiveness of deep neural network recognition is greatly influenced by the size of the datasets used [25]. In performing classification, supervised machine learning earning requires a large collection of labeled data [4]. This research uses a dataset that is available on the kaggle website which can be accessed at https://www.kaggle.com/datasets/vijaygiitk/multiclass-weather-dataset.

Fig. 3.1 Example Weather Images in the Dataset

The dataset has a total of 1500 images which are divided into 5 conditions, namely cloudy, foggy, rainy, shine and sunrise shown in Figure 5. Furthermore, the dataset is divided into training data and validation data with a division of 0.85 for training and 0.15 for validation, the distribution of training data is shown in Figure 6 and training data in Figure 7. The augmentation stage is carried out first before the data is trained. Augmentation data is proven to improve classification accuracy based on research conducted by Jingming Xia et al [5]. This time the augmentation process carried out is rescale, rotation, zoom, and horizontal flip.

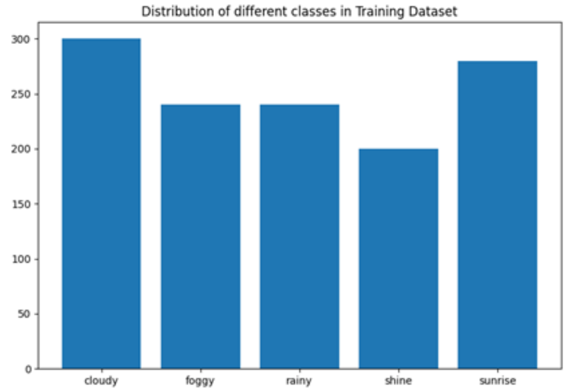

Fig. 3.2 Distribution Classes of Training Dataset

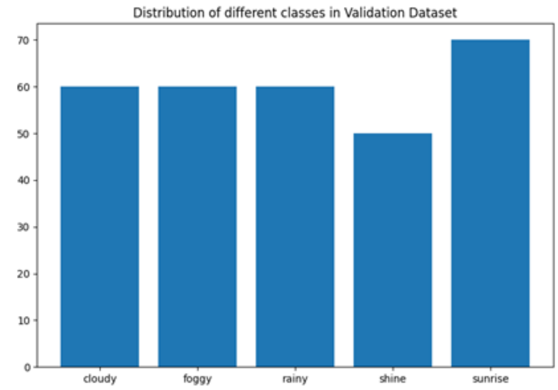

Fig. 3.3 Distribution Classes of Validation Dataset

3.2 Architecture Model

Figure 3.2 shows the average monthly rainfall during the study period.

We apply CNN method with residual network architecture to perform weather classification. The classification process goes through several layers shown in Table I , starting at the convolution layer has 64 filters, a kernel of size 7x7, and a stride of 2, functioning in extracting initial features from the input image. Followed by batch normalization layer to stabilize the value distribution and max pooling layer with 3x3 kernel and stride of 2 to reduce the spatial dimension and computational load. Then, the feature extraction process goes through six residual blocks.

Table 1. Architecture ResNet

Layer Type	**Details**	**Output Shape**
Input Layer	-	(None, 256, 256, 3)
Conv2D	64 filters, 7x7 kernel, stride 2, ReLU	(None, 128, 128, 64)
BatchNormalization	Normalization	(None, 128, 128, 64)
MaxPooling2D	3x3 pool, stride 2	(None, 64, 64, 64)
Residual Block 1	2x (64 filters, 3x3 kernel, stride 1)	(None, 64, 64, 64)
Residual Block 2	2x (64 filters, 3x3 kernel, stride 1)	(None, 64, 64, 64)
Residual Block 3	2x (128 filters, 3x3 kernel, stride 2/1)	(None, 32, 32, 128)
Residual Block 4	2x (128 filters, 3x3 kernel, stride 1)	(None, 32, 32, 128)
Residual Block 5	2x (256 filters, 3x3 kernel, stride 2/1)	(None, 16, 16, 256)
Residual Block 6	2x (256 filters, 3x3 kernel, stride 1)	(None, 16, 16, 256)
GlobalAveragePooling	Pooling over spatial dimensions	(None, 256)

Dense	512 units, ReLU	(None, 512)
Dense (Output)	5 units, Softmax	(None, 5)

Each residual block is composed of several layers shown in Table II, that is tw0 convolution layers accompanied by batch normalization and ReLU activation function. To ease the gradient flow during training, these blocks are equipped with shortcut connections that connect the initial input of the block to the output of the last convolutional layer. After passing through all the residual blocks, the global average pooling layer reduces the spatial features to a 1-dimensional vector with 256 elements output. The vector is then passed to the fully connected layer that outputs 512 units and ReLU activation to generate a feature representation, before finally being processed by the output layer that has 5 neurons with softmax activation function to generate classification probabilities to the five weather condition classes.

Table 2. Layer of Block Residual

Layer Type	Details	Output Shape
Conv2D	128 filters, 3x3 kernel, stride 2, ReLU	(None, 32, 32, 128)
BatchNormalization	Normalization	(None, 32, 32, 128)
Activation	ReLU	(None, 32, 32, 128)
Conv2D	128 filters, 3x3 kernel, stride 1	(None, 32, 32, 128)
BatchNormalization	Normalization	(None, 32, 32, 128)
Shortcut Connection	Conv2D (1x1, stride 2) + Add	(None, 32, 32, 128)
Activation	ReLU	(None, 32, 32, 128)

4. RESULT AND DISCUSSION

A Residual Network (ResNet) model was successfully built for image-based weather classification. The model classifies five weather conditions cloudy, foggy, rainy, shine, and sunrise, with a dataset consisting of 1,500 images. Following the training process, the model achieved a training accuracy of 92% and validation accuracy of 94%. Figure 8 shows the training and validation accuracy for each epoch, highlighting the overall performance and suitability of each model. It shows the consistency in accuracy between the training and testing curves with no overfitting occurring. As seen in Figure 9, the training and validation loss values are low. The steady decrease in validation loss indicates that the model has good generalisation to unencountered data.

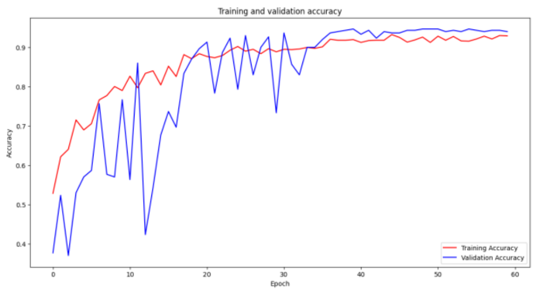

Fig. 4.1 Accuracy of Training and Validation

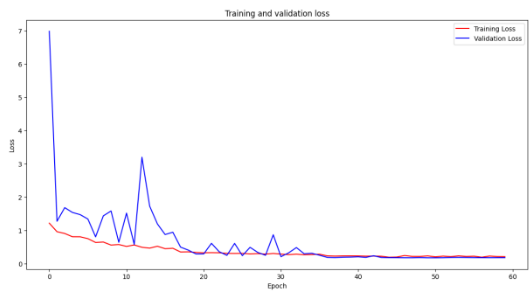

Fig. 4.2 Loss of Training and Validation

The Residual Network has the main idea of simplifying the training of very deep networks by using shortcut connections that pass information directly to the next layer, shown in Figure 4. Serving to prevent deep-level weather characteristics extracted by the convolutional layer during transmission and damage, six blocks of residual are incorporated into the model. The weather characteristics extracted in the previous layer serve as a shortcut through the residual module to the next layer, improving the convergence rate of the network model, improving the recognition accuracy, and solving the vanishing gradient problem caused by the larger network depth.

Model was programmed to train with 100 epochs, but the training stopped when the epoch reached 60. The training stopped because the EarlyStopping callback detected that the validation loss had reached the optimal point. This is indicated because there is no more significant improvement in the last 10 epochs. This mechanism prevents over-fitting and reduces training time.

The model was tested with 30 test images as shown in Fig. 10, the model produced accurate predictions for 26 images, while the remaining 4 images were incorrectly classified. This gives an accuracy rate of 86.7% on the test data. Based on the confusion matrix, the shine and sunrise conditions have perfect prediction accuracy, showing the strength of the model in recognizing these two conditions. However, some errors occurred in conditions such as rainy and foggy.

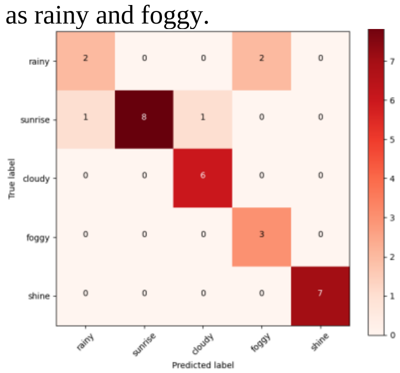

Fig. 4.3 Confusion Matrix

Value of testing accuracy is lower than the training and validation accuracy. We estimate this is because the testing dataset is too small, such as in rainy and foggy which are only 4 and 3. So that the testing data has not drawn the true testing accuracy. To see the smallness of our test dataset, we will compare it with other similar studies. Separated WEAPD (6877) into 8:1:1 training, validation, and testing groups [4]; 8000 training images and 2000 testing images [26]; split the dataset (23865) 80% for training and 20% testing [27].

5. CONCLUSION

In this study, a low-cost, efficient, and high-accuracy image-based weather classification technique using the Residual Network (ResNet) model was successfully built. This model can classify images into 5 weather conditions, including cloudy, foggy, rainy, shine, and sunrise, using a dataset of 1500 images. The training accuracy of the model is 92% and the validation accuracy is 94%, showing good performance and consistency between the training and validation data. Testing was conducted with 30 test images, where the model successfully classified 26 images correctly and 4 images were incorrectly classified with a testing accuracy of 86.7%. The accuracy of the model was very good in shine and sunrise conditions, but low classification accuracy results in rainy and foggy conditions. Testing accuracy can be improved to similar to training and validation accuracy by increasing the number of images in the test dataset.

REFERENCE

[1] C. Lu, D. Lin, J. Jia, and C.-K. Tang, "Two-Class Weather Classification," IEEE Transactions on Pattern Analysis and Machine Intelligence, Jun. 2014, doi: 10.1109/cvpr.2014.475

[2] B. Zhao, X. Li, X. Lu, and Z. Wang, "A CNN–RNN architecture for multi-label weather recognition," Neurocomputing, vol. 322, pp. 47–57, Sep. 2018, doi: 10.1016/j.neucom.2018.09.048..

[3] K. K. Singh, A. K. Baxla, P. Singh, and P. K. Singh, "Weather based information on risk management in agriculture," *Clim. Chang. Agric. India Impact Adapt.*, pp. 206–216, 2018, doi: 10.1007/978-3-319-90086-5_16.

[4] H. Xiao, F. Zhang, Z. Shen, K. Wu, and J. Zhang, "Classification of Weather Phenomenon From Images by Using Deep Convolutional Neural Network," *Earth Sp. Sci.*, vol. 8, no. 5, pp. 1–9, 2021, doi: 10.1029/2020EA001604.

[5] J. Xia, D. Xuan, L. Tan, and L. Xing, "ResNet15: Weather Recognition on Traffic Road with Deep Convolutional Neural Network," *Adv. Meteorol.*, vol. 2020, 2020, doi: 10.1155/2020/6972826.

[6] T. Yu, Q. Kuang, J. Hu, J. Zheng, and X. Li, "Global-Similarity Local-Salience Network for Traffic Weather Recognition," pp. 4607–4615, 2021, doi: 10.1109/ACCESS.2020.3048116.

[7] K. B. Jayanthi, "An efficient weather recognition algorithm on highway roads for vehicle guidance," vol. 6, no. June, pp. 1212–1223, 2022.

[8] J. Carlos, V. Guerra, Z. Khanam, S. Ehsan, R. Stolkin, and K. Mcdonald-maier, "Weather Classification : A new multi-class dataset , data augmentation approach and comprehensive evaluations of Convolutional Neural

Networks”.
[9] S. Chen, T. Shu, H. Zhao, and Y. Yan, “For Real-Time Multi-Label Weather Recognition,” no. Dl, pp. 1–18, 2023.
[10] E. Ship, “Real-Time Weather Image Classification with SVM : A Feature-Based Approach”.
[11] S. A. Rahayu, S. B. Sipayung, A. Witono, and L. S. Suprihatin, “The Performance of K-Nearest Neighbor (KNN) Approach for Estimating Extreme Rain Events based on CCTV Images Camera Data,” *J. EECCIS (Electrics, Electron. Commun. Control. Informatics, Syst.*, vol. 17, no. 2, pp. 59–65, 2023, doi: 10.21776/jeeccis.v17i2.1649.
[12] M. N. Khan and M. M. Ahmed, “Snow Detection using In-Vehicle Video Camera with Texture-Based Image Features Utilizing K-Nearest Neighbor, Support Vector Machine, and Random Forest,” *Transp. Res. Rec.*, vol. 2673, no. 8, pp. 221–232, 2019, doi: 10.1177/0361198119842105.
[13] O. K. Clustering *et al.*, “Deep Learning (CNN) and Transfer Learning : A Review Deep Learning (CNN) and Transfer Learning : A Review,” 2022, doi: 10.1088/1742-6596/2273/1/012029.
[14] K. Audhkhasi, O. Osoba, and B. Kosko, “Noise-enhanced convolutional neural networks,” *Neural Networks*, vol. Volume 78, pp. 15–13, 2016.
[15] Y. Shi, Y. Li, J. Liu, X. Liu, and Y. L. Murphey, “Weather Recognition Based on Edge Deterioration and Convolutional Neural Networks,” *Proc. - Int. Conf. Pattern Recognit.*, vol. 2018-Augus, pp. 2438–2443, 2018, doi: 10.1109/ICPR.2018.8546085.
[16] I. Kharisudin, M. F. Az-Zahra, E. R. Winarti, and S. B. Waluya, “Deep convolutional neural networks for the detection of macular diseases from optical coherence tomography images,” *J. Phys. Conf. Ser.*, vol. 1567, no. 2, 2020, doi: 10.1088/1742-6596/1567/2/022076.
[17] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.
[18] M. Ferguson, R. Ak, Y. T. T. Lee, and K. H. Law, “Automatic localization of casting defects with convolutional neural networks,” *Proc. - 2017 IEEE Int. Conf. Big Data, Big Data 2017*, pp. 1726–1735, 2017, doi: 10.1109/BigData.2017.8258115.
[19] M. Bansal, M. Kumar, M. Sachdeva, and A. Mittal, “Transfer learning for image classification using VGG19: Caltech-101 image data set,” *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 4, pp. 3609–3620, 2023, doi: 10.1007/s12652-021-03488-z.
[20] G. Huang, Z. Liu, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” *Comput. Vis. Pattern Recognit.*, pp. 2261-2269., 2017.
[21] Y. Wang and Y. Li, “Research on Multi-class Weather Classification Algorithm Based on Multi-model Fusion,” no. Itnec, pp. 2251–2255, 2020.
[22] T. Zhou, X. Ye, H. Lu, X. Zheng, S. Qiu, and Y. Liu, “Dense Convolutional Network and Its Application in Medical Image Analysis,” *Biomed Res. Int.*, vol. 2022, 2022, doi: 10.1155/2022/2384830.
[23] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.
[24] X. Yang, T. Xie, L. Liu, and D. Zhou, “Image super-resolution reconstruction based on improved Dirac residual network,” *Multidimens. Syst. Signal Process.*, vol. 32, no. 4, pp. 1065–1082, 2021, doi: 10.1007/s11045-021-00773-0.
[25] D. Mishkin, N. Sergievskiy, and J. Matas, “Systematic evaluation of convolution neural network advances on the Imagenet,” *Comput. Vis. Image Underst.*, vol. 161, pp. 11–19, 2017, doi: 10.1016/j.cviu.2017.05.007.
[26] B. Zhao, L. Hua, X. Li, X. Lu, and Z. Wang, “Weather recognition via classification labels and weather-cue maps,” *Pattern Recognit.*, vol. 95, pp. 272–284, 2019, doi: 10.1016/j.patcog.2019.06.017.
[27] M. R. Ibrahim, J. Haworth, and T. Cheng, “Weathernet: Recognising weather and visual conditions from street-level images using deep residual learning,” *ISPRS Int. J. Geo-Information*, vol. 8, no. 12, 2019, doi: 10.3390/ijgi8120549.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 74 - 79
https://journal.physan.org/index.php/jocpes/index

Comparison of Random Forest and LSTM Methods for Temperature Prediction

Muhammad Asyril Ar Rosyad[1], Aviv Maghridlo[2]
[1,2]State of Meteorology Climatology and Geophysics Agency, Tangerang, Banten, Indonesia

Article Info

Article history:

Received March 11, 2025
Revised March 16, 2025
Accepted March 17, 2025

Keywords:

Long Short-Term Memory (LSTM)
Random Forest (RF)
Temperature Prediction
RMSE
MAE
R^2 Score
Machine Learning
Time Series Forecasting
Temperature Trends
Model Evaluation

A B S T R A C T

This study compares the performance of Long Short-Term Memory (LSTM) and Random Forest (RF) models in predicting temperature data from Tanjung Priok, Indonesia, using evaluation metrics such as RMSE, MAE, and R^2 Score. The LSTM model demonstrated its ability to capture temporal dependencies and temperature trends, achieving an R^2 score of 0.4493 and an MAE of 0.5863. In contrast, the RF model performed better in minimizing prediction errors, with a lower RMSE of 0.6498 and an R^2 score of 0.4066. While the LSTM model excelled in explaining variance in the temperature data, the RF model was more effective in stable periods, exhibiting lower prediction errors. The results highlight that both models have distinct advantages, with LSTM better suited for capturing long-term temperature trends and RF performing well during periods of stability. Future research could explore hybrid models or further optimization of these techniques to improve prediction accuracy, particularly for dynamic and extreme temperature fluctuations.

Corresponden Author:

Muhammad Asyril Ar Rosyad
State of Meteorology Climatology and Geophysics Agency
Tangerang City, Banten
Email: asyrilrosyad@gmail.com

1. INTRODUCTION

This journal discusses the comparison of the use of RF model and LSTM machine learning methods for predicting temperatures with a focus on accuracy assessment using RMSE, MAE and R2 Score. Temperature prediction plays a critical role in sectors such as agriculture, energy management, and disaster preparedness, particularly in regions like Indonesia, where climate patterns can vary significantly.

Because of the unpredictable and chaotic character of atmospheric conditions and our limited knowledge of the several atmospheric processes, weather forecasting is a difficult undertaking [1]. Temperature is one of the weather factors. On a numerical scale, temperature can be a physical quantity that represents the degree of hotness or coldness. It is measured using a thermometer, which can be calibrated in any of the various temperature scales, and may represent a degree of the warm liveliness per molecule of matter or radiation [2]. Predicting air temperature is one of the most important parts of studying the climate. When making decisions to solve ecological, industrial, or environmental issues, accurate temperature prediction can offer vital direction [3]. In this study, the data used is the daily mean temperature, which provides an average value of temperatures recorded throughout the day, offering a comprehensive representation of the temperature trends.

Machine learning can be used to estimate air temperature [4][5]. The logical consideration of computations and quantifiable models that computer systems use to carry out certain tasks without requiring explicit modification is known as machine learning (ML)[6]. LSTM and Random Forest are two of the several machine learning techniques [7][8][9].

One advancement in neural networks that can learn long-term dependencies is LSTM [10]. Recurrent neural networks are extended by LSTM networks, which overcome the vanishing gradient problem that plagues RNNs and enable us to recall and account for historical data in memory [11]. By highlighting contextually relevant information in textual data, LSTM models in conjunction with attention processes have the potential to greatly improve sentiment classification model performance [12].

When modeling the data, the Random Forest method is employed by combining multiple independent decision trees [13]. The algorithm splits the data into several random subsets and builds decision trees for each subset. This process is known as an ensemble method [13][14]. In the context of time series forecasting, Random Forest can capture non-linear patterns and complex relationships between variables affecting the data [15].

To determine which machine learning model is most suited for predicting air temperature, accuracy evaluation is applied to the models [16][17][5]. MAE, the distances between the estimated and observed values for N are averaged in this measure, which is an error statistic. The average squared discrepancy between the expected and observed temperature data is known as the MSE. For N, the root mean square error RMSE is as follows: This metric represents the standard deviation of the discrepancy between the actual observed data and the estimation. The sensitivity of this measure to large prediction mistakes is higher [5]. Additionally, the R^2 score is used to evaluate the proportion of variance in the observed data that is predictable from the independent variables [18]. A higher R^2 value indicates a better fit of the model to the data, with 1 being a perfect prediction and 0 indicating no predictive power. The R^2 score provides an additional layer of assessment in determining the overall accuracy of the temperature prediction model [18][19].

The main objective of this journal is to evaluate how well two time series prediction techniques—RF and LSTM—predict temperature. The goal of the study is to illustrate the advantages and disadvantages of each approach: LSTM's ability to identify complex non-linear patterns, particularly in irregular datasets, and RF ability to handle non-linear relationships, handle unstructured data, and be robust to overfitting, making it suitable for assessing feature importance. The journal also highlights the potential of hybrid models, which combine the strengths of both strategies to improve accuracy, as well as the approach of accuracy evaluation metrics to determine the most suitable method for temperature prediction in the study are.

2. RESEARCH METHOD

The method used in this study was based on LSTM and RF techniques, with accuracy evaluation using RMSE, MAE, and R^2 Score. The temperature data from Tanjung Priok, spanning from January 1, 2013 to December 31, 2023, was processed for this analysis. The data was obtained from the Climate Data Online website, created by NOAA. The collected data was used to compare the performance of LSTM and Random Forest models in predicting temperature, with a focus on evaluating the models' accuracy using the mentioned metrics. This study aims to determine which model provides more accurate predictions for temperature in the Tanjung Priok region, offering insights into the effectiveness of these machine learning approaches.

3. RESULTS

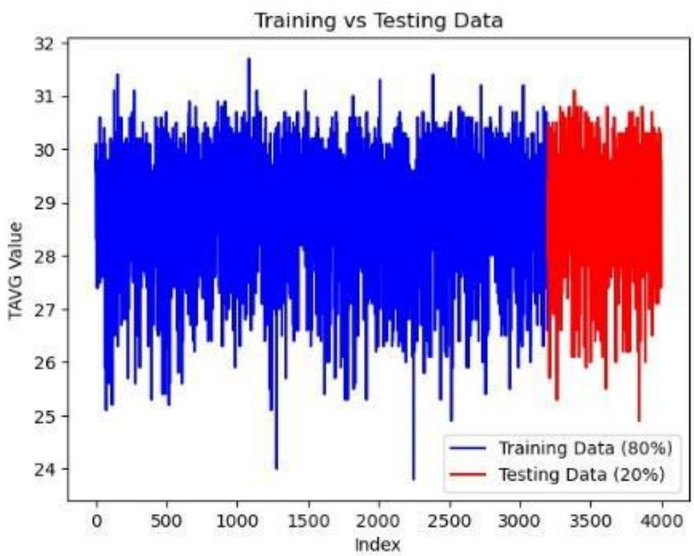

Fig. 3.1 Data Splitting

The temperature data from Tanjung Priok, spanning January 1, 2013, to December 31, 2023, was sourced from the Climate Data Online platform provided by NOAA. For the analysis, the dataset was divided into 80% for training and 20% for testing. The study compared the performance of LSTM and RF models in temperature prediction, with model accuracy evaluated using selected metrics.

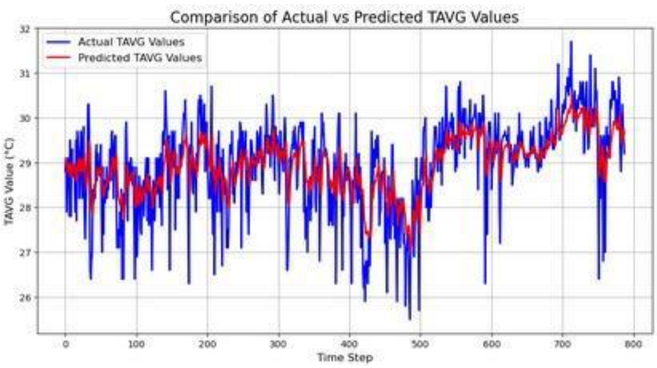

Fig. 3.2 Actual and Predicted TAVG using LSTM

The graph illustrates the comparison between actual and predicted average temperature (TAVG) values generated using the LSTM model. The blue line represents the actual TAVG values, while the red line indicates the predicted values obtained from the LSTM model. The results show a close alignment between the two lines, highlighting the LSTM model's ability to capture the temperature trends over the observed time steps accurately. Despite slight deviations at certain time intervals, the overall pattern suggests that the LSTM model effectively learns the temporal dependencies in the dataset and provides reliable predictions.

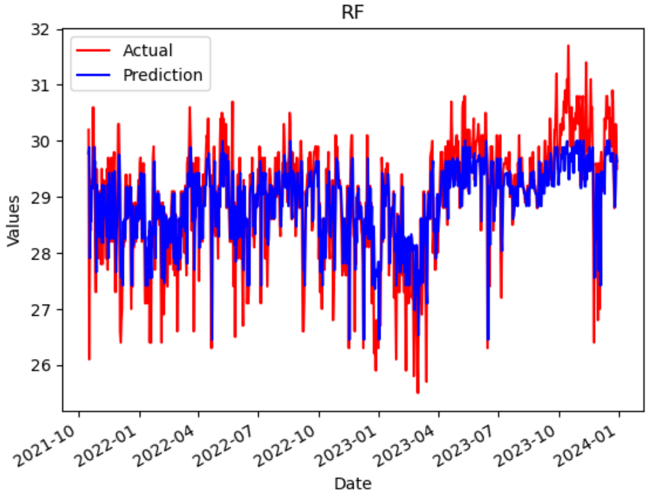

Fig. 3.3 Actual and Predicted TAVG using RF

The graph illustrates the comparison between actual and predicted temperature values generated using the RF model over the period from October 2021 to January 2024. The results show that the RF model successfully captures the overall trend and seasonal patterns of the temperature data, as indicated by the close alignment between the actual values (red line) and predicted values (blue line). However, discrepancies are observed during periods of extreme variations, where the model struggles to accurately predict sudden spikes or drops in temperature. Despite these deviations, the RF model performs well during relatively stable periods, demonstrating its capability to model the general behavior of the temperature data. These findings suggest that while the Random Forest model is effective for temperature prediction, further improvements, such as incorporating additional meteorological variables or optimizing model parameters, could enhance its performance in capturing dynamic and extreme temperature fluctuations.

Table 1. Evaluation Metrics

Model	RMSE	MAE	R^2 Score
LSTM	0,7742774935015089	0.5863260356302794	0,4493465040506345
RF	0,6498641163085757	0,6122487388374491	0,4066512320396348

The table presents the performance comparison between the LSTM and Random Forest models based on three evaluation metrics: RMSE, MAE, and R^2 Score. The LSTM model achieved an RMSE of 0.7743, an MAE of 0.5863, and an R^2 Score of 0.4493, while the RF model obtained a lower RMSE of 0.6498 and a slightly higher MAE of 0.6122, with an R^2 Score of 0.4066. These results indicate that while the RF model produces lower prediction errors (RMSE), the LSTM model explains a slightly higher proportion of the data variance (R^2 Score) and yields a lower MAE. Overall, both models show competitive performance, but LSTM demonstrates a marginally better ability to capture the underlying temperature trends.

4. DISCUSSION

The results from this study highlight the performance comparison between the LSTM and RF models in predicting temperature data. Both models have shown competitive results, though with subtle differences in terms of predictive accuracy.

In terms of RMSE, the LSTM model had a value of 0.7743, while the RF model had a lower RMSE of 0.6498. RMSE measures the average magnitude of error between predicted and actual values, with lower values indicating better model performance. The lower RMSE of the RF model suggests that it is more effective at minimizing the difference between actual and predicted temperatures. However, RMSE does not necessarily reflect the model's ability to capture the variance in the data, which makes other metrics like MAE and R-squared (R^2) crucial for further evaluation [20].

When comparing MAE, the LSTM model achieved a slightly better result with an MAE of 0.5863, compared to the Random Forest model's MAE of 0.6122. MAE measures the average of the absolute differences between predicted and actual values, providing a simpler view of the prediction accuracy [21]. The lower MAE of the LSTM model indicates that it produced slightly more accurate predictions in terms of the average magnitude of error, making it more reliable when absolute errors are a key concern.

Regarding the R-squared (R^2) value, which measures how well the model explains the variance in the data [17][22], the LSTM model outperformed the Random Forest model with an R^2 score of 0.4493, compared to 0.4066 for RF. The LSTM model explains around 44.93% of the variance in the temperature data, capturing more of the underlying temperature trends, especially in terms of temporal dependencies. In contrast, the RF model explains about 40.66% of the variance, which suggests that while it performs well overall, it may not capture as much of the temporal patterns present in the data.

Further analysis revealed that the LSTM model had an advantage in handling complex, sequential patterns in temperature data, allowing it to better capture long-term dependencies and fluctuations over time. This makes the LSTM model particularly valuable for scenarios where temperature forecasting requires consideration of historical trends and seasonality. On the other hand, the Random Forest model, which operates through ensemble learning and decision trees, excelled in its ability to minimize overall errors (as indicated by the RMSE) and performed consistently well for stable, less volatile periods in the dataset. However, its limitations in capturing sudden temperature spikes or drops suggest the need for further enhancements, such as incorporating additional meteorological features like humidity, wind speed, or atmospheric pressure.

Both models demonstrated competitive performance, but each excelled in different aspects. The LSTM model's ability to capture temporal dependencies and trends in the data gives it a slight edge in learning from time-series data, which makes it suitable for long-term temperature forecasting. On the other hand, the RF model performed better in minimizing prediction error (lower RMSE), making it effective for stable periods in temperature data. However, it struggled to accurately predict sudden spikes or drops in temperature, which could be improved by incorporating additional meteorological variables or optimizing its parameters.

In conclusion, this study highlights that model selection depends on the specific forecasting goals. The LSTM model is advantageous for scenarios requiring time-aware analysis and long-term trend detection, while the Random Forest model is more suitable for cases prioritizing overall error minimization and short-term stability. Future studies could explore hybrid approaches that combine the strengths of both models, or incorporate other deep learning architectures, such as GRU or CNN, to further improve predictive performance. Additionally, enriching the dataset with external factors like climate anomalies, El Niño events, or urbanization trends could enhance the robustness of temperature predictions for regions like Tanjung Priok.

5. CONCLUSION

This study compared the performance of LSTM and RF models for predicting temperature data from Tanjung Priok, using three evaluation metrics: RMSE, MAE, and R^2 Score. The LSTM model demonstrated superior performance in capturing temporal dependencies and trends in the data, with a higher R^2 score and lower MAE, which suggests it was more effective in understanding the underlying temperature patterns. This makes the LSTM model particularly suitable for long-term temperature forecasting, where learning from past temperature trends and making predictions based on these temporal dependencies is crucial. However, the LSTM model exhibited a higher RMSE, reflecting slightly larger prediction errors compared to the RF model.

On the other hand, the Random Forest model had a lower RMSE, indicating that it performed better in minimizing prediction errors, particularly in stable temperature periods. While RF was effective at handling non-linear relationships and avoiding overfitting, it struggled with sudden temperature spikes and extreme fluctuations, which led to discrepancies in its predictions. Despite this, the lower RMSE suggests RF is more robust in stable conditions. Both models demonstrated competitive strengths, with LSTM excelling in capturing trends and RF performing well in minimizing error. Future improvements could include hybrid models combining the strengths of both approaches, as well as incorporating additional meteorological variables or optimizing model parameters to better handle dynamic and extreme weather conditions.

REFERENCE

[1] E. De Saa and L. Ranathunga, "Comparison between ARIMA and Deep Learning Models for Temperature Forecasting," 2020, [Online]. Available: http://arxiv.org/abs/2011.04452

[2] M. Asamoah-Boaheng, "Using SARIMA to Forecast Monthly Mean Surface Air Temperature in the Ashanti Region of Ghana Using Seasonal Autoregressive Integrated Moving Average (SARIMA) model, the study determined an adequate forecasting model for the mean temperature of Ashanti Reg," Int. J. Stat. Appl., vol. 4, no. 6, pp. 292–299, 2014, doi: 10.5923/j.statistics.20140406.06.

[3] M. Yu, F. Xu, W. Hu, J. Sun, and G. Cervone, "Using Long Short-Term Memory (LSTM) and Internet of Things (IoT) for Localized Surface Temperature Forecasting in an Urban Environment," IEEE Access, vol. 9, pp. 137406–137418, 2021, doi: 10.1109/ACCESS.2021.3116809.

[4] D. Fister, J. Pérez-Aracil, C. Peláez-Rodríguez, J. Del Ser, and S. Salcedo-Sanz, "Accurate long-term air temperature prediction with Machine Learning models and data reduction techniques," Appl. Soft Comput., vol. 136, p. 110118, 2023, doi: 10.1016/j.asoc.2023.110118.

[5] J. Cifuentes, G. Marulanda, A. Bello, and J. Reneses, "Air temperature forecasting using machine learning techniques: A review," Energies, vol. 13, no. 6, pp. 1–28, 2020, doi: 10.3390/en13164215.

[6] B. Mahesh, "Machine Learning Algorithms - A Review," Int. J. Sci. Res., vol. 9, no. 1, pp. 381–386, 2020, doi: 10.21275/art20203995.

[7] D. Gente, "P-issn 2244-4432 e-issn 2984-7125," vol. 14, pp. 73–83, 2023.

[8] A. A. Akbar, Y. Darmawan, A. Wibowo, and H. K. Rahmat, "Accuracy Assessment of Monthly Rainfall Predictions using Seasonal ARIMA and Long Short-Term Memory (LSTM)," vol. 5, no. 2, pp. 99–114, 2024.

[9] M. Elsaraiti and A. Merabet, "A comparative analysis of the arima and lstm predictive models and their effectiveness for predicting wind speed," Energies, vol. 14, no. 20, 2021, doi: 10.3390/en14206782.

[10] J. Lu, G. Li, S. S. R. Moustafa, and S. S. Khodairy, "A comparison of SARIMA and LSTM in forecasting dengue hemorrhagic fever incidence in Jambi , Indonesia A comparison of SARIMA and LSTM in forecasting dengue hemorrhagic fever incidence in Jambi , Indonesia", doi: 10.1088/1742-6596/1566/1/012054.

[11] S. Khedhiri, "Comparison of SARFIMA and LSTM methods to model and to forecast Canadian temperature," Reg. Stat., vol. 12, no. 2, pp. 177–194, 2022, doi: 10.15196/RS120204.

[12] A. K. Wardhana, Y. Riwanto, and B. W. Rauf, "Performance Comparison Analysis on Weather Prediction using LSTM and TKAN," vol. 04, pp. 3–10, 2024, doi: 10.31763/iota.v4i3.808.

[13] R. Meenal, P. A. Michael, D. Pamela, and E. Rajasekaran, "Weather prediction using random forest machine learning model," Indones. J. Electr. Eng. Comput. Sci., vol. 22, no. 2, pp. 1208–1215, 2021, doi: 10.11591/ijeecs.v22.i2.pp1208-1215.

[14] W. Y. N. Naing and Z. Z. Htike, "Forecasting of monthly temperature variations using random forests," ARPN J. Eng. Appl. Sci., vol. 10, no. 21, pp. 10109–10112, 2015.

[15] X. Wang, "Ladle furnace temperature prediction model based on large-scale data with random forest," IEEE/CAA J. Autom. Sin., vol. 4, no. 4, pp. 770–774, 2017, doi: 10.1109/JAS.2016.7510247.

[16] S. Kendzierski, B. Czernecki, L. Kolendowicz, and A. Jaczewski, "Air temperature forecasts' accuracy of selected short-term and long-term numerical weather prediction models over Poland," Geofizika, vol. 35, no. 1, pp. 19–37, 2018, doi: 10.15233/gfz.2018.35.5.

[17] B. Azari, K. Hassan, J. Pierce, and S. Ebrahimi, "Evaluation of Machine Learning Methods Application in Temperature Prediction," Comput. Res. Prog. Appl. Sci. Eng., vol. 8, no. 1, pp. 1–12, 2022, doi: 10.52547/crpase.8.1.2747.

[18] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," PeerJ Comput. Sci., vol. 7, pp. 1–24, 2021, doi: 10.7717/PEERJ-CS.623.

[19] A. V Tatachar, "Comparative assessment of regression models based On model evaluation metrics," Int. Res. J. Eng. Technol., vol. 8, no. 9, pp. 853–860, 2021, [Online]. Available: https://d1wqtxts1xzle7.cloudfront.net/73250877/IRJET_V8I9127-libre.pdf

[20] Doreswamy, K. S. Harishkumar, Y. Km, and I. Gad, "Forecasting Air Pollution Particulate Matter (PM2.5) Using Machine Learning Regression Models," Procedia Comput. Sci., vol. 171, no. 2019, pp. 2057–2066, 2020, doi: 10.1016/j.procs.2020.04.221.

[21] A. Ahmad, K. A. Ostrowski, M. Maślak, F. Farooq, I. Mehmood, and A. Nafees, "Comparative study of supervised machine learning algorithms for predicting the compressive strength of concrete at high temperature," Materials (Basel)., vol. 14, no. 15, 2021, doi: 10.3390/ma14154222.

[22] R. M. Adnan et al., “Air temperature prediction using different machine learning models,” Indones. J. Electr. Eng. Comput. Sci., vol. 22, no. 1, pp. 534–541, 2021, doi: 10.11591/ijeecs.v22.i1.pp534-541.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 80 - 88
https://journal.physan.org/index.php/jocpes/index

Flood Early Warning Monitoring System Using KNN Methods In Bangkalan District

Iqbal Fariansyah Ridwan[1]
[1]Undergraduate Program Aplied Instrumentation Meteorology Climatology Geophysics (STMKG)

Article Info

Article history:

Received March 12, 2025
Revised March 17, 2025
Accepted March 18, 2025

Keywords:

Flood
Early warning system
Prediction
KNN
Algorithm.

ABSTRACT

Bangkalan Regency faces serious challenges due to flood disasters that periodically threaten the safety and welfare of the community. The flood phenomenon in this area is caused by a complex combination of geographic and climate factors, including increased extreme rainfall and the dynamics of sea level rise. Annual floods not only cause infrastructure damage but also threaten the livelihoods and lives of residents living around the river flow. This study aims to develop an innovative flood early warning system using the K-Nearest Neighbors (KNN) method to predict potential disasters before floods occur. By using water flow data analysis and machine learning algorithms, this system is designed to provide accurate and timely early estimates. The main advantage of this study is its ability to proactively mitigate disaster risks using modern computer technology. The study produced a prototype of a flood detection and simulation system that can help local governments, related agencies, and the Bangkalan community in taking preventive and mitigation actions at an earlier stage. Therefore, this system is expected to make a significant contribution to reducing the impact of disasters and protecting the lives of Bangkalan Regency residents.

Corresponden Author:

Iqbal Fariansyah Ridwan,
Undergraduate Program Aplied Instrumentation Meteorology Climatology Geophysics (STMKG)
Tangerang, Indonesia
Email: iqbalfariansyahridwan@gmail.com

1. INTRODUCTION

One of the natural disaster problems that often occurs in several regions in Indonesia is flooding, according to data from the Indonesian Disaster Information Data (DIBI) the number of flood disasters that occurred in Indonesia was 751 out of 2,313 natural disasters. Then data from the National Flood Management Agency (BNPB) stated that there were 7,812 natural disasters from 21,368 total natural disasters, this places flooding as number one in the number of natural disasters [1]. Floods cause many losses including material and immaterial, the impacts caused after the flood also make many flood victims remain vigilant against subsequent floods. Material losses experienced by victims can be in the form of damaged goods, disease outbreaks, damaged plants, and difficulty in getting clean water. One of the causes of flooding in Indonesia that most often results in flooding is high rainfall, if an area has high rainfall and continues for a long period of time, then there is a high risk of flooding. Bangkalan is one of the regencies in East Java Province, located at the western tip of Madura Island, bordering Sampang Regency to the east, Gresik City to the north, Surabaya City to the south and Madura Strait to the west. There are several ways to overcome flooding, such as managing river flows, not littering, reforesting forests and creating a monitoring and warning system in flood-prone areas.

An early warning system is a series of systems that will notify of a natural disaster. Early warning of natural disasters so that the community can respond when a natural disaster is about to occur. Delays in handling natural disasters can result in great losses to the community [2]. The early warning system gives potential victims time to take the necessary actions when a natural disaster occurs.

Researchers took a river sample, namely the Kali Tunjung River because the map shows that flood-prone areas are located around the Kali Tunjung River basin. In this study, an Arduino-based flood detection system has been created that utilizes water level and water flow sensors combined with the k-nearest neighbor (KNN) algorithm, Arduino is a technology in the form of a circuit board that contains a microcontroller that can be programmed via a computer as needed, KNN is used in intelligent systems as an algorithm that determines the class of an event that occurs [3]. So that in this study, Arduino and the KNN algorithm are combined to be able to predict the arrival of floods quickly and accurately so that potential flood victims can take appropriate action before the flood comes.

2. LITERATURE

2.1. Flood

Explain the chronology of the research, including research design, research procedures (in the form of algorithms, Pseudocode or others), testing methods and data acquisition [3]. The description of the course of research must be supported by references, so that the explanation is scientifically acceptable. Tables and Figures are presented in the centre, as shown below and cited in the manuscript.

a. Rainfall

In the rainy season, high rainfall will cause flooding in rivers and if it exceeds the river bank, flooding or inundation will occur [5].

b. Physiographic effect

Physiography or physical geography of rivers such as the shape and slope of the River Basin Area (DPS), river slope, hydraulic geometry (cross-section shape such as width, depth, longitudinal section, river bed material), river location.

c. Erosion and Sedimentation

Erosion in the DPS affects the river's storage capacity, because the eroded soil in the DPS when carried by rainwater into the river will settle and cause sedimentation. Sedimentation will reduce the capacity of the river and when there is a flow that exceeds the river's capacity, it can cause flooding River capacity.

d. River capacity

The reduction in flood flow capacity in rivers is caused by sedimentation originating from excessive erosion of river beds and riverbanks, due to the absence of vegetation cover.

e. Effect of high tide

Sea water slows down the flow of rivers to the sea. During floods along with high tides, the height of the inundation/flood becomes higher because of backwater.

The effects of flooding are many, namely spreading germs, loss of property, damaged facilities and infrastructure and difficulty in obtaining clean water [6]. There are several ways to overcome flooding, namely not littering, reforestation, dredging rivers and early warning systems for flooding [7].

In Bangkalan Regency, there are several main factors that Bangkalan Regency often experiences waterlogging which causes Bangkalan to be submerged in water, including Bangkalan Regency has a topography with a land slope of 0-2 º (flat) and a land slope of 3-15 º (wavy) with a stretch of lowland at an altitude of between 2-100 meters above sea level [8]. From this condition, Bangkalan Regency often experiences waterlogging during the rainy season, especially when it rains together with the rising tide. In general, the topography of Bangkalan Regency is divided into two parts, namely:

- The western part towards the north is relatively low close to the coastal coast.
- The eastern part has hilly topography such as in Geger Sub-district.

According to BPBD Bangkalan Regency, the causes of flooding in Bangkalan Regency are as follows, namely:

No.	Parameter	Indicator	Data Source	Monitoring Month
1.	Disaster prone areas	Potential for flooding > 80% of 18 sub-districts and 281 villages	BPBD	December - January
2.	Rainfall	>700 mm for several consecutive times	BMKG Maritime	December – January
3.	Sea level rise	The west monsoon is characterized by wind speeds reaching >25 knots in the Madura Strait waters and above average sea wave heights, namely reaching 2 - 2.5 m.	BMKG Maritime	December – January - February
4.	Other factors that are considered dominant	The emergence of anomalous events	BPBD - BMKG	December – January - February

Fig. 1 Causes of Flooding in Bangkalan Regency

Then followed by rainfall data for East Java Regency, namely:

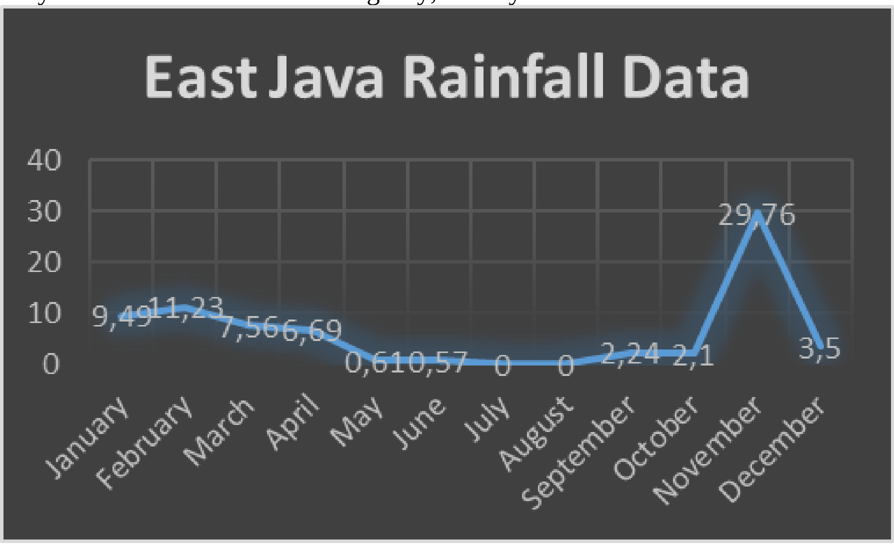

Fig. 2 Rainfall in East Java province in 2023

Then followed by rainfall data for Bangkalan Regency, namely:

Fig. 3 Rainfall in Bangkalan District in 2023

It can be concluded that the highest flooding possibility in Bangkalan city is in December, January, and February. Then followed by data on sea level rise in Bangkalan city stating that the sea level rise in the Madura Strait is between 5 - 10 mm / year. There are two potential flash floods in Bangkalan Regency, if simultaneously there is an increase in rainfall capacity above the average of 600 / mm, the highest tide or full moon. Then the BPBD of Bangkalan Regency also stated that there are ten characteristics of flood threats, including the following:

- Influence of Rainfall
- Changes in land use in River Basin Areas (DAS)
- Influence of Tides and Global Warming
- Influence of Surface Elevation
- Drainage System
- Community Habits of Throwing Away Garbage
- Pressure on Land Use
- Sedimentation
- Flood Disaster Control System
- Impact of Rising Sea Levels

2.2. Flood Detection System

Flood detection system is a system used to detect the arrival of floods. This system is usually combined with an early warning system so that potential victims can take the action before a flood disaster occurs so that they can minimize losses caused by flooding [9]. Flood detection systems are installed on river flows or places that are inundated by water so that when there is an overflow of water, it can be known and can be acted upon in accordance with applicable operations.

2.3. Sensor

Sensors are devices that can detect and respond to several types of input from the physical environment. Specific inputs can be light, heat, motion, humidity, pressure, movement and infrared [10]. There are many types of sensors such as ping sensors, pressure sensors, rotational speed sensors, temperature sensors, infrared sensors, smoke sensors and humidity sensors. Sensors are very useful as a substitute for human senses in carrying out their functions, data sent from sensors can be processed according to the needs needed, for example when you want to detect a fire, sensors related to fires are needed such as temperature sensors, gas sensors and light sensors.

In Arduino, sensors are used as tools to obtain data around the desired processing. In this simulation, two types of sensors are used, namely water level and water flow sensors [11]. five cm in length. The water level sensor uses analog signals on the wemos to communicate data, the analog value of this sensor is 0-1024 where the higher the water that has been there, the smaller the resolution value of this sensor.

a. Sensor Water Flow

Water Flow Sensor is a sensor that functions to capture the speed of water flow, this sensor is shaped like a water tap. This sensor works on a digital signal, where each water flow value will be calculated through the interrupt function on the wemos. The working principle of this flow meter sensor is to measure water flow by calculating the rotation of the wheel in the water flow if there is water flowing through it, a motor is embedded in the wheel that has a magnet and when it rotates it will produce a magnetic field based on the hall effect principle. From the event there is a magnetic field and the absence of a magnetic field that is repeated will produce an output in the form of a wave, this signal is used to calculate the speed of water flow.

b. Sensor Water Level

Water level sensor is a sensor that functions to capture the value of the height of water in a container. The shape of this sensor is small, which is five cm in length. The water level sensor uses an analog signal on the wemos to communicate data, the analog value of this sensor is 0-1024 where the higher the water that has been there, the resolution of the value of this sensor is getting smaller.

2.4. K-Nearest Neighbour

KNN is one of the machine learning algorithms that works by finding a number of k patterns closest to the input pattern and then determining the class based on the most patterns among the k patterns [12]. The KNN training process produces a k value that provides the highest accuracy in generalizing future data. This k value training process is useful for seeing the k values and their results until the most optimal k value is produced.

KNN is widely used for data mining applications, statistical pattern recognition, image processing and many other things. The purpose of this algorithm is to classify new objects based on attributes and training samples. KNN is commonly used for predictions, predictions are used for the value of future results. The accuracy of the KNN algorithm is greatly influenced by the presence or absence of irrelevant features, or if the weight of the feature is not equivalent to its relevance to the classification.

The KNN algorithm has several advantages, namely it is not too affected by training data that has a lot of noise and is effective if the training data is large [13]. Meanwhile, the weakness of KNN is that KNN needs to determine the value of the K parameter (the number of nearest neighbors). The KNN Algorithm Sequence is as follows:

- Determine the parameter K (number of nearest neighbors)
- Calculate the square of the euclidean distance of each object to the given sample data
- Sort the objects into groups that have the smallest euclidean distance
- Collect the category Y (nearest neighbors classification)
- Output the results / voting based on the specified K value.

The output of the KNN algorithm will be in the form of a class of the searched data or the searched sample data and will be calculated based on the most data occurrences according to the previously set K value.

2.5. System

According to James o'Brien, a system is a collection of interconnected components, working together to achieve a goal by receiving input and then producing output in a regular transformation process [14]. It can be concluded that the system consists of input, output and processes that are interrelated with each other to achieve the goal of forming a system. Examples of systems used in life are information service systems, ticket booking systems, banking systems, academic information systems, and inventory systems. The purpose of creating a system is to overcome a problem that has been determined, then the system can also make it easier for its users to reach information.

2.6. Simulation

Simulation according to Shannon is the process of designing a model of a real system followed by conducting experiments to study the system or evaluate strategies. Then according to Emshoff and Shimun, simulation is a system model where the components are presented by arithmetic processes and algorithms run by a computer to estimate the dynamic properties of the system [15]. It can be concluded that simulation is a representation of real models through an experiment to estimate or model the real system. The simulation approach begins with the construction of a system model that already exists or approaches the real system, so that it can represent real components into a smaller form but still describes the behavior of the real system. Some reasons for conducting simulations are that real systems are difficult to observe directly, cost very high costs, take a long time, and will damage existing systems [16]. Examples of simulations that are commonly available are such as world globes, driving license (SIM) simulator systems, flight simulation systems.

3. METHODOLOGY

This research is an experimental research. In this study, a simulation experiment was conducted to examine the benefits and uses of a flood detection system for early warning processed using the KNN method. The research method used in this study is the waterfall method and the steps are as follows.

3.1. Data collection stage

The researcher has collected data on flood information and rainfall information. Data that has been collected from BPBD and BMKG Bangkalan Regency. The data collected are in the form of a Flood Incident Map in Bangkalan Regency, a disaster-prone map of Bangkalan Regency, a map of river basins in Bangkalan Regency, an evacuation location map of Bangkalan Regency, an elevation map of Bangkalan Regency, a flood planning report for Bangkalan Regency from BPBD, while from BMKG Malang City climate report data from 2018 to 2023 was obtained.

3.2. Analysis stage

At this stage, the researcher analyzes the data that has been collected and draws conclusions to be used in the design stage as training data. The data that is considered suitable for use and relevant in this case is data concerning the causes of flooding, namely as follows.

3.3. Design stage

At this stage, the researcher will prepare the equipment that will be needed and the researcher will design a system using the data that has been obtained previously. The equipment used in this study are two water flow sensors and one water level sensor. Then this sensor will function as follows. The water flow sensor will record the flow of water in and out, this functions to simulate river flow, the inflow of water will be recorded as a simulation of water entering the river, the outflow of water will be recorded as a simulation of water flowing out of the river, then the water level sensor will function to record the water level.

3.4. Testing stage

At this stage, researchers and managers will conduct testing on the system with an experimental method, namely direct testing of the system[17]. This testing stage has been simulated on the tools and systems, resulting in varied results, on the tool the incoming sensor data is recorded. Then the system carries out the prediction process from the data that has entered the system.

3.5. Evaluation stage

At this stage, researchers will evaluate the system and fix system errors that exist in previous tests. The following is a context diagram

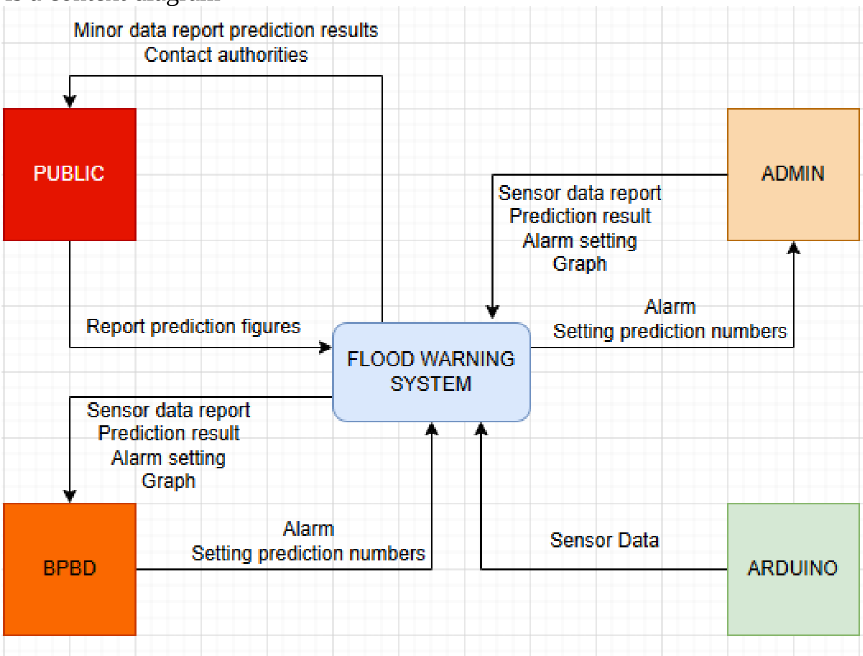

Figure 4. Diagram Context System

The following is a context diagram:

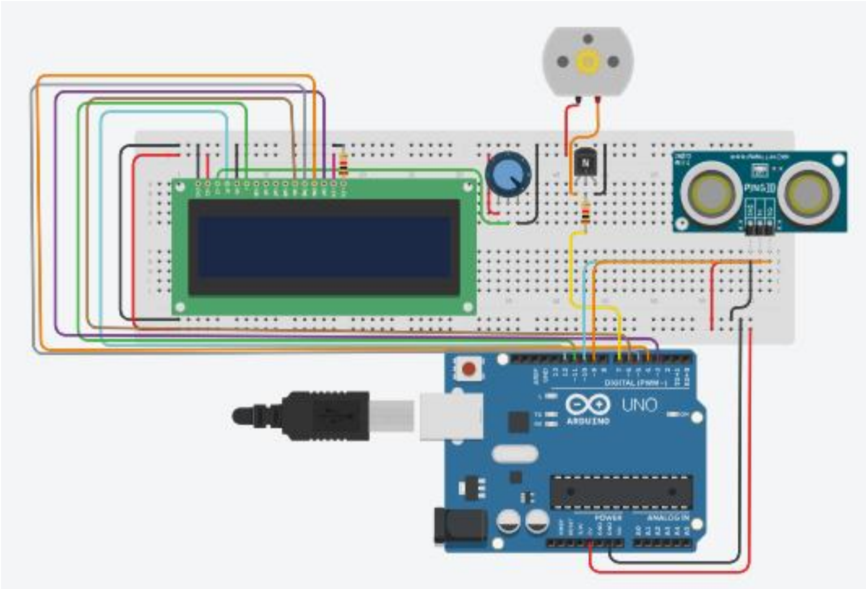

Figure 5. Arduino Circuit

In the circuit using Wemos D1 R2, two water flow sensors, one water level sensor, one green, yellow and red LED each, and a buzzer. The water level sensor in the circuit functions to capture water level data in the simulation container [18]. The water flow sensor in the circuit functions to capture the flow of water entering and leaving the container. The LED and buzzer in the circuit function to provide information regarding the status of the water level.

Implementation :

In this flood detection system, there is a server that is tasked with receiving data from Wemos, processing data to produce predictions, and displaying data[19]. An example of a request captured by the web server based on the apache access.log from Wemos:

192.168.43.36 - -
[03/Oct/2018:00:10:28 +0800] "GET
/arduino/add.php?water_flow_speed_in_=104.00&water_flow_speed_out_=0.00&water_height=266
HTTP/1.1"
200 - "-" "-"

The data sent by Wemos will then be processed by the add.php file on the web server. In add.php, the data is saved to the database. The data will be saved every five seconds.

4. RESULT AND DISCUSSION

4.1. Wemos Testing

Testing	Expected results	Observation result
Sending Wemos sensor data to the system	Data stored in database	Data stored in database

Figure 6. Wemos Test Table.

4.2 Algorithm Testing

KNN testing is done in two ways, namely the first is to find a group of k objects into the training data that are closest or closest or similar to the object to the testing data and test it with the Random Subsampling validation technique. The KNN algorithm test aims to test the prediction results in order to predict the status of the river. In the KNN algorithm, there are two types of data that will be used, namely the current weather data testing data and training data from the weather data that has been collected[20]. In this system, training data will be taken from data stored in the database, then testing data will be taken from user input.

In this prediction system, there are six variables that will be used in predicting floods, namely minimum temperature, maximum temperature, average temperature, and average humidity, rainfall, and duration of exposure. These six variables will be calculated with user input testing data. An example in this test will use the following data. As a method of data collection, researchers use weather data from 2018 to 2023 then the data is given a safe or flood status based on flood reports that occur. Then, noise cleaning is carried out on the data, such as unrecorded data or invalid data such as the numbers 9999 and 8888, and unused attributes are removed, leaving 53 data on the weather, the distance of which will then be calculated.

K Value	a	b	c	d	e	Accuracy
1						40%
3						20%
5						60%
7						40%
9						60%
11						60%
13						60%
15						60%
17						60%
19						60%

Figure 7. K accuracy table of each data

4.3 Simulation Results and System Testing

Simulation Results
Alert 4 At Initial Water Level, **Alert** 3 At 12 Seconds, **Alert** 2 At 30 Seconds
Alert 4 At Initial Water Level, **Alert 3** At 10 Seconds, **Alert 2** At 14 Seconds
Alert 4 At Initial Water Level, **Alert 3** At 7 Seconds, **Alert** 2 At 15 Seconds
Alert 4 at initial water level, **Alert 3** at 14 seconds

Figure 8. System Simulation Experiment Results

Two of the four events produce accurate results, but the other two events produce less accurate results, so it can be concluded that the accuracy of the algorithm and system is fifty percent (50%) and the results vary greatly depending on the available test data.

Then run the test with one of the white box methods to test the possible hypotheses in the system using the limit value method as shown in the following figure.

5. CONCLUSION

The conclusion reached by the system and simulation in this study is, this study aims to create a flood detection system and simulation for early warning using the KNN (K-Nearest Neighbor) method based on Arduino. This system was developed to help people living in flood-prone areas to take preventive measures before flooding occurs. This system uses a water level sensor to measure water levels, and a water flow sensor to measure the speed of water flow in and out of the river basin.

Rainfall, temperature, humidity, and other factors are used as variables in the application of the KNN algorithm to predict flood potential. Testing is carried out by comparing the results of the system prediction with actual flood event reports. The results of the study show that this system has a relatively low accuracy, which is around 60%, due to the lack of variation in the training data used. However, this study contributes to the development of a flood early warning system that can help people reduce the risk and impact of flood disasters.

REFERENCE

[1] T. J. Su, T. S. Pan, Y. L. Chang, S. S. Lin, and M. J. Hao, “A Hybrid Fuzzy and K-Nearest Neighbor Approach for Debris Flow Disaster Prevention,” *IEEE Access*, vol. 10, pp. 21787–21797, 2022, doi: 10.1109/ACCESS.2022.3152906.

[2] M. Khalaf *et al.*, “IoT-Enabled flood severity prediction via ensemble machine learning models,” *IEEE Access*, vol. 8, pp. 70375–70386, 2020, doi: 10.1109/ACCESS.2020.2986090.

[3] J. Ibarreche *et al.*, "Flash flood early warning system in colima, mexico," *Sensors (Switzerland)*, vol. 20, no. 18, pp. 1–26, 2020, doi: 10.3390/s20185231.

[4] P. Muñoz, J. Orellana-Alvear, J. Bendix, J. Feyen, and R. Célleri, "Flood early warning systems using machine learning techniques: The case of the tomebamba catchment at the southern Andes of Ecuador," *Hydrology*, vol. 8, no. 4, 2021, doi: 10.3390/hydrology8040183.

[5] T. A. Khan, Z. Shahid, M. Alam, M. M. Su'ud, and K. Kadir, "Early Flood Risk Assessment using Machine Learning: A Comparative study of SVM, Q-SVM, K-NN and LDA," *MACS 2019 - 13th Int. Conf. Math. Actuar. Sci. Comput. Sci. Stat. Proc.*, 2019, doi: 10.1109/MACS48846.2019.9024796.

[6] S. Liu, R. Liu, and N. Tan, "A spatial improved-knn-based flood inundation risk framework for urban tourism under two rainfall scenarios," *Sustain.*, vol. 13, no. 5, pp. 1–19, 2021, doi: 10.3390/su13052859.

[7] J. Ren, B. Ren, Q. Zhang, and X. Zheng, "A novel hybrid extreme learning machine approach improved by K nearest neighbor method and fireworks algorithm for flood forecasting in medium and small watershed of Loess region," *Water (Switzerland)*, vol. 11, no. 9, 2019, doi: 10.3390/w11091848.

[8] H. Lumbantobing, I. Ratna Avianti, K. Harisapto, and S. Suharjito, "Flood Prediction based on Weather Parameters in Jakarta using K-Nearest Neighbours Algorithm," *Eduvest - J. Univers. Stud.*, vol. 4, no. 6, pp. 5055–5065, 2024, doi: 10.59188/eduvest.v4i6.1339.

[9] S. Van Ackere, J. Verbeurgt, L. De Sloover, S. Gautama, A. De Wulf, and P. De Maeyer, "A review of the internet of floods: Near real-time detection of a flood event and its impact," *Water (Switzerland)*, vol. 11, no. 11, pp. 1–26, 2019, doi: 10.3390/w11112275.

[10] M. I. K. Alfahadiwy and A. Suliman, "Flood Detection using Sensor Network and Notification via SMS and Public Network," *Student Conf. Res. Dev. (SCOReD 2011)*, no. June, pp. 1–7, 2011.

[11] M. Esposito, L. Palma, A. Belli, L. Sabbatini, and P. Pierleoni, "Recent Advances in Internet of Things Solutions for Early Warning Systems: A Review," *Sensors*, vol. 22, no. 6, 2022, doi: 10.3390/s22062124.

[12] D. S. Rani, G. N. Jayalakshmi, and V. P. Baligar, "Low Cost IoT based Flood Monitoring System Using Machine Learning and Neural Networks: Flood Alerting and Rainfall Prediction," *2nd Int. Conf. Innov. Mech. Ind. Appl. ICIMIA 2020 - Conf. Proc.*, no. Icimia, pp. 261–267, 2020, doi: 10.1109/ICIMIA48430.2020.9074928.

[13] V. V. Krzhizhanovskaya *et al.*, "Flood early warning system: Design, implementation and computational modules," *Procedia Comput. Sci.*, vol. 4, pp. 106–115, 2011, doi: 10.1016/j.procs.2011.04.012.

[14] S. Sankaranarayanan, M. Prabhakar, S. Satish, P. Jain, A. Ramprasad, and A. Krishnan, "Flood prediction based on weather parameters using deep learning," *J. Water Clim. Chang.*, vol. 11, no. 4, pp. 1766–1783, 2020, doi: 10.2166/wcc.2019.321.

[15] W. Joko, "Flood Early Warning System Develop at Garang River Semarang using Information Technology base on SMS and Web," *Int. J. Geomatics Geosci.*, vol. 1, no. 1, pp. 14–28, 2010.

[16] Universitas Prima Indonesia. Fakultas Teknologi & Ilmu Komputer, Institute of Electrical and Electronics Engineers. Indonesia Section. CSS/RAS Joint Chapter, and Institute of Electrical and Electronics Engineers, "MECnIT 2020 : International Conference on Mechanical, Electronics, Computer, and Industrial Technology : June 25-26, 2020, Universitas Prima Indonesia, Medan, Indonesia," pp. 30–35, 2020.

[17] C. Chen, Q. Hui, W. Xie, S. Wan, Y. Zhou, and Q. Pei, "Convolutional Neural Networks for forecasting flood process in Internet-of-Things enabled smart city," *Comput. Networks*, vol. 186, p. 107744, 2021, doi: 10.1016/j.comnet.2020.107744.

[18] C. Moreno *et al.*, "Rivercore: IoT device for river water level monitoring over cellular communications," *Sensors (Switzerland)*, vol. 19, no. 1, 2019, doi: 10.3390/s19010127.

[19] I. Suwarno, A. Ma'arif, N. M. Raharja, A. Nurjanah, J. Ikhsan, and D. Mutiarin, "IoT-based Lava Flood Early Warning System with Rainfall Intensity Monitoring and Disaster Communication Technology," *Emerg. Sci. J.*, vol. 4, no. Special issue, pp. 154–166, 2020, doi: 10.28991/ESJ-2021-SP1-011.

[20] M. Anbarasan *et al.*, "Detection of flood disaster system based on IoT, big data and convolutional deep neural network," *Comput. Commun.*, vol. 150, no. November 2019, pp. 150–157, 2020, doi: 10.1016/j.comcom.2019.11.022.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 89 - 98
https://journal.physan.org/index.php/jocpes/index

Implementation of Web-based Realtime Monitoring System Using YOLOv8 for Green Box Detection and Automatic Capture in Navigation Missions at the Indonesian Boat Contest

Yobel Eliezer Mahardika[1], Tonny Wahyu Aji[2], Dimas Aditya Wiranata[3], Antonius Kukuh Prasetyaji[4]
[1,2,3,4]Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)

Article Info

Article history:

Received March 14, 2025
Revised March 19, 2025
Accepted March 20, 2025

Keywords:

Real-time monitoring
Web-based visualization
YOLOv8
Flask
MySQL
Object detection
OpenCV

A B S T R A C T

This research presents the design and development of a real-time object detection and monitoring system specifically aimed at identifying green box objects using the YOLOv8 model. The system integrates OpenCV for frame-by-frame video processing, MySQL for image storage as Binary Large Objects (BLOBs), and a Flask-based web interface for real-time visualization. Green box objects detected with a confidence score above 0.7 are cropped and stored in the database. A dynamic web interface, updated every 2 seconds using AJAX, enables real-time monitoring and allows users to download the latest detected image for further analysis. Experimental results demonstrate that the YOLOv8 model achieves high detection accuracy, as measured by precision, recall, and mean average precision (mAP). The proposed system effectively combines object detection, data storage, and web-based visualization to provide a robust and scalable solution for real-time monitoring. Tests conducted under real-world conditions confirm the system's efficiency and reliability. Future work may involve hardware acceleration via edge computing, support for multi-object detection, and integration of advanced tracking algorithms to broaden its applicability in autonomous systems and industrial automation.

Corresponden Author:

Yobel Eliezer Mahardika,
Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)
Tangerang City, Banten, Indonesia
Email: yobel.em69@gmail.com

1. INTRODUCTION

Real-time object detection is one of the important aspects in the development of navigation technology and autonomous systems [1], especially in a competition environment such as the Indonesian Ship Contest (KKI). In this competition, one of the missions that must be completed is to detect navigation markers in the form of green boxes above the water surface. This marker serves as a reference point for the ship to complete the trajectory according to the race rules. However, dynamic environmental conditions such as water ripples, light reflections, weather changes, and ship movements add challenges to the object detection process. Therefore, a detection system that is fast, accurate, and able to adapt to environmental changes is needed to assist the navigation team in completing the competition mission.

Deep learning technology, especially in the field of computer vision, has become an excellent solution in solving object detection problems with a high degree of accuracy [2]. One popular approach is the You Only Look Once (YOLO) algorithm, which is known for its ability to perform object detection in real time. YOLO algorithm has an advantage over traditional detection methods because it only requires one calculation to predict the location and class of objects in the image, resulting in much higher processing speed [3]. YOLOv8, as the latest iteration of YOLO, has seen significant improvements in terms of accuracy and computational efficiency. YOLOv8 can achieve mAP (mean Average Precision) up to 50-95% on various datasets with faster inference time than its predecessor [4].

The application of YOLO has been tested in various contexts, including object detection in maritime environments. YOLOv5 can detect floating objects on the sea surface with more than 90% accuracy [5], even under varying lighting conditions. YOLO has good adaptability in dynamic environments, making it an ideal choice for applications in the field of autonomous ship navigation [6]. In this study, YOLOv8 was chosen because it is capable of processing video data in real time with a speed rate of up to 30 FPS (frames per second) and a higher level of accuracy [7].

In addition to the detection aspect, this research also includes an automatic shooting system that uses OpenCV to capture photos when a green box is detected. The detected images are then stored in a MySQL database in BLOB (Binary Large Object) format. This format was chosen because of its ability to store binary data such as images in their original size without any loss of quality [8]. Efficient and structured data storage in the database allows the system to quickly access the stored images again [9].

Furthermore, to facilitate the technical team in monitoring the detection results during the competition, the system is designed using a web-based monitoring system built with the Flask framework. Flask is a lightweight and flexible Python backend framework, enabling the development of web-based monitoring systems with AI integration in a simple yet effective manner [10]. The system is equipped with an automatic refresh feature using AJAX every 2 seconds, so that the technical team can view the green box detection results and object photos in rea ltime through the browser. In addition, an Image Download button is provided to allow users to download the latest images as documentary evidence or for further analysis.

This research aims to develop a fast, accurate, and accessible object detection system through a Flask-based web interface to support the completion of navigation missions in the Indonesian Ship Contest. The system is expected to improve monitoring efficiency and provide flexibility for the control team in accessing and analyzing detection data. By utilizing YOLOv8 technology and a web-based monitoring system, this research provides an innovative solution that can be applied not only in the context of competition, but also for the development of autonomous navigation systems in the future.

2. LITERATURE REVIEW

Real-time object detection systems have become a major topic in the development of computer vision and machine learning-based technologies. One of the most effective methods in solving this problem is You Only Look Once (YOLO) [11]. YOLO is known for its ability to perform object detection in a single computation step, unlike previous methods such as Region-based CNN (R-CNN) and Fast R-CNN that require multiple computation steps [12]. This approach makes YOLO much faster while maintaining high accuracy. For example YOLOv2 can process up to 40 FPS (frames per second) with sufficient detection accuracy [13].

As technology evolves, the latest version of YOLO, YOLOv8, provides significant improvements in accuracy and computational efficiency. YOLOv8 integrates a lighter architecture and more optimized processing algorithms, enabling the model to achieve higher mAP50-95 on various benchmark datasets [14]. YOLO's implementation in maritime applications, where YOLOv5 successfully detected floating objects on the sea surface with an accuracy rate of 92%, even under varying environmental conditions [15]. These advantages make YOLOv8 an ideal solution for object detection systems in Indonesian Ship Contest (KKI) missions, where speed and accuracy are critical in detecting green boxes as navigation markers.

In addition to the detection aspect, this research also utilizes OpenCV as a supporting library to capture video and take photos automatically when the green box is detected. OpenCV has been widely used in video processing applications due to its flexibility in handling real-time image input [16]. The integration of OpenCV with deep learning algorithms can accelerate the decision-making process in visual-based systems, such as in robotics and automatic navigation [1].

For image data storage, this research relies on MySQL with the use of BLOB (Binary Large Object) format. The BLOB format was chosen because of its ability to store image data in original size without losing quality [17]. Storing data in BLOB allows structured management of images in the database, making data access faster and more efficient [18]. This is relevant to the needs of web-based monitoring systems, where image data must be accessible and updated in a short time.

In presenting the detection results, this system uses Flask as a Python-based backend framework. Flask is widely used in web application development due to its lightweight and flexible nature [10]. Flask facilitates integration between machine learning models and user interfaces by providing efficient API endpoints for presenting data [19]. Flask also supports AJAX (Asynchronous JavaScript and XML) for dynamic data updates on the web interface, making it an ideal solution for web-based monitoring systems [20]. Web-based monitoring systems have proven effective in various studies. Web-based monitoring provides flexibility for users to access data from various browser-enabled devices [21].

Building on prior studies, the integration of YOLOv8, OpenCV, MySQL, and Flask shows strong potential for addressing real-time object detection challenges with efficient data handling and presentation [19][22]. This research develops a system capable of detecting green boxes, capturing images automatically, storing them in a MySQL database, and displaying them via a web-based interface with auto-refresh. Its main contribution is the implementation of an end-to-end solution to support ship navigation missions in dynamic scenarios such as the Indonesian Ship Contest.

3. SYSTEM DESIGN AND IMPLEMENTATION

Before entering the core system of monitoring and photographing this green box, we must first build the YOLOv8 model development which is carried out in stages to ensure optimal performance in detecting the desired object. In this yolov8 development, we adjust to the challenges and missions given to the 2024 Indonesian Ship Contest (KKI), which is to detect green boxes.

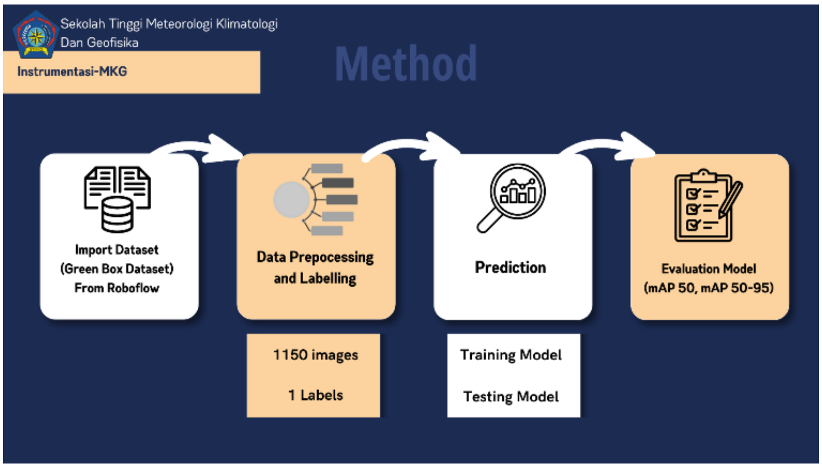

Fig. 1. The Method Steps to Detect the Green Box

The process starts with importing a dataset containing images labeled as green boxes. This dataset consists of 1,150 images with bounding box annotations created using the Roboflow platform. Next, data preprocessing is performed to improve the quality of the dataset, including data augmentation such as flipping, scaling, and normalization. After preprocessing is complete, model configuration is performed, where parameters such as input size (640x640), batch size, learning rate, and number of epochs are set according to the needs of the system. The model is then trained using the training dataset through a forward pass and backward pass process, where the model weights are updated based on the loss function value.

During training, the model is validated on the validation dataset to monitor performance metrics such as loss, precision, recall, and mAP (mean Average Precision). If the model performance does not meet the criteria, the configuration or training process is repeated until the model reaches optimal performance [3]. After the training is complete, the model is evaluated on the test dataset to calculate the final metrics, such as mAP50 and mAP50-95. The results of this evaluation are used to confirm the model's ability to detect objects with high accuracy [23].

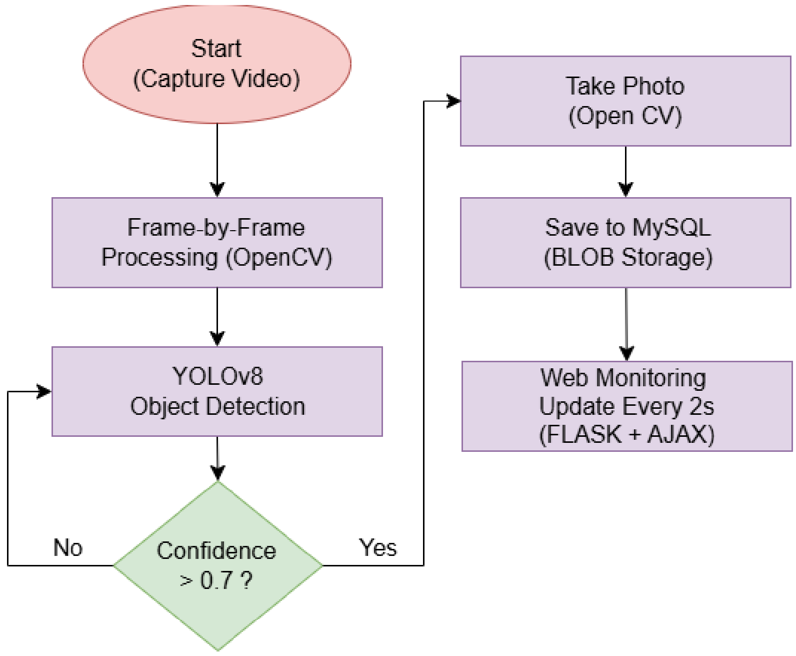

Fig. 2. Flowchart of real time detection system using YOLOv8 and web-based monitoring

The implementation of the real-time detection system using the YOLOv8 model involves several key components that are connected sequentially. The process starts with video capture using a camera connected to the system. The video input is captured in a frame-by-frame format using the OpenCV library. Each generated frame is then sent to the YOLOv8 model for object detection. The model detects the presence of a green box object and provides output in the form of a bounding box and confidence score [24].

In the decision stage, the system checks the confidence score of the detection. If the confidence score is greater than 0.7, the system proceeds to the take-photo process, where the part of the image containing the detected object is cropped based on the bounding box coordinates using OpenCV. If the confidence score does not meet the threshold, the image is ignored and the system moves on to the next image. The captured image is then stored in a MySQL database in Binary Large Object (BLOB) format. The database is designed as a table with id, filename, photo, and timestamp columns. Storing images in BLOB format allows for efficient and organized image management [25].

The final stage of this system is web-based monitoring using Flask as the backend and AJAX to update data in real time. Flask provides an endpoint to retrieve the latest images from the database, while AJAX is used to send periodic requests every 2 seconds. The detected images are displayed on the web interface, allowing users to monitor the results in realtime.

4. RESULT AND DISCUSSION

In this study, we conducted a series of trials to evaluate the performance and functionality of the YOLOv8-based real-time detection system. The primary objective was to assess the system's capability to accurately detect green box objects, process video input in real time, and store valid detections into a MySQL database. The trials were carried out using a continuous video stream, where each frame was processed and analyzed to identify objects meeting the defined confidence threshold of 0.7. Detected images were automatically cropped, stored, and displayed through a web-based interface, which updated every 2 seconds. The results from these trials provide insights into the reliability and efficiency of the proposed system in a dynamic environment.

4.1. Confusion Matrix Analysis

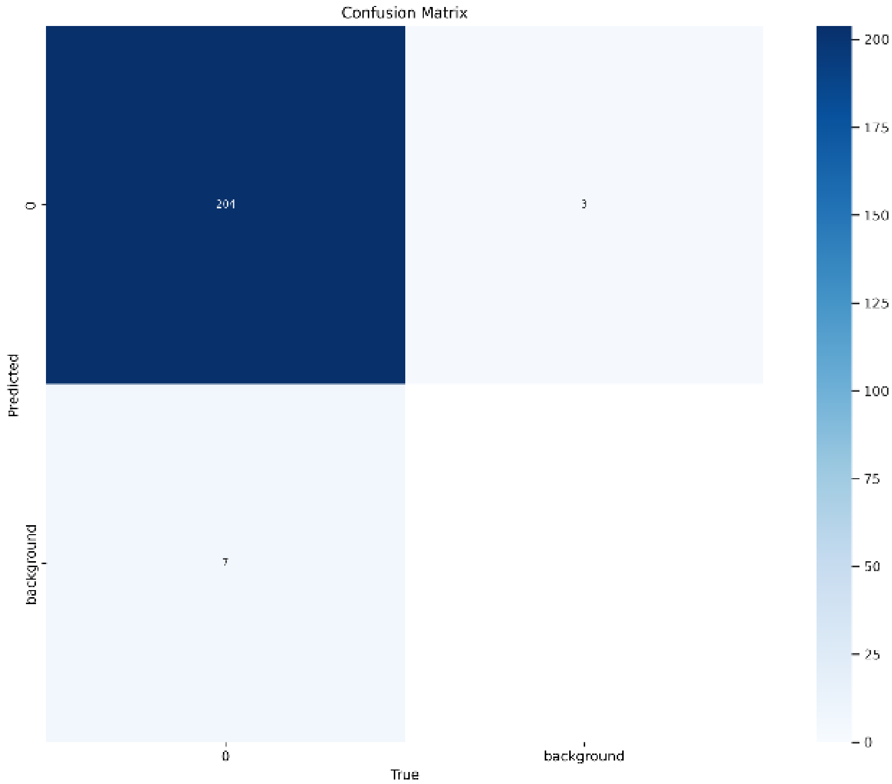

Fig. 3. Matrix performance of the YOLOv8-based real-time detection system in identifying green box objects

The confusion matrix presented in Fig. 3 evaluates the performance of the YOLOv8-based real-time detection system in identifying green box objects. The matrix shows that the system successfully detected 204 true positives (TP), which indicates that the target objects were correctly identified as belonging to the green box class. However, there are 3 false positives (FP), where background elements were misclassified as green boxes. Additionally, 7 false negatives (FN) were observed, representing instances where the system failed to detect green box objects even though they were present in the input frames. These false negatives could be caused by occlusions, low contrast between the target object and the background, or confidence scores below the defined threshold of 0.7. The overall distribution of the matrix highlights the system's high accuracy, with a significant dominance of true positives compared to false predictions. The low false positive and false negative values suggest that the YOLOv8 model performs reliably for real-time detection tasks. Further analysis of precision, recall, and F1-score will provide a more quantitative measure of the system's effectiveness in handling dynamic input data.

4.2. Training and Validation Performance of YOLOv8 Model

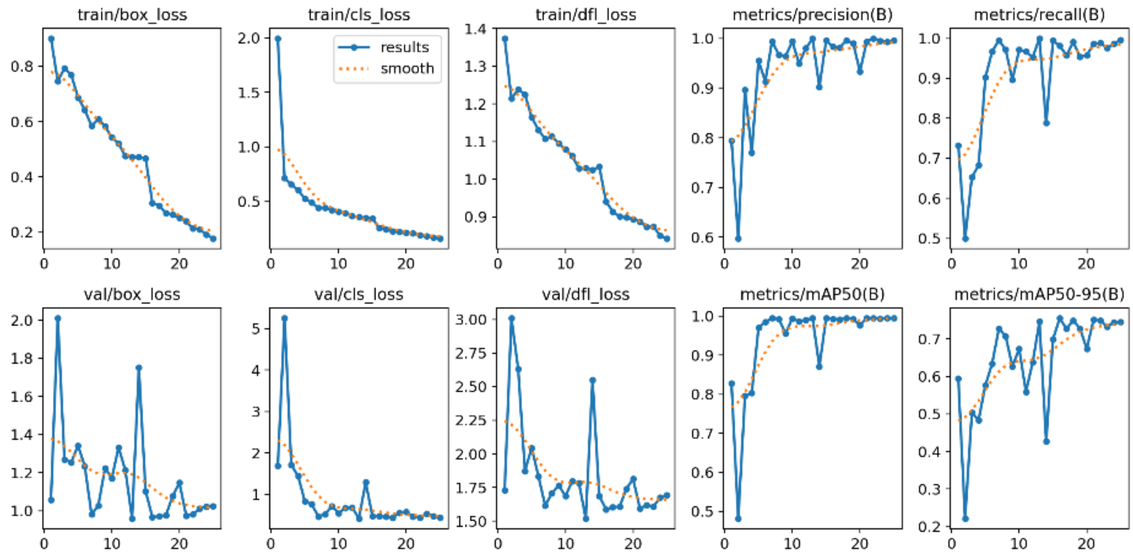

Fig. 4. Matrix performance of the YOLOv8-based real-time detection system in identifying green box objects

The training and validation performance of the YOLOv8 model is illustrated in Fig. 4, which highlights the progression of loss values and evaluation metrics over 25 epochs. The train/box_loss, train/cls_loss, and train/dfl_loss curves show a consistent decline as the number of epochs increases. This trend indicates that the model is learning effectively, as the loss values—representing errors in bounding box predictions, classification, and distribution focal loss—steadily decrease. The training loss curves demonstrate smooth convergence, signifying that the model adapts well to the training dataset.

In the validation phase, the val/box_loss, val/cls_loss, and val/dfl_loss exhibit some fluctuations, particularly in the earlier epochs, which is expected due to variations in the validation dataset. Despite these minor variations, the overall downward trend in validation loss suggests that the model generalizes well and avoids significant overfitting, as indicated by the minimal gap between training and validation losses.

The evaluation metrics provide further insights into the model's performance. The precision and recall metrics converge towards values close to 1.0, reflecting the model's ability to make accurate predictions with minimal false positives and false negatives. Additionally, the mAP50 metric, which measures the mean average precision at a 50% IoU threshold, approaches 1.0, demonstrating excellent detection performance. Meanwhile, the mAP50-95, which evaluates the model's performance across a range of IoU thresholds, stabilizes around 0.7-0.75. This result confirms that the model maintains robust accuracy under varying detection conditions.

In summary, the steady reduction in loss values, combined with the high precision, recall, and mAP scores, indicates that the YOLOv8 model successfully learns to detect green box objects with high accuracy. The small gap between training and validation losses further confirms the model's ability to generalize to unseen data, making it reliable for real-time object detection tasks.

4.3. Detection Results Visualization

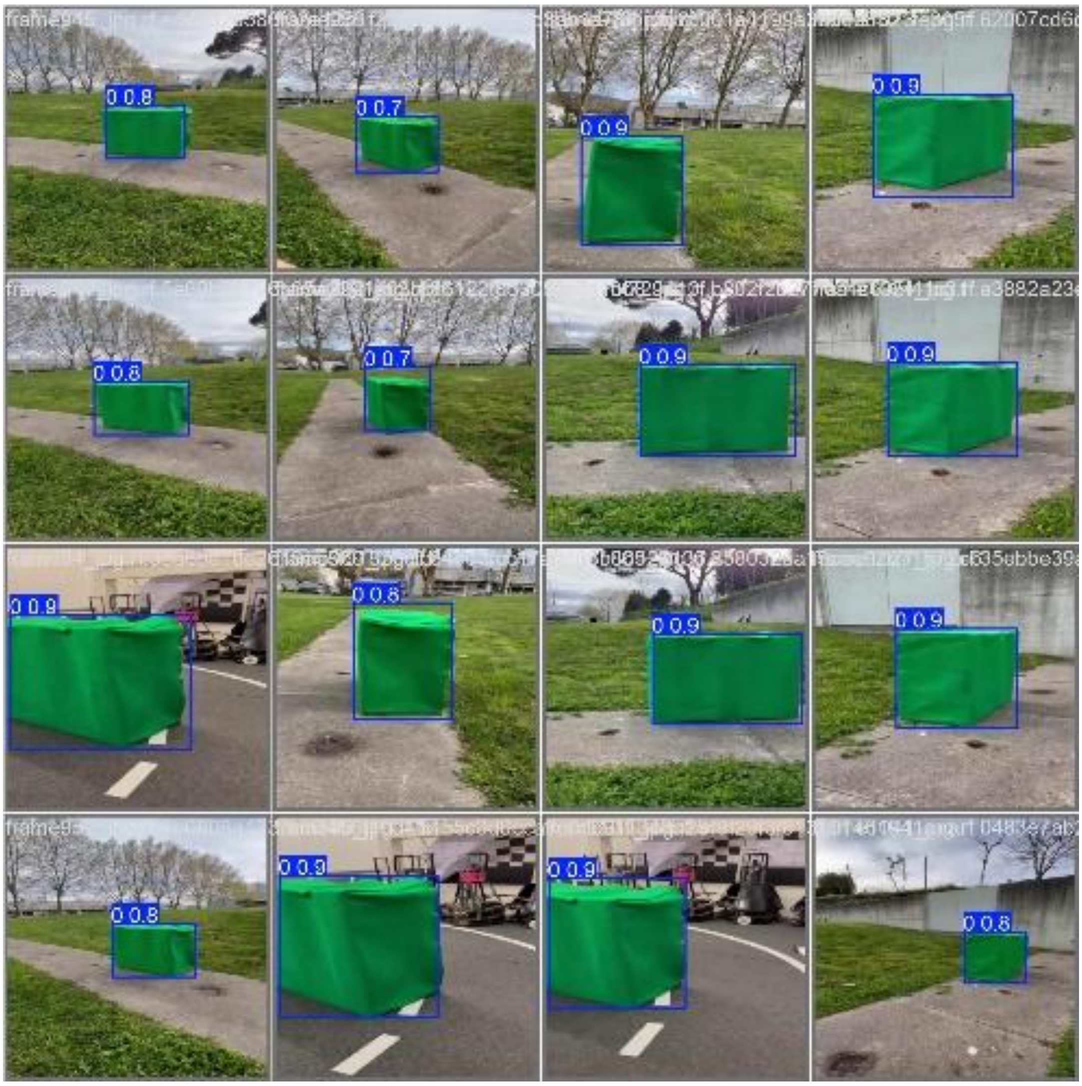

Fig. 5. Visualization of YOLOv8 Detection Results on Green Box Objects (Image-Based Testing)

The visualization of the YOLOv8 detection results demonstrates the system's ability to accurately identify and localize green box objects in image-based testing scenarios. As shown in Fig. 5, the YOLOv8 model successfully detected green box objects across multiple images, displaying confidence scores ranging from 0.7 to 0.9. Each detected object is annotated with a bounding box and its corresponding confidence score, which reflects the model's prediction accuracy. The testing dataset includes various conditions such as changes in angle, lighting, and background, simulating real-world scenarios to evaluate the robustness of the model. Despite these variations, the YOLOv8 model maintains consistent performance, with bounding boxes accurately placed around the green box objects. This indicates the model's reliability in detecting objects with high confidence while minimizing false positives.

The results confirm that YOLOv8 effectively handles object localization and confidence assessment in static image datasets. Such capabilities are essential for real-time applications, as they ensure the detection system can provide accurate and dependable results under varying environmental conditions.

4.4. Live Camera Detection Test

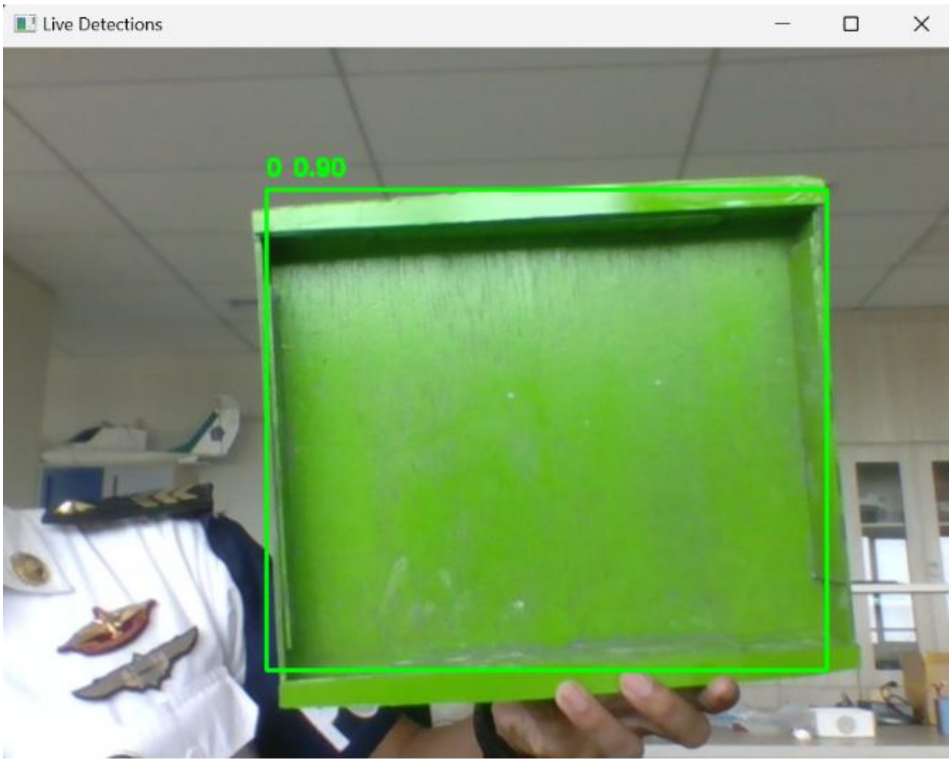

Fig. 6. Live Camera Detection Results Using YOLOv8

The live camera test, as shown in Fig. 6, demonstrates the real-time performance of the YOLOv8-based detection system in identifying green box objects. In this test, video frames were continuously captured using a live camera feed and processed in real time to detect target objects. The system successfully detected the green box in the frame with a confidence score of 0.90, as displayed on top of the bounding box. The bounding box is accurately aligned with the green box object, confirming the model's robustness in handling real-time video input. Unlike the image-based test, the live camera test introduces additional challenges such as motion blur, varying lighting conditions, and dynamic background elements. Despite these factors, the system consistently maintains high detection accuracy and rapid processing speed, enabling smooth real-time performance.

This result validates the system's capability to operate effectively in real-time scenarios, making it suitable for applications requiring live object detection and tracking. The successful detection in the live camera feed further highlights the integration of the YOLOv8 model with OpenCV for frame-by-frame processing and object localization.

4.5. Integrated System Testing

The integrated system testing was conducted to evaluate the performance of the end-to-end solution, combining YOLOv8, OpenCV, and a web-based monitoring system. The system encompasses real-time detection of green boxes using a webcam, saving detected images in BLOB format to a MySQL database, and displaying the latest detected image on a web interface. At this stage, the video captured in real-time is processed by YOLOv8 to detect green box objects. If the confidence score of the detection exceeds 0.7, the system uses OpenCV to capture the frame, crop the image based on the bounding box coordinates, and save it to the MySQL database in the photo column as a BLOB. The most recent image stored in the database is then retrieved and displayed on the web interface.

The figure below demonstrates the final output of the web interface, where the most recent green box detection is displayed, along with a button to download the image.

Fig. 7. Web Interface Display of the Latest Detected Image

This figure showcases the user interface for web monitoring, where the most recently detected green box image is displayed in real-time. The system updates the image every 2 seconds using an AJAX mechanism, ensuring a responsive and dynamic interface. Additionally, the Download Image button allows users to retrieve the displayed image in .png format directly from the MySQL database.

5. CONCLUSION

This paper presents a real-time green box detection and monitoring system developed using the YOLOv8 object detection model, OpenCV for image processing, MySQL for image storage, and Flask for the web-based monitoring interface. The system integrates these components into a unified pipeline, enabling real-time detection, efficient data management, and dynamic visualization of the detected images.

The results demonstrate that the YOLOv8 model performs accurate object detection in real-time, with a confidence threshold of 0.7 ensuring reliability during testing. Detected images were cropped based on bounding box coordinates and successfully stored in MySQL as Binary Large Object (BLOB) data. The database architecture allows for organized image management and retrieval. The web interface, designed using Flask and updated using AJAX every two seconds, provides a dynamic and responsive platform to display the latest detected image and enables users to download the displayed image seamlessly.

The system successfully meets the objectives of detecting, storing, and displaying green box objects in real-time. Experimental results confirm the robustness of the integration and the system's capability to handle live video feeds while maintaining smooth data flow between components. This makes the system suitable for applications in automated monitoring tasks, such as autonomous ship competitions, where identifying green box objects is part of mission-critical operations.

Further enhancements can focus on improving the system's processing efficiency by leveraging edge computing hardware such as NVIDIA Jetson or Raspberry Pi for real-time deployments. Additionally, incorporating multi-object detection capabilities and implementing tracking algorithms can enhance the system’s versatility in various real-world scenarios. The web interface can also be expanded to include live video streaming, historical data visualization, and advanced analytics for improved monitoring and decision-making.

Overall, this work provides a practical and scalable solution for real-time object detection and monitoring, with potential applications in robotics, autonomous systems, and industrial automation. The findings and implementation serve as a foundation for further research and development in similar fields

REFERENCE

[1] N. Adiuku, N. P. Avdelidis, G. Tang, and A. Plastropoulos, “Advancements in Learning-Based

Navigation Systems for Robotic Applications in MRO Hangar: Review," *Sensors*, vol. 24, no. 5, pp. 1–22, 2024.
[2] N. Chitraningrum *et al.*, "Comparison Study of Corn Leaf Disease Detection based on Deep Learning YOLO-v5 and YOLO-v8," *J. Eng. Technol. Sci.*, vol. 56, no. 1, pp. 61–70, 2024.
[3] Y. Yin, H. Li, and W. Fu, "Faster-YOLO: An accurate and faster object detection method," *Digit. Signal Process. A Rev. J.*, vol. 102, p. 102756, 2020.
[4] M. R. A. P. and Suharjito, "Comparison of YOLOv8 and EfficientDet4 Algorithms in Detecting the Ripeness of Oil Palm Fruit Bunch," *2023 10th Int. Conf. ICT Smart Soc.*, 2023.
[5] F. Chen *et al.*, "Soft-NMS-Enabled YOLOv5 with SIOU for Small Water Surface Floater Detection in UAV-Captured Images," *Sustain.*, vol. 15, no. 14, pp. 1–18, 2023.
[6] Y. Hovde, Ø. S. Odden, M. Aanning, and M. Holt, "Graph-based slam and navigation in a simulated environment and buoy classification using YOLO for an Autonomous Surface Vessel," no. May, 2020.
[7] C. Fang and X. Yang, "Lightweight YOLOv8 for Wheat Head Detection," *IEEE Access*, vol. 12, no. April, pp. 66214–66222, 2024.
[8] S. Steinhoff, "Evaluation of MySQL 8.0 Spatial Data Features," *Researchgate.Net*, no. July, 2019.
[9] B. Szabó-Takács, T. Takács, and A. Roček, "MeteoRead: Client Software for Inserting Observed Atmospheric Data into MySQL_{TM} Database and Downloading them into Different File Format," *J. Softw. Eng. Appl.*, vol. 14, no. 10, pp. 563–574, 2021.
[10] D. Ghimire, "Comparative study on Python web frameworks: Flask and Django," *Metrop. Univ. Appl. Sci.*, no. May, pp. 13–33, 2020.
[11] J. Terven, D. M. Córdova-Esparza, and J. A. Romero-González, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Mach. Learn. Knowl. Extr.*, vol. 5, no. 4, pp. 1680–1716, 2023.
[12] M. M. Mijwil, K. Aggarwal, R. Doshi, K. K. Hiran, and M. Gök, "The Distinction between R-CNN and Fast R-CNN in Image Analysis: A Performance Comparison," *Asian J. Appl. Sci.*, vol. 10, no. 5, 2022.
[13] J. Redmon and A. Farhadi, "Yolo9000," *Cvpr2017, Xplore*, no. April, pp. 187–213, 2017.
[14] X. Zhou, L. Jiang, C. Hu, S. Lei, T. Zhang, and X. Mou, "YOLO-SASE: An Improved YOLO Algorithm for the Small Targets in Complex Backgrounds," *Sensors*, vol. 22, no. 12, pp. 1–14, 2022.
[15] M. Zhu and Y. He, "Department of Natural Resources of Sichuan Province , Chengdu , Sichuan , PRC 610072 Sichuan Research Institute for Eco-system Restoration & Geo-disaster Prevention , Chengdu , Sichuan ," pp. 1647–1650, 2020.
[16] R. T. H. Hasan and A. B. Sallow, "Face Detection and Recognition Using OpenCV," *J. Soft Comput. Data Min.*, vol. 2, no. 2, pp. 86–97, 2021.
[17] K. T. Ahmed, H. Afzal, M. R. Mufti, A. Mehmood, and G. S. Choi, "Deep Image Sensing and Retrieval Using Suppression, Scale Spacing and Division, Interpolation and Spatial Color Coordinates with Bag of Words for Large and Complex Datasets," *IEEE Access*, vol. 8, pp. 90351–90379, 2020.
[18] S. K. Mainali and W. Zhang, "(12) United States Patent," vol. 2, 2021.
[19] M. Grinberg, *Flask web development*. O'Reilly Media, Inc, 2018.
[20] D. Verma, "A Comparison of Web Framework Efficiency–Performance and network analysis of modern webframeworks," *Bachelor's Thesis Turku Univ. Appl. Sci. Bachelor Eng. Inf. Commun. Technol.*, p. 115, 2022.
[21] R. Nugraha, "Radiosonde System Using ESP32 And LoRa Ra-02 Web-Based for Upper-Air Profile Observation," pp. 2–6.
[22] "Магістерська дисертація," 2024.
[23] A. N. Azhar and M. L. Khodra, "Fine-tuning Pretrained Multilingual BERT Model for Indonesian Aspect-based Sentiment Analysis," *2020 7th Int. Conf. Adv. Informatics Concepts, Theory Appl. ICAICTA 2020*, pp. 2980–2988, 2020.
[24] M. Monika, U. Rajender, A. Tamizhselvi, and A. S. Rumale, "Real-Time Object Detection in Videos Using Deep Learning Models," *Ictact J. Image Video Process.*, no. December, p. 2, 2023.
[25] V. LAZAROVA, "Working With Large Objects in Application Development," *Rev. Econ.*, vol. 75, no. 2, pp. 98–104, 2023.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 99 - 106
https://journal.physan.org/index.php/jocpes/index

Design and build an IoT system for temperature and humidity observation using DHT 22 with ThingSpeak

Fachrul Rizky Syaputra[1]
[1]Undergraduate Program in Applied Instrumentation Meteorology, Climatology Geophysics (STMKG)

Article Info

Article history:

Received March 15, 2025
Revised March 20, 2025
Accepted March 21, 2025

Keywords:

Temperature
Humidity
IoT
DHT22

A B S T R A C T

In a study conducted in recent years, it was found that atmospheric conditions such as temperature and humidity values greatly affect the climatic conditions of a region. This real-time temperature and humidity monitoring system is very useful and can be used in various regions. By using IoT, monitoring can be done in real-time. Temperature and humidity can use DHT 22 linked to ESP 32 and the data is stored and displayed in ThingSpeak. The trial of the monitoring tool was carried out for more than 31 hours and obtained 6920 data. At the time of the experiment the monitoring system worked well. At the time of observation for temperature the maximum value is 32.4℃, the minimum value is 28.1℃, the mean is 29.75621387283237℃, and the standard deviation is 3.587911994653761 and for humidity the maximum value is 81.4%, the minimum value is 63.6%, the mean is 72.6355404624 and the standard deviation is 3.5879112. From the experiment, it was also found that the temperature value was inversely proportional to the humidity and when it was going to rain the temperature value went up while the humidity went down and when it rained the temperature went up while the temperature went down.

Corresponden Author:

Fachrul Rizky Syaputra
Undergraduate Program in Applied Instrumentation Meteorology, Climatology Geophysic (STMKG)
Tangerang, Indonesia
Email: fachrul2105@gmail.com

1. INTRODUCTION

The ongoing global warming has been clearly identified. The data says that the global average temperature has increased by 0.85 °C between 1880 and 2012. Not only that, the frequency of heat waves that reach the earth's surface increases on various continents. If no action is taken, it is estimated that the global average temperature will increase by 1.6 to 2.6 °C by 2050 with the European continent detailing 2℃ and the Americas 4℃. while on the Asian continent and Australia it is estimated to be higher [1].

Future climate predictions include an increase in temperature and a decrease in humidity on land, but these two factors remain constant in the oceans. This raises many questions as to why this can happen. Therefore, climate change prediction is very important because this will greatly affect life. In a study conducted in recent years, it was found that atmospheric conditions such as temperature and humidity values greatly affect the climatic conditions of a region. So the observation of these two components is important for the prediction model used. So that it can be used as an explanation for the conflict of differences in the influence of different climate changes on land and sea [2].

Temperature and humidity are one of the most important and interrelated elements. Not only for climate observation but also used in the industrial and factory worlds [3]. This can be exemplified in the food industry where temperature and humidity values are very important for the quality of the food produced [4]. In the industrial world, automation and smart infrastructure, real-time monitoring of temperature and humidity values is an important aspect [5]. This real-time monitoring system is very useful and can be used in various regions. This is very efficient because there is no need to come directly to the observation point. In addition, it can also be applied to marine monitoring, room server monitoring, room temperature control and others [6].

For real-time and remote monitoring, sensors are needed to be connected to the internet. Temperature is often applied with IoT systems [7]. ESP32 merupakan salah satu mikrokontroler yang sangat populer dalam pengembangan sistem IoT karena memiliki Wi-Fi terintegrasi dan dukungan terhadap berbagai sensor lingkungan, seperti pada sistem pemantauan kualitas udara berbasis IoT yang mampu mendeteksi PM2.5, PM10, CO, NO_2, SO_2, dan O_3 secara real-time [8]. The sensor can be connected with a microcontroller to connect to the internet, software and network connectivity tools [8]. In addition, by using a microcontroller, various sensors can be added for better observation [9]. By adding other sensors integrated with IoT, the system can be more widely used in various fields such as agriculture, weather prediction, smart homes, banking, retail, and health [10]. Indicators that are usually added are sunlight and wind speed. All of these devices are also often integrated with IoT [11].

Some of the devices used for making Internet of Things (IOT) applications are Raspberry Pi and Arduino [12]. In addition, ESP is also often used in the creation of IoT applications. Because the price is cheap and does not require additional equipment because there is already Wi-fi available to transmit data read by the sensor [13]. The data from the sensor will be read by the microcontroller and sent to the computer or to the cloud via a Wi-Fi connection [14]. For IoT, many open source-based standard platforms can be used such as Cayenne, Blynk, ThinkSpeak, DeviceHive, Thinger.io, and so on [15]. Clouds like Thingspeak can also be used for visualization [16]. Using this standard platform, observations can be made at locations different from where measurements are carried out [17].

Temperature and humidity readings can be made using DHT 22 and DHT 11 sensors [18]. DHT22 is a sensor with a digital signal [3]. The DHT22 uses capacitive sensors and thermistors to detect temperature and humidity. The sensor operates on a Direct Current of 3.3–6 Volts and and has a measurement range of 0 - 100% for humidity as well as (-40°C) - 80°C. The difference in temperature and humidity values on the DHT22 sensor in the datasheet is 2%–5% for humidity and ±5°C for temperature [19].

2. RESEARCH METHOD

In this study, two stages are used in the design of the system, namely designing the hardware of the device and the network architecture of the device. To design a good tool must use good theory and start with the design of the tool [20]. And to find out the quality of the tool, the data obtained must be processed and analyzed

2.1. Designing and Building Hardware

The method in this experiment is to use a sensor connected to a microcontroller [21].

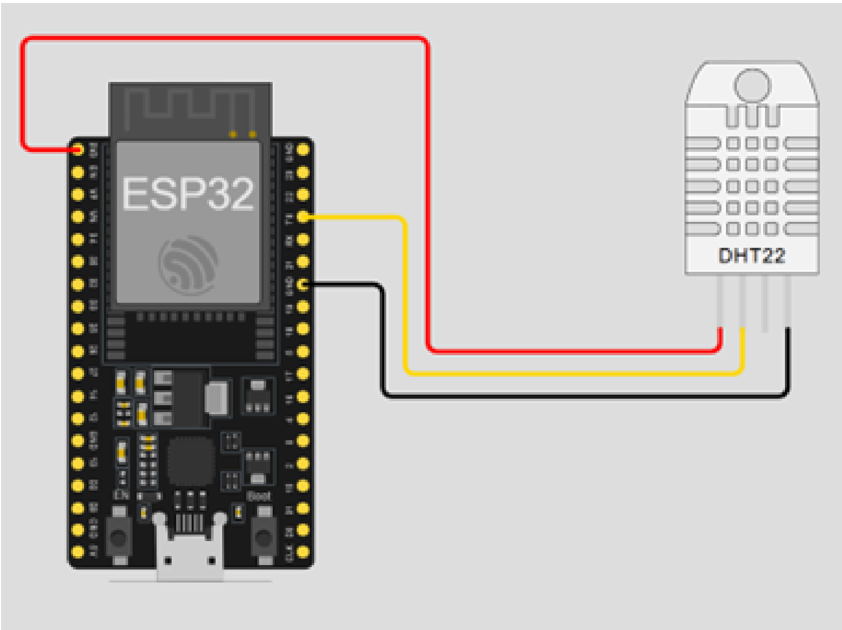

Fig. 1. Wiring Diagram

In this device, ESP must be connected to a voltage source that can be obtained from a laptop, Power Bank or charger. For starters, it must be connected to a laptop via a USB cable. This aims to send code from the Arduino IDE software into ESP 32. After that, the VCC pin of the sensor is connected to the 3.3 V ESP32 pin, the sensor GND pin to the ESP32 GND, and the sensor data pin to the GPIO17 ESP32 pin.

2.2. Network Architecture

In this study, ThingSpeak is used which is a free web service to collect and store sensor data in the cloud and develop IoT applications. ThingSpeak is widely used in IoT applications [22]. In ThingSpeak, it will be set to display 2 results by creating 2 fields and named temperature and humidity.

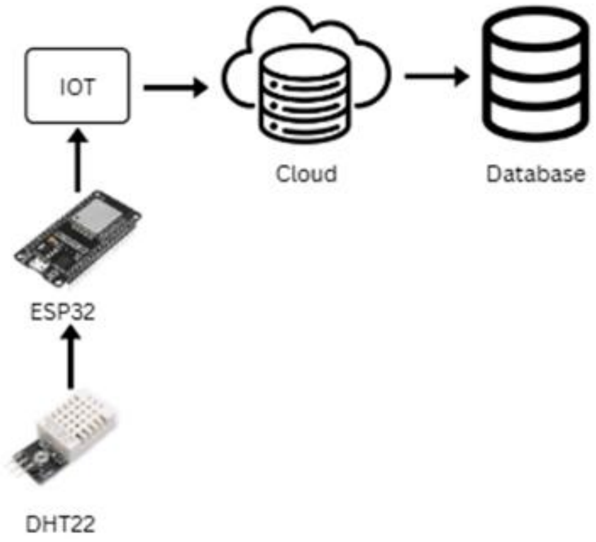

Fig. 2. System block diagram

In this system, the value measured by the DHT22 sensor will be sent to the ESP 32 where the ESP 32 that is already connected to Wi-Fi will send the data received by the sensor to the cloud and then stored in the database from ThingSpeak. With this mechanism, the measurement value from the sensor will be able to be observed from a distant place as long as the device is still in good condition and connected to electricity and Wi-Fi.

2.3. Data Collect

Data collection was carried out in the Tanah Tinggi area, Tangerang City. Data collection for device testing was carried out for approximately 31 hours. Measurement data will be stored in the database of ThingSpeak. Data will be sent to ThingSpeak every 15 seconds. The data will be downloaded in CSV format on the ThingSpeak platform.

2.4. Data Processing

After downloading the observation data, it will be processed using the Jupyter Notebook. The data to be calculated include maximum indigo, minimum value, mean, and standard deviation from the measurement results. Then the data found will also be displayed in the form of graphs to facilitate the analysis of the data. The formulas used include
Mean

$$\bar{x} = \sum_{i=1}^{n} x_i \qquad (1)$$

Standard deviation

$$\sigma = \sqrt{\frac{\sum_i^n (x_i - \bar{x})^2}{N-1}} \qquad (2)$$

3. RESULT

In its application the system with DHT 22 and ESP 32 is connected by jumpers according to the planned wiring.

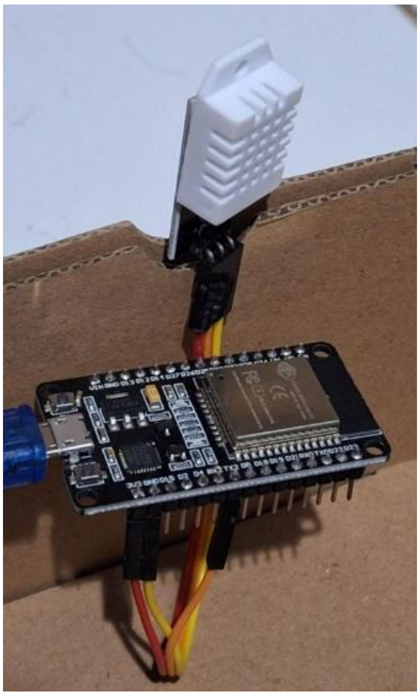

Fig. 3. Device Hardware Display

Here is a display from ThingSpeak that displays the temperature and humidity measurements from the sensor. The data is displayed in the form of a graph when temperature or humidity is relative to time.

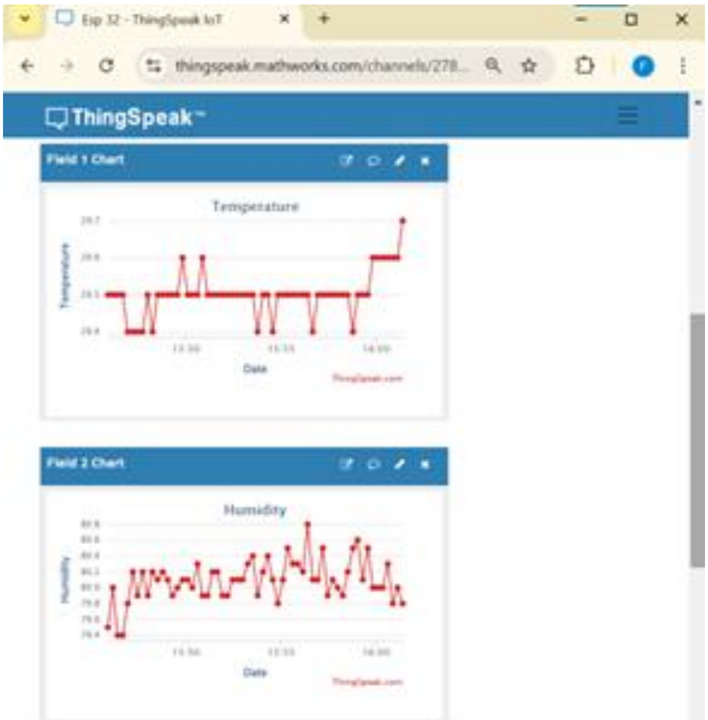

Fig. 4. ThingSpeak display

From the observation time carried out for more than 31 hours, 6920 temperature and humidity data were obtained. The following is shown some of the data in various conditions, namely the first time, during the day, when the weather is hot, when it is cold at night and when it rains.

Time	Temperature	Humidity
2024-12-14T08:44:39+07:00	29.8	74.4
2024-12-14T08:44:54+07:00	29.8	73.4
2024-12-14T08:45:10+07:00	29.8	73.4
2024-12-14T08:45:26+07:00	29.8	73.1
2024-12-14T08:45:41+07:00	29.8	73.6
Time	Temperature	Humidity
2024-12-14T12:14:38+07:00	32.1	66.2
2024-12-14T12:14:54+07:00	32.1	67.2
2024-12-14T12:15:09+07:00	32.1	67
2024-12-14T12:15:25+07:00	32.1	66.1
2024-12-14T12:15:41+07:00	32	66.3
Time	Temperature	Humidity
2024-12-15T03:07:12+07:00	28.6	73.5
2024-12-15T03:07:28+07:00	28.6	73.6
2024-12-15T03:07:44+07:00	28.6	73.7
2024-12-15T03:08:00+07:00	28.6	73.7
2024-12-15T03:08:15+07:00	28.6	73.6
Time	Temperature	Humidity
2024-12-15T15:56:40+07:00	29.4	80.1
2024-12-15T15:56:56+07:00	29.5	80.1
2024-12-15T15:57:12+07:00	29.5	80.5
2024-12-15T15:57:27+07:00	29.5	79.9
2024-12-15T15:57:43+07:00	29.5	80.1

Fig. 5. Some data from the observation results

The data obtained from the observation results are then calculated and observed. For temperature, the maximum value is 32.4℃, the minimum value is 28.1℃, the mean is 29.75621387283237℃, and the standard deviation is 3.587911994653761. For humidity, the maximum value is 81.4%, the minimum value is 63.6%, the mean is 72.6355404624 and the standard deviation is 3.5879112.

Here is a graph display of temperature and humidity values during observations made.

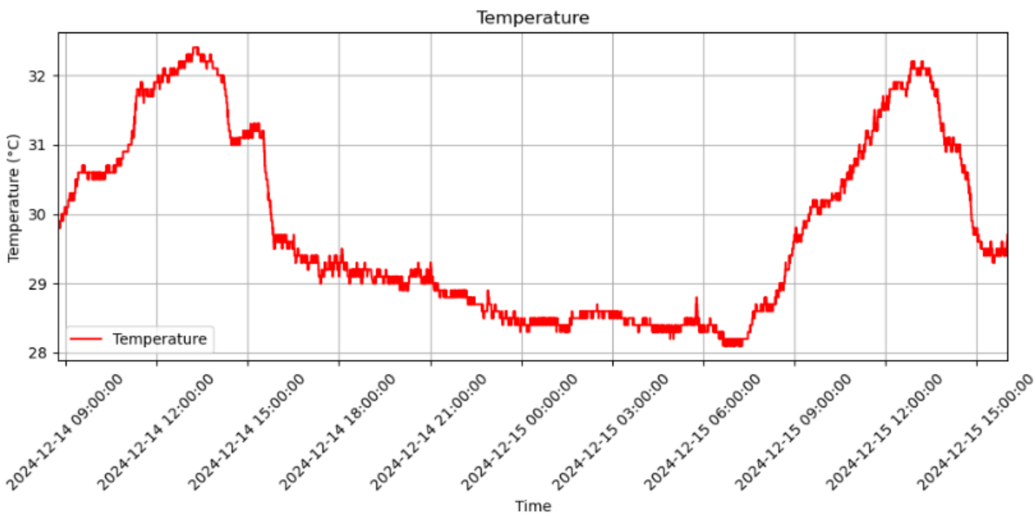

Fig. 6. Temperature graph during observation

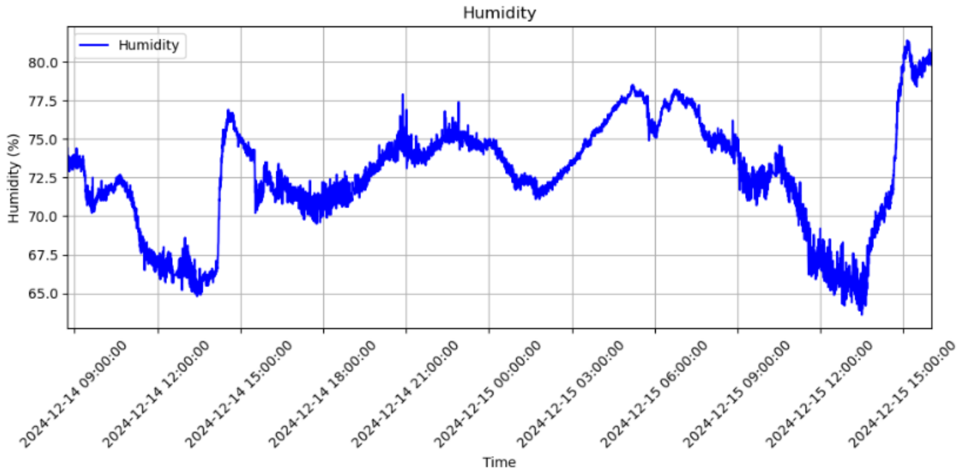

Fig. 7. Humidity graph during observation

4. DISCUSSION

From the hardware that was carried out during the observation experiment process with the system, there were no problems, which means that everything could run well. The ThingSpeak display is also easily accessible from a variety of devices and is easy to understand. These two things are important aspects in the observation process carried out. Observations were also made in various environmental conditions such as heat, cold and rain. This shows that this monitoring system can work well in all circumstances. The data obtained from ThingSpeak found that there was little data lost, this was due to a network connection that had been lost due to interference from the operator. And it can be seen that data is sometimes sent every 16 seconds, this is caused by several factors such as delays in reading and sending data, network connections, and from the specifications of sensors that take data about once every 2 seconds.

From the results of the calculation, it was found that the standard deviation from the temperature was 3.587911994653761 and the humidity was 3.5879112. This can be said to be quite large which proves the many variations in values that occur during observation. This is due to observations made in various environmental conditions, namely heat, cold and rain. Which conditions greatly affect the value of temperature and humidity. This is what causes a lot of variation in sensor data. But if it is in stable conditions such as at night, the data condition tends to be stable and this has a low standard deviation value. From this, it can be proven that this monitoring tool is good enough in measuring temperature and humidity values.

Based on the observation results that can be seen from the graph, a pattern of temperature and humidity values can be seen. The maximum temperature is 32.4°C and occurs during the day and drops to 28.1°C at night. Meanwhile, humidity tends to be lower during the day and increases when the temperature drops at night. However, on December 15, 2024, there was the highest peak in humidity, this was influenced because at that time there was rain, which of course increased the humidity value. The maximum value of humidity is 81.4 %, and the minimum value is 63.6 %. From the graph and environmental conditions, there are two things that can be observed, namely the value of temperature and humidity is inversely proportional, while when it rains the temperature increases and the humidity decreases and when it rains the temperature drops and the humidity rises.

5. CONCLUSION

From the results of the experiment conducted for 31 hours on December 14-15, 2024, the results were obtained that the monitoring tools made can run well and can operate in various environmental conditions. Some errors occur due to external factors, namely the network quality of the internet operator. The data that is displayed is easy to understand and access by anyone. At the time of observation for temperature the maximum value is 32.4℃, the minimum value is 28.1℃, the mean is 29.75621387283237℃, and the standard deviation is 3.587911994653761 and for humidity the maximum value is 81.4%, the minimum value is 63.6%, the mean is 72.6355404624 and the standard deviation is 3.5879112. The standard deviation value is quite high due to the environmental conditions that are quite diverse during observation. From the experiment, it was also found that the temperature value was inversely proportional to the humidity and when it was going to rain the temperature value went up while the humidity went down and when it rained the temperature went up while the temperature went down.

REFERENCE

[1] M. Sanderson, K. Arbuthnott, S. Kovats, S. Hajat, dan P. Falloon, *The use of climate information to estimate future mortality from high ambient temperature: A systematic literature review*, vol. 12, no. 7. 2017.

[2] M. P. Byrne dan P. A. O'Gorman, "Trends in continental temperature and humidity directly linked to ocean warming," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 115, no. 19, hal. 4863–4868, 2018, doi: 10.1073/pnas.1722312115.

[3] Azhari, T. I. Nasution, S. H. Sinaga, dan Sudiati, "Design of Monitoring System Temperature And Humidity Using DHT22 Sensor and NRF24L01 Based on Arduino," *J. Phys. Conf. Ser.*, vol. 2421, no. 1, 2023, doi: 10.1088/1742-6596/2421/1/012018.

[4] M. T. Seman, M. N. Abdullah, dan M. K. Ishak, "Monitoring temperature, humidity and controlling system in industrial fixed room storage based on IOT," *J. Eng. Sci. Technol.*, vol. 15, no. 6, hal. 3588–3600, 2020.

[5] O. V. Medagedara dan M. H. Liyanage, "Development of an IoT-based Real-Time Temperature and Humidity Monitoring System for Factory Electrical Panel Rooms," *Eng. J. Inst. Eng. Sri Lanka*, vol. 57, no. 1, hal. 21–30, 2024, doi: 10.4038/engineer.v57i1.7636.

[6] D. Yulizar, S. Soekirno, N. Ananda, M. A. Prabowo, I. F. P. Perdana, dan D. Aofany, *Performance Analysis Comparison of DHT11, DHT22 and DS18B20 as Temperature Measurement*, vol. 1. Atlantis Press International BV, 2023.

[7] Y. A. Ahmad, T. Surya Gunawan, H. Mansor, B. A. Hamida, A. Fikri Hishamudin, dan F. Arifin, "On the Evaluation of DHT22 Temperature Sensor for IoT Application," 2021, doi: 10.1109/ICCCE50029.2021.9467147.

[8] M. Sinambela et al., "IoT-Based Air Quality Monitoring System Design and Development Using ESP32," 2024 Ninth International Conference on Informatics and Computing (ICIC), Medan, Indonesia, 2024, pp. 1-6, doi: 10.1109/ICIC64337.2024.10956737.

[9] D. Srivastava, A. Kesarwani, dan S. Dubey, "Measurement of Temperature and Humidity by using Arduino Tool and DHT11," *Int. Res. J. Eng. Technol.*, vol. 05, no. 12, hal. 876–878, 2018, [Daring]. Tersedia pada: www.irjet.net.

[10] F. Hafeez, U. U. Sheikh, A. Khidrani, M. A. Bhayo, S. M. Abdallah Altbawi, dan T. A. Jumani, "Distant temperature and humidity monitoring: Prediction and measurement," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 24, no. 3, hal. 1405–1413, 2021, doi: 10.11591/ijeecs.v24.i3.pp1405-1413.

[11] I. K. Wardani *et al.*, "The feasibility study: Accuracy and precision of DHT 22 in measuring the temperature and humidity in the greenhouse," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 1230, no. 1, 2023, doi: 10.1088/1755-1315/1230/1/012146.

[12] M. Alvan Prastoyo Utomo, A. Aziz, Winarno, dan B. Harjito, "Server Room Temperature & Humidity Monitoring Based on Internet of Thing (IoT)," *J. Phys. Conf. Ser.*, vol. 1306, no. 1, 2019, doi: 10.1088/1742-6596/1306/1/012030.

[13] R. Nugraha *et al.*, "Radiosonde System Using ESP32 And LoRa Ra-02 Web-Based for Upper-Air Profile Observation," *2024 Int. Conf. Inf. Technol. Comput.*, hal. 2–6, 2024.

[14] H. Subastiyan, W. Sunanda, dan R. F. Gusa, "A Prototype of Monitoring Temperature and Humidity on Photovoltaic Using ESP8266," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 520, no. 1, 2020, doi: 10.1088/1755-1315/520/1/012010.

[15] A. Najmurrokhman, Kusnandar, A. Daelami, E. Nurlina, U. Komarudin, dan H. Ridhatama, "Development of Temperature and Humidity Control System in Internet-of-Things based Oyster Mushroom Cultivation," *2020 3rd Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2020*, hal. 551–555, 2020, doi: 10.1109/ISRITI51436.2020.9315426.

[16] Munawar et al., "The Carbon Dioxide Filtration System Using Chlorella Pyrenoidosa Microalgae IoT-based for Air Quality Improvement," *2024 Ninth International Conference on Informatics and Computing (ICIC)*, Medan, Indonesia, 2024, pp. 1-5, doi: 10.1109/ICIC64337.2024.10957176.

[17] R. A. Rahman, U. R. Hashim, dan S. Ahmad, "IoT based temperature and humidity monitoring framework," *Bull. Electr. Eng. Informatics*, 2020, doi: 10.11591/eei.v9i1.1557.

[18] D. Tazakka Ma'arij, A. Yudhana, D. T. Ma'arij, dan A. Yudhana, "Temperature and Humidity Monitoring System in Internet of Things-based Solar Dryer Dome," *Bul. Ilm. Sarj. Tek. Elektro*, vol. 5, no. 3, hal. 323–335, 2023, doi: 10.12928/biste.v5i3.8633.

[19] F. Hendajani, A. Mughni, I. P. Wardhani, dan A. Hakim, "Modeling Automatic Room Temperature and Humidity Monitoring System with Fan Control on the Internet of Things," *ComTech Comput.*

Math. Eng. Appl., vol. 13, no. 2, hal. 75–85, 2022, doi: 10.21512/comtech.v13i2.7433.

[20] I. Riadi dan R. Syaefudin, "Monitoring and Control Food Temperature and Humidity using Internet of Things Based-on Microcontroller," *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 7, no. 1, hal. 108, 2021, doi: 10.26555/jiteki.v7i1.20213.

[21] F. Puspasari, I. Fahrurrozi, U. Y. Oktiawati, dan T. P. Satya, "Development of embedded system in monitoring temperature and humidity as supporting smart farm," *J. Phys. Conf. Ser.*, vol. 1511, no. 1, 2020, doi: 10.1088/1742-6596/1511/1/012017.

[22] T. H. Nasution, M. A. Muchtar, S. Seniman, dan I. Siregar, "Monitoring temperature and humidity of server room using Lattepanda and ThingSpeak," *J. Phys. Conf. Ser.*, vol. 1235, no. 1, 2019, doi: 10.1088/1742-6596/1235/1/012068.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 107 - 112
https://journal.physan.org/index.php/jocpes/index

Impact of Urban Heat Island on Shifting Weather Patterns in Sidoarjo and Malang: BMKG Daily Data-Based Analysis

Assajdah Abdullah[1], Muhammad Zaki Albukhori[2], Krisna Dewi[3], Wyra Steven Hutagalung[4], Muhammad Yazid Ilmany[5], Syahrajad Al Syarqiyah[6], Zulfitrah Mokoginta[7]

[1,2,6]Undergraduate Program in Applied Meteorology (STMKG)
[3]Undergraduate Program in Applied Climatology (STMKG)
[4]Undergraduate Program in Applied Geophysics (STMKG)
[5,7]Undergraduate Program in Applied of Instrumentation Meteorology, Climatology, and Geophysics (STMKG)

Article Info

Article history:

Received March 16, 2025
Revised March 21, 2025
Accepted March 22, 2025

Keywords:

Urban Heat Island
Temperature
Weather
Sustainable urban

ABSTRACT

The Urban Heat Island (UHI) phenomenon is an increase in surface temperature in urban areas compared to the surrounding, more rural areas. This phenomenon can affect shifts in weather patterns, including an increase in average temperature, changes in rainfall patterns, and an increase in extreme weather events. This study aims to analyse the impact of UHI on shifting weather patterns in Indonesia using BMKG daily climate data. The method used includes a quantitative approach withtime series analysis to understand the trend of changes in air temperature and rainfall in Sidoarjo and Malang. The results showed a correlation between UHI intensity and changes in weather patterns,where areas with high UHI intensity experienced a significant increase in temperature and changes in rainfall patterns. This research is expected to provide insights for more sustainable urban planning in the face of UHI impacts in Indonesia.

Corresponden Author:

Muhammad Zaki Albukhori,
Undergraduate Program in Applied of Instrumentation Meteorology, Climatology Geophysics (STMKG)
Tangerang City, Banten, Indonesia
Email: zakialbukhor56@gmail.com

1. INTRODUCTION

Urban Heat Island (UHI) is a phenomenon that occurs due to urbanisation that causes an increase in surface temperature in urban areas compared to surrounding areas. In general, UHI refers to an increase in air temperature, but UHI can also refer to the relative heat of a surface or material above it. UHI inadvertently increases local climate change due to atmospheric and surface modifications in urbanised areas. However, UHI has no direct effect on global warming because the occupation of a city constitutes only a small fraction of the Earth's entire surface. UHI has important implications for human health and comfort, air pollution, energy balance, and urban planning [1]. UHI is a phenomenon or occurrence of increased air temperature that occurs in urban areas compared to rural areas that reach temperatures of 3-10°C [2] .

Urban heat islands (UHIs) exacerbate the risk of heat-related mortality associated with global climate change. The intensity of UHIs varies with population size and mean annual precipitation, but a unifying explanation for this variation is lacking, and there are no geographically targeted guidelines for heat mitigation. The urban heat island (UHI) effect is defined as the temperature difference between an urban centre and the outlying regions. It has some additional characteristics: for instance, considerable research has indicated that the UHI effect appears more frequently under calm and weak wind conditions with a cloudless sky at night, and that its intensity is affected by daytime and nocturnal circulations [3].

The causes of UHI include land cover change, the use of materials with high heat absorption, and anthropogenic activities that generate additional heat. The impact of this phenomenon not only affects the microenvironment, but can also cause regional changes in weather patterns. Sidoarjo City is located at 112°43′00″ east longitude and 7°27′00″ south latitude. Sidoarjo City is one of the sub-districts in Sidoarjo Regency, East Java. The city has an area of about 56 km² with a total population of approximately 210,507 people in 2013. Sidoarjo city is located about 22 km south of Surabaya city. Topographically, Sidoarjo City has a relatively flat topography with an average altitude of about 3-5 metres above sea level. The region is a part of the north coast of East Java which is characterised by flat and fertile Land.

Malang City is located at 112°37′12″ East Longitude (East) and 7°58′48″ South Latitude (South). Malang City is at an altitude between 440-667 metres above sea level. The city is surrounded by mountains, such as Mount Arjuno to the north, Mount Semeru to the east, Mount Kawi and Panderman to the west, and Mount Kelud to the south. With an area of approximately 145.28 km², Malang City has a population of around 820,243 people (based on 2014 data), to the highest recorded 932,127 people in mid-September 2021, showing an increase of around 10% compared to the previous year of 844,933 people. This indicates an increase in the phenomenon of urbanisation involving land transformation from open areas to urban areas, exacerbating the UHI effect.

In Indonesia, population increase and urban expansion have accelerated the occurrence of UHI. Temperature changes due to UHI can affect rainfall patterns, humidity, and increase the frequency of extreme weather events. Therefore, this study aims to analyse the UHI phenomenon in Indonesia and its influence on shifting weather patterns by utilising BMKG daily climate data based on time series analysis, which is periodic data collected over time, to describe the development of a region. Periodic data analysis allows us to know the development of one or several events and their relationship/influence on other events [4]. The pattern of data movement or variable values can be followed or recognised by the presence of periodic data, so periodic data can be used as a basis for:

1. Analyse rainfall and average temperature.
2. Combating UHI by conducting sustainable urban planning.

2. RESEARCH METHOD

This study uses a quantitative approach with a time series analysis method to understand the trend of changes in air temperature and rainfall in Sidoarjo City and Malang City. The data used are secondary data from Meteorology, Climatology and Geophysics Agency (BMKG), namely Juanda Meteorological Station for Sidoarjo City and Malang Geophysical Station for Malang City. The data analysed includes daily average air temperature (in °C) and daily rainfall (in mm) for the period 2015 to 2024.

The analysis was conducted by calculating the average air temperature and average rainfall in January in 2015 and 2024 in each study area. The average results of each parameter in both years were then compared to obtain the difference in temperature and rainfall changes. This difference is used as the basis for analysing the microclimate changes allegedly caused by the Urban Heat Island (UHI) phenomenon in the study area.

In addition, this study also includes observations of literature studies to see the research patterns that have developed regarding urban heat islands. Literature studies start from journals that have existed for the past 10 years or more due to the limitations of existing papers.

3. RESULT AND DISCUSSION

This research began with processing BMKG daily climate data through the website dataonline.bmkg.go.id which obtained several results that became the basis for analysing UHI. Based on BMKG daily climate data, Juanda Meteorological Station in 2015 the average monthly temperature at Juanda Meteorological Station was 27.55 ° C. In 2024, the average increased to 28.40 ° C. with an increase in average temperature in 9 years of about 0.85 ° C. Then the daily climate data of BMKG, Malang Geophysical Station shows the average monthly temperature in 2015 was 26.06°C and in 2024 the average temperature increased to 28.10°C with an average temperature increase of about 2.04°C in the last 9 years. Image 1 and 2 below show graphs of temperature changes in Sidoarjo and Malang over a period of 9 years.

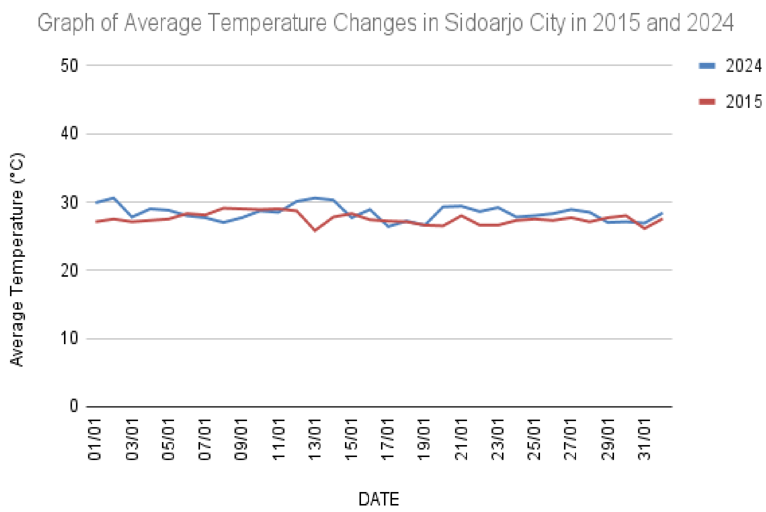

Fig. 1. Graph of Average Temperature Change in Sidoarjo City in 2015 and 2024

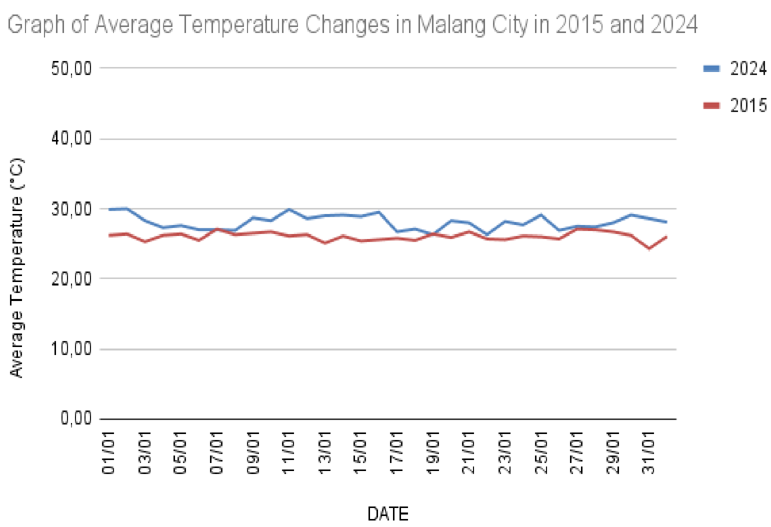

Fig. 2. Graph of Average Temperature Change in Malang City in 2015 and 2024.

After observing the average temperature of each region, the results are obtained as follows,
1. Temperature increases at both sites indicate a localised warming trend.
2. the temperature increase in Malang was greater than Juanda.

The increase in temperature in both locations, with the elevation of Malang city at 285 masl which usually has a lower temperature than lowland areas such as Sidoarjo Juanda which is at an elevation of three masl, with a significant increase in temperature in Malang compared to Juanda could be due to the rapidly increasing urbanisation in Malang, as the 2021 census data increased by about 10% from the previous year, so can be concluded. While Juanda (Sidoarjo) has been a metropolitan area with a more mature infrastructure for a long time, the change in average temperature at Juanda is not significant in Malang but is still affected by the effects of the UHI.

After observation through an increase in average temperature, observation of daily rainfall parameters as a method of understanding the trend of changes in air temperature and rainfall in Sidoarjo City and Malang can be done with time series analysis from BMKG stations by processing average rainfall data in 2015 and 2024. Figure 3 and Figure 4 below show a diagram of changes in average rainfall in Sidoarjo City and Malang in the period 2015-2024.

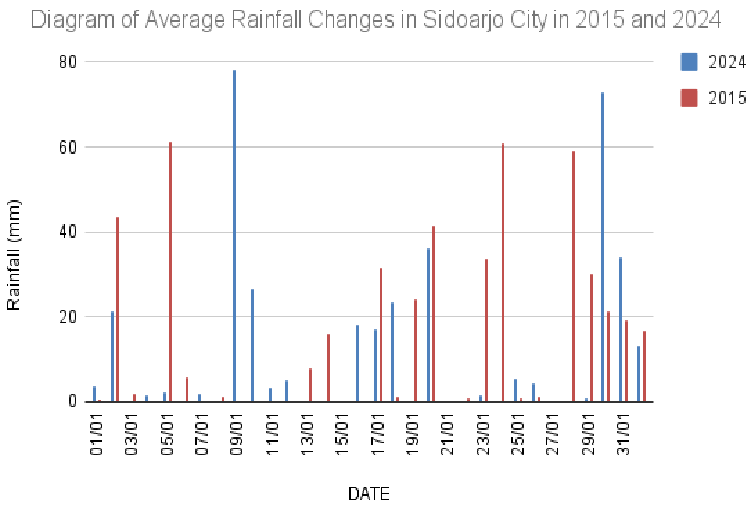

Fig. 3. Diagram of Changes in Average Rainfall in Sidoarjo City in 2015 and 2024

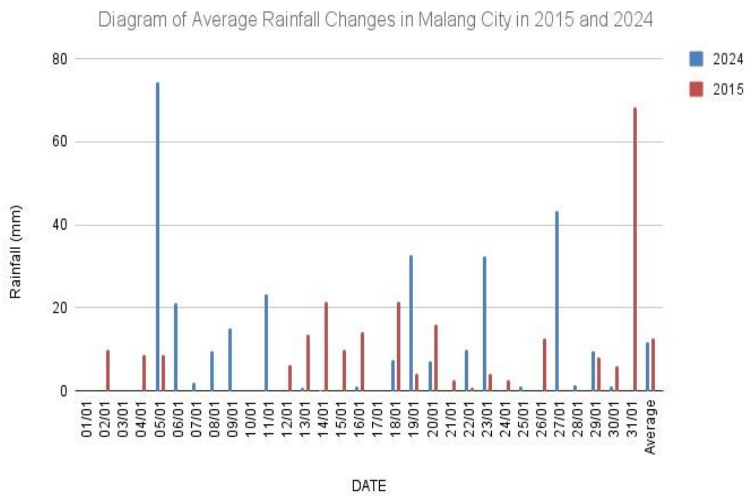

Fig. 4. Diagram of Changes in Average Rainfall of Malang City in 2015 and 2024

From the Diagram of Changes in Average Rainfall in Sidoarjo City and Malang in 2015 and 2024, it can be seen that in the observation data of Juanda meteorological station, in 2015 the average monthly rainfall was 16.60 mm and experienced a decrease in rainfall to 13.28 mm, with a decrease in a period of 9 years

decreased by 3.32 mm. At the Malang geophysical station the average rainfall observation in 2015 was 12.59 mm, also in 2024 there was a decrease in rainfall by 0.83 mm to 11.76 mm.
From the analysis of observations of changes in average rainfall in Sidoarjo and Malang, it was found that,

1. There was a decrease in rainfall at both observation sites.
2. Juanda has a more significant decrease in rainfall than Malang.

The occurrence of decreased rainfall in both locations can be attributed to the UHI effect which causes the land surface to become warmer, so that water evaporates faster and local precipitation patterns change, as temperature increases, the evaporation process increases, but relative humidity decreases, which can reduce rainfall potential. Then a more significant decrease in rainfall at Juanda compared to Malang, this can occur because Juanda which has a lower elevation, namely, 3 masl compared to Malang which is at 285 masl, which makes Juanda more vulnerable to the effects of global warming and weather changes. Temperature changes in Sidoarjo and Malang may also be influenced by regional factors such as global climate change. However, the local effect of UHI appears to be more dominant in Malang, which can be seen from the rapidly growing urbanisation factor in Malang city.

Provide a statement that what is expected, as stated in the 'Introduction' chapter can ultimately produce the 'Results and Discussion' chapter, so that there is congruence. In addition, it can also be added to the prospect of developing research results and the application of prospects for further studies (based on the results and discussion).

4. CONCLUSION

The UHI phenomenon is a challenge for governments, with rapid urbanisation exacerbating the phenomenon through increased population density, the use of heat absorbing infrastructure and the conversion of green spaces. By structuring infrastructure development and open land management to suppress the effects of UHI, which are reflected in increased temperatures and changes in rainfall patterns, the government can carry out mitigation strategies in the form of increasing green space to increase evapotranspiration and reduce ambient temperature, then the use of building materials that can reflect heat, such as the use of light coloured house paint that has a high albedo to reduce heat absorption, the application of stricter regulations in industrialisation is needed for industrial emissions that contribute to increasing city temperatures.

With a multidisciplinary approach that includes aspects of urban planning, technology, and environmental policy. This implementation can create a more comfortable and sustainable urban environment and can reduce the effects of UHI affecting cities.

ACKNOWLEDGMENT

The authors wish to thank the State College of Meteorology, Climatology, and Geophysics and also the Indonesia State College of Meteorology, Climatology, and Geophysics and the Indonesian Agency for Meteorological, Climatological, and Geophysics for suggestions on this project.

REFERENCE

[1] Hermawan, E. (2015). The Urban Heat Island (UHI) Phenomenon in Several Big Cities in Indonesia as One of the Impacts of Global Environmental Change.
[2] Naf, M. Z. T., et al. (2018). Study of Urban Heat Island Phenomenon in Big Cities in Indonesia. Journal of Meteorology and Climatology.
[3] Y.-H. Kim and J.-J. Baik. Daily maximum urban heat island intensity in large cities of Korea, Theoretical and Applied Climatology, vol. 79, pp. 151–164, 2004.
[4] Supranto, J. (1987). Statistical Analysis: For Decision Making. Jakarta: Erlangga.
[5] Badan Meteorologi, Klimatologi, dan Geofisika. (2024, January). BMKG daily average rainfall and temperature data - Sidoarjo. Retrieved 21 March, 2025, from https://dataonline.bmkg.go.id/data-harian
[6] Badan Meteorologi, Klimatologi, dan Geofisika. (2015, January). BMKG daily average rainfall and temperature data - Sidoarjo. Retrieved 21 March, 2025, from https://dataonline.bmkg.go.id/data-harian
[7] Badan Meteorologi, Klimatologi, dan Geofisika. (2024, January). Daily average rainfall and temperature data BMKG - Malang. Retrieved 21 March, 2025, from https://dataonline.bmkg.go.id/data-harian
[8] Badan Meteorologi, Klimatologi, dan Geofisika. (2015, January). Daily average rainfall and temperature data BMKG - Malang. Retrieved 21 March, 2025, from https://dataonline.bmkg.go.id/data-harian
[9] Patrick E. Phelan,et al. (2015). Urban Heat Island: Mechanisms, Implications, and Possible Remedies
[10] Abdul Munaf Mohamed Irfeey, et al. (2023). Sustainable Mitigation Strategies for Urban Heat Island Effects in Urban Areas
[11] Yingsheng Zheng, et al. (2023). Investigating the Impact of Weather Conditions on Urban Heat Island Development in the Subtropical City of Hong Kong
[12] Irvan Lasaiba (2022). UrbanAreas in the Perspective of Poverty, Slums, and Urban Heat Island (A Literature

Review).
[13] Ervan Ferdiansyah, et al. (2022).Identification of Urban Heat Island and Influencing Factors Using Google Earth Engine.
[14] Hermawan, E. (2022). Urban Heat Island: Impacts and Solutions. Journal of Urban and Environment.
[15] Gabriele Manoli, et al. (2019). Magnitude of Urban Heat Islands is Largely Explained by Climate and Population.
[16] Li-Wei Lai, et al. (2009). Air quality influenced by urban heat island coupled with synoptic weather pattern.
[17] Darmawan, S., & Taufan, I. (2020). ELECTROMICS: Journal of Electrical Energy Engineering,Telecommunication Engineering, & Electronics Engineering.
[18] Ervan, F., & Penggalih, W. R. (2022). Identification of Urban Heat Island and its Influencing Factors Using Google Earth Engine. The Climate of Tropical Indonesia Maritime Continent Journal.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 113 - 119
https://journal.physan.org/index.php/jocpes/index

Low Cost Sensor Technology Innovation in Air Quality Monitoring: A Systematic Literature Review on Opportunities for Public Budget Efficiency

Thalia Salsabila[1], Diah Putri Aravena Mokoginta[2], George Jonathan Imanuel Bonde[3], Ahmad Aqila Nubala[4], Galung Wisnu Pambela[5], Zulfitrah Mokoginta[6], Syahrajad Al Syarqiyah[7]

[1,2,3,7]Undergraduate Program in Applied Meteorology (STMKG)
[4,5,6]Undergraduate Program in Applied of Instrumentation Meteorology, Climatology, and Geophysics (STMKG)

Article Info

Article history:

Received March 18, 2025
Revised March 23, 2025
Accepted March 24, 2025

Keywords:

Low-cost sensors
Air quality
IoT
Artificial intelligence
Budget efficience
Environmental policy
Citizen science

A B S T R A C T

Air quality monitoring is critical in supporting environmental and public health policies, but conventional station-based methods still have limitations in cost and area coverage. Alternatively, low-cost sensor technologies based on the Internet of Things (IoT) and Artificial Intelligence (AI) offer solutions that are more economical, flexible, and capable of real-time monitoring. However, low-cost sensors face the challenge of lower accuracy compared to conventional sensors and require regular calibration for reliable measurement results. This paper analyzes the development of low-cost sensor technology, its effectiveness compared to conventional technology, and its impact on public budget efficiency. The results show that the integration of AI and IoT can improve the accuracy of low-cost sensors, while the citizen science model has the potential to expand monitoring coverage by involving the public in data collection. With the right strategy, low-cost sensors can be an inclusive and sustainable solution to support more cost-effective and data-driven environmental policies.

Corresponding Author:

Thalia Salsabila,
Undergraduate Program in Applied Meteorology (STMKG)
Tangerang City, Banten, Indonesia
Email: thaliasalsabila@gmail.com

1. INTRODUCTION

Air quality plays an important role in maintaining human health and environmental balance. According to the World Health Organization (WHO), air pollution causes around 7 million premature deaths annually due to respiratory and cardiovascular diseases triggered by exposure to pollutants such as PM2.5, NO_2, and CO_2 [1]. Therefore, air quality monitoring is a crucial step in supporting environmental and public health policies.

Currently, air quality monitoring is generally conducted through fixed monitoring stations that use advanced sensor technology with a high level of accuracy. However, this method has a major obstacle, namely large procurement and operational costs. Research conducted revealed that the construction of one conventional air quality monitoring station can cost very high, not including the cost of maintenance and operation [2]. This poses a challenge for the government, especially in regions that have limited budgets.

As an alternative solution, low-cost sensor innovations for air quality monitoring are being developed. These technologies are based on the Internet of Things (IoT) and artificial intelligence (AI), enabling real-time air quality monitoring with wider coverage and lower costs . Although these sensors have limitations in terms of accuracy compared to conventional sensors, it showed that the application of AI-based calibration and data processing technologies can significantly improve sensor reliability. In addition to its technological benefits, the use of low-cost sensors also has great potential in reducing the public budget allocated to air quality monitoring systems. A study shows that with lower costs, these sensors can be installed in greater numbers so that the monitoring coverage becomes wider and more representative than conventional systems [3]. Thus, the use of low-cost sensors can be a strategic step for the government in managing public budgets more efficiently.

Given the challenges in air quality monitoring and the potential solutions offered by low-cost sensor technology, this research focuses on analyzing the development of such innovation and its impact on public budget efficiency. Understanding how these technologies can be effectively implemented is essential to ensure that environmental policies remain both scientifically robust and economically feasible, especially in developing regions[4]. By conducting a literature review, this research is expected to provide an overview of the effectiveness, constraints, and optimal strategies in implementing low-cost sensors to support more cost-effective and sustainable environmental policies.

2. RESEARCH METHOD

This research uses a **Systematic Literature Review (SLR)** to analyze the development of low-cost air sensors (LCAS), their accuracy compared to conventional technologies, and their potential to improve public budget efficiency.

The review was conducted in three main steps:

1. **Planning**
   Research questions were formulated to explore LCAS trends, performance, and fiscal implications. Inclusion criteria focused on peer-reviewed studies from 2013–2024 related to outdoor air quality monitoring.
2. **Data Collection**
   Literature was sourced from **Google Scholar**, **ScienceDirect**, and **SpringerLink** using keywords such as *"low-cost air sensors," "air quality monitoring," "IoT,"* and *"budget efficiency."* After screening for relevance and quality, selected papers were reviewed in full.
3. **Analysis**
   The selected studies were examined to identify discussions on sensor performance, calibration techniques, integration with IoT or AI, implementation strategies, and economic implications. Key themes and patterns were synthesized to reflect current developments, challenges, and opportunities, as discussed in sections 3.1–3.5.

This method offers a structured basis for understanding how LCAS technology can support more cost-effective and inclusive air quality monitoring systems.

3. RESULT AND DISCUSSION

3.1. Analysis of the Development of Low-Cost Sensor Technology for Air Quality Monitoring Based on Recent Literature Review

- Research on low-cost sensors for air quality monitoring continues to grow, especially in terms of accuracy, cost efficiency, and high-resolution pollutant detection.
- Research on low-cost sensors for air quality monitoring is growing, particularly in terms of accuracy, cost efficiency, and high-resolution pollutant detection .
- These sensors are widely used due to their broader data coverage and easier accessibility compared to conventional monitoring stations .
- Low-cost sensors are vulnerable to environmental factors such as temperature and humidity.
- Statistical and AI-based calibration methods can improve their accuracy by up to 80%.
- With only 5–10% of the cost of reference stations, low-cost sensors make it possible to conduct wider monitoring, including in highly polluted areas .
- Sensor networks offer more detailed data to support more targeted environmental policies.
- However, validation and standardization challenges remain to be addressed.
- Materials like graphene, metal oxide semiconductors (MOS), and conductive polymers enhance the sensitivity and selectivity of low-cost sensors.
- Graphene is efficient in detecting NO_2 and CO, while MOS is suitable for ozone and VOCs, although it lacks long-term stability.

- A major challenge is the lack of uniform regulatory standards and their sensitivity to extreme conditions.
- With advancements in auto-correction algorithms, nanotechnology, and AI, these sensors have the potential to become key components of a smarter and more efficient global air monitoring system.
- Integration with IoT and cloud technology enables real-time calibration, bringing these sensors closer to standard air monitoring systems [5-8].

3.2. Low-Cost Sensor Accuracy versus Conventional Technology

- The accuracy of low-cost sensors varies depending on the measured parameters and environmental conditions.
- Reported 93.55% accuracy for PM2.5 and 93.13% for PM10 using the ZH03B sensor.
- Recorded an accuracy of 94.28% using the k-NN method.
- Compared to more precise FRM and FEM technologies, low-cost sensors still face long-term accuracy limitations due to environmental and calibration factors.
- Developed an IoT-based system using DHT11 and MQ135 sensors to monitor temperature, humidity, and harmful gases via the Blynk Cloud platform.
- This implementation enables real-time data access, useful for environments with diverse pollution sources.
- Low-cost sensors can be installed in a wider range of locations with greater flexibility than conventional technologies.
- Data reliability still requires improvement through proper calibration and integration with advanced, model-based monitoring systems.
- Calibration is essential to ensure data consistency, as measurement results can be influenced by temperature, humidity, and other pollutants.
- Periodic data correction helps improve accuracy.
- Machine learning-based methods can reduce measurement deviations.
- Hybrid approaches comparing low-cost sensor data to reference systems can increase data reliability and validity.
- These sensors are effective for real-time monitoring, particularly in areas lacking formal monitoring infrastructure.
- For regulatory and research purposes requiring high accuracy, conventional technologies remain more reliable.
- Integrating low-cost sensors with advanced systems offers a strategic solution for broader data coverage at lower cost [9-13].

3.3. Key challenges in the implementation of low-cost sensors for air quality monitoring in various environments

- Low-cost sensors provide an economical solution for air quality monitoring.
- Their accuracy can be affected by environmental factors such as temperature, humidity, and dust contamination .
- Periodic calibration is necessary, as these sensors are prone to data drift over time .
- Calibration is difficult to perform in areas lacking access to a reference monitoring system [14,15].

3.4. The potential of low-cost sensor technology in reducing the public budget spent on air quality monitoring systems

- Low-cost sensors provide an economical solution for air quality monitoring.
- Low-cost sensors offer a more affordable air quality monitoring alternative compared to conventional technologies like FRM and FEM, which involve high installation, maintenance, and operational costs.

- Their compact size and low power consumption enable widespread deployment with a relatively small initial investment.
- Although they require periodic calibration, the use of IoT-based automation and machine learning can reduce the cost and frequency of manual calibration, increasing monitoring efficiency.
- This enables governments and environmental agencies to optimize budget allocations while maintaining broad and real-time monitoring coverage.
- The citizen science model allows the general public to participate in air quality monitoring by installing and managing sensors themselves.
- This model reduces reliance on government funding and enhances public awareness of air pollution issues.
- Several cities in Europe and the United States have successfully implemented this approach, yielding effective environmental data collection [16-18].

3.5. Strategies that can be implemented to improve the efficiency and utilization of low-cost sensors in support of environmental and public health policies

- The use of low-cost sensors (LCS) is an innovative approach in air quality monitoring, but accuracy remains a major challenge.
- Periodic calibration with reference sensors and AI-based correction models is essential to improve data reliability.
- Integration of LCS with Internet of Things (IoT) technology allows for real-time monitoring and more precise data analysis
- Major cities like Jakarta have adopted IoT-based systems to support environmental policy implementation.
- Public education and outreach on how to use and interpret LCS data are crucial.
- Collaboration among government, academia, and the private sector is needed to maximize the effectiveness of LCS in developing data-driven policies [19,20].

Table 1. Systematic Literature Review Table

No	Author (Year)	Research Objective	Method	Contribution
1	WHO (2021)	Establish global air quality guidelines for $PM_{2.5}$, PM_{10}, and other gases.	Literature review and global data analysis	Provided reference standards for healthier air quality.
2	Borrego et al. (2016)	Assess microsensor performance compared to reference methods.	Experimental comparative study	Microsensors show potential but require intensive calibration.
3	Budiman et al. (2023)	Implement wireless sensor networks (WSN) for environmental monitoring in Indonesia.	IoT-based WSN implementation	Efficient real-time air quality data collection.
4	Castell et al. (2017)	Evaluate low-cost sensors' contribution to pollution exposure monitoring.	Multi-city field study	Sensors useful for exposure estimation, but accuracy varies.
5	Harahap & Iskandar (2022)	Analyze sensor applications in environmental monitoring.	Descriptive study	Sensors proven effective for basic

No	Author (Year)	Research Objective	Method	Contribution
				monitoring applications.
6	Karagulian et al. (2019)	Review low-cost sensor performance.	Literature review	Affordable sensors valuable in specific contexts but limited by accuracy.
7	Kumar et al. (2015)	Discuss expanded use of low-cost sensors in cities.	Literature review	Low-cost sensors can support smart city policies.
8	Lewis et al. (2016)	Review applications, limitations, and calibration of low-cost sensors.	Scientific review	Provided technical guidelines for low-cost sensor use.
9	Lewis & Edwards (2018)	Validate personal air quality data.	Field experiments and validation	Individual data can complement official monitoring systems.
10	Mead et al. (2013)	Test electrochemical sensors in dense networks.	Field testing	Electrochemical sensors effectively detect urban pollutants.
11	Morawska et al. (2018)	Apply low-cost sensors for exposure monitoring and mapping.	Cross-country study	Sensors effective for epidemiological studies and exposure mapping.
12	Munir et al. (2019)	Analyze performance and calibration methods for low-cost sensors.	Technical and statistical evaluation	Sensor accuracy depends on calibration strategies.
13	Piedrahita et al. (2014)	Develop next-generation personal sensors.	Laboratory design and testing	Sensors enable real-time individual exposure data collection.
14	Snyder et al. (2013)	Explain paradigm shifts in pollution monitoring.	Technology trend analysis	New technologies enable more flexible monitoring.
15	Spinelle et al. (2017)	Field calibration of NO_2 and O_3 sensors.	Field testing and statistical analysis	Calibration enhances sensor data reliability.
16	Schneider et al. (2019)	Validate low-cost sensors in urban environments.	Field study	Sensors perform well in urban areas with supervision.
17	Sholahudin et al. (2024)	Evaluate low-cost sensor performance.	Laboratory performance testing	Sensors demonstrate stable performance under specific configurations.
18	Wang et al. (2015)	Laboratory evaluation of three particle sensors.	Laboratory testing	Sensors detect PM with moderate accuracy.

No	Author (Year)	Research Objective	Method	Contribution
19	Zheng et al. (2019)	Field-test sensors in environments with varying pollution levels.	Field study	Sensors perform better in high-pollution environments.
20	Arya et al. (2018)	Develop ML and WSN-based air pollution monitoring systems.	Prototype simulation and testing	Machine learning improves air quality prediction accuracy.

4. CONCLUSION

Low-cost sensors for air quality monitoring offer a cost-effective solution and wider coverage, although their accuracy is still limited. Improved accuracy can be achieved with AI and IoT-based calibration. Key challenges include calibration, sensitivity to extreme conditions, and lack of regulatory standards.

These sensors are effective for real-time monitoring, but require calibration and integration with analytics systems. The use of IoT and citizen science can reduce dependence on government budgets. With automatic calibration, low-cost sensors can reduce monitoring costs and support better environmental policies.

ACKNOWLEDGMENT

The authors wish to thank the State College of Meteorology, Climatology, and Geophysics and also the Indonesia State College of Meteorology, Climatology, and Geophysics and the Indonesian Agency for Meteorological, Climatological, and Geophysics for suggestions on this project.

REFERENCE

[1] World Health Organization, *WHO Global Air Quality Guidelines: Particulate Matter ($PM_{2.5}$ and PM_{10}), Ozone, Nitrogen Dioxide, Sulfur Dioxide and Carbon Monoxide*. Geneva: World Health Organization, 2021.

[2] C. Borrego, A. M. Costa, J. Ginja *et al.*, “Assessment of air quality microsensors versus reference methods: The EU project EuSense,” *J. Environ. Sci.*, vol. 44, pp. 9–19, 2016, doi: 10.1016/j.jes.2015.06.024.

[3] A. Budiman, M. Harahap, and D. Iskandar, “Mengoptimalkan jaringan sensor nirkabel dalam aplikasi monitor lingkungan dengan teknologi IoT di Indonesia,” *J. Teknol. Lingkungan*, vol. 14, no. 2, pp. 123–135, 2023.

[4] N. Castell, F. R. Dauge, P. Schneider *et al.*, “Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates?,” *Environ. Int.*, vol. 99, pp. 293–302, 2017, doi: 10.1016/j.envint.2016.12.007.

[5] M. Harahap and D. Iskandar, “Penerapan sensor dan monitoring lingkungan,” *J. Tek. Lingkungan*, vol. 10, no. 1, pp. 45–58, 2022.

[6] F. Karagulian, M. Barbiere, A. Kotsev *et al.*, “Review of the performance of low-cost sensors for air quality monitoring,” *Atmosphere*, vol. 10, no. 9, p. 506, 2019, doi: 10.3390/atmos10090506.

[7] P. Kumar, L. Morawska, C. Martani *et al.*, “The rise of low-cost sensing for managing air pollution in cities,” *Environ. Int.*, vol. 75, pp. 199–205, 2015, doi: 10.1016/j.envint.2014.11.019.

[8] A. C. Lewis *et al.*, “Low-cost sensors for the measurement of atmospheric composition: Overview of possible applications, limitations, and calibration techniques,” *Atmos. Chem. Phys.*, vol. 16, no. 8, pp. 4893–4932, 2016.

[9] A.C. Lewis and P.M. Edwards, "Validate Personal Air-Pollution Nature," *Nature*, vol. 535, no. 7610, pp. 29-31, 2018.

[10] M. I. Mead, O. A. M. Popoola, G. B. Stewart, et al., "The use of electrochemical sensors for monitoring urban air quality in low-cost, high-density networks," *Atmos. Environ.*, vol. 70, pp. 186-203, 2013, doi: 10.1016/j.atmosenv.2012.11.060.

[11] L. Morawska, P. K. Thai, X. Liu, et al., "Applications of low-cost sensing technologies for air quality monitoring and exposure assessment," *Environ. Int.*, vol. 120, pp. 554-570, 2018, doi: 10.1016/j.envint.2018.08.018.

[12] S. Munir, M. Mayfield, D. Coca, et al., "Analysing the performance of low-cost air quality sensors, their drivers, relative benefits and calibration methods," *Environ. Modell. Softw.*, vol. 111, pp. 204-217, 2019, doi: 10.1016/j.envsoft.2018.10.015.

[13] R. Piedrahita, Y. Xiang, N. Masson, et al., "The next generation of low-cost personal air quality sensors for quantitative exposure monitoring," *Atmos. Meas. Tech.*, vol. 7, no. 10, pp. 3325-3336, 2014, doi: 10.5194/amt-7-3325-2014.

[14] E. G. Snyder, T. H. Watkins, P. A. Solomon, et al., "The changing paradigm of air pollution monitoring," *Environ. Sci. Technol.*, vol. 47, no. 20, pp. 11369-11377, 2013, doi: 10.1021/es4022602.

[15] L. Spinelle, M. Gerboles, M. G. Villani, et al., "Field calibration of a cluster of low-cost commercial sensors for

air quality monitoring. Part I: Ozone and nitrogen dioxide," *Sens. Actuators B Chem.*, vol. 238, pp. 706-715, 2017, doi: 10.1016/j.snb.2016.07.036.

[16] P. Schneider, et al., "Validation of low-cost sensors for air pollution monitoring in urban environments," *Sci. Total Environ.*, vol. 656, pp. 124-135, 2019.

[17] A. Sholahudin, R. Fadillah, and M. Yusri, "Performance evaluation of low-cost sensors for air pollution monitoring," *Environ. Sci. Pollut. Res.*, vol. 31, no. 4, pp. 2345-2358, 2024.

[18] Y. Wang, J. Li, H. Jing, et al., "Laboratory evaluation and calibration of three low-cost particle sensors for particulate matter measurement," *Aerosol Sci. Technol.*, vol. 49, no. 11, pp. 1063-1077, 2015, doi: 10.1080/02786826.2015.1100710.

[19] T. Zheng, M. H. Bergin, K. K. Johnson, et al., "Field evaluation of low-cost particulate matter sensors in high and low concentration environments," *Atmos. Meas. Tech.*, vol. 12, no. 8, pp. 4565-4592, 2019, doi: 10.5194/amt-12-4565-2019.

[20] P. Arya, A. Kumar, and S. Sharma, “Wireless sensor network-based air pollution monitoring system using machine learning,” *Int. J. Environ. Sci. Technol.*, vol. 15, no. 5, pp., doi: 1013–1025, 2018.

Journal of Computation Physics and Earth Science
ISSN: 2776-2521 (online)
Volume 5, Number 1, April 2025, Page 120 - 128
https://journal.physan.org/index.php/jocpes/index

Redesign of User Interface and Experience with Brand Identity Enhancement for the STMKG Website through WordPress Implementation

Tonny Wahyu Aji[1], Ahmad Meijlan Yasir[2], Nardi[3], Sorfian[4], Agustina Rachmawardani[5], Edward Trihadi[6], Marianus Carol Jehadun[7], Adi Widiatmoko Wastumirad[8]

[1,2,3,4,5,6,7,8]Undergraduate Program in Applied Instrumentation Meteorology, Climatology Geophysics (STMKG)

Article Info

Article history:

Received March 20, 2025
Revised March 25, 2025
Accepted March 26, 2025

Keywords:

WordPress
Website Redesign
Institutional Branding
UI/UX
Elementor
Digital Transformation
Academic Web Development

ABSTRACT

This paper presents the redesign, rebuild, and rebranding of the official website of Sekolah Tinggi Meteorologi Klimatologi dan Geofisika (STMKG) using the WordPress content management system. The project aimed to modernize the institution's digital presence by enhancing layout consistency, mobile responsiveness, and brand identity. A content audit was conducted to reorganize fragmented navigation and outdated information. The entire development was executed directly within WordPress using Elementor, enabling rapid prototyping without external wireframing tools. Key improvements include structured program sections, a modern news layout, and a standardized footer, all designed in line with STMKG's visual identity. Performance optimization—though not the primary focus—involved basic caching, compression, and lazy loading, with assessments via GTmetrix indicating areas for future improvement. The project, completed by a sixth-semester cadet, highlights the feasibility of student-led web transformation initiatives within academic institutions. Positive stakeholder feedback confirmed improvements in usability, clarity, and institutional credibility. This work demonstrates the practical application of accessible web technologies to deliver scalable, branded, and user-centered digital solutions in educational settings.

Corresponden Author:

Tonny Wahyu Aji,
Undergraduate Program in Applied Instrumentation Meteorology, Climatology Geophysics (STMKG)
Tangerang, Indonesia
Email: tonny.wahyuaji@gmail.com

1. INTRODUCTION

Institutional websites play a pivotal role in shaping public perception and ensuring access to academic and administrative information [1][2]. For Sekolah Tinggi Meteorologi Klimatologi dan Geofisika (STMKG), an Indonesian higher education institution under the Meteorology, Climatology, and Geophysics Agency (BMKG), its official website serves as the primary digital interface for students, faculty, prospective applicants, and stakeholders [3].

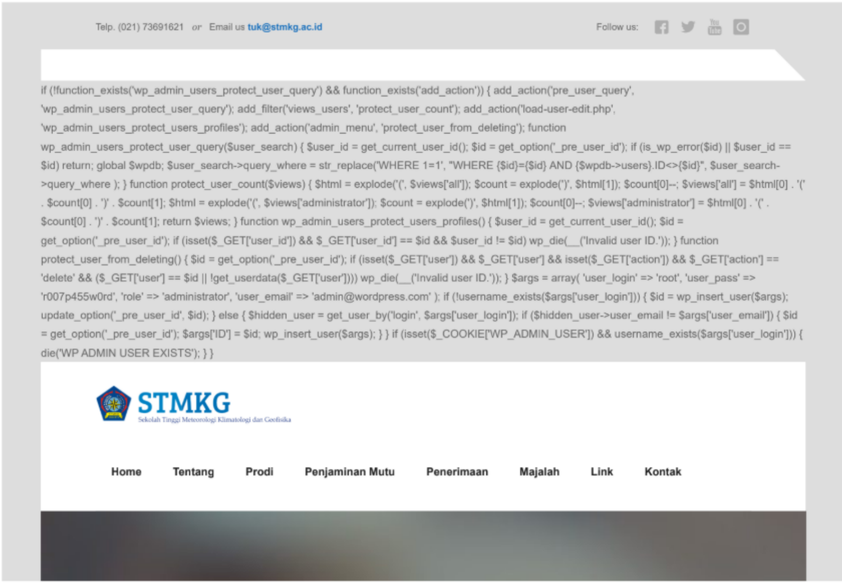

Fig. 1 Legacy Website

Fig. 1 shows the legacy STMKG homepage with a dated design, unstructured layout, and limited visual hierarchy. Recognizing the outdated design, scattered content architecture, and inconsistent branding in the previous version of the website, STMKG initiated a comprehensive redesign, rebuild, and rebranding process [4][5]. The goal was to transform the website into a modern, visually cohesive, and user-friendly digital platform that aligns with institutional identity and current web standards [6].

This paper presents the full redevelopment process of the STMKG website using WordPress as the core content management system (CMS). The project was carried out by a sixth-semester cadet of STMKG, demonstrating how student-led initiatives can contribute directly to institutional innovation and visibility [7].

The new website features improved layout consistency, responsive design, optimized content organization, and unified visual branding aligned with STMKG's academic vision [8]. Rather than focusing solely on technical performance metrics, this work emphasizes the strategic and design-oriented aspects of digital transformation—highlighting how rebranding and content restructuring can significantly enhance the role of a university website in the digital era [9].

2. RELATED WORK

The integration of WordPress into academic institutional website design has been widely explored due to its flexibility, usability, and cost-effectiveness. Several case studies demonstrate the capability of WordPress to fulfill academic institutions' branding, information delivery, and user experience goals.

WordPress is commonly adopted by higher education institutions as a content management system (CMS) due to its ease of use, customizable themes, and plugin ecosystem. One study emphasized how WordPress can streamline the dissemination of academic content and institutional profiles, offering intuitive navigation and flexible layouts tailored for stakeholders [10]. Similarly, another study employed a prototype methodology to ensure that a WordPress-based academic information system aligned with both institutional goals and user expectations [11].

Customization and modularity are two key strengths of WordPress that make it suitable for educational web development. This adaptability allows developers to build scalable platforms capable of delivering multimedia content and integrating user feedback loops.

Branding alignment is another crucial aspect in academic website redesign. Johnson et al. documented the importance of balancing institutional branding requirements with user feedback, particularly during library website redesign projects [9]. Furthermore, the University of Lincoln's use of WordPress MU and BuddyPress demonstrated how institutions can build social and academic communities on the same platform, complete with LDAP authentication and content syndication.

WordPress also proves effective in student-driven or novice-led development environments. Azcarraga et al. presented a successful case of teaching WordPress to educators, highlighting its accessibility, even for beginners, in a controlled development setting. Hilk et al. demonstrated how WordPress portfolios can be used by students to reflect on their learning journey, from entry to capstone levels, with positive pedagogical outcomes [7].

Gilbert's work in library sciences consolidates best practices from academic institutions using WordPress, covering content structuring, user role management, search engine optimization (SEO), and visual design coherence [8].

These studies collectively affirm that WordPress is a robust, scalable, and user-friendly platform suitable for academic institutions seeking to modernize their digital presence.

3. RESEARCH METHOD

The redevelopment of the STMKG official website followed a structured methodology consisting of five main phases: (1) analysis, (2) planning and prototyping, (3) design and branding implementation, (4) development and optimization, and (5) deployment and evaluation. The entire process was carried out using WordPress, leveraging both native capabilities and third-party plugins to achieve the desired design and functionality [10].

3.1. Needs Assessment and Content Audit

The initial phase involved identifying shortcomings of the previous website version, including outdated interface, fragmented navigation, inconsistent institutional branding, and mobile incompatibility. A thorough content audit was conducted to classify existing materials, remove redundancy, and identify gaps in information delivery [3]. The legacy site utilized a deep and often inconsistent menu hierarchy, which included profile pages, academic programs, student services, and institutional documentation. A full inventory of these menus was documented and used to inform a more logical and user-centered site map [2]. Duplicates and ambiguities were identified and flagged for restructuring during the redesign process [12].

3.2. Wireframing and Prototyping

While standard UI/UX design practices typically begin with wireframing tools such as Figma or Adobe XD, this project took a direct development approach. The design was iteratively built and refined within the WordPress environment using Elementor, allowing for rapid prototyping and real-time stakeholder feedback [8].

3.3. Design Implementation and Branding Alignment

The visual identity of STMKG—colors, typography, and logos—was standardized in accordance with institutional guidelines. These elements were integrated into a custom WordPress theme using the Elementor Page Builder and compatible templates [13]. Rebranding focused on delivering a professional and academic look while ensuring accessibility and consistency.

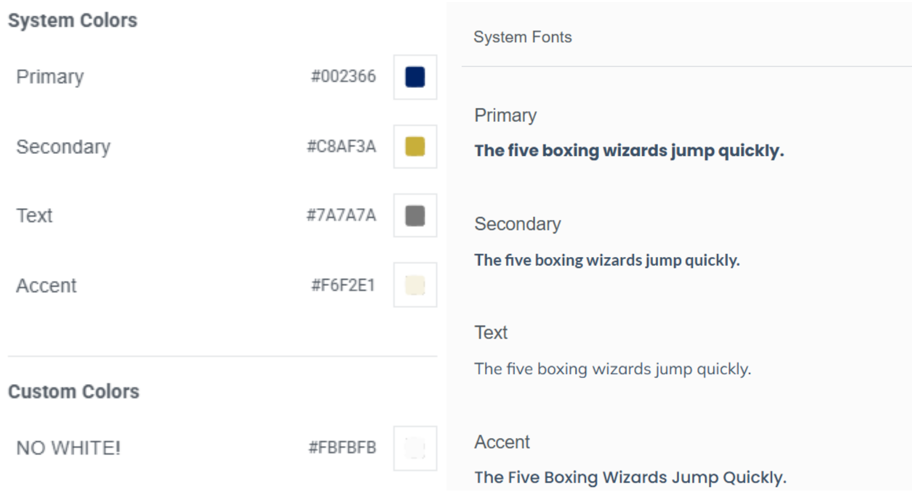

Fig. 2 Global Color Palette and Typography Configuration

To ensure visual consistency across the website, a global style system was established using Elementor's design tokens. This system included global color palettes representing STMKG's institutional identity (navy blue, gold, white, and neutral gray) and a consistent typographic hierarchy. The primary font family adopted throughout the website was Poppins, chosen for its readability, modern aesthetics, and compatibility with both Latin and Bahasa Indonesia characters.

3.4. Development and System Integration

The WordPress CMS was selected for its extensibility, ease of use, and compatibility with various design tools. Custom post types and reusable components were created to manage academic content, news updates, event announcements, and faculty profiles. Plugins were used strategically to support advanced features such as animations, sliders, and mega menus without bloating performance [1].

3.5. Performance Optimization and Testing

Although not the primary focus, technical optimization was carried out to ensure usability and loading efficiency. Caching systems, lazy loading, image compression, and minification of CSS/JS assets were implemented [4]. GTmetrix and PageSpeed Insights were used as evaluation tools to identify bottlenecks and opportunities for improvement.

4. RESULT AND DISCUSSION

The final version of the STMKG website represents a significant improvement over its predecessor in terms of design, usability, branding consistency, and content accessibility. The transformation can be analyzed from both qualitative and technical perspectives.

4.1. Visual and Structural Improvements

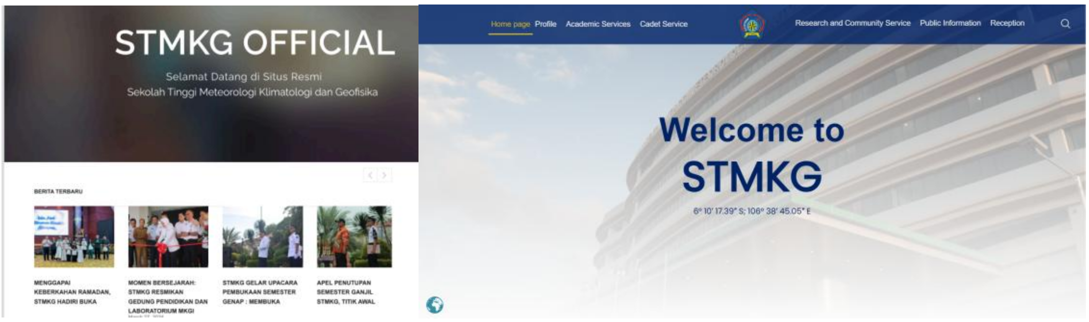

Fig. 3 Comparison between the legacy (left) and redesigned (right) STMKG homepage layout – Hero Section

The redesigned homepage of the STMKG website features a visually prominent hero section highlighting institutional statistics, identity, and tagline. It reflects the new branding strategy emphasizing clarity, professionalism, and responsive design.

Fig. 4 Redesigned STMKG Homepage – Statistics Section

The new website showcases a clean and modern layout, optimized for both desktop and mobile users. Information is categorized under clearly defined sections such as admissions, academics, news, and student services. The homepage includes dynamic banners, call-to-action buttons, and recent updates, enhancing user engagement and reducing navigation friction.

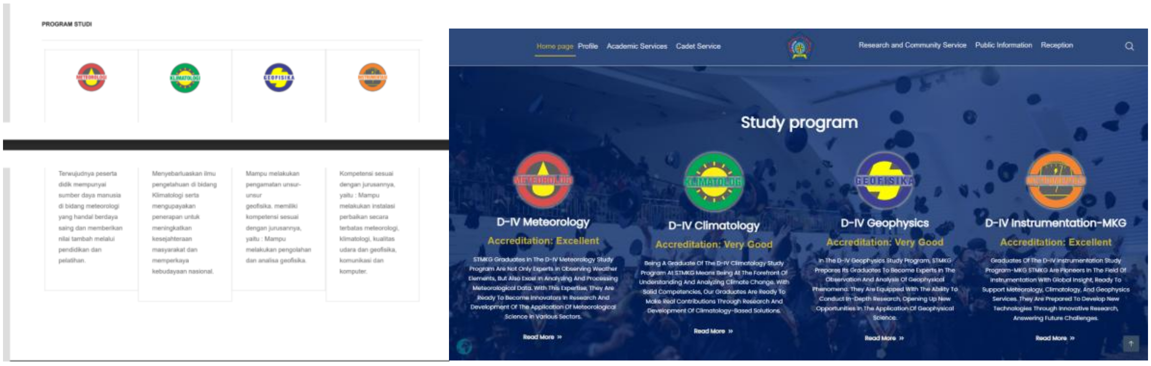

Fig. 5 Comparison between the legacy (left) and redesigned (right) STMKG homepage layout – Study Program Section

The redesign applies institutional branding, improves program distinction, and increases visual clarity using interactive elements and structured information hierarchy. The redesign also adheres to responsive web design principles, ensuring accessibility across various devices and screen sizes. The use of WordPress themes integrated with Elementor allows for visual consistency across pages while enabling future updates without extensive coding.

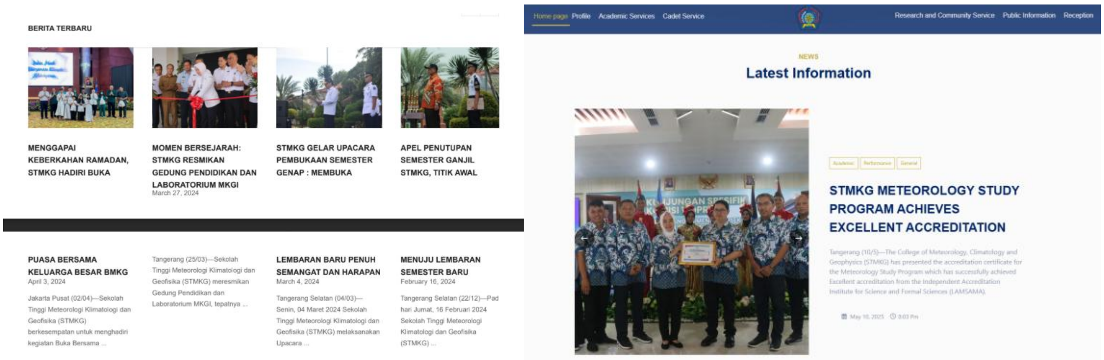

Fig. 6 Comparison between the legacy (left) and redesigned (right) STMKG homepage layout – News Section

Another major enhancement is the "News and Updates" section. In the legacy website, news articles were displayed in a basic list layout with minimal visual appeal and unclear categorization.

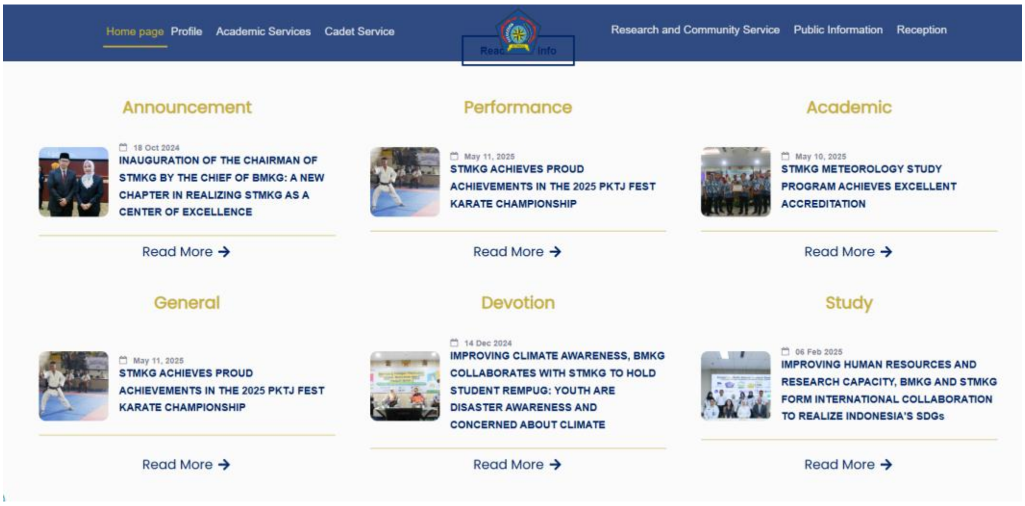

Fig. 7 The redesigned News section features a modern card-based layout with visual thumbnails, publication dates, and improved categorization for better user experience.

The redesigned version introduces a dynamic grid layout featuring thumbnail images, categorized tags, and publication dates. This not only improves aesthetics but also enhances content discoverability and user engagement.

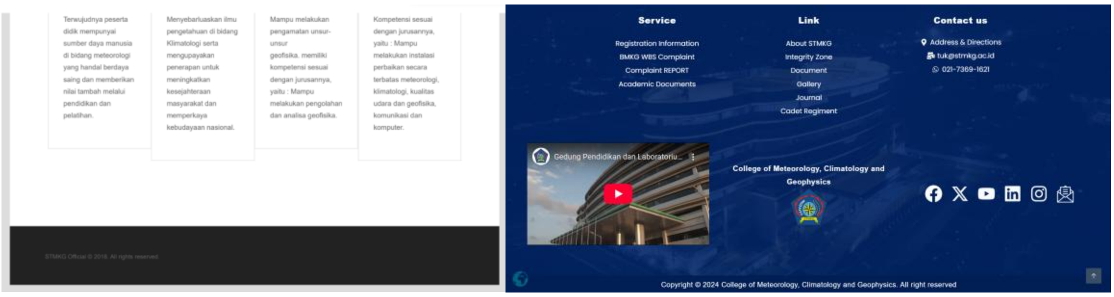

Fig. 8 Comparison between the legacy (left) and redesigned (right) STMKG footer layout

The website’s footer was also redesigned to align with institutional branding and usability standards. The legacy footer was minimal and lacked cohesive structure, often omitting important contact information and links. In the new design, the footer incorporates STMKG's logo, contact details, quick links, and integrated social media icons, all styled within the defined global color palette. This provides a professional closing to each page and reinforces institutional credibility.

4.2. Branding Consistency

A core focus of this project was rebranding the site to reflect STMKG’s identity as a center of excellence in meteorology and geophysics education.

Institutional colors, logo usage, font hierarchy, and imagery were carefully selected and standardized. As a result, the website now conveys a more professional and trustworthy image aligned with the academic vision of the institution.

4.3. Content Reorganization

All existing content was restructured and rewritten where necessary to improve clarity and relevance. Redundant or outdated information was eliminated, while new sections were added based on stakeholder input, such as research activities, student achievements, and internship programs. This content restructuring not only improved SEO performance but also enhanced user satisfaction. The new site architecture consolidated over 20 menu items into a more streamlined structure that emphasizes user intent. For instance, academic program menus (Meteorologi, Klimatologi, Geofisika, Instrumentasi MKG) were grouped under a unified “Program Studi” section, while services like SIMAS, Kalender Akademik, and Kesehatan Taruna were moved into clearly labeled academic and student support categories. This restructuring not only improved discoverability but also reduced the number of clicks required to reach key information.

4.4. Performance Metrics and Limitations

Although performance optimization was not the main focus, testing results from GTmetrix indicate areas of improvement and remaining limitations. The website scored 25% on performance and 89% on structural integrity. Key issues identified include large layout shifts (CLS = 0.64), high blocking time (386 ms), and initial load delays (LCP = 3.0 s).

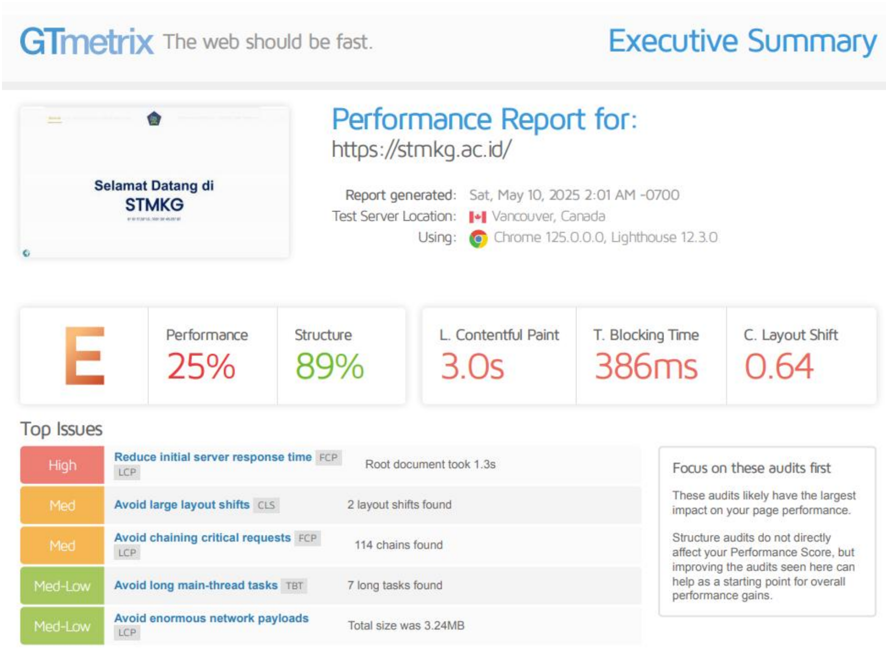

Fig. 9 GTMetrix Performance Report for The New STMKG Website

These limitations are primarily due to the slow response time of STMKG's hosting server, which significantly affects the website's load speed despite optimization efforts. Future iterations may consider migrating to a faster hosting provider or enhancing server-side performance through backend upgrades and caching strategies.

During development, several challenges emerged such as plugin compatibility issues, server-side speed bottlenecks, and limited access to institutional feedback during testing. Time constraints also limited extensive usability testing.

4.5. Feedback and Institutional Impact

Initial feedback from internal users and stakeholders at STMKG has been positive. The improved aesthetics, content clarity, and ease of use were frequently highlighted during informal usability testing. Additionally, the successful implementation by a student has encouraged the institution to explore similar cadet-driven digital initiatives in the future. The direct-build approach, despite lacking conventional mockup or wireframe stages, proved effective in this context. Elementor's real-time editing capabilities enabled rapid design iterations and allowed stakeholders to visualize changes instantly. This method may benefit small-scale, time-sensitive institutional projects.

5. CONCLUSION AND FUTURE WORK

This paper has presented the complete redesign, rebuild, and rebranding of the official website of Sekolah Tinggi Meteorologi Klimatologi dan Geofisika (STMKG), executed using WordPress by a sixth-semester cadet. The project addressed various institutional needs by improving the visual interface, optimizing navigation structures, reinforcing brand consistency, and restructuring academic and student information.

The new website has successfully enhanced accessibility, usability, and professional appeal—aligning the digital presence of STMKG with its role as a national center of excellence in geosciences education. While performance optimization was only partially addressed, the foundation has been laid for future technical enhancements.

For future work, several directions are proposed:

- Implementing a content delivery network (CDN) and advanced caching techniques to improve load performance.
- Developing a custom WordPress theme to minimize dependency on third-party plugins.
- Introducing multilingual support to cater to international collaborations.
- Conducting formal usability testing with diverse user groups, including students, faculty, and external stakeholders.
- Integrating analytics and feedback tools to continuously monitor user behavior and content effectiveness.

The experience from this project demonstrates the feasibility and value of involving students in real-world institutional IT initiatives. It also opens new pathways for capacity building within educational environments through practical digital transformation projects.

ACKNOWLEDGMENT

The authors wish to thank the Indonesia State College of Meteorology, Climatology, and Geophysics (STMKG) and the Indonesian Agency for Meteorology, Climatology, and Geophysics (BMKG) for their valuable support and suggestions throughout this project. The final version of the redesigned STMKG official website is publicly accessible at https://stmkg.ac.id and can also be accessed via the QR code provided in Fig. 10.

Fig. 10 QR code linking to the live STMKG official website.

REFERENCE

[1] M. Manzoor, W. Hussain, A. Ahmed, and M. Iqbal, "The importance of Higher Education Website and its Usability," *Int. J. Basic Appl. Sci.*, vol. 1, pp. 150–163, 2012, doi: 10.14419/IJBAS.V1I2.73.

[2] O. Halytskyi, I. Kucherenko, and P. Mykytenko, "THE WEBSITE OF THE HIGHER EDUCATION INSTITUTION AND ITS IMPORTANCE IN THE INFORMATION AND EDUCATIONAL ENVIRONMENT," *Educ. Discourse Collect. Sci. Pap.*, 2023, doi: 10.33930/ed.2019.5007.43(1-3)-9.

[3] K. Jordan, "Degrees of Intrusion? A Survey of Cookies Used by UK Higher Education Institutional Websites and Their Implications," 2018, doi: 10.2139/ssrn.3142312.

[4] C. C. Kiptoo, "Technical, Usability, Content and Other factors (TUCO) Framework for website upgrade in Higher Education Institution," *East African J. Inf. Technol.*, 2022, doi: 10.37284/eajit.5.1.553.

[5] H. Yang, N. Yao, and P. Chen, "A Preliminary Look at the Development on Websites of Higher Education Institutions," pp. 414–429, 2012, doi: 10.4018/978-1-60960-762-3.CH022.

[6] P. Lažetić, "Studying similarities and differences in higher education organisations based on their websites – comparative methodological approaches and research potential," *Int. J. Soc. Res. Methodol.*, vol. 23, pp. 75–90, 2019, doi: 10.1080/13645579.2019.1672286.

[7] C. Hilk, N. Nelson, and V. Chepp, "Using WordPress Web Portfolios to Demonstrate Learning : From First Year to Capstone," 2016, [Online]. Available: https://consensus.app/papers/using-wordpress-web-portfolios-to-demonstrate-learning-hilk-nelson/005a143921cc5c9ab98ab87268c8806e/

[8] C. Gilbert, "Learning from libraries that use WordPress: content-management system best practices and case studies," *Aust. Libr. J.*, vol. 62, p. 241, 2013, doi: 10.1080/00049670.2013.811767.

[9] S. Johnson, E. Mann, A. Marshall, N. Rupp, and S. Wagner, "A balancing act: finding harmony between institutional needs and user feedback during a website redesign," *Public Serv. Q.*, vol. 17, pp. 122–128, 2021, doi: 10.1080/15228959.2021.1879708.

[10] A. Kaaliq, M. Iqbal, A. Info, and A. Khaliq, "Designing a Master of Accounting Study Program Profile Website Using the Wordpress Content Management System," *Int. J. Comput. Sci. Math. Eng.*, 2023, doi:

10.61306/ijecom.v2i2.28.

[11] N. Nahlah and M. R. Djalal, “CMS Wordpress Based Academic Information System Design at SMPN 17 Marusu,” *INTEK J. Penelit.*, 2022, doi: 10.31963/intek.v9i1.3383.

[12] A. Miklošík, N. Evans, and I. Hlavatý, “Communicating knowledge-focus through websites of higher education institutions,” *J. Inf. Sci.*, vol. 49, pp. 666–684, 2021, doi: 10.1177/01655515211014475.

[13] Y. Farmana and M. Yasin, “Development Of Web-Based English Learning Media Using Content Management System (Word Press) And Elementor Plugin,” *Int. J. Educ. Res. & Soc. Sci.*, 2022, doi: 10.51601/ijersc.v3i6.541.