ISSN 2355-3286

ULTIMA Computing

Jurnal Sistem Komputer

Engineering | Vol. 18 | No. 1 | Page 1 - 60 | June 2026 | ISSN 2355-3286

UMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA

ISSN 2355-3286

EDITORIAL BOARD

EDITORIAL ADDRESS

Universitas Multimedia Nusantara (UMN)
Jl. Scientia Boulevard
Gading Serpong
Tangerang, Banten - 15811
Indonesia
Phone. (021) 5422 0808
Fax. (021) 5422 0800
Email : ultimacomputing@umn.ac.id

Ultima Computing : Jurnal Sistem Komputer is a Journal of Computer Engineering Study Program, Universitas Multimedia Nusantara which presents scientific research articles in the field of Computer Engineering and Electrical Engineering as well as current theoretical and practical issues, including Edge Computing, Internet-of-Things, Embedded Systems, Robotics, Control System, Network and Communication, System Integration, as well as other topics in the field of Computer Engineering and Electrical Engineering. The Ultima Computing : Jurnal Sistem Komputer is published regularly twice a year (June and December) and is jointly managed by the Computer Engineering and Electrical Engineering Study Program at Universitas Multimedia Nusantara.

Call for Papers

International Journal of New Media Technology (IJNMT) is a scholarly open access, peer-reviewed, and interdisciplinary journal focusing on theories, methods and implementations of new media technology. Topics include, but not limited to digital technology for creative industry, infrastructure technology, computing communication and networking, signal and image processing, intelligent system, control and embedded system, mobile and web based system, and robotics. IJNMT is published twice a year by Faculty of Engineering and Informatics of Universitas Multimedia Nusantara in cooperation with UMN Press.

Ultimatics : Jurnal Teknik Informatika is the Journal of the Informatics Study Program at Universitas Multimedia Nusantara which presents scientific research articles in the fields of Analysis and Design of Algorithm, Software Engineering, System and Network security, as well as the latest theoretical and practical issues, including Ubiquitous and Mobile Computing, Artificial Intelligence and Machine Learning, Algorithm Theory, World Wide Web, Cryptography, as well as other topics in the field of Informatics.

Ultima Computing : Jurnal Sistem Komputer is a Journal of Computer Engineering Study Program, Universitas Multimedia Nusantara which presents scientific research articles in the field of Computer Engineering and Electrical Engineering as well as current theoretical and practical issues, including Edge Computing, Internet-of-Things, Embedded Systems, Robotics, Control System, Network and Communication, System Integration, as well as other topics in the field of Computer Engineering and Electrical Engineering.

Ultima InfoSys : Jurnal Ilmu Sistem Informasi is a Journal of Information Systems Study Program at Universitas Multimedia Nusantara which presents scientific research articles in the field of Information Systems, as well as the latest theoretical and practical issues, including database systems, management information systems, system analysis and development, system project management information, programming, mobile information system, and other topics related to Information Systems.

ISSN 2355-3286

FOREWORD

ULTIMA Greetings!

Ultima Computing : Jurnal Sistem Komputer is a Journal of Computer Engineering and Electrical Engineering at Multimedia Nusantara University which presents scientific research articles in the field of Computer Systems as well as the latest theoretical and practical issues, including Edge Computing, Internet-of-Things, Embedded Systems, Robotics, Control Systems, Network and Communication, System Integration, and other topics in the field of Computer Engineering and Electrical Engineering.

In this June 2026 edition, Ultima Computing enters the 1st Edition of Volume 18. In this edition there are six scientific papers from researchers, academics and practitioners in the fields of Computer Engineering and Electrical Engineering. Some of the topics raised in this journal are: Controlling and Monitoring Milk Pasteurization using Fuzzy Logic integrated with the Internet of Things (IoT), Early Detection of Non-Melanoma Skin Lesions: A ResNet50 and SVM-Based Deep Learning Approach, Feature Selection Benchmarks for Breast Cancer Diagnosis: A Comparative Machine Learning Study, Design and Development of an Automated Gymnastics Movement Evaluation Device, Enhancing Energy Visibility: A Real-Time Power Monitoring System via Web Scraping and NoSQL, and Design and Development of a Water Filling System Using An Arduino Mega 2560 Based on Flowmeter Sensor.

On this occasion we would also like to invite the participation of our dear readers, researchers, academics, and practitioners, in the field of Engineering and Informatics, to submit quality scientific papers to: International Journal of New Media Technology (IJNMT), Ultimatics : Jurnal Teknik Informatics, Ultima Infosys: Journal of Information Systems and Ultima Computing: Journal of Computer Systems. Information regarding writing guidelines and templates, as well as other related information can be obtained through the email address ultimacomputing@umn.ac.id and the web page of our Journal here.

Finally, we would like to thank all contributors to this June 2026 Edition of Ultima Computing. We hope that scientific articles from research in this journal can be useful and contribute to the development of research and science in Indonesia.

June 2026,

Monica Pratiwi, S.ST., M.T.
Editor-in-Chief

ISSN 2355-3286

TABLE OF CONTENT

ISSN 2355-3286

Controlling and Monitoring Milk Pasteurization using Fuzzy Logic integrated with the Internet of Things (IoT)

Lisa Yihaa Roodhiyah[1], Nashrullah Afif Zuhma[2]
[1,2]Automation Engineering Technology, Department of Industrial Technology, Vocational School, Universitas Diponegoro, Semarang, Indonesia
[1]lisayihaa@lecturer.undip.ac.id
[2]afifzhm@students.undip.ac.id

Accepted December 14, 2025
Approved February 06, 2026

***Abstract*— Precise temperature control in nonlinear thermal processes with fluctuating conditions remains a challenge for traditional control methods. This research presents a Mamdani-type fuzzy logic control system integrated with an IoT architecture for milk pasteurization using the Low-Temperature Long-Time (LTLT) method, which is widely used in Indonesia for small-scale dairy production. The controller takes milk temperature and volume as linguistic inputs and outputs a continuous PWM signal to regulate heater power. Unlike traditional on–off systems or model-based PID controllers, this fuzzy logic approach does not rely on an explicit mathematical model and remains effective across different milk volumes without retuning. Experiments with 3 L, 5 L, and 8 L of milk show that the controller keeps temperatures close to the 64°C target, with averages of 64.11°C, 64.07°C, and 64.03°C, respectively. Max overshoot is limited to 1.56%, 0.29%, and 0.19%, while high-temperature stability is demonstrated by standard deviations of 0.26, 0.08, and 0.12, indicating robustness. Furthermore, it functions at a lower average PWM duty cycle compared to on–off control, leading to smoother operation and improved efficiency. This system can handle nonlinear thermal processes with varying loads and is supported by real-time IoT connectivity monitoring.**

Index Terms*— Fuzzy logic control; Mamdani inference system; Milk pasteurization; Low-temperature long-time (LTLT); Internet of Things (IoT)*

I. INTRODUCTION

Milk is a vital source of protein, an essential component of feed formulations for both children and adults. During processing to become consumed milk, raw milk is pasteurized. This is because pasteurization creates an ideal environment that prevents rapid growth of harmful microorganisms, especially when stored at or below room temperature. Pasteurization is a method developed by the French scientist Louis Pasteur in the 1800s. The process, which involves heating milk to a high temperature and then rapidly cooling it, extends its shelf life, a process known as Extending Shelf Life (ESL) [1]. Milk pasteurization primarily aims to eliminate harmful bacteria and reduce spoilage bacteria, thereby extending milk shelf life. Therefore, pasteurization is a crucial step in milk processing, ensuring microbial control and preserving food quality for consumers by removing pathogens that cause disease, preventing souring, and maintaining nutrient quality. There are various methods for pasteurizing milk, which can be categorized into two groups: traditional and modern methods [1]. Traditional methods included thermal pasteurization, such as low-temperature short-time (LTST), low-temperature long-time (LTLT), high-temperature short-time (HTST), and flash pasteurization, all of which use temperatures below the boiling point. Then modern methods include high-temperature (HT) and UHF pasteurization [2], [3], [4].

Among those pasteurization methods, high-temperature short-time (HTST) and low-temperature long-time (LTLT) pasteurization are the most widely used. In Indonesia, a survey indicates that the cattle-farming community and milk processing industry still rely on a slow-heating method to enhance milk flavor. As a result, LTLT pasteurization is deemed suitable for Indonesia's milk industry, as it promotes effective blending of milk and its flavor [5]. LTLT is the process of pasteurizing milk by heating it to 65°C and holding it at that temperature for 30 minutes [6]. Maintaining the exact temperature is vital in the LTLT pasteurization method because achieving that temperature enables inactivation of non-spore-forming pathogens, such as psychrotrophic spoilage bacteria, including Gram-negative Pseudomonas [4], and Coxiella burnetii, the most heat-resistant pathogen in raw milk [6]. Additionally, LTLT showed no significant changes in nutrient levels, except for a slight loss of Vitamin A and Vitamin C. This type of milk can be stored for up to 2-3 days before spoilage due to putrefactive organisms rather than acid formation [7].

Several control system approaches have been developed to optimize milk pasteurization outcomes in prior research, including model predictive control [8], Proportional-Integral-Derivative (PID) control [3], and artificial intelligence Internet of Things (AIoT) approaches [8]. However, those methods have several limitations. The model predictive control method faces inherent challenges, including noisy signals, model accuracy issues, and hardware constraints. Then, the PID approach requires tuning of several parameters, including proportional (P), integral (I), and derivative (D), which are appropriate for a given volume. Fixed PID gains are inadequate when process conditions change significantly, such as when a volume change happens [9]. Subsequently, the AIoT approach proved inefficient for higher-capacity applications, as it required more time to reach the setpoint [8]. Therefore, this study employs an alternative approach: fuzzy logic, which is more efficient for larger volumes when reaching the setpoint temperature. Fuzzy logic control was also used in the previous study; however, in that study, the controller was a PLC, which is well-suited to industrial production processes. In this study, the controller is an ESP32, which is more appropriate for the cattle-farming community. Several fuzzy logic algorithms exist, such as Mamdani-type, Sugeno-type, and Tsukamoto-type [10], [11]. However, this study uses Mamdani-type fuzzy logic because it provides high interpretability and clear linguistic transparency. In contrast, Sugeno-type fuzzy logic relies on mathematical functions in its consequents, which are harder to interpret [11]. Mamdani is better suited to ill-defined nonlinear systems than Tsukamoto, which requires monotonic functions and crisp outputs for each rule, thereby limiting flexibility for complex nonlinear behavior. Mamdani systems also offer a richer representation of uncertainty than Sugeno systems, which directly convert outputs into crisp values—this can improve computational efficiency but reduces explicit uncertainty representation. Additionally, Mamdani is often the preferred choice when introducing fuzzy logic or when ease of acceptance among operators and engineers is essential, as its control architecture is straightforward, which covers fuzzification, inference, aggregation, and defuzzification [11].

The earlier pasteurization system that employed fuzzy logic [17] did not account for the efficiency of the power heater used to warm the milk. To address this in the current study, specific parameters—milk temperature and milk volume—have been selected as inputs for the fuzzy logic controller. This approach aims to optimize the power heater's efficiency while ensuring the milk reaches the standard temperature required by the LTLT process. According to the references, IoT systems facilitate real-time monitoring and control of essential process parameters in dairy processing, such as milk pasteurization. This integration can greatly improve operational efficiency, product quality, and safety [16]. In this study, the IoT system is employed to oversee the LTLT process and track real-time physical parameters of milk, ensuring that the standard LTLT temperature is achieved while preserving heater efficiency. The monitoring and control dashboard on the PC display is used to communicate with the ESP32 microcontroller via TCP/IP over Wi-Fi within a local area network.

II. METHODS

Fuzzy logic is a mathematical framework for managing uncertainty and vagueness, making it helpful in solving complex problems in engineering, artificial intelligence, and decision-making [12]. Fuzzy logic enables modeling a system using fuzzy sets and rules that describe its behavior. Three common methods of deductive inference in fuzzy systems that rely on linguistic rules are (1) Mamdani systems, (2) Sugeno models, and (3) Tsukamoto models [11], [13]. In fuzzy logic, truth values range from entirely true to completely false. The completely true value is explicitly 1, while the completely false value is explicitly 0 [13]. In Fuzzy Logic, instead of simply giving a yes or no answer, the truth value or membership indicates a matter of degree. The Mamdani method was first introduced by Ebrahim Mamdani in 1975. It is among the most popular fuzzy inference techniques because its results are both visualizable and linguistically understandable. In this approach, each fuzzy rule has an antecedent (IF) and a consequent (THEN). The inference process employs the min operator to assess the truth degree in the antecedent and the max operator to combine the results in the consequent. Generally, the Mamdani-type fuzzy logic method consists of a fuzzifier, a fuzzy inference module containing rules, and a defuzzifier. The steps of the fuzzy logic method are shown in Fig.1.

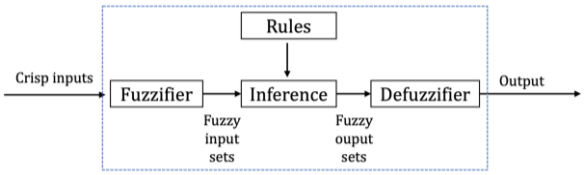


Fig. 1. The Fuzzy Logic Step of the Mamdani-type

The fuzzy logic control method implemented in this system is based on a Mamdani fuzzy inference system, which comprises three main stages: fuzzification, inference using fuzzy rules, and defuzzification. This controller regulates heater power to maintain milk temperature at 63–65 °C for 30 minutes, in accordance with the LTLT pasteurization method. The first step of the Mamdani fuzzy system is fuzzification. Fuzzification is the process of converting crisp input values into fuzzy values using membership functions. In this system, two input variables (milk volume and temperature parameters) and one output variable (heater power percentage) are defined. The milk

volume input is obtained from the ultrasonic sensor HY-SRF05, and the temperature data is collected from the DS18B20 sensor. The milk tank has a capacity of 10 liters. The milk used in this study is from cattle farmers and was purchased for this project. Then, the physical parameters of milk, which are temperature and the milk volume, are used as the input parameters of the membership function of fuzzy. Triangular or trapezoidal membership functions are used to represent each input variable. The triangular membership function is defined as shown in Equation 1, where a triangular fuzzy set is characterized by the parameters a, b, and c. While μ(x) is a fuzzy degree (μ ϵ [0,1]). Equation 1 is used when the membership function has a single peak [13].

$$\mu(x) = \begin{cases} 0, & x \le a \text{ or } x \ge c \\ \frac{x-a}{b-a}, & a < x < b \\ \frac{c-x}{c-b}, & b < x < c \end{cases} \quad (1)$$

For a trapezoidal fuzzy set with parameters a, b, c, and d, the membership function is given as shown in Equation 2 [13]. Equation 2 is used to represent stable operating regions, such as the desired temperature range (63 – 65) °C).

$$\mu(x) = \begin{cases} 0, & x \le a \text{ or } x \ge d \\ \frac{x-a}{b-a}, & a < x < b \\ 1, & b < x < c \\ \frac{d-x}{d-c}, & c < x < d \end{cases} \quad (2)$$

Fuzzification involves creating a fuzzy set through membership functions. The input variable classifications, including milk volume and temperature, are listed in Table I. The justification for determining the membership function range in Table I is based on expert knowledge, ensuring that the LTLT process can be operated at the standard temperature of 65 °C while maintaining the heater's efficiency.

TABLE I. MEMBERSHIP CLASSIFICATION

Input Variable Classifications			
Temperature parameters	**Temperature Range (° C)**	**Milk Volume**	**Milk Volume Range (%)**
Cold	-6 – 20	Empty	0 - 20
Normal	18 – 64	Low	20 – 45
Warm	63.5 – 65.5	Medium	45 – 70
Hot	65 – 80	Full	65 - 100

In Table I, the Cold parameters indicate the initial raw milk conditions before heating. The Normal parameters indicate the temperature at which the milk begins to warm toward the target pasteurization temperature. Warm parameters show when the milk is adequately heated and nearing the ideal temperature. Lastly, Hot parameters indicate that the milk has reached or exceeded the pasteurization temperature, requiring controlled heating to maintain a steady temperature and avoid overheating. The graphical display of both membership functions is shown in Fig. 2.

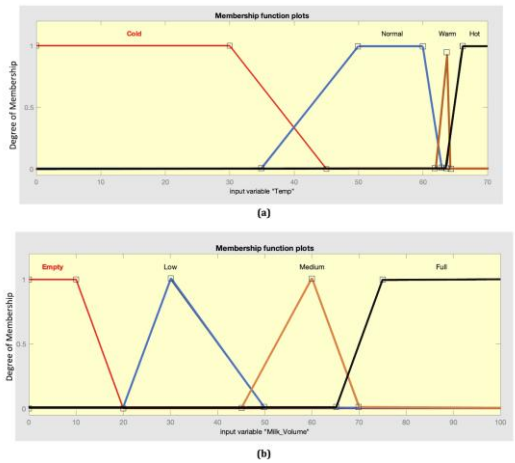


Fig. 2. The graphical display of the membership function for (a) the temperature input and (b) the milk volume input (small box indicates the parameter point)

If the fuzzy inference system has numerous input variables, meaning the rule antecedents have many components, then fuzzy logic operators are used to combine the membership values and produce a single outcome that reflects the rule's result. The if-then rules are generated based on the membership values of each fuzzy set within the input variables [13], [14]. The general fuzzy rule is expressed as shown in Equation 3.

$$IF\ (T\ is\ A_i)\ AND\ (L\ is\ B_j)\ THEN\ (P\ is\ C_k) \quad (3)$$

A_i and B_j denote fuzzy sets of temperature and milk volume, respectively, and C_k denotes a fuzzy set of heater power. The degree of activation (firing strength) of each rule is calculated using the minimum operator, as shown in Equation 4 [11], [13].

$$\alpha_r = \min\left(\mu_{A_i}(T), \mu_{B_j}(L)\right) \quad (4)$$

Where α_r is the firing strength of rule r, $\mu_{A_i}(T)$ is the membership degree of temperature, and $\mu_{B_j}(T)$ is the membership degree of milk volume. Next, the outputs of all activated rules are combined using the maximum (max) operator, as shown in Equation 5. This aggregated fuzzy set represents the final fuzzy output before defuzzification [11], [13].

$$\mu_{output}(z) = \max(\alpha_1, \alpha_2, \cdots, \alpha_n) \quad (5)$$

The fuzzy logic rule used in this study is presented in Table II.

TABLE II. THE RULE OF FUZZY LOGIC IN THIS STUDY

No.	**Input Variable Classifications**		
	Temperature parameters	**Milk Volume**	**Output**
1	Hot	Empty	Level 0
2	Hot	Low	Level 0
3	Hot	Medium	Level 0

No.	Input Variable Classifications		
	Temperature parameters	Milk Volume	Output
4	Hot	Full	Level 0
5	Warm	Empty	Level 0
6	Warm	Low	Level 1
7	Warm	Medium	Level 1
8	Warm	Full	Level 1
9	Normal	Empty	Level 0
10	Normal	Low	Level 2
11	Normal	Medium	Level 3
12	Normal	Full	Level 3
13	Cold	Empty	Level 0
14	Cold	Low	Level 2
15	Cold	Medium	Level 2
16	Cold	Full	Level 2

Description: Output represents the percentage of the heater power, which can be described as follows: Level 0: 0%, Level 1: 20 – 45%, Level 2: 50 – 70%, Level 3: 85 – 100%

The output of the fuzzy logic controller is heater power, expressed as a Pulse Width Modulation (PWM) signal. The output fuzzy sets (Level 0, Level 1, Level 2, and Level 3) represent the percentage of the PWM duty cycle associated with the heater power condition.

The last step of Mamdani fuzzy logic implementation is defuzzification. Defuzzification converts the aggregated fuzzy output into a crisp numerical value. Several defuzzification methods can be used, including Bisector of Area (BOA), Center of Area (COA), Mean of Maximum (MOM), Smallest of Maximum (SOM), and Largest of Maximum (LOM) [13], [15]. In this study, the COA method was used for defuzzification. The general equation of the COA method is shown in Equation 6

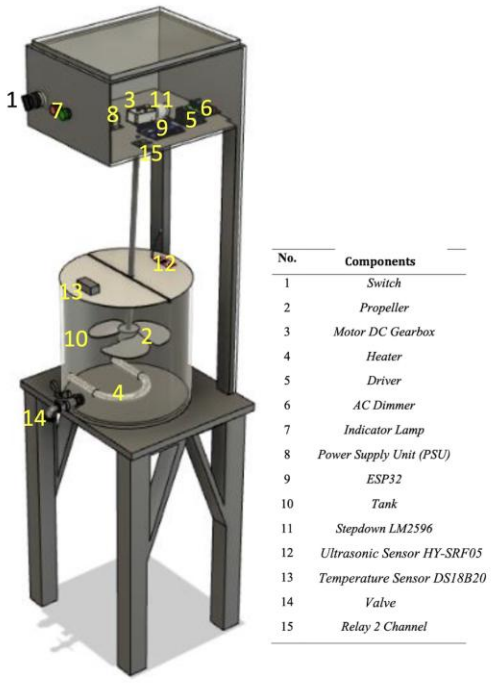
$$z_{COA} = \frac{\int_Z \mu_A(z) \cdot z \, dz}{\int_Z \mu_A(z) dz} \qquad (6)$$

Where Z_{COA} is crisp output, $\mu_A(z)$ is the membership function of the aggregated output z.

$$z_{trapezoid} = \frac{a+2b+2c+d}{6}, \qquad (7)$$

$$z_{triangle} = \frac{a+b+c}{3}. \qquad (8)$$

The crisp value Z obtained from defuzzification is then converted into a PWM duty cycle, which is applied to the AC Light Dimmer to control the heater power. This process is repeated continuously in real time to ensure stable temperature control during pasteurization. The process has been monitored using an IoT system, which is connected to the hardware via TCP/IP over Wi-Fi within a local area network. Fig. 3 shows the 3D model of the hardware system. Then the monitoring system is shown in Fig. 4. Then, The overall steps are illustrated in the flowchart in Fig. 5.



No.	Components
1	*Switch*
2	*Propeller*
3	*Motor DC Gearbox*
4	*Heater*
5	*Driver*
6	*AC Dimmer*
7	*Indicator Lamp*
8	*Power Supply Unit (PSU)*
9	*ESP32*
10	*Tank*
11	*Stepdown LM2596*
12	*Ultrasonic Sensor HY-SRF05*
13	*Temperature Sensor DS18B20*
14	*Valve*
15	*Relay 2 Channel*

Fig. 3. The 3D design of the system

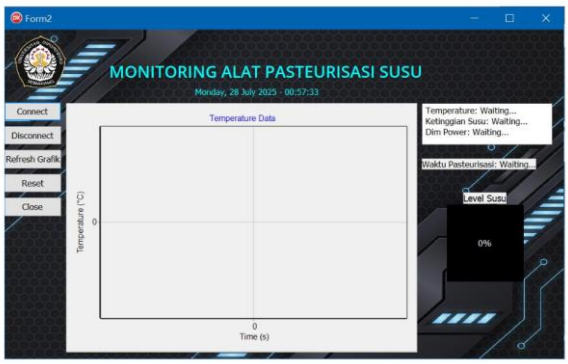


Fig. 4. The Flowchart of the system

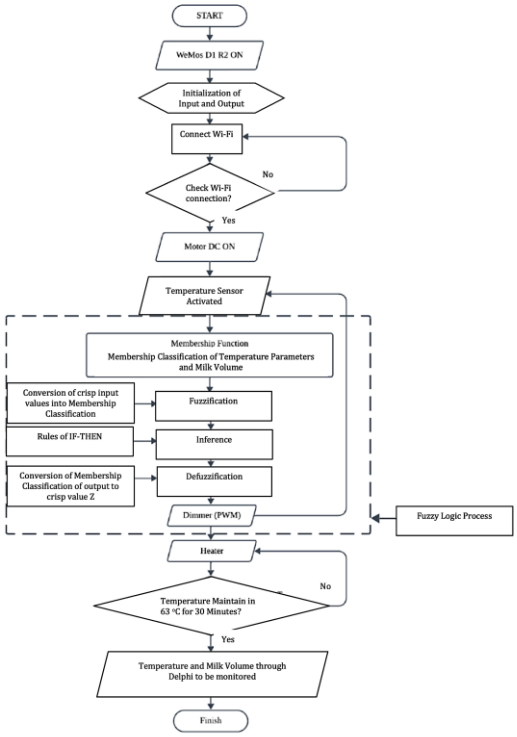


Fig. 5. The monitoring dashboard of the system

The block diagram of the proposed pasteurization system is presented in Fig. 6. Fig. 6(a) presents the hardware and IoT system, while Fig. 6(b) shows how the milk, fuzzy logic, and IoT components interact within this framework. The communication of the IoT components in this study uses TCP/IP over Wi-Fi within a local area network.

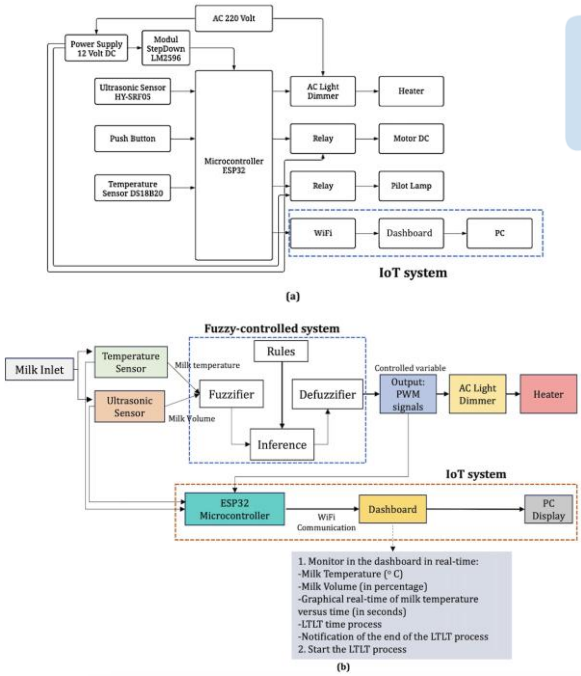


Fig. 6. The block diagram of the proposed pasteurization system, including (a) the block diagram of hardware and IoT system, and (b) the block diagram of the interaction among the milk, fuzzy logic, and IoT components within this framework

III. RESULTS AND DISCUSSION

This study tests the membership results using three milk-volume categories: 3 liters, 5 liters, and 8 liters. The 3-liter category indicates low milk volume, 5 liters represents medium volume, and 8 liters signifies full volume. A case-scenario approach to fuzzy set membership was used to optimize the membership function, which was subsequently analyzed. The case-scenario process for fuzzy set membership is presented in Table III. The results of the case-scenario in Table III are shown in Fig. 7 for milk volumes of 3 liters, 5 liters, and 8 liters.

TABLE III. THE CASE-SCENARIO MEMBERSHIP OF FUZZY LOGIC IN THIS STUDY

No.	Trials	Temperature Parameters (°C)	Membership in Temperature Parameters Fuzzy Set	Milk Volume (Liters)	Membership in Milk Volume Fuzzy Set	Output Parameters	Membership in Output Parameters
1.	Case Scenario 1	Cold	(-6, 0, 10, 20)	Empty	(0, 0, 10, 20)	Level 0	(0, 0, 0, 0)
		Normal	(18, 30, 45, 64)	Low	(20, 30, 30, 45)	Level 1	(10, 25, 40, 60)
		Warm	(63.5, 64, 64, 65.5)	Medium	(45, 50, 50, 70)	Level 2	(60, 70, 80, 90)
		Hot	(65, 66, 75, 80)	Full	(65, 75, 100, 100)	Level 3	(90, 100, 100, 100)
2.	Case Scenario 2	Cold	(-6, 0, 10, 20)	Empty	(0, 0, 10, 20)	Level 0	(0, 0, 0, 0)
		Normal	(18, 30, 45, 63.56)	Low	(20, 30, 30, 45)	Level 1	(30, 40, 40, 50)
		Warm	(63, 63.69, 63.69, 64.5)	Medium	(45, 50, 50, 70)	Level 2	(60, 70, 80, 90)
		Hot	(64, 66, 75, 80)	Full	(65, 75, 100, 100)	Level 3	(90, 100, 100, 100)
3.	Case Scenario 3 (Final)	Cold	(-6, 0, 20, 45)	Empty	(0, 0, 10, 20)	Level 0	(0, 0, 0, 0)
		Normal	(35, 50, 50, 63)	Low	(20, 30, 30, 50)	Level 1	(30, 40, 40, 50)
		Warm	(62, 63.94, 63.94, 64.06)	Medium	(45, 60, 60, 70)	Level 2	(50, 60, 60, 70)
		Hot	(63.69, 66, 75, 80)	Full	(65, 75, 100, 100)	Level 3	(85, 90, 95, 100)

The results of the case-scenario in Table III are shown in Fig. 7 for milk volumes of 3 liters, 5 liters, and 8 liters. The comparison results show that the best membership parameters come from case scenario 3 (the Final parameters), as indicated by the blue line. As shown in Fig. 7(a), for a 3-liter milk volume, the average temperature is 64.11°C, with a minimum overshoot of 1.56% and a standard deviation of 0.26. In comparison, Case scenario 1 reaches 65.77°C, with a 4.00% overshoot and a standard deviation of 0.40. Similarly, Case scenario 2 reaches an average temperature of 64.44°C, with a 2.34% overshoot and a standard deviation of 0.37. For a 5-liter milk volume, the comparison results also indicate that the best membership parameters are from Case scenario 3 (the Final parameters), again marked by the blue line. As shown in Fig. 7(b), the average temperature is 64.07°C, with a minimum overshoot of 0.29% and a standard deviation of 0.08. In comparison, Case scenario 1 reaches 65.49°C, with a 3.03% overshoot and a standard deviation of 0.34. Similarly, Case scenario 2 reaches an average temperature of 64.63°C, with a 1.75% overshoot and a standard deviation of 0.20. For an 8-liter milk volume, the comparison results also demonstrate that the best membership parameters are from Case scenario 3 (the Final parameters), as indicated by the blue line.

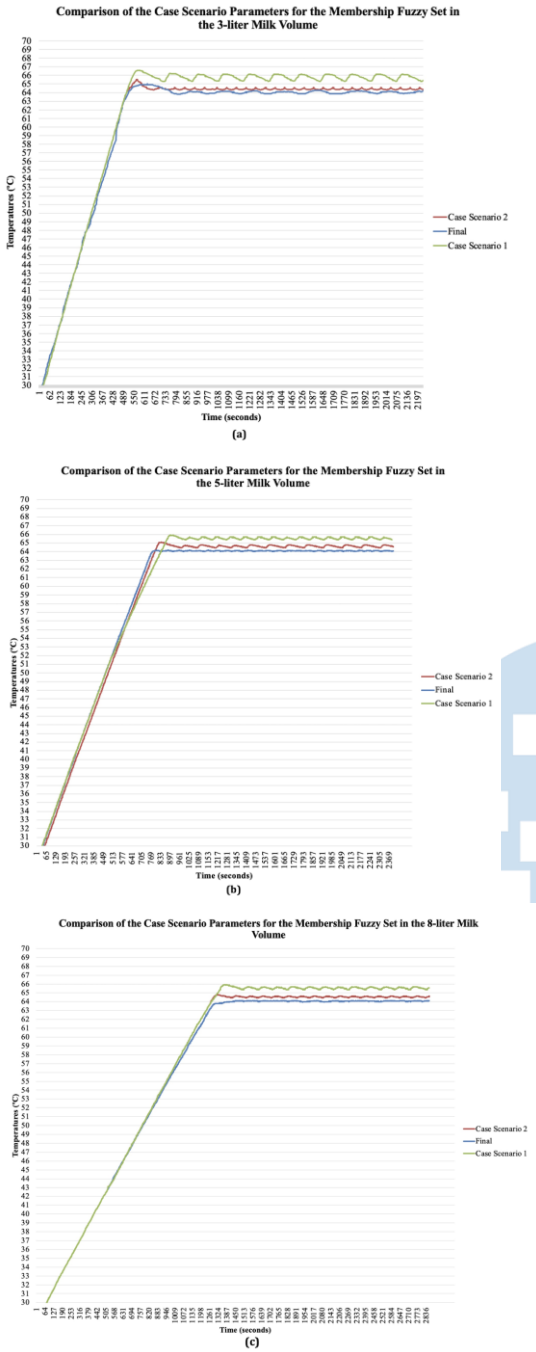


Fig. 7. The comparison result of the case-scenario parameters for the membership fuzzy set in (a) 3-liter milk volume, (b) 5-liter milk volume, and (c) 8-liter milk volume (the red line indicates case sceanrio 1, the blue line indicates case scenario 3 (Final).

As shown in Fig. 7(c), the average temperature is 64.03°C, with a minimum overshoot of 0.19% and a standard deviation of 0.12. In comparison, Case scenario 1 reaches 65.47°C, with a 3.03% overshoot and a standard deviation of 0.37. Similarly, Case scenario 2 reaches an average temperature of 64.52°C, with a 1.27% overshoot and a standard deviation of 0.27. Therefore, the optimal membership values for the fuzzy set parameters are listed in Table IV.

TABLE IV. THE OPTIMUM MEMBERSHIP OF FUZZY SET PARAMETERS

Input Variable Classifications			
Temperature parameters	**Temperature Range (° C)**	**Milk Volume**	**Milk Volume Range (%)**
Cold	-6 – 45	Empty	0 - 20
Normal	35 – 63	Low	20 – 50
Warm	62 – 64.06	Medium	45 – 70
Hot	63.69 – 80	Full	65 - 100

The performance of the Mamdani fuzzy logic control system developed in this study was quantitatively assessed and compared with the traditional on–off control method using several key performance metrics, including average temperature, set-point temperature tracking, temperature overshoot, rise time, standard deviation of temperature, and average PWM output of the AC light dimmer. These metrics were chosen because they directly indicate the effectiveness, stability, and efficiency of the LTLT pasteurization process. The set-point temperature in this study is 64 °C. Testing milk volumes of 3 liters, 5 liters, and 8 liters using a Mamdani fuzzy logic control system demonstrated improved temperature regulation compared with the traditional on-off control method, particularly in reducing overshoot as shown in Table V. The experimental results show that the Mamdani fuzzy logic controller maintains an average operating temperature that closely aligns with the target set-point of 63–65 °C. In various test volumes, the average temperature under fuzzy control remains reliably centered around the desired set-point, indicating effective tracking performance.

In contrast, the on-off control system exhibits a larger difference between the average temperature and the setpoint. Because the control action is binary (either fully ON or fully OFF), the system's temperature fluctuates more significantly around the target, causing the average temperature to drift outside the ideal pasteurization range. This indicates that the on-off controller lacks the resolution needed for precise temperature control in continuous thermal processes. Additionally, Mamdani fuzzy logic requires a lower PWM duty cycle than conventional on-off control, as shown in Table V. A comprehensive comparison of key performance metrics is presented in Table V.

TABLE V. PERFORMANCE OF MAMDANI FUZZY LOGIC WITH OPTIMAL FUZZY SET MEMBERSHIP PARAMETERS

Temperature Control System	Milk Volume (liters)	Key performance metrics					
		Average Temperature During Fuzzy Control (°C)	Set-point Temperature (°C)	Temperature Overshoot (%)	Rise Time (seconds)	Standard Deviation of Temperature	Average PWM Dimmer Duty Cycle During Fuzzy Control (%)
Mamdani Fuzzy Logic (This Study)	3	64.11	64	1.56	180	0.26	23.81
	5	64.07	64	0.29	210	0.08	37.89
	8	64.03	64	0.19	310	0.12	50.15
Traditional on-off control method	3	65.67	64	9.17	-	1.66	27.58
	5	64.28	64	1.56	-	0.45	35.99
	8	64.15	64	1.27	-	0.39	52.54

This study compares the Mamdani fuzzy logic controller to a PID control system [3] used for industrial milk pasteurization. Results indicate that the Mamdani controller is slower than the PID, with rise and settling times of 0.177 and 0.34 seconds, respectively, and no overshoot in ideal simulation conditions at a milk flow rate of 325 L/min [3]. Nonetheless, the Mamdani controller emphasizes long-term temperature stability and robustness amid varying milk volumes, as shown in Table V, without depending on mathematical models.

IV. Conclusion

This study demonstrates that a Mamdani-type fuzzy logic control system, combined with an Internet of Things (IoT) framework, can effectively oversee and regulate the Low-Temperature Long-Time (LTLT) milk pasteurization process. Using milk volume and temperature as input parameters, the proposed fuzzy controller can smoothly adjust heater power and keep the process temperature within the critical range of 63–65 °C for 30 minutes, ensuring microbial safety while maintaining milk quality.

Experimental results with different milk volumes (3 L, 5 L, and 8 L) indicate that the optimized fuzzy membership parameters markedly enhance control performance. The Mamdani fuzzy logic controller achieves temperatures that closely follow the target setpoint while minimizing overshoot and reducing temperature fluctuations, as evidenced by lower standard deviations than traditional on–off control methods. These findings demonstrate that fuzzy logic provides superior stability and accuracy for continuous thermal processes such as LTLT pasteurization. In addition, the fuzzy logic approach requires a lower average PWM duty cycle for the AC light dimmer than the on–off controller, indicating more efficient energy usage and smoother actuator operation. Although the fuzzy controller exhibits slower rise and settling times than PID control reported in prior industrial-scale studies, it offers greater robustness to variations in milk volume. It does not rely on an accurate mathematical model of the process. This makes the proposed system particularly suitable for small-scale, community-based milk-processing applications, such as those commonly found in the Indonesian cattle-farming sector.

Overall, combining Mamdani Fuzzy Logic with IoT-based monitoring provides a practical, stable, and energy-efficient solution for LTLT milk pasteurization. Future efforts could expand the system to achieve higher capacities, incorporate adaptive or hybrid control methods, and evaluate long-term reliability and scalability in real industrial settings.

ACKNOWLEDGMENT

The authors are grateful for the support from the Modeling and Control Laboratory of the Automation Engineering Technology Program, Vocational School, Universitas Diponegoro, which provided the necessary facilities for this research. Sincere thanks are extended to the mentor (Mr. Fakhruddin Mangkusasmito) for their guidance throughout this project. The authors also wish to acknowledge their colleagues for helpful discussions and their families for their unwavering support.

References

[1] M. Chitnis, S. Bhatawdekar, and A. A. Shinde, "Automated Pasteurizer Unit for On-Farm Pasteurization of Milk," *International Journal of Emerging Technology and Advanced Engineering*, vol. 5, no. 4, pp. 304–306, 2015, [Online]. Available: https://www.ijetae.com

[2] J. A. S. Leite *et al.*, "The effect of microwave-assisted heating on bioactive and immunological compounds in donor human milk," *LWT*, vol. 161, p. 113306, May 2022, doi: 10.1016/j.lwt.2022.113306.

[3] A. O. Amole, O. E. Olabode, D. O. Akinyele, and S. G. Akinjobi, "Optimal Temperature Control Scheme for Milk Pasteurization Process Using Different Tuning Techniques for a Proportional Integral Derivative Controller," *Iranian Journal of Electrical and Electronic Engineering*, vol. 18, no. 3, p. 2170, 2022, doi: 10.22068/IJEEE.18.3.2170.

[4] A. Šalaševičius, D. Uždavinytė, M. Visockis, P. Ruzgys, and S. Šatkauskas, "Comparative Analysis of Pulsed Electric Fields (PEF) and Traditional Pasteurization Techniques: Comparative Effects on Nutritional Attributes and Bacterial Viability in Milk and Whey Products," *Applied Sciences*, vol. 13, no. 22, p. 12127, Nov. 2023, doi: 10.3390/app132212127.

[5] Y. Triwidyastuti, M. Nizar, H. Harianto, and J. Jusak, "Pengendali Suhu pada Proses Pasteurisasi Susu dengan Menggunakan Metode PID dan Metode Fuzzy Sugeno," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 6, no. 4, p. 355, Jul. 2019, doi: 10.25126/jtiik.2019641068.

[6] A. Rabbani, M. Ayyash, C. D. C. D'Costa, G. Chen, Y. Xu, and A. Kamal-Eldin, "Effect of Heat Pasteurization and Sterilization on Milk Safety, Composition, Sensory Properties, and Nutritional Quality," *Foods*, vol. 14, no. 8, p. 1342, Apr. 2025, doi: 10.3390/foods14081342.

[7] K. K. Dash *et al.*, "A comprehensive review on heat treatments and related impact on the quality and microbial safety of milk and milk-based products," *Food Chemistry Advances*, vol. 1, p. 100041, Oct. 2022, doi: 10.1016/j.focha.2022.100041.

[8] Mokh. S. Hadi, B. S. R. Sugiono, M. A. Mizar, A. Witjoro, and M. Irvan, "Enhancing low-temperature long-time milk pasteurization process with a C4.5 algorithm-driven AIoT system for real-time decision-making," *J Food Process Eng*, vol. 47, no. 4, Apr. 2024, doi: 10.1111/jfpe.14606.

[9] A. I. Dounis, P. Kofinas, C. Alafodimos, and D. Tseles, "Adaptive fuzzy gain scheduling PID controller for maximum power point tracking of photovoltaic system," *Renew Energy*, vol. 60, pp. 202–214, Dec. 2013, doi: 10.1016/j.renene.2013.04.014.

[10] L. N. Pattanaik, R. Agrawal, and L. Kumari, "Model of a fuzzy autonomation system for a steel wire roll mill," in *2016 3rd International Conference on Recent Advances in Information Technology (RAIT)*, IEEE, Mar. 2016, pp. 274–278. doi: 10.1109/RAIT.2016.7507915.

[11] T. J. Ross, *Fuzzy Logic with Engineering Applications*. in Third Edition. Chichester, West Sussex: John Wiley & Sons, Ltd., 2010.

[12] J. Li and Z. Gong, "SISO Intuitionistic Fuzzy Systems: IF-${t}$ -Norm, IF-${R}$ -Implication, and Universal Approximators," *IEEE Access*, vol. 7, pp. 70265–70278, 2019, doi: 10.1109/ACCESS.2019.2918169.

[13] S. N. Putri and D. R. S. Saputro, "Construction fuzzy logic with curve shoulder in inference system mamdani," *J Phys*

Conf Ser, vol. 1776, no. 1, p. 012060, Feb. 2021, doi: 10.1088/1742-6596/1776/1/012060.

[14] A. A. I. D. Fibriayora, G. K. Gandhiadi, N. K. T. Tastrawati, And I. P. E. N. Kencana, "Application Of Mamdani Fuzzy Method To Determine Round Bread Production At Pt Vanessa Bakery," E-Jurnal Matematika, vol. 8, no. 3, p. 204, Aug. 2019, doi: 10.24843/MTK.2019.v08.i03.p254.

[15] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. Upper Saddle River, NJ, USA: Prentice Hall, 1997.

[16] A. M. A. Hamad and D. A. B. Awad, "Integration of IoT in Dairy Facilities," in Next-Generation Technologies in Dairy Processing and Production, IGI Global, 2025, pp. 139–170. doi: 10.4018/979-8-3693-9949-1.ch005.

[17] B. Uzun and N. Allahverdı, "Fuzzy Control System for Predicting The Quality of Thermally Processed Raw Milk," International Journal of Applied Mathematics Electronics and Computers, vol. 9, no. 3, pp. 79–84, Sep. 2021, doi: 10.18100/ijamec.980645.

ISSN 2355-3286

Early Detection of Non-Melanoma Skin Lesions: A ResNet50 and SVM-Based Deep Learning Approach

Ray Louie D'Angelito[1], Josephine Larissa Rachmadiana[2], Muhammad Nur Rohman[3], Timothy Wirjantoro Harjanto[4], I Nyoman Julian Sanjaya[5], Yuri Pamungkas[6*]
[1,2,3,4,5,6] Department of Medical Technology, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[1]5049231008@student.its.ac.id ; [2]5049231105@student.its.ac.id ; [3]5049231040@student.its.ac.id ; [4]5049231098@student.its.ac.id ; [5]5049231059@student.its.ac.id ; [6*]yuri@its.ac.id
* Corresponding Author

Accepted on February 11, 2026
Approved on May 11, 2026

***Abstract*—Skin cancer is a rapidly growing global health issue that demands efficient and accurate early detection. While much research focuses on aggressive types like melanoma, basal cell carcinoma, and squamous cell carcinoma, there is also a critical need for non-invasive classification of non-melanoma skin lesions, including actinic keratosis, dermatofibroma, nevus, and vascular lesions. These conditions can also progress to more severe malignancies or indicate underlying health issues. This study presents a computer-aided detection method for these skin conditions by employing deep learning techniques, specifically a ResNet50-based Convolutional Neural Network (CNN), alongside a Support Vector Machine (SVM) classifier. The aim is to improve diagnostic accuracy and accessibility through image data processing and feature extraction. The main contribution of this research is the application of deep learning for automated classification of non-melanoma skin lesions, with the goal of enhancing early detection. The models were trained and evaluated using the International Skin Imaging Collaboration (ISIC) dataset, with two test scenarios to assess their performance. In Test 4, the CNN demonstrated superior results, achieving F1-scores of 44.70% for actinic keratosis, 85.25% for dermatofibroma, 78.76% for nevus, and a perfect 100.00% for vascular lesion. In comparison, the SVM model achieved lower F1-scores: 21.88% for actinic keratosis, 27.91% for dermatofibroma, 62.46% for nevus, and 70.58% for vascular lesion. The results highlight the effectiveness of deep learning, particularly CNNs, in automated dermatological diagnosis. These findings lay the groundwork for future web and mobile applications that could support early skin cancer detection and clinical decision-making.**

***Index Terms*—Skin Cancer Detection; Deep Learning; Convolutional Neural Network (CNN); Support Vector Machine (SVM); Non-Melanoma Skin Lesions**

I. INTRODUCTION

Skin cancer is among the most prevalent malignancies worldwide, with cases steadily increasing each year [1]. Although clinical and research efforts have largely concentrated on highly aggressive types such as melanoma, basal cell carcinoma, and squamous cell carcinoma, the significance of non-melanoma skin lesions (including actinic keratosis, dermatofibroma, nevus, and vascular lesions) remains critical [2]. These lesions are not only widespread but can also progress into more severe malignancies or reflect underlying systemic diseases if not detected in time.

Actinic keratosis, for instance, is a pre-cancerous lesion that may advance to squamous cell carcinoma when left untreated [3]. Dermatofibroma, while usually benign, often resembles malignant tumors under dermoscopy and can complicate diagnosis [4]. Nevi, commonly referred to as moles, are mostly harmless but certain atypical types have malignant potential and can evolve into melanoma [5]. Likewise, vascular lesions can signal deeper vascular or systemic abnormalities [6]. These examples underline the necessity of accurate and early diagnosis for improving patient outcomes and reducing mortality.

The research problem lies in the persistent difficulty of differentiating these lesions at an early stage due to their overlapping morphological characteristics, which can lead to misdiagnosis and delayed treatment [7]. Conventional diagnostic methods, including dermoscopic evaluation, rely heavily on the expertise of dermatologists and may be subject to inter-observer variability [8]. Therefore, there is a pressing need for automated, reliable, and scalable solutions to assist clinicians in improving diagnostic accuracy.

To address this challenge, deep learning has emerged as a powerful solution, with Convolutional Neural Networks (CNNs) proving highly effective in analyzing medical images [9]. CNNs automatically extract complex visual features, making them particularly well-suited for dermatological image classification [10]. State-of-the-art models such as ResNet50 have shown robust performance in image recognition tasks due to their residual connections that

overcome the vanishing gradient problem, allowing for deeper and more accurate networks [11]. In parallel, Support Vector Machines (SVMs) have also been utilized in prior research, demonstrating strong performance in distinguishing between benign and malignant lesions by finding optimal separating hyperplanes in high-dimensional feature spaces [12].

This study builds upon these advances by employing ResNet50 to classify actinic keratosis, dermatofibroma, nevus, and vascular lesions using a publicly available dataset from the International Skin Imaging Collaboration (ISIC) [13]. By comparing the deep learning approach with SVMs, the research seeks to evaluate the strengths and limitations of both methods in the context of multiclass classification of non-melanoma lesions.

The novelty of this study lies in its specific focus on non-melanoma skin lesions, which have often received less attention compared to melanoma, while leveraging advanced deep learning architecture and benchmarking it against traditional machine learning methods [14],[15]. This dual approach provides a more comprehensive perspective on automated lesion detection and highlights the feasibility of integrating such methods into accessible diagnostic platforms.

The contribution of the research is twofold: first, it provides an in-depth evaluation of ResNet50 and SVM performance in the classification of four clinically significant non-melanoma skin lesions, and second, it offers insights into the potential integration of these computational models into web- and mobile-based applications for wider accessibility. Ultimately, this work contributes to the ongoing effort of enhancing early detection and supporting dermatologists in clinical decision-making.

II. Research Methods

This research proposes an early detection method for non-melanoma skin lesions using a ResNet50-based CNN and Support Vector Machine (SVM) classifier. The process begins with dataset preparation, where images are resized to 224x224 pixels, followed by data augmentation techniques such as flipping, rotation, and color shifting to enhance model robustness. Image segmentation isolates the lesions, focusing the model on relevant features. The ResNet50-based CNN extracts features, which are then classified into categories like nevus, dermatofibroma, actinic keratosis, and vascular lesions using SVM. The model's performance is evaluated through metrics like accuracy, precision, recall, F1-score, AUC, and specificity, with Grad-CAM visualization used to highlight the areas of the image the model focuses on, improving interpretability. This approach enables accurate, automated classification, assisting clinicians in early detection and diagnosis, as illustrated in **Figure 1**.

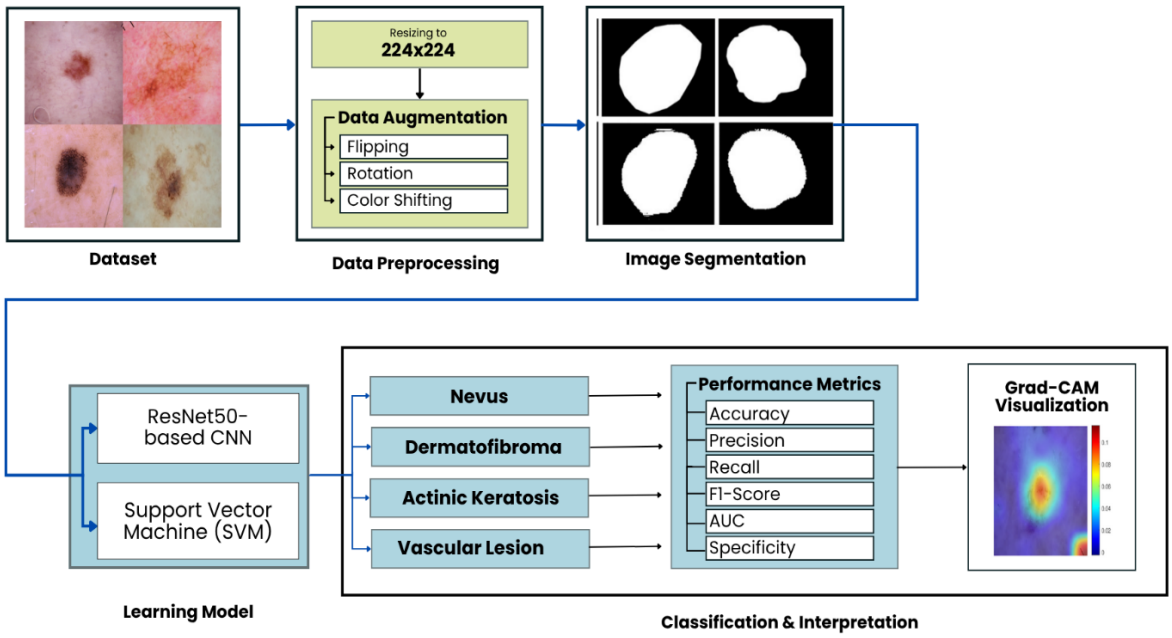


Fig. 1. Methodology of the research

A. Related Works

The early detection of non-melanoma skin lesions has been a focal point in recent dermatological research, with significant progress achieved through the use of deep learning techniques. Akinrinade et al. (2025) demonstrated that deep learning models, specifically CNNs and GANs, combined with transfer learning and few-shot learning, improve skin cancer detection, particularly melanoma [16]. Their work showed how these models can address dataset imbalances and small data challenges, highlighting the value of using data augmentation techniques. Their findings indicate that the use of transfer learning with ResNet50 achieved an impressive accuracy of 96%, making it a viable model for skin cancer detection in clinical settings.

Similarly, Erbay et al. (2025) explored the combination of CNNs and Vision Transformers (ViTs) with ensemble learning approaches, such as hard and soft voting, to improve skin lesion classification [17]. Their study, based on the HAM10000 dataset, found that ensemble methods combining CNNs and ViTs yielded the best performance, achieving an accuracy of 89% and an F1-score of 0.80. This ensemble learning approach, which leverages the strengths of multiple models, has shown promise in enhancing classification accuracy, especially when applied to large, imbalanced datasets.

Another study by Zareen et al. (2024) employed a hybrid CNN-RNN architecture combining ResNet-50 and LSTM to classify skin lesions [18]. This approach achieved an impressive accuracy of 99.06% on the ISIC dataset. The incorporation of RNNs for temporal learning enabled this model to handle various skin cancer types effectively. The high accuracy and superior performance of this hybrid model underline the potential of combining different deep learning techniques for more robust and accurate skin lesion classification.

In a different approach, Nalamwar et al. (2025) introduced a multidimensional CapsNet with attention mechanism, achieving an accuracy of 97.63% with high precision and recall (98.11% and 98.73%, respectively) [19]. This model outperformed traditional CNNs and GoogLeNet, offering a significant improvement in skin cancer classification. The use of attention mechanisms in CapsNets allows the model to focus on important features, which is crucial for detecting skin lesions that exhibit subtle variations.

Manickavasagam et al. (2025) applied CNN-based transfer learning, enhanced with harmonic brown bear optimization, to the SIIM-ISIC Melanoma dataset [20]. Their model achieved an accuracy of 91.75%, with a true positive rate (TPR) of 93.76% and a true negative rate (TNR) of 89.77%. The use of optimization techniques to enhance transfer learning models resulted in superior skin cancer detection performance, reducing false positives and improving overall diagnostic accuracy.

The use of hybrid models combining CNN and ViT has also shown promising results. Pacal et al. (2025) introduced a CNN-ViT hybrid model with focal self-attention, yielding accuracy rates of 92.54% on the ISIC 2019 dataset and 95.01% on the HAM10000 dataset [21]. Their approach demonstrated that hybrid models are well-suited for real-time deployment, improving both accuracy and efficiency in skin lesion classification.

In summary, these works illustrate the significant strides made in deep learning for skin cancer detection, with a variety of models and approaches contributing to improved accuracy and robustness. The integration of multiple AI techniques, including CNNs, ViTs, RNNs, and ensemble methods, has proven to be effective in addressing the challenges of skin lesion classification, particularly in non-melanoma skin lesions. This research paves the way for the development of advanced AI-driven systems for the early detection of skin cancers, improving clinical decision-making and patient outcomes. A summary of related works is presented in **Table I**.

TABLE I. RESEARCH RELATED TO SKIN CANCER DETECTION BASED ON DEEP AND MACHINE LEARNING

Author(s)	Year	AI Method	Dataset	Metrics Evaluation Value	Key Findings*
Akinrinade, et al [16]	2025	Deep Learning (CNN, GAN, Transfer Learning)	ISIC 2016, ISIC 2017, HAM10000	Accuracy: 96%, Sensitivity: 91.1%, Specificity: 95.7% (ResNet50), AUC: 0.99	Deep learning models improve skin cancer detection, especially melanoma.
Erbay, et al [17]	2025	CNN, Vision Transformer, Ensemble Learning	HAM10000	Accuracy: 0.89% (ensemble), F1-score: 0.80	Ensemble learning improves skin lesion classification accuracy.
Zareen, et al [18]	2024	Hybrid CNN-RNN (ResNet-50 + LSTM)	ISIC (9000 images)	Accuracy: 99.06%	Achieved high accuracy in classifying skin cancer types.
Nalamwar, et al [19]	2025	Multidimensional CapsNets with Attention	HAM10000	Accuracy: 97.63%, Precision: 98.11%, Recall: 98.73%, F1-score: 98.42%	Outperforms CNN and GoogLeNet with high precision and recall.
Manickavasagam, et al [20]	2025	CNN-based Transfer Learning (DenseNet)	SIIM-ISIC Melanoma, Skin Cancer	Accuracy: 91.75%, TPR: 93.76%, TNR: 89.77%	Superior performance with enhanced precision and reduced false positives.

Author(s)	Year	AI Method	Dataset	Metrics Evaluation Value	Key Findings*
Pacal, et al [21]	2025	CNN-ViT Hybrid	ISIC 2019, HAM10000	ISIC 2019: Accuracy: 92.54%, Precision: 90.41%, Recall: 87.68%, F1: 88.86%; HAM10000: Accuracy: 95.01%, Precision: 94.70%, Recall: 92.11%, F1: 93.34%	Superior performance, high accuracy, and real-time deployment.
ThangaPurni, et al [22]	2025	Hybrid ARP-ViT-CNN	HAM10000	Accuracy: 98.2%, Precision: 0.94, Recall: 0.96, F1-score: 0.95	Outperforms CNN-only and ViT-only models with high accuracy.
Khattar, et al [23]	2025	Hybrid DeepCNN	HAM10000	PSNR: 45.56, MSE: 2.43, SSIM: 0.9781	Hybrid method improves image quality, enhancing segmentation and classification.
Reis, et al [24]	2024	MABSCNET (CNN + ViT)	ISIC 2020, ISIC 2018, Kaggle	ISIC 2020: Accuracy: 92.74%, ISIC 2018: 100%, Kaggle: 94.24%	Hybrid model outperforms CNN and ViT in skin cancer detection.
Nanda, et al [25]	2025	CNN, GoogleNet, ResNet	HAM10000	Accuracy: CNN 73%, GoogleNet 71%, ResNet 47%; Precision, Recall, F1-Score, AUC-ROC	CNN with mean sampling performs best, addressing class imbalance.

* "Key Findings" refers to the primary methodological or performance-related insight reported by each study that is most relevant to the classification of skin lesions using deep or machine learning.

B. Dataset

The dataset used in this study was obtained from the ISIC 2019 Challenge on Kaggle [26]. The ISIC dataset is one of the most widely recognized benchmarks in dermatological image analysis, compiled through a collaboration between leading skin cancer research institutions worldwide. It was specifically developed to support the training and evaluation of automated skin lesion classification systems, providing standardized dermoscopic images with expert-validated annotations [13]. The dataset spans a broad spectrum of skin conditions, enabling controlled evaluation of model performance across both common and rare lesion types. Its use in this study ensures comparability with prior work and provides a reliable foundation for assessing the proposed method. A curated subset of 962 images was selected to focus on four specific skin lesion types: actinic keratosis (329 images), dermatofibroma (122 images), nevus (369 images), and vascular lesion (142 images). To meet the input requirements of the ResNet50 model, the images were resized to 224×224 pixels [27]. To address class imbalance and enhance the model's robustness, data augmentation techniques including horizontal and vertical flipping, random rotation, and color shifting (adjustments to brightness, contrast, and saturation) were applied [28],[29]. Among these, flipping and rotation are expected to be particularly beneficial for minority classes such as dermatofibroma and vascular lesion, as they generate geometrically diverse variants of lesions whose appearance is orientation-independent — meaning a dermatofibroma looks the same regardless of rotation, so each rotated copy is a genuinely useful training sample. Color shifting, while useful overall, is likely to be most impactful for vascular lesions, which are characterized by distinct reddish hues that benefit from intensity variation to prevent the model from relying solely on color cues. Future work should consider class-specific augmentation strategies, such as applying higher rotation multipliers or synthetic oversampling (e.g., SMOTE) exclusively to the underrepresented classes, to more directly address the imbalance rather than applying a uniform augmentation pipeline across all classes. The final dataset consists of 2,357 images, representing both benign and malignant skin conditions. Although the original ISIC dataset includes a variety of skin conditions like basal cell carcinoma, melanoma, and pigmented benign keratosis, this study narrowed its focus to the four clinically significant conditions mentioned above. This targeted approach allows for a more detailed evaluation of the model's performance on these specific lesion types.

C. Feature Extraction and Selection

In this study, two different approaches to obtaining features from images were employed, depending on the classifier. For the CNN-based model, the process is one of automatic feature extraction: the network learns and derives hierarchical representations directly from raw pixel data without any manual intervention. Through multiple convolutional layers, CNNs gradually develop abstract and discriminative features, starting from basic visual elements like edges and textures to more complex semantic patterns [30]. This process eliminates the need for manual feature engineering, making traditional methods like MRMR (Minimum Redundancy Maximum Relevance) unnecessary for the CNN model.

For the Support Vector Machine (SVM) classifier, handcrafted feature extraction was performed manually using standard image processing techniques. A total of 15 features were derived, covering first-order statistics, shape descriptors, and texture features. No automated feature selection algorithm (such as MRMR) was applied; instead, all 15 features were used collectively as input to the SVM, representing a deliberate feature extraction (not selection) strategy. These features typically include morphological properties (such as

area and perimeter), intensity distributions (like mean, standard deviation, and histogram width), and texture features (e.g., Gray Level Co-occurrence Matrix (GLCM) and Gray Level Size Zone Matrix (GLSZM)). Although the specific number of features or the use of a feature selection algorithm like MRMR was not explicitly detailed for the SVM, it is assumed that a comprehensive set of relevant features was extracted and used, which is a common approach in traditional machine learning for image analysis.

D. Classifiers

This research compares the performance of two distinct machine learning approaches for classifying skin lesions:

1) CNN (ResNet50 Architecture)

The ResNet50 model was chosen for its strong performance in complex image classification tasks. Its architecture enables the training of very deep networks, addressing the vanishing gradient problem through residual connections [31]. In this study, the ResNet50 model was fine-tuned and trained on preprocessed dermatoscopic images to classify them into four categories of skin lesions: actinic keratosis, dermatofibroma, nevus, and vascular lesion.

2) Support Vector Machine (SVM)

SVM is a well-established supervised machine learning algorithm known for its effectiveness in high-dimensional data classification [32],[33]. The SVM works by creating a hyperplane, or a set of hyperplanes, in a high-dimensional space that separates data points into distinct classes. For this research, the SVM classifier was trained on handcrafted features extracted from the dermatoscopic images, allowing for a comparative analysis with the deep learning approach.

E. Metrics Evaluation

The performance of both the CNN and SVM classification models was thoroughly evaluated using standard metrics derived from the confusion matrix. The confusion matrix (**Figure 2**) offers a comprehensive view of the model's predictions, clearly distinguishing between correct and incorrect classifications for each class.. For multi-class classification, the metrics were calculated using a one-vs-all approach for each class. This method ensures that the metrics are computed independently for each class, addressing the issue of imbalanced class distributions [34],[35].

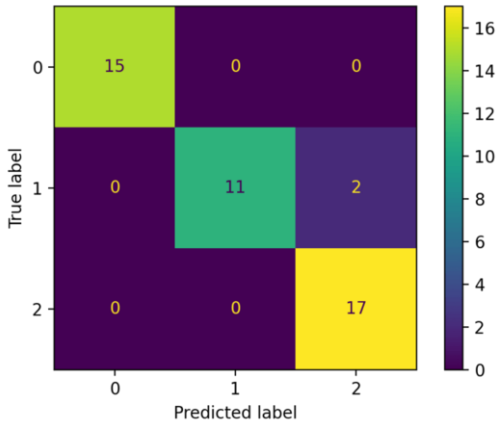


Fig. 2. Example of Confusion Matrix [36]

The fundamental counts extracted for each class (actinic keratosis, dermatofibroma, nevus, vascular lesion) are:

a. True Positives (TP): Instances correctly predicted as positive for the class.

b. True Negatives (TN): Instances correctly predicted as negative for the class.

c. False Positives (FP): Instances incorrectly predicted as positive for the class (Type I error).

d. False Negatives (FN): Instances incorrectly predicted as negative for the class (Type II error).

From these counts, the following performance metrics were calculated:

1) Accuracy (Acc): The overall proportion of correct predictions across all classes [37].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (1)$$

2) Precision (Prec): The proportion of correctly predicted positive instances out of all instances predicted as positive for a specific class [38].

$$Precision = \frac{TP}{TP + FP} \times 100 \quad (2)$$

3) Recall / Sensitivity (Sens): The proportion of actual positive instances that were correctly identified by the model [39].

$$Sensitivity = \frac{TP}{TP + FN} \times 100 \quad (3)$$

4) Specificity (Spec): The proportion of actual negative instances correctly identified by the model (true negative rate) [40].

$$Specificity = \frac{TN}{TN + FP} \times 100 \quad (4)$$

5) F1-Score: The harmonic mean of Precision and Recall, providing a balanced measure of the model's accuracy, especially useful in cases of imbalanced class distributions [41].

$$F1 = 2 \times \frac{precision \times sensitivity}{precision + sensitivity} \times 100 \quad (5)$$

6) Area Under the Receiver Operating Characteristic Curve: A comprehensive measure of a model's ability to distinguish between classes, representing the probability that the model will rank a randomly chosen positive example higher than a randomly chosen negative example [42].

III. Results and Discussions

A. Dataset Overview and Preprocessing

The study utilized a comprehensive skin cancer dataset designed to capture the variability of four distinct classes: actinic keratosis, dermatofibroma, nevus, and vascular lesion. These conditions, though categorized under skin cancer, vary significantly in clinical presentation and histopathological features, making them challenging for automated classification. The dataset included images that were preprocessed to 224×224 pixels, which is the optimal size for input into the ResNet-50 architecture. The class distribution of the dataset is presented in **Table II**. This preprocessing step was crucial as it ensured that the images were consistent in dimensions and ready for efficient feature extraction, thereby improving the performance of the convolutional neural network (CNN) model [43].

TABLE II. Dataset Distribution

Class	Number of Images	Percentage of Total
Actinic Keratosis	329	34.2%
Dermatofibroma	122	12.7%
Nevus	369	38.4%
Vascular Lesion	142	14.8%
Total	**962**	**100%**

The dataset demonstrates a clear imbalance in the distribution of classes, with nevus comprising the largest proportion (38.4%), followed by actinic keratosis (34.2%), vascular lesions (14.8%), and dermatofibroma (12.7%) as the smallest class. This uneven distribution presents a challenge for machine learning algorithms, as models tend to exhibit bias toward the majority class. In practical terms, this means that the model may be more inclined to predict the larger classes (nevus and actinic keratosis) accurately, while struggling to correctly classify images from the smaller classes, especially dermatofibroma.

B. Training Performance Analysis

The training curve for the 30% training configuration demonstrates excellent convergence characteristics. The validation accuracy curve shows steady improvement throughout the epochs, reaching a plateau at approximately 90.38%, with minimal signs of overfitting. Notably, the smoothed training accuracy curve (represented by the blue line) illustrates rapid learning during the early epochs, suggesting that the model quickly adapts to the data. The validation points, depicted as black dots, show consistent performance without significant fluctuations, which indicates that the model is generalizing well. Furthermore, the loss curves show an inverse relationship with accuracy, which is expected. The training loss decreases steadily as the model learns, while the validation loss stabilizes, signifying that the model has achieved a state of effective generalization. These observations suggest that the model's hyperparameters were well-chosen and that transfer learning from the pretrained ResNet-50 weights contributed to the efficient convergence of the model. The steady performance and stable learning curves indicate that the model is benefiting from both the training data and pretrained weights, making this configuration quite effective [44]. The training progress for this configuration is shown in **Figure 3**.

The 25% training configuration produced the most remarkable results, achieving the highest validation accuracy of 90.69% despite using less training data than the 30% configuration. Interestingly, this result challenges the conventional assumption that more training data always leads to better performance. The convergence pattern for the 25% configuration shows superior stability when compared to the 30% configuration. The validation accuracy progression is smoother, with fewer oscillations, indicating that the model is learning more consistently. This suggests that the 25% configuration strikes an optimal balance, providing enough training data for the model to learn effectively while avoiding the risks of overfitting that often accompany larger datasets. The training loss curve also demonstrates a more consistent descent, while the validation loss stabilizes at a lower level compared to the 30% configuration. These observations confirm that the 25% configuration leads to better generalization capability, making it a counterintuitive but effective choice. This outcome underscores the importance of carefully selecting the amount of training data, as the model can still achieve high performance with less data when overfitting is controlled [45]. A plausible explanation for the superior stability of the 25% configuration over the 30% configuration lies in the interaction between dataset size and class

imbalance. With a larger test set (30%), the model is exposed to a higher absolute number of majority-class samples (nevus and actinic keratosis) during training, which may amplify the existing class imbalance and introduce additional noise into the gradient updates. This can cause the model to overfit to the dominant class distributions while reducing its generalization capacity for minority classes such as dermatofibroma. The 25% configuration, by retaining a slightly larger proportion of data for testing, inadvertently creates a more balanced training dynamic — the model receives just enough exposure to each class to develop robust representations without becoming biased toward any single distribution. The smoother validation loss curve observed in the 25% configuration supports this interpretation, as it indicates more consistent gradient updates and fewer conflicting learning signals across epochs. The training progress for this configuration is shown in **Figure 4**.

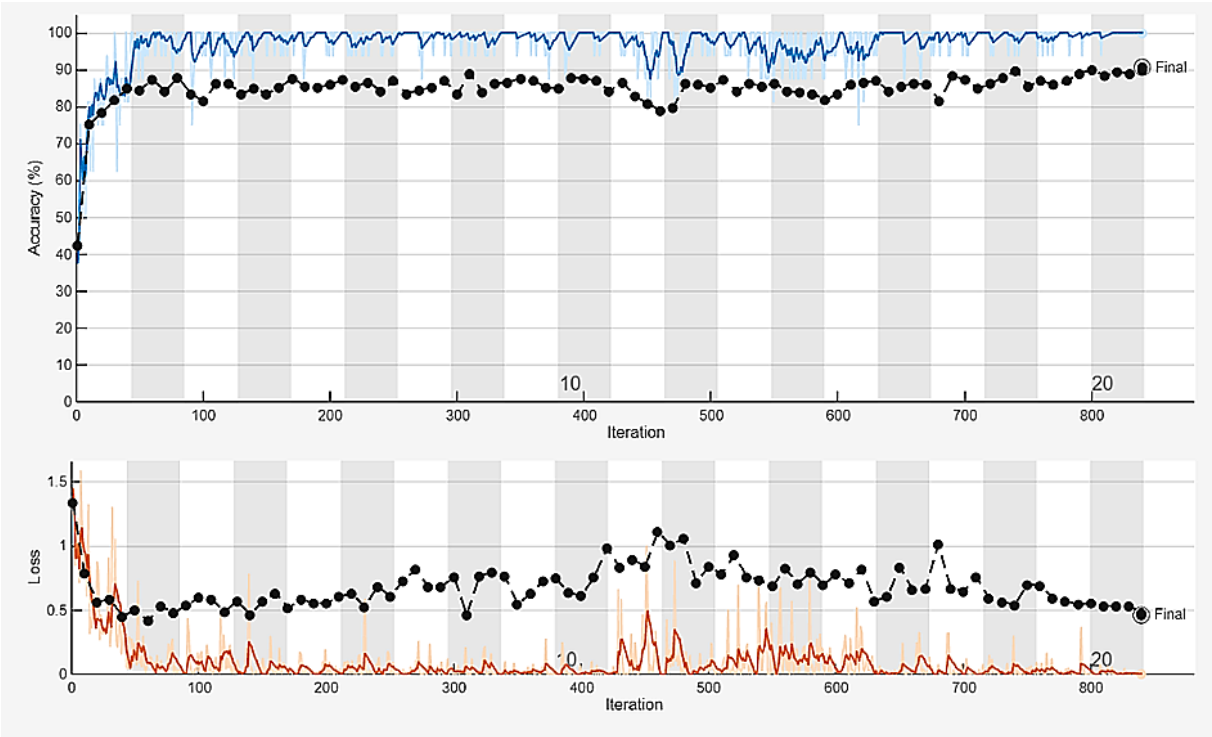


Fig. 3. Training progress of the CNN model with 30% data, showing accuracy and loss curves, with a final validation accuracy of 90.38%.

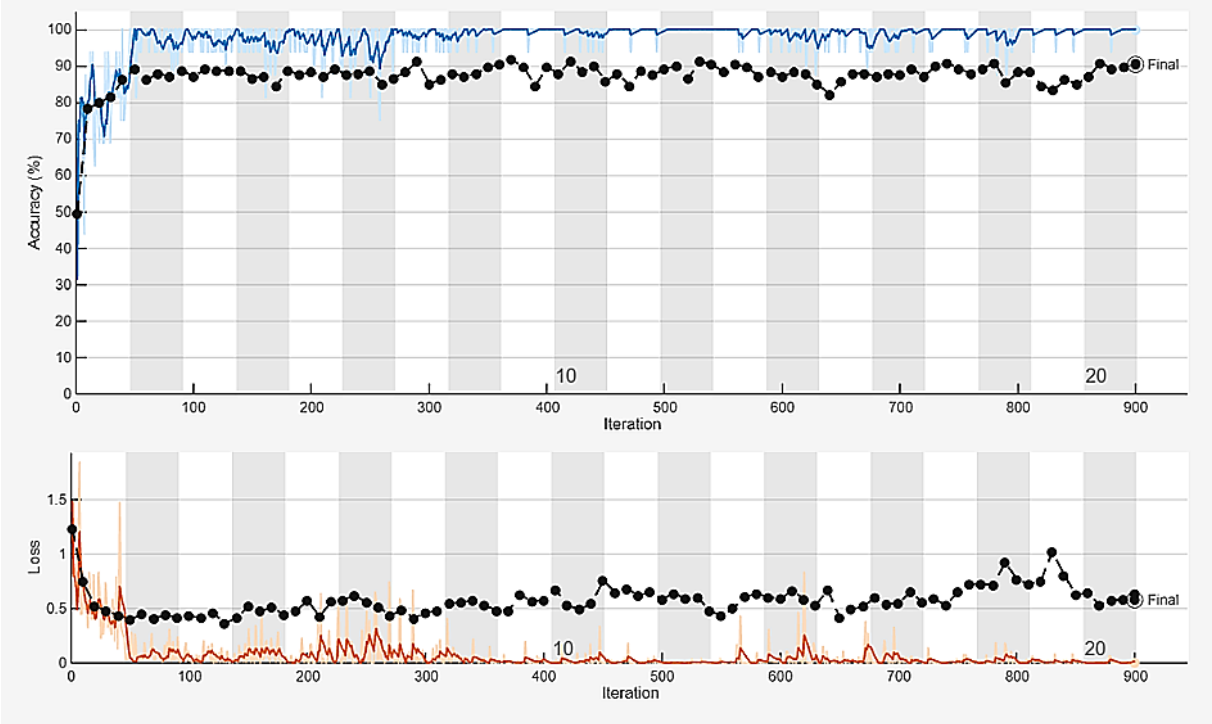


Fig. 4. Training progress of the CNN model with 25% data, showing accuracy and loss curves, with a final validation accuracy of 90.69%.

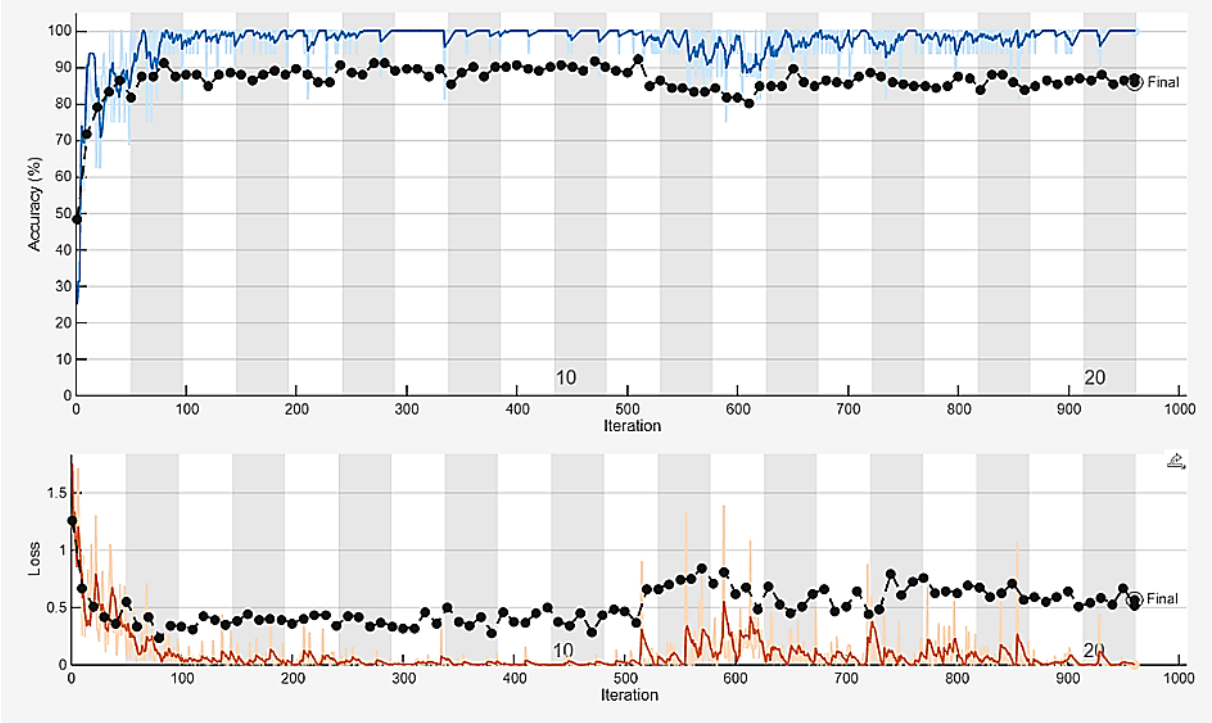


Fig. 5. Training progress of the CNN model with 20% data, showing accuracy and loss curves, with a final validation accuracy of 85.94%.

The 20% training configuration, which uses the least amount of training data, shows the expected performance degradation, with the model achieving a validation accuracy of 85.94%. This result highlights the limitations that come with using smaller datasets, as the model is constrained in terms of the amount of information it can learn. Despite this, the convergence pattern remains stable and consistent, indicating that the model architecture is robust even with minimal training data. The training curves exhibit more pronounced initial learning phases, which suggests that the model must extract the maximum amount of information from

the limited examples available. The validation accuracy plateaus earlier than with larger training sets, which implies that the model reaches its learning capacity constraints sooner. Even though the performance is lower compared to configurations with more data, the consistent and stable convergence pattern serves as evidence that the ResNet-50 architecture is still effective under these conditions. The model continues to perform reliably, demonstrating its robustness and suitability for handling smaller datasets [46]. The training progress for this configuration is shown in **Figure 5**.

The training metrics reveal several unexpected patterns that challenge conventional machine learning assumptions and provide deeper insights into the relationship between data size, training time, and model performance. Among the configurations tested, the 25% training data configuration achieved the highest validation accuracy of 90.69%, despite requiring the longest training time of 24 minutes and 54 seconds. This suggests that the model underwent more complex optimization dynamics with this configuration, possibly due to the data's balance between sufficient complexity for the model to learn and avoiding the overfitting risks associated with larger datasets. The longer training time, while initially seeming inefficient, may have provided the necessary learning steps for the model to fine-tune its weights, leading to the highest accuracy [47].

In contrast, the 30% configuration demonstrated the best efficiency ratio, achieving a validation accuracy of 90.38% in the shortest relative training time. This configuration completed the training process more quickly while maintaining a competitive accuracy score, indicating that the model was able to extract useful features from the larger dataset more rapidly, without sacrificing performance. The efficiency of the 30% configuration underscores the importance of not just raw training time, but the rate at which a model can learn and generalize from the data, reflecting a well-balanced trade-off between accuracy and speed.

On the other hand, the 20% training configuration, although achieving the lowest accuracy of 85.94%, completed the highest number of iterations (960), which suggests that the model underwent more frequent weight updates due to the smaller batch sizes. With fewer examples per batch, the model was updated more frequently, allowing for finer adjustments to the weights, albeit with less data to generalize from. This configuration highlights how the frequency of weight updates can impact the learning process, though it ultimately may not be sufficient for achieving high accuracy without more data.

These results indicate that optimal performance is not solely dependent on the volume of training data, but on the intricate balance between the amount of data, the training dynamics, and the model's capacity to learn and generalize. Simply increasing the training dataset size does not guarantee better outcomes; rather, it is essential to consider how data size interacts with training time, batch size, and the model's ability to capture and generalize features from the data. Thus, the findings suggest that a nuanced approach to training, considering not just the quantity of data but also the configuration of training parameters, can lead to more efficient and effective model performance. **Table III** summarizes the performance of each training configuration. Note that the test set size percentages (30%, 25%, 20%) correspond to training set sizes of 70%, 75%, and 80% of the total data, respectively

TABLE III. TRAINING CONFIGURATION PERFORMANCE SUMMARY

Test Set Size (Training Set Size)	Final Validation Accuracy	Training Time	Total Iterations	Epochs Completed
30% (70%)	90.38%	16 min 34 sec	900	20/20
25% (75%)	90.69%	24 min 54 sec	840	20/20
20% (80%)	85.94%	15 min 50 sec	960	20/20

C. *Feature Extraction Analysis*

The handcrafted feature extraction process was designed to capture key characteristics of the data, yielding 15 comprehensive features categorized into three primary groups. These features are crucial for gaining insights into the statistical, geometric, and textural properties of the analyzed data. The first category, First-order Statistical Features, consists of five features: Mean, Median, Standard Deviation, Skewness, and Kurtosis. These features provide an overall understanding of the data's distribution, central tendency, variability, and shape. The Mean and Median give insights into centrality, while Standard Deviation reflects data spread. Skewness and Kurtosis assess the symmetry and the extremities of the distribution, respectively [48].

The second category, Shape-based Features, includes four features that describe the geometric properties of the shape under analysis. These include Area, Perimeter, Eccentricity, and Solidity. Area and Perimeter give quantitative measurements of size and boundary length, respectively. Eccentricity measures how elongated the shape is, indicating its deviation from circularity, while Solidity quantifies the compactness of the shape by comparing the area to the convex hull area. These shape-based features are particularly useful for recognizing and differentiating different forms and structures.

Finally, the Texture Features group, derived from the Gray Level Co-occurrence Matrix (GLCM), consists of six features: Contrast, Correlation, Energy, Homogeneity, Coefficient of Variation, and Histogram Width. These features capture the surface characteristics of the shape, with Contrast reflecting intensity differences between neighboring pixels, Correlation assessing linear dependencies, and Energy providing information about texture uniformity. Homogeneity indicates how smooth or regular the texture is, while Coefficient of Variation offers a normalized measure of variation, and Histogram Width quantifies the range of intensity values. Together, these features provide a rich set of descriptors, enabling detailed texture analysis.

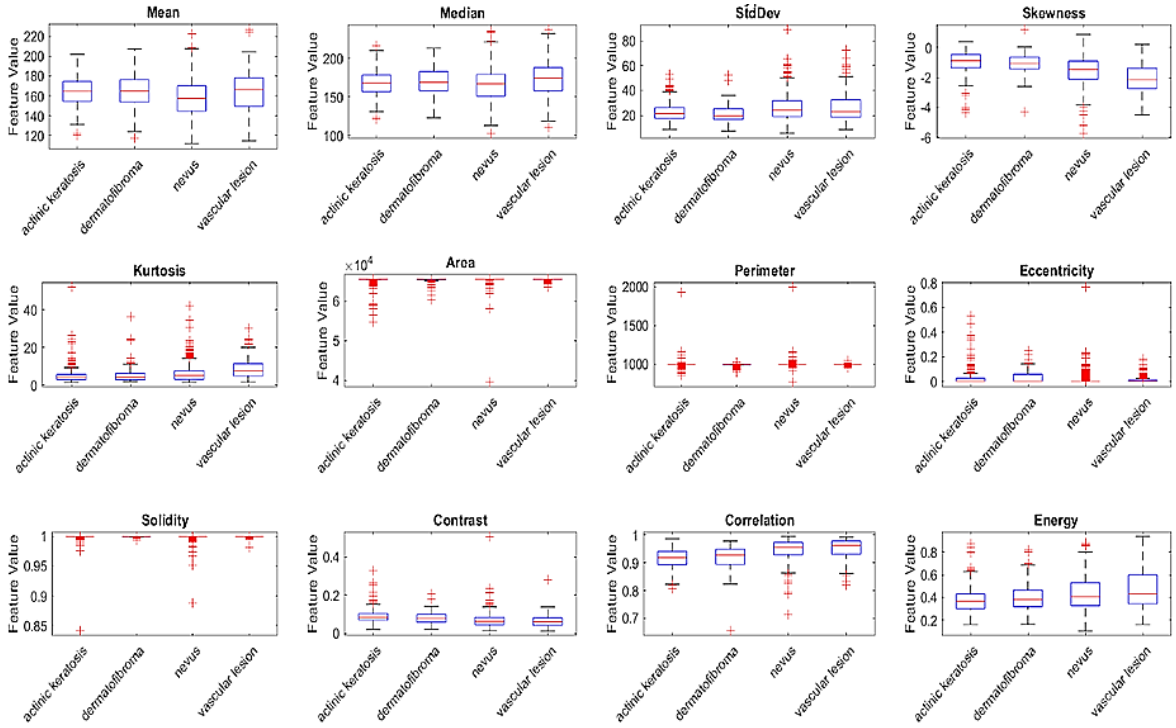


Fig. 6. Feature distribution analysis across four skin cancer types: (a) first-order statistical features (mean, median, standard deviation, skewness, kurtosis), (b) shape-based features (area, perimeter, eccentricity, solidity), and (c) texture features (contrast, correlation, energy, homogeneity, coefficient of variation, histogram width).

Figure 6 presents the feature distribution analysis, which reveals distinct morphological characteristics across the four skin cancer types.. The box plots demonstrate varying degrees of feature discrimination, with several features showing clear separability between classes. Notably, the skewness and kurtosis features exhibit pronounced differences, particularly for vascular lesions which show consistently different distributions. The presence of outliers (red crosses) indicates natural variability within each class, which is expected in medical imaging data. The area and perimeter features show relatively similar distributions across classes, suggesting shape-based features alone may not provide sufficient discrimination. The texture features (contrast, correlation, energy) demonstrate more promising discriminative potential, with vascular lesions and nevus showing distinct patterns [49].

TABLE IV. FEATURE STATISTICS SUMMARY

Feature	Actinic Keratosis	Dermatofibroma	Nevus	Vascular Lesion
Mean Intensity	164.70 ± 15.10	165.52 ± 16.37	158.24 ± 19.50	165.35 ± 19.21
Median	168.44 ± 16.81	169.89 ± 17.26	166.24 ± 21.19	173.30 ± 20.43
Standard Deviation	22.25 ± 7.51	21.17 ± 7.22	26.49 ± 10.83	26.79 ± 12.14
Skewness	-0.96 ± 0.67	-1.08 ± 0.69	-1.59 ± 0.96	-2.11 ± 0.95
Kurtosis	4.88 ± 4.00	5.35 ± 4.53	6.29 ± 4.91	8.95 ± 5.66
Area	65304.34 ± 1108.33	65251.25 ± 753.35	65353.79 ± 1454.92	65459.23 ± 244.81
Perimeter	1000.30 ± 55.19	990.79 ± 18.89	1000.97 ± 56.05	996.81 ± 9.97
Eccentricity	0.026 ± 0.064	0.032 ± 0.053	0.015 ± 0.054	0.018 ± 0.039
Solidity	0.999 ± 0.009	1.000 ± 0.001	0.999 ± 0.007	1.000 ± 0.002
Contrast	0.088 ± 0.034	0.080 ± 0.031	0.067 ± 0.039	0.062 ± 0.033

Correlation	0.916 ± 0.036	0.919 ± 0.044	0.947 ± 0.037	0.949 ± 0.037
Energy	0.379 ± 0.113	0.403 ± 0.128	0.437 ± 0.158	0.473 ± 0.180
Homogeneity	0.957 ± 0.014	0.962 ± 0.014	0.968 ± 0.016	0.970 ± 0.015
Coefficient of Variation	0.136 ± 0.046	0.130 ± 0.049	0.171 ± 0.075	0.165 ± 0.081
Histogram Width	162.84 ± 35.25	150.71 ± 42.85	147.81 ± 31.42	156.03 ± 38.08

The analysis of **Table IV** reveals distinctive feature statistics that play a crucial role in discriminating between four skin cancer types, highlighting several significant patterns. Firstly, the progression of skewness shows a clear trend from actinic keratosis (-0.96) to vascular lesion (-2.11), indicating increasingly left-skewed intensity distributions. This suggests that vascular lesions exhibit more concentrated high-intensity regions, a characteristic that differentiates them from the other lesion types. Secondly, the variation in kurtosis further distinguishes vascular lesions, which exhibit the highest kurtosis value of 8.95 ± 5.66. This indicates that the intensity distribution of vascular lesions is more peaked with heavy tails, reflecting their distinctive vascular patterns.

In terms of texture, the correlation feature displays notable separation between actinic keratosis/dermatofibroma (0.916-0.919) and nevus/vascular lesion (0.947-0.949), suggesting differences in the smoothness of the textures across these lesions. The energy distribution also shows a progressive increase, from actinic keratosis (0.379) to vascular lesion (0.473), implying varying degrees of textural uniformity. Vascular lesions, with the highest energy value, demonstrate the most uniform texture patterns, further contributing to their classification. Finally, the contrast patterns show that actinic keratosis has the highest contrast value (0.088), which may explain its consistently high classification performance across all configurations, as it presents more distinct boundaries compared to other lesion types. These statistical features provide valuable insights into the characteristics of each skin lesion type and explain the variations in classification performance across different types. By understanding these patterns, we can better interpret the significance of each feature in the classification task [50].

D. Classification Performance Analysis

The confusion matrices reveal critical insights into model performance and potential overfitting. **Figure 7** presents the confusion matrices for the 30% training configuration. The training set confusion matrix (left) shows perfect classification with 100% accuracy across all classes, indicating that the model has completely learned the training data patterns. However, the test set performance (right) shows more realistic results with 90.38% overall accuracy. The test confusion matrix reveals class-specific performance variations: actinic keratosis shows excellent performance with minimal misclassifications, while dermatofibroma exhibits more confusion with other classes. The diagonal dominance in the test matrix indicates good overall classification capability, but the off-diagonal elements reveal specific class confusions that require attention for clinical deployment

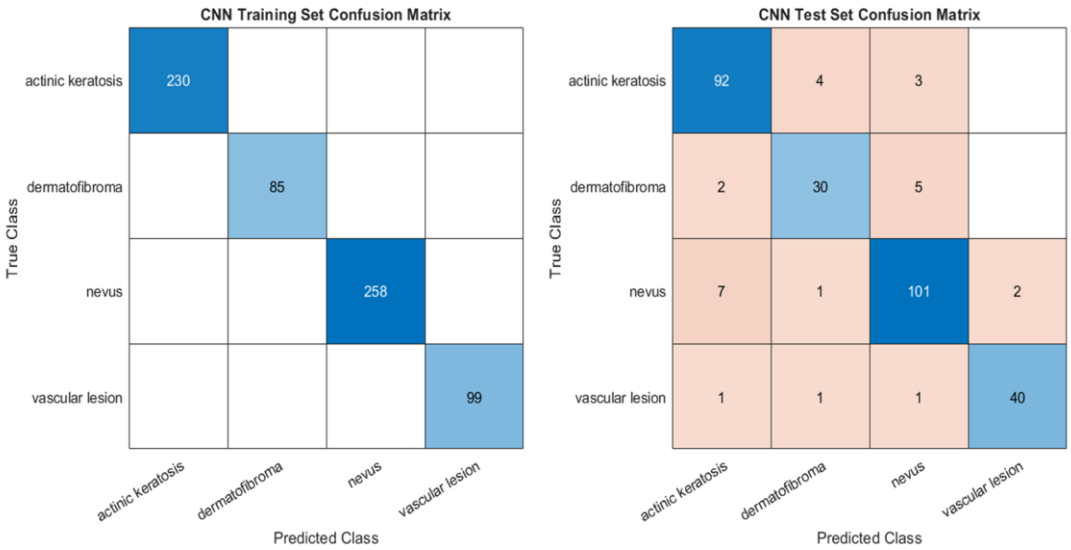


Fig. 7. Confusion matrices for CNN model trained on 30% of data: (a) training set, (b) test set.

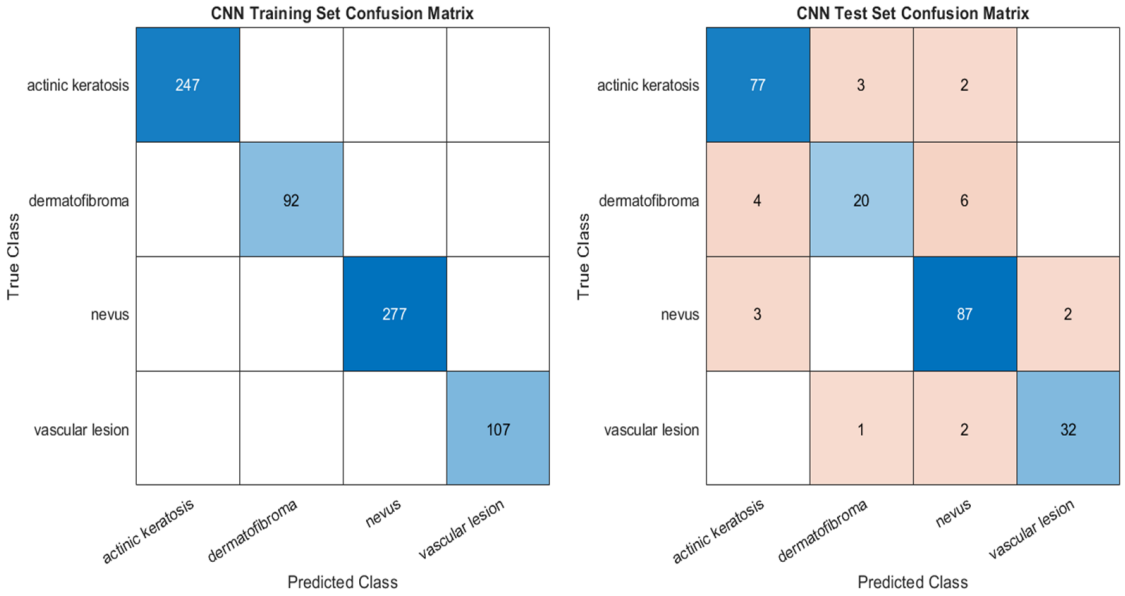


Fig. 8. Confusion matrices for CNN model trained on 25% of data: (a) training set, (b) test set.

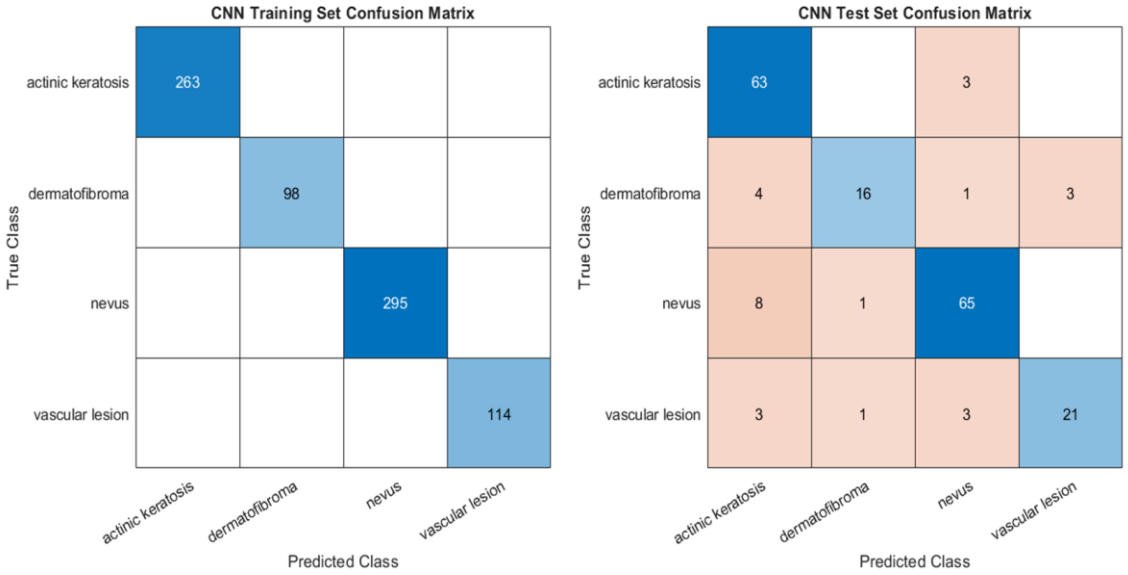


Fig. 9. Confusion matrices for CNN model trained on 20% of data: (a) training set, (b) test set.

As shown in **Figure 8**, the 25% training configuration demonstrates the most balanced performance between training and test sets. The training confusion matrix shows excellent but not perfect performance, suggesting reduced overfitting compared to the 30% configuration. The test set confusion matrix achieves the highest validation accuracy (90.69%) with improved class balance. Notably, the dermatofibroma classification shows improvement compared to the 30% configuration, indicating that the reduced training data prevents overfitting to class-specific noise. The overall diagonal dominance is stronger, with fewer misclassifications across all classes, validating the optimal training data size hypothesis.

As shown in **Figure 9**, the 20% configuration shows the expected performance degradation due to limited training data, achieving 85.94% test accuracy. The training confusion matrix indicates that the model struggles to achieve perfect training performance, suggesting insufficient data for complete pattern learning. The test confusion matrix shows increased misclassifications, particularly for minority classes like dermatofibroma. However, the performance degradation is relatively modest (4.75% decrease from optimal), indicating that the ResNet-50 architecture maintains reasonable performance even with limited training data. This finding is crucial for clinical settings where large training datasets may not be available.

TABLE V. PERFORMANCE METRICS - SCENARIO 1 : 30% TEST DATA (70% TRAINING)

Method	Accuracy (%)	Precision	Recall	Sensitivity	Specificity	F1-Score	AUC
CNN (ResNet-50)	90.69	0.901	0.895	0.895	0.966	0.898	0.979
SVM (RBF Kernel)	60.07	0.576	0.448	0.448	0.844	0.434	0.801

Table V presents the performance metrics for Scenario 1. The first scenario demonstrates the clear superiority of the CNN approach over SVM across all evaluation metrics. The CNN achieves 90.69% accuracy compared to SVM's 60.07%, representing a 30.62 percentage point improvement. The precision difference (0.901 vs 0.576) indicates that the CNN produces far fewer false positives, crucial for clinical applications where false alarms can lead to unnecessary procedures. The recall disparity (0.895 vs 0.448) shows that the CNN identifies significantly more actual positive cases, critical for cancer detection where missing cases has severe consequences. The AUC values (0.979 vs 0.801) demonstrate the CNN's superior ability to discriminate between classes across all decision thresholds.

TABLE VI. PERFORMANCE METRICS - SCENARIO 2: 25% TEST DATA (75% TRAINING)

Method	Accuracy (%)	Precision	Recall	Sensitivity	Specificity	F1-Score	AUC
CNN (ResNet-50)	90.38	0.897	0.866	0.866	0.965	0.879	0.969
SVM (RBF Kernel)	62.92	0.581	0.491	0.491	0.858	0.494	0.805

Table VI presents the performance metrics for Scenario 2. The second scenario shows slight variations in performance metrics while maintaining the CNN's substantial advantage over SVM. The CNN achieves 90.38% accuracy with well-balanced precision (0.897) and recall (0.866), indicating robust performance across different test data compositions. The high specificity (0.965) demonstrates excellent ability to correctly identify negative cases, minimizing false positive rates. The F1-score of 0.879 represents a balanced harmonic mean between precision and recall, crucial for imbalanced datasets. The SVM performance shows marginal improvement in this scenario (62.92% vs 60.07%), but remains substantially below clinically acceptable level.

TABLE VII. PERFORMANCE METRICS - SCENARIO 3: 20% TEST DATA (80% TRAINING)

Method	Accuracy (%)	Precision	Recall	Sensitivity	Specificity	F1-Score	AUC
CNN (ResNet-50)	85.94	0.869	0.812	0.812	0.948	0.834	0.970
SVM (RBF Kernel)	57.81	0.518	0.439	0.439	0.839	0.431	0.772

Table VII presents the performance metrics for Scenario 3. The third scenario demonstrates the impact of limited training data on CNN performance while maintaining superiority over SVM. The CNN accuracy drops to 85.94%, representing a 4.75 percentage point decrease from optimal performance, indicating graceful degradation with reduced training data. The precision (0.869) and recall (0.812) remain at clinically acceptable levels, suggesting that the model maintains diagnostic utility even with minimal training examples. The AUC remains high (0.970), indicating preserved discriminative capability across decision thresholds. The SVM shows the poorest performance in this scenario, emphasizing the importance of sufficient training data for traditional machine learning approaches.

TABLE VIII. CNN CLASS-WISE PERFORMANCE SUMMARY ACROSS ALL TRAINING CONFIGURATIONS

Training Set	Class	Training Accuracy	Test Recall	Test Precision*	Test Samples
30%	Actinic Keratosis	100%	95.5%	0.902	66
30%	Dermatofibroma	100%	66.7%	0.833	24
30%	Nevus	100%	87.8%	0.918	74
30%	Vascular Lesion	100%	75.0%	0.952	28
25%	Actinic Keratosis	100%	93.9%	0.753	82
25%	Dermatofibroma	100%	66.7%	0.538	30

Training Set	Class	Training Accuracy	Test Recall	Test Precision*	Test Samples
25%	Nevus	100%	94.6%	0.578	92
25%	Vascular Lesion	100%	91.4%	0.455	35
20%	Actinic Keratosis	100%	92.9%	0.808	99
20%	Dermatofibroma	100%	81.1%	0.889	37
20%	Nevus	100%	91.0%	0.903	111
20%	Vascular Lesion	100%	93.0%	0.875	43

Table VIII presents the class-wise performance summary. The analysis reveals distinct patterns across the different training configurations, which are closely linked to the dataset characteristics. For Actinic Keratosis, the recall remains consistently high (92.9%-95.5%) across all configurations, indicating strong detection capability. However, the precision shows more variation (0.753-0.902), which suggests the presence of some false positive variability. Dermatofibroma, on the other hand, demonstrates a more challenging classification profile, with consistently lower recall values (66.7%-81.1%) across configurations. This pattern is likely a result of its smaller dataset size (122 images), underlining the impact of class imbalance on performance. Nevus shows stable performance with high recall (87.8%-94.6%) but experiences variable precision (0.578-0.918). The relatively larger dataset size (369 images) provides a sufficient number of training examples, ensuring consistent recall, though precision fluctuates depending on the training configuration. Lastly, Vascular Lesions exhibit a dramatic improvement in performance as the training set size increases, with recall progressing from 75.0% to 93.0%. This suggests that vascular lesions benefit significantly from the inclusion of more training examples. Notably, the consistent 100% training accuracy across all classes and configurations indicates a potential for overfitting, emphasizing the need to rely on test set performance metrics for a more realistic and generalized assessment of the model's effectiveness.

TABLE IX. CNN CLASS-WISE DETAILED PERFORMANCE (30% CONFIGURATION)

Class	Precision	Recall	Sensitivity	Specificity	F1-Score	AUC
Actinic Keratosis	0.902	0.929	0.929	0.948	0.915	0.972
Dermatofibroma	0.833	0.811	0.811	0.976	0.822	0.969
Nevus	0.918	0.910	0.910	0.950	0.914	0.975
Vascular Lesion	0.952	0.930	0.930	0.992	0.941	0.999

Table IX presents the detailed performance metrics for the 30% configuration, highlighting the optimal class-wise performance characteristics.. Vascular Lesions achieve the best overall performance, with an F1-score of 0.941 and a perfect AUC of 0.999, demonstrating near-perfect discrimination capabilities. The high precision (0.952) and specificity (0.992) indicate an excellent ability to avoid false positives, further confirming the robustness of the model for this class. Actinic Keratosis and Nevus show comparable performance, with F1-scores of 0.915 and 0.914, respectively. Both classes demonstrate a balance between precision and recall, indicating a robust classification capability that is well-suited for clinical applications. Dermatofibroma, despite having the lowest F1-score (0.822), exhibits excellent specificity (0.976) and precision (0.833), which suggests a low false positive rate. However, the lower recall (0.811) indicates that some cases are missed, which can be directly attributed to the limited number of training examples. Overall, all classes achieve AUC values above 0.969, reflecting excellent discriminative capability across all decision thresholds. These results validate the effectiveness of the ResNet-50 architecture in skin cancer classification.

TABLE X. SVM CLASS-WISE PERFORMANCE (25% CONFIGURATION)

Class	Precision	Recall	Sensitivity	Specificity	F1-Score	AUC
Actinic Keratosis	0.753	0.744	0.744	0.873	0.748	0.893
Dermatofibroma	0.538	0.233	0.233	0.971	0.326	0.753
Nevus	0.578	0.848	0.848	0.615	0.687	0.825
Vascular Lesion	0.455	0.139	0.139	0.971	0.213	0.751

Table X presents the SVM class-wise performance under the 25% configuration. The analysis reveals several limitations of traditional machine learning approaches in handling complex medical image classification tasks. For Actinic Keratosis, the SVM achieves the best performance with an F1-score of 0.748, though this is still substantially lower than the performance of CNN (0.915). The balanced precision and recall suggest that handcrafted features can provide some level of discriminative capability for this class, but they fall short in comparison to deep learning methods. Nevus demonstrates high recall (0.848) but

poor precision (0.578), indicating that the SVM tends to over-classify images as nevus. This behavior reflects the majority class bias commonly seen in imbalanced datasets, where the model over-represents the more frequent class. For Dermatofibroma and Vascular Lesion, the performance is extremely poor, with F1-scores of 0.326 and 0.213, respectively. The very low recall values (0.233 and 0.139) further highlight that the SVM fails to identify most positive cases for these classes, struggling particularly with the underrepresented examples. The particularly poor performance on dermatofibroma (F1: 0.326) and vascular lesion (F1: 0.213) can be attributed to several compounding factors. First, both classes are severely underrepresented in the dataset (122 and 142 images respectively), meaning that the handcrafted feature distributions for these classes overlap considerably with those of the majority classes. Second, the 15 extracted features — while comprehensive — are derived from global image statistics (e.g., mean intensity, GLCM contrast) that fail to capture the subtle local morphological patterns that distinguish dermatofibroma and vascular lesions from nevus and actinic keratosis. Dermatofibroma, in particular, is known dermoscopically for its variable and non-specific appearance [4], making it difficult for any fixed feature set to reliably characterize it. Vascular lesions, while visually distinctive to trained clinicians, exhibit intensity and texture distributions that may overlap with other lesion types when reduced to scalar statistics. In contrast, the CNN implicitly learns spatially localized, hierarchical features through its convolutional layers, enabling it to detect subtle distinguishing patterns that handcrafted features cannot capture. This fundamental architectural difference explains the large performance gap between CNN and SVM specifically for these two classes.

E. Model Interpretability Through Grad-CAM Analysis

The Gradient-weighted Class Activation Mapping (Grad-CAM) visualizations (**Figure 10**) provide valuable insights into the CNN model's decision-making process, highlighting the regions of dermatoscopic images that the model focuses on when making classification decisions. Red regions, indicating the highest activation, show that the model concentrates its attention on the lesion areas for actinic keratosis and vascular lesions, which are correctly predicted with high confidence (scores ranging from 0.92 to 1.00). However, dermatofibroma exhibits scattered activation patterns, reflecting the challenges the model faces in accurately classifying this class. For actinic keratosis, strong central activation is observed (score: 1.00), signaling that the model reliably focuses on the most relevant features. In contrast, nevus misclassification (predicted as vascular lesion, score: 0.62) is associated with peripheral activation, indicating overlap of features between the two classes. Clear activation patterns are evident for both actinic keratosis (score: 1.00) and vascular lesion (score: 0.98), demonstrating robust feature localization. On the other hand, dermatofibroma misclassification (predicted as nevus, score: 0.69) shows diffuse activation, suggesting that the model struggles to localize distinct features for this class.

From a clinical perspective, Grad-CAM visualizations offer a practical pathway for integrating the model's decisions into real-world diagnostic workflows. In a clinical setting, a dermatologist could use the Grad-CAM heatmap as a secondary confirmation layer: after the model produces a classification, the physician can examine whether the highlighted activation region corresponds to the visually suspicious area of the lesion. If the model's focus aligns with the clinically relevant region, it increases confidence in the prediction. Conversely, if the activation is scattered or concentrated outside the lesion boundary — as observed in several dermatofibroma cases — the clinician is alerted to treat the model's output with caution and apply independent judgment. This human-in-the-loop approach preserves clinical authority while leveraging the model's pattern recognition capabilities. Future deployment of this system in web or mobile applications could render Grad-CAM overlays alongside the classification result, enabling transparent and accountable AI-assisted diagnosis even in low-resource healthcare environments.

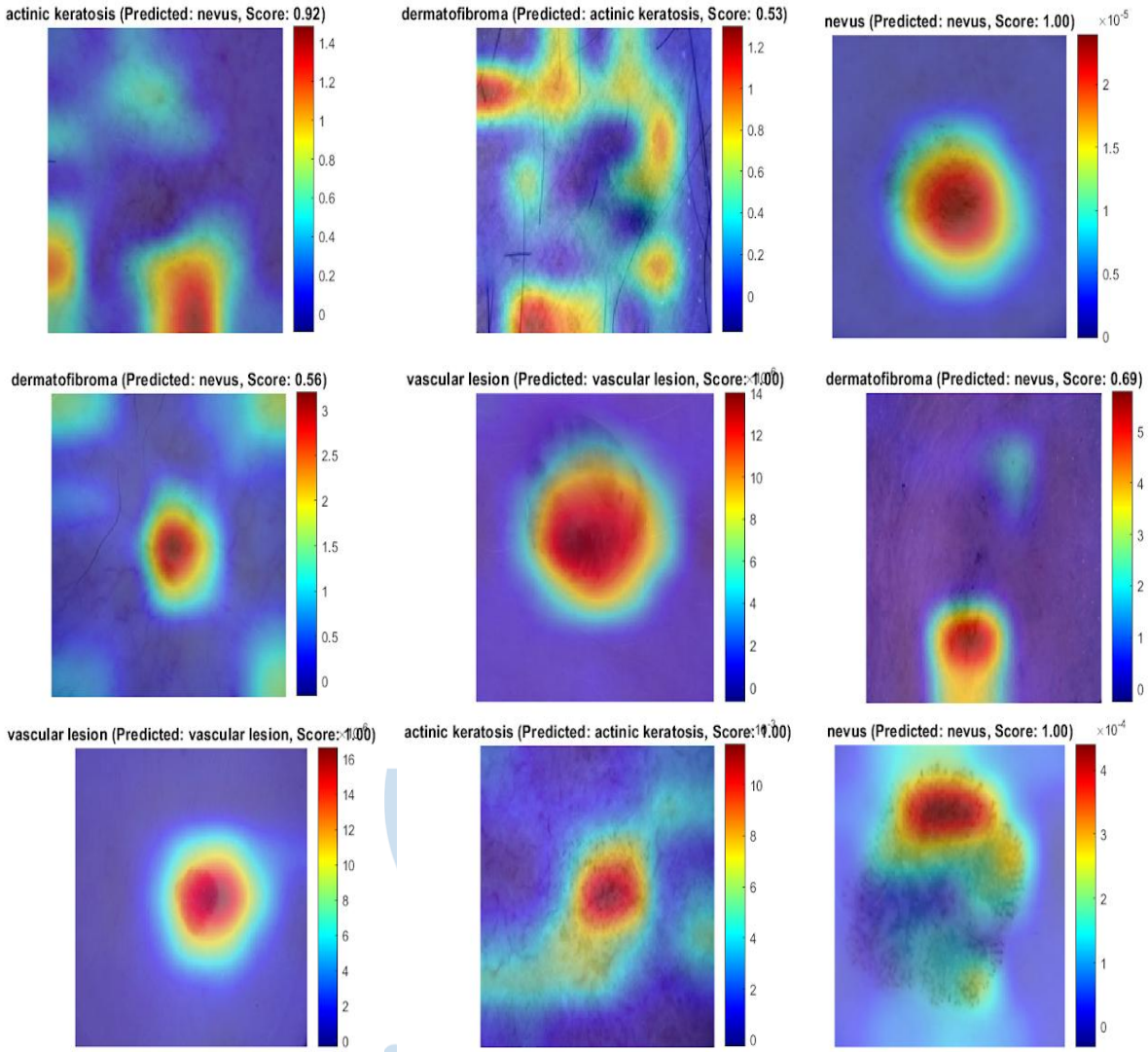


Fig. 10. Grad-CAM visualization for different skin cancer types

The key finding of this study (defined here as the primary result that departs from conventional expectations and carries the greatest practical implication) is the unexpected result that the 25% training set configuration achieved the highest validation accuracy of 90.69%, surpassing both the 30% (90.38%) and 20% (85.94%) configurations. This counterintuitive outcome holds important implications for skin cancer classification using CNN architectures. It suggests that there is an optimal balance between generalization and overfitting. The 25% configuration provided enough data for effective learning without introducing excessive noise or overfitting, which may have occurred with the larger 30% dataset. Despite having more training data, the 30% configuration showed slight performance degradation, indicating that additional data may have led to overfitting in some cases. This finding emphasizes the need for careful consideration of training data size, where too much data may not necessarily lead to better performance if it leads to overfitting.

Training efficiency was also a key aspect of the analysis, revealing notable patterns. The 30% configuration provided the highest accuracy/time ratio, making it the most efficient in terms of computational resources, while the 25% configuration, although achieving the highest absolute accuracy, required longer training times. The 20% configuration demonstrated good efficiency but at the cost of compromised accuracy, reinforcing the importance of finding a balance between accuracy and computational efficiency, especially in resource-constrained environments where time and hardware resources are limited.

Class-specific performance showed consistent patterns across all training configurations. Actinic Keratosis demonstrated consistently high recall rates (>92%) across all configurations, reflecting its reliable detection capabilities. Nevus showed robust performance with recall rates ranging from 87.8% to 94.6%, indicating stable classification. Vascular Lesion saw dramatic improvement in recall, progressing from 75% to 93% as the training set size increased,

suggesting that larger datasets significantly improved its classification accuracy. In contrast, Dermatofibroma consistently exhibited the lowest performance (66.7% - 81.1%) across all configurations, presenting the primary challenge in classification. This consistent underperformance is likely due to dataset imbalance, as dermatofibroma has the smallest number of images (122 compared to 369 for nevus). This issue suggests that targeted data augmentation or specialized preprocessing techniques are necessary to address the imbalance and improve classification accuracy [51].

Feature analysis provided additional insights into the discriminative characteristics of different skin cancer types [52]. Statistical features, such as skewness and kurtosis, revealed clear distinctions between skin cancer types [53]. Vascular Lesions exhibited the highest kurtosis, indicating distinct intensity distribution patterns, while Actinic Keratosis showed the highest contrast, which may explain its consistently strong classification performance [54]. Additionally, texture-based features such as correlation coefficients and contrast patterns were instrumental for classification [55]. Vascular Lesions and Nevus exhibited the highest correlation, suggesting smoother texture patterns, further aiding in their differentiation.

Clinically, the achieved performance metrics offer substantial potential for diagnostic support, with accuracy levels exceeding 90% in the optimal configurations. This makes the model suitable for use in clinical settings, especially as a screening tool in resource-limited environments. The finding that 25% of the available training data yields optimal results has significant implications for clinical deployment, reducing annotation costs, accelerating deployment, and improving accessibility for institutions with limited resources. Furthermore, this multi-class classification approach provides a more comprehensive diagnostic tool compared to many studies focused solely on binary melanoma detection, aligning more closely with clinical practice where multiple lesion types must be diagnosed.

Finally, the technical contributions of this study are noteworthy. The use of transfer learning with ResNet-50 for skin cancer classification proved highly effective. The modification of the final layers of the pretrained model to accommodate four classes while maintaining the feature extraction capabilities of ResNet-50 allowed for effective learning despite the smaller dataset. The consistently high training accuracy across all configurations further validated the effectiveness of knowledge transfer in deep learning. Moreover, the feature engineering approach, incorporating 15 handcrafted features, combined first-order statistics, shape descriptors, and texture analysis to provide comprehensive lesion characterization. This enhanced the model's ability to discriminate between different skin cancer types, offering valuable insights into the classifier's decision-making process and making it a useful tool for clinical applications.

IV. CONCLUSION

This study highlights the potential of automated skin cancer classification using CNN architectures, achieving clinically relevant performance with optimal resource utilization. The central finding of this study (that using only 25% of the data for testing (75% for training) yields the highest validation accuracy of 90.69%) challenges traditional assumptions and offers a practical breakthrough for resource-constrained environments like rural or low-resource healthcare settings. By using a smaller dataset, this method reduces training time and computational costs while maintaining high diagnostic accuracy. The multi-class classification system effectively distinguishes between four skin cancer types, demonstrating the robustness of CNNs in complex medical problems. The clinical applicability of this approach is a key strength, with the model's performance making it suitable for real-world diagnostic support, especially in underserved areas where specialized expertise is scarce. This research enables wider access to advanced skin cancer detection, potentially saving lives in remote regions. From a clinical impact perspective, the study emphasizes cost-effective solutions that can be rapidly deployed. The 25% training configuration is recommended for skin cancer classification systems due to its proven ability to deliver high accuracy with minimal resources. Future improvements, such as targeted dermatofibroma detection through advanced data augmentation, will further enhance the system's diagnostic capabilities, making it adaptable to various skin conditions. Additionally, the Grad-CAM analysis demonstrated that the model's decision-making is grounded in clinically relevant regions of the dermoscopic image. For actinic keratosis and vascular lesions, the model consistently activated over the lesion core, reflecting high-confidence and well-localized predictions. In contrast, the diffuse and scattered activation patterns observed for dermatofibroma indicate that the model has not yet learned sufficiently discriminative features for this class, likely due to the limited number of training samples. These interpretability insights are valuable not only for validating model behavior but also for guiding future data collection efforts — specifically targeting dermatofibroma cases to improve activation focus. Incorporating Grad-CAM into the final diagnostic system would enable clinicians to visually verify the model's reasoning before accepting its output, thereby enhancing trust and safety in clinical deployment. Ultimately, this research supports the adoption of AI-powered diagnostic tools, balancing precision, efficiency, and ease of implementation, with the potential to revolutionize dermatology.

ACKNOWLEDGMENT

The authors would like to acknowledge the Department of Medical Technology, Institut Teknologi Sepuluh Nopember, for the facilities and support in this research.

REFERENCES

[1] A. H. Roky, M. M. Islam, A. M. F. Ahasan, M. S. Mostaq, M. Z. Mahmud, M. N. Amin, and M. A. Mahmud, "Overview of skin cancer types and prevalence rates across continents," *Cancer Pathogenesis and Therapy*, vol. 3, no. 2, Aug. 2024, doi: https://doi.org/10.1016/j.cpt.2024.08.002.

[2] N. F. Kormos, I. D. Paval, C. M. Petrenciu, A. S. Vizitiu, and A. L. Baican, "Diagnostic Challenges and Influencing Factors in Non-Melanoma Skin Cancers: A Retrospective Analysis of Basal Cell Carcinoma and Squamous Cell Carcinoma Cases," *Cancer Reports*, vol. 8, no. 9, Sep. 2025, doi: https://doi.org/10.1002/cnr2.70332.

[3] J. R. Thamm, S. Schuh, and J. Welzel, "Epidemiology and Risk Factors of Actinic Keratosis. What Is New for the Management for Sun-Damaged Skin," *Dermatology Practical & Conceptual*, vol. 14, no. S1, p. e2024146S, Aug. 2024, doi: https://doi.org/10.5826/dpc.1403s1a146s.

[4] L. Wan, A. Park, L. Almatroud, and A. Khachemoune, "Dermatofibroma: Reappraisal and Updated Review," *Clinical Cosmetic and Investigational Dermatology*, vol. Volume 18, pp. 1873–1887, Aug. 2025, doi: https://doi.org/10.2147/ccid.s526191.

[5] P. S.-Sienkiewicz, D. J.-Lewandowska, J. Calik, G. T.-Kołodziejczyk, and P. Mańkowski, "Nevi and Melanoma in Children: What to Do in Daily Medical Practice: Encyclopedia for Pediatricians and Family Doctors," *Diagnostics*, vol. 14, no. 18, pp. 2004–2004, Sep. 2024, doi: https://doi.org/10.3390/diagnostics14182004.

[6] G. S. Colafati, E. Piccirilli, A. Marrazzo, A. Carboni, A. Diociaiuti, M. E. Hachem, F. Esposito, M. Zama, M. Rollo, C. Gandolfo, and P. Tomà, "Vascular lesions of the pediatric orbit: A radiological walkthrough," *Frontiers in Pediatrics*, vol. 10, Nov. 2022, doi: https://doi.org/10.3389/fped.2022.734286.

[7] F. Cassalia, A. Danese, E. Cocchi, S. Vaienti, A. Bolzon, L. Franceschin, R. Mazzetto, F. Caroppo, D. Melandri, and A. B. Fortina, "Congenital Dermatofibrosarcoma Protuberans—An Update on the Ongoing Diagnostic Challenges," *Cancers*, vol. 17, no. 1, p. 158, Jan. 2025, doi: https://doi.org/10.3390/cancers17010158.

[8] K. Nawaz, A. Zanib, I. Shabir, J. Li, Y. Wang, T. Mahmood, and A. Rehman, "Skin cancer detection using dermoscopic images with convolutional neural network," *Scientific Reports*, vol. 15, no. 1, Mar. 2025, doi: https://doi.org/10.1038/s41598-025-91446-6.

[9] M. Li, Y. Jiang, Y. Zhang, and H. Zhu, "Medical image analysis using deep learning algorithms," *Frontiers in Public Health*, vol. 11, no. 1273253, Nov. 2023, doi: https://doi.org/10.3389/fpubh.2023.1273253.

[10] H. K. Jeong, C. Park, S. W. Jiang, M. Nicholas, S. Chen, R. Henao, and M. Kheterpal, "Image Quality Assessment Using Convolutional Neural Network in Clinical Skin Images," *JID Innovations*, vol. 4, no. 4, pp. 100285–100285, Apr. 2024, doi: https://doi.org/10.1016/j.xjidi.2024.100285.

[11] A. R. Gunukula, H. D. Gupta, and V. S. Sheng, "Detecting AI-Generated Images Using a Hybrid ResNet-SE Attention Model," *Applied Sciences*, vol. 15, no. 13, pp. 7421–7421, Jul. 2025, doi: https://doi.org/10.3390/app15137421.

[12] J. Alcaraz, M. Labbé, and M. Landete, "Support Vector Machine with feature selection: A multiobjective approach," *Expert Systems with Applications*, vol. 204, p. 117485, Oct. 2022, doi: https://doi.org/10.1016/j.eswa.2022.117485.

[13] B. Cassidy, C. Kendrick, A. Brodzicki, J. J.-Korjakowska, and M. H. Yap, "Analysis of the ISIC image datasets: Usage, benchmarks and recommendations," *Medical Image Analysis*, vol. 75, p. 102305, Jan. 2022, doi: https://doi.org/10.1016/j.media.2021.102305.

[14] A. K. Nambisan, A. Maurya, N. Lama, T. Phan, G. Patel, K. Miller, B. Lama, J. Hagerty, R. Stanley, and W. V. Stoecker, "Improving Automatic Melanoma Diagnosis Using Deep Learning-Based Segmentation of Irregular Networks," *Cancers*, vol. 15, no. 4, pp. 1259–1259, Feb. 2023, doi: https://doi.org/10.3390/cancers15041259.

[15] M. Rahman, S. M. N. Nobel, P. K. Meena, V. P. Meena, J. Bahadur, and A. Appaji, "Accelerated and Precise Skin Cancer Detection through an Enhanced Machine Learning Pipeline for Improved Diagnostic Accuracy," *Results in Engineering*, pp. 104168–104168, Jan. 2025, doi: https://doi.org/10.1016/j.rineng.2025.104168.

[16] O. Akinrinade and C. Du, "Skin Cancer Detection Using Deep Machine Learning Techniques," *Intelligence-Based Medicine*, pp. 100191–100191, Dec. 2024, doi: https://doi.org/10.1016/j.ibmed.2024.100191.

[17] H. Erbay, Y. M. Abulgasim, D. E. Özer, and F. Ertürk, "Enhancing multi-class skin lesion diagnosis through ensemble learning of CNN and transformer architectures," *Engineering Science and Technology, an International Journal*, vol. 70, p. 102145, Jul. 2025, doi: https://doi.org/10.1016/j.jestch.2025.102145.

[18] S. S. Zareen, G. Sun, M. Kundi, S. F. Qadri, and S. Qadri, "Enhancing Skin Cancer Diagnosis with Deep Learning: A Hybrid CNN-RNN Approach," *Computers, materials & continua/Computers, materials & continua (Print)*, vol. 79, no. 1, pp. 1497–1519, Jan. 2024, doi: https://doi.org/10.32604/cmc.2024.047418.

[19] S. R. Nalamwar and S. U. Belgamwar, "Multidimensional CapsNets attention-gated approach for skin cancer detection and classification," *Computers & Electrical Engineering*, vol. 127, pp. 110573–110573, Aug. 2025, doi: https://doi.org/10.1016/j.compeleceng.2025.110573.

[20] M. Manickavasagam, V. V. Rani, U. K. Giri, and B. Maram, "Skin Cancer Detection using Harmonic Brown Bear Optimization enabled Transfer Learning," *Computational Biology and Chemistry*, vol. 119, pp. 108551–108551, Jun. 2025, doi: https://doi.org/10.1016/j.compbiolchem.2025.108551.

[21] I. Pacal, B. Ozdemir, J. Zeynalov, H. Gasimov, and N. Pacal, "A novel CNN-ViT-based deep learning model for early skin cancer diagnosis," *Biomedical Signal Processing and Control*, vol. 104, p. 107627, Jan. 2025, doi: https://doi.org/10.1016/j.bspc.2025.107627.

[22] J.S. ThangaPurni and M. Braveen, "Unified ARP-ViT-CNN system: Hybrid deep learning approach for segmenting and classifying multiple skin cancer lesions," *Array*, pp. 100515–100515, Sep. 2025, doi: https://doi.org/10.1016/j.array.2025.100515.

[23] S. Khattar, R. Kaur, and A. Kumar, "DeepHybrid-CNN: A Hybrid Approach for Pre-processing of Skin Cancer Images," *Computerized Medical Imaging and Graphics*, pp. 102611–102611, Aug. 2025, doi: https://doi.org/10.1016/j.compmedimag.2025.102611.

[24] H. C. Reis and V. Turk, "Fusion of transformer attention and CNN features for skin cancer detection," *Applied soft computing*, pp. 112013–112013, Jul. 2024, doi: https://doi.org/10.1016/j.asoc.2024.112013.

[25] P. Nanda, D. Rout, and S. Kumari, "Multi-Class Skin Cancer Detection Using CNN-Architecture Based Deep Learning Models," *Procedia Computer Science*, vol. 260, pp. 226–235, Jan. 2025, doi: https://doi.org/10.1016/j.procs.2025.03.197.

[26] C. H.-Pérez, M. Combalia, S. Podlipnik, N. C. F. Codella, V. Rotemberg, A. C. Halpern, O. Reiter, C. Carrera, A. Barreiro, B. Helba, S. Puig, V. Vilaplana, and J. Malvehy, "BCN20000: Dermoscopic Lesions in the Wild," *Scientific Data*, vol. 11, no. 1, Jun. 2024, doi: https://doi.org/10.1038/s41597-024-03387-w.

[27] S. Phiphitphatphaisit and O. Surinta, "Deep feature extraction technique based on Conv1D and LSTM network for food image recognition", *Eng Appl Sci Res*, vol. 48, no. 5, pp. 581–592, Jul. 2021, doi: https://doi.org/10.14456/easr.2021.60.

[28] K. Dunphy, M. N. Fekri, K. Grolinger, and A. Sadhu, "Data Augmentation for Deep-Learning-Based Multiclass Structural Damage Detection Using Limited Information," *Sensors*, vol. 22, no. 16, p. 6193, Aug. 2022, doi: https://doi.org/10.3390/s22166193.

[29] T. Islam, M. S. Hafiz, J. R. Jim, M. M. Kabir, and M. F. Mridha, "A systematic review of deep learning data augmentation in medical imaging: Recent advances and future research directions," *Healthcare Analytics*, vol. 5, pp. 100340–100340, Jun. 2024, doi: https://doi.org/10.1016/j.health.2024.100340.

[30] P. K. Mall, P. K. Singh, S. Srivastav, V. Narayan, M. Paprzycki, T. Jaworska, and M. Ganzha, "A comprehensive review of deep neural networks for medical image processing: Recent developments and future opportunities," *Healthcare Analytics*, vol. 4, p. 100216, Dec. 2023, doi: https://doi.org/10.1016/j.health.2023.100216.

[31] V. Singh, A. Baral, R. Kumar, S. Tummala, M. Noori, S. V. Yadav, S. Kang, and W. Zhao, "A Hybrid Deep Learning Model for Enhanced Structural Damage Detection: Integrating ResNet50, GoogLeNet, and Attention Mechanisms," *Sensors*, vol. 24, no. 22, pp. 7249–7249, Nov. 2024, doi: https://doi.org/10.3390/s24227249.

[32] R. Guido, S. Ferrisi, D. Lofaro, and D. Conforti, "An Overview on the Advancements of Support Vector Machine Models in Healthcare Applications: A Review," *Information*, vol. 15, no. 4, p. 235, Apr. 2024, doi: https://doi.org/10.3390/info15040235.

[33] S. A. Salleh, N. Khalid, N. Danny, N. A. M. Zaki, M. Ustuner, Z. A. Latif, and V. Foronda, "Support Vector Machine (SVM) and Object Based Classification in Earth Linear Features Extraction: A Comparison," *Revue Internationale de Geomatique*, vol. 33, no. 0, pp. 183–199, Jun. 2024, doi: https://doi.org/10.32604/rig.2024.050723.

[34] J. S. A.-Ruiz and M. Michalak, "Classification performance assessment for imbalanced multiclass data," *Scientific Reports*, vol. 14, no. 1, May 2024, doi: https://doi.org/10.1038/s41598-024-61365-z.

[35] A. Venčkauskas, J. Toldinas, and N. Morkevičius, "Improving Multi-Class Classification for Recognition of the Prioritized Classes Using the Analytic Hierarchy Process," *Applied Sciences*, vol. 15, no. 13, pp. 7071–7071, Jun. 2025, doi: https://doi.org/10.3390/app15137071.

[36] Pierian Training, "Confusion Matrix with Scikit-Learn and Python - Pierian Training," *Pierian Training*, May 12, 2023. https://pieriantraining.com/confusion-matrix-with-scikit-learn-and-python.

[37] S. Farhadpour, T. A. Warner, and A. E. Maxwell, "Selecting and Interpreting Multiclass Loss and Accuracy Assessment Metrics for Classifications with Class Imbalance: Guidance and Best Practices," *Remote Sensing*, vol. 16, no. 3, pp. 533–533, Jan. 2024, doi: https://doi.org/10.3390/rs16030533.

[38] E. Richardson, R. Trevizani, J. A. Greenbaum, H. Carter, M. Nielsen, and B. Peters, "The receiver operating characteristic curve accurately assesses imbalanced datasets," *Patterns*, vol. 5, no. 6, pp. 100994–100994, May 2024, doi: https://doi.org/10.1016/j.patter.2024.100994.

[39] H. Talebi, A. K. Bardsiri, and V. K. Bardsiri, "Developing a Hybrid Machine Learning Model for Employee Turnover Prediction: Integrating LightGBM and Genetic Algorithms," *Journal of Open Innovation Technology Market and Complexity*, pp. 100557–100557, May 2025, doi: https://doi.org/10.1016/j.joitmc.2025.100557.

[40] S. A. Hicks *et al.*, "On evaluation metrics for medical applications of artificial intelligence," *Scientific Reports*, vol. 12, no. 1, p. 5979, Apr. 2022, doi: https://doi.org/10.1038/s41598-022-09954-8.

[41] M. Cristina, J. Braet, and J. Springael, "Performance Metrics for Multilabel Emotion Classification: Comparing Micro, Macro, and Weighted F1-Scores," *Applied Sciences*, vol. 14, no. 21, pp. 9863–9863, Oct. 2024, doi: https://doi.org/10.3390/app14219863.

[42] S. Bhat, A. Mansoor, B. Georgescu, A. B. Panambur, F. C. Ghesu, S. Islam, K. Packhäuser, D. R.-Salas, S. Grbic, and A. Maier, "AUCReshaping: improved sensitivity at high-specificity," *Scientific Reports*, vol. 13, no. 1, p. 21097, Nov. 2023, doi: https://doi.org/10.1038/s41598-023-48482-x.

[43] A. Mjahad, A. P.-Aguado, L. L.-Serrano, and A. R.-Muñoz, "Optimizing Image Feature Extraction with Convolutional Neural Networks for Chicken Meat Detection Applications," *Applied Sciences*, vol. 15, no. 2, p. 733, Jan. 2025, doi: https://doi.org/10.3390/app15020733.

[44] L. D'hooge, M. Verkerken, T. Wauters, F. De Turck, and B. Volckaert, "Investigating Generalized Performance of Data-Constrained Supervised Machine Learning Models on Novel, Related Samples in Intrusion Detection," *Sensors*, vol. 23, no. 4, p. 1846, Feb. 2023, doi: https://doi.org/10.3390/s23041846.

[45] M. Imani, A. Beikmohammadi, and H. R. Arabnia, "Comprehensive Analysis of Random Forest and XGBoost Performance with SMOTE, ADASYN, and GNUS Under Varying Imbalance Levels," *Technologies*, vol. 13, no. 3, p. 88, Feb. 2025, doi: https://doi.org/10.3390/technologies13030088.

[46] N. Sebastian and B. Ankayarkanni, "Enhanced ResNet-50 with Multi-Feature Fusion for Robust Detection of Pneumonia in Chest X-Ray Images," *Diagnostics*, vol. 15, no. 16, p. 2041, Aug. 2025, doi: https://doi.org/10.3390/diagnostics15162041.

[47] K. Zhu, J. Wang, X. Hu, X. Xie, and G. Yang, "Improving Generalization of Adversarial Training via Robust Critical Fine-Tuning," *arXiv (Cornell University)*, Jan. 2023, doi: https://doi.org/10.48550/arxiv.2308.02533.

[48] T. Guo, "Extreme Precipitation Strongly Impacts the Interaction of Skewness and Kurtosis of Annual Precipitation Distribution on the Qinghai–Tibetan Plateau," *Atmosphere*, vol. 13, no. 11, p. 1857, Nov. 2022, doi: https://doi.org/10.3390/atmos13111857.

[49] A. Ray, S. Sarkar, F. Schwenker, and R. Sarkar, "Decoding skin cancer classification: perspectives, insights, and advances through researchers' lens," *Scientific Reports*, vol. 14, no. 1, Dec. 2024, doi: https://doi.org/10.1038/s41598-024-81961-3.

[50] L. Wen, Z. Xiao, X. Xu, and B. Liu, "Disaster Recognition and Classification Based on Improved ResNet-50 Neural Network," *Applied Sciences*, vol. 15, no. 9, p. 5143, May 2025, doi: https://doi.org/10.3390/app15095143.

[51] S. Lee and S. Lee, "Efficient Data Augmentation Methods for Crop Disease Recognition in Sustainable Environmental Systems," *Big Data and Cognitive Computing*, vol. 9, no. 1, p. 8, Jan. 2025, doi: https://doi.org/10.3390/bdcc9010008.

[52] T. Khater, S. Ansari, S. A. Mahmoud, A. J. Hussain, and H. Tawfik, "Skin Cancer Classification Using Explainable Artificial Intelligence on Pre-extracted Image Features," *Intelligent systems with applications*, vol. 20, pp. 200275–200275, Nov. 2023, doi: https://doi.org/10.1016/j.iswa.2023.200275.

[53] A. Naeem, T. Anees, M. Khalil, K. Zahra, R. A. Naqvi, and S.-W. Lee, "SNC_Net: Skin Cancer Detection by Integrating Handcrafted and Deep Learning-Based Features Using Dermoscopy Images," *Mathematics*, vol. 12, no. 7, pp. 1030–1030, Mar. 2024, doi: https://doi.org/10.3390/math12071030.

[54] G. Fredman, S. R. Wiegell, M. Haedersdal, and G. R. Untracht, "Vascular feature identification in actinic keratosis grades I-III using dynamic optical coherence tomography with automated, quantitative analysis," *Archives of Dermatological Research*, vol. 316, no. 7, Jun. 2024, doi: https://doi.org/10.1007/s00403-024-03022-z.

[55] C. Soare, E. C. Cozma, A. M. Celarel, A. M. Rosca, M. Lupu, and V. M. Voiculescu, "Digitally Enhanced Methods for the Diagnosis and Monitoring of Treatment Responses in Actinic Keratoses: A New Avenue in Personalized Skin Care," *Cancers*, vol. 16, no. 3, pp. 484–484, Jan. 2024, doi: https://doi.org/10.3390/cancers16030484

ISSN 2355-3286

Feature Selection Benchmarks for Breast Cancer Diagnosis: A Comparative Machine Learning Study

R. Rossa Alfi Nur[1], Nashir Abbas Husaini[2], Moch. Arjunnaja[3], Az-Zahra Batrisyia Juniarto[4], Yuri Pamungkas[5*]
[1,2,3,4,5] Department of Medical Technology, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[1]5049231083@student.its.ac.id ; [2]5049231080@student.its.ac.id ; [3]5049231034@student.its.ac.id ; [4]5049231079@student.its.ac.id ; [5*]yuri@its.ac.id
* Corresponding Author

Accepted on February 13, 2026
Approved on June 03, 2026

Abstract—Breast cancer remains one of the most common causes of death among women, making early and precise detection essential. Yet conventional diagnosis can be limited by specialist shortages, cost, and slow workflows. We therefore assess machine-learning classification with feature selection to streamline diagnosis. Our contribution is a comparative benchmark of feature-selection strategies and classifiers on the WDBC dataset. We evaluated five models (SVM, neural-networks, decision tree, bagged-tree, and boosted-tree). Chi-square (χ^2), mRMR, and ReliefF selected 5, 10, 15, and 30 features, and performance was measured across multiple train–test splits using accuracy, precision, recall, specificity, and F1-score. SVM was overall the top performer and stable across splits. The best SVM setting reached 97.81% accuracy, with strong precision and F1-score, indicating reliable benign–malignant separation. Neural-networks usually ranked second but were more sensitive to the split. Bagged Tree generally improved on a single decision tree, while Boosted Tree showed mixed gains depending on the subset. ReliefF and mRMR often matched or exceeded χ^2 with smaller subsets, showing that careful feature reduction can retain accuracy while lowering dimensionality. In conclusion, combining effective feature selection with an appropriate classifier improves breast cancer classification, and SVM with a compact feature set is a practical choice.

Index Terms—Breast cancer classification; Machine learning; Feature selection; WDBC dataset; Support Vector Machine

I. INTRODUCTION

Cancer has become one of the most critical global health challenges today. Different forms of cancer, including lung, cervical, thyroid, and breast cancer, have generated widespread concern worldwide. Among women in particular, breast cancer stands out as one of the most common and life-threatening malignancies. [1]. Breast cancer is the most frequently diagnosed cancer among women, with concerning rates of occurrence and death. It is estimated that around 10% of women will develop breast cancer at some stage in their lifetime [2]. In the ranking of cancer-related mortality among women, breast cancer occupies the second position after lung cancer. This indicates that, despite not being the primary cause of cancer deaths, breast cancer remains a substantial contributor to female cancer mortality and represents a significant global public health burden [3]. Statistically, breast cancer contributes to 25% of all cancer cases and represents 12% of total new cancer cases diagnosed in women [4]. Medically, breast cancer is defined as uncontrolled cell proliferation originating from breast tissue [5]. This abnormal proliferation forms a mass called a tumor. Tumor classification is divided into two main categories: benign tumors which are non-cancerous, and malignant tumors which are cancerous [6]. Benign tumors have characteristics of slow growth, clear boundaries, and no ability to spread to other tissues or organs. In contrast, malignant tumors exhibit rapid and uncontrolled growth, possess poorly defined margins, and have the capacity to metastasize, spreading to distant organs or tissues throughout the body [7].

World Health Organization (WHO) data shows that breast cancer trends are experiencing significant increases not only in developed countries but also in middle- and low-income countries. Data indicate that more than 2.3 million females received a breast cancer diagnosis in 2020, with the number of deaths estimated at approximately 685,000 cases worldwide. WHO projections show an alarming increase, where the number of deaths from breast cancer increased from 685,000 to 963,000 in 2021 [8]. Breast cancer detection can be performed through various medical examination modalities, including breast ultrasonography, physical examination by professional medical personnel, biopsy (tissue sampling for histopathological analysis), mammography, and breast MRI [9]. Biopsy is considered the gold standard among all diagnostic techniques for breast cancer diagnosis confirmation. Radiologists and specialist doctors then interpret

comprehensive examination results to determine cell malignancy status [10].

However, there are several significant limitations in conventional diagnostic systems. First, limited availability of specialist doctors becomes a barrier to healthcare access. Second, most tumors in early stages do not show clear clinical manifestations. Third, the entire diagnostic process requires relatively long time. Fourth, high medical examination costs become a barrier for some communities [11]. Yet, early breast cancer detection can prevent patients from unnecessary therapy and significantly improve prognosis. The limitations in conventional diagnostic systems drive researchers to develop Computer-Aided Diagnosis (CAD) systems capable of detecting tumors and providing accurate and rapid results without dependence on doctor or radiologist interpretation [12]. Machine learning, as a branch of AI, has been widely applied for disease prediction based on historical data and learning patterns. This technology implements various approaches for performance optimization, including optimization, statistical, and probabilistic techniques.

Comparative studies have demonstrated that experienced specialist physicians achieve a diagnostic accuracy of approximately 79% in detecting breast cancer, whereas machine learning based approaches have been reported to attain accuracy levels of up to 91%, indicating their potential to enhance diagnostic performance [13]. These results show that machine learning has superior capabilities in preventing and detecting diseases compared to conventional methods. Thus, machine learning implementation has great potential to reduce mortality rates from cancer. In recent years, different approaches for forecasting breast cancer have emerged. Classification techniques such as Random Forest (RF), Support Vector Machine (SVM), AdaBoost Classifier, K-Nearest Neighbors (KNN), and XGBoost Classifier have been used in various recent literature. This study employs the Wisconsin Diagnostic Breast Cancer (WDBC) obtained from Kaggle and applies multiple machine learning based classification algorithms to systematically classify breast cancer types in patients with suspected malignancy. Five classification models used in this study include Decision Tree (DT), SVM, Neural Networks, Bagged Tree, and Boosted Tree. This research also implements three different feature selection methods, namely Chi-square (χ^2), Minimum Redundancy Maximum Relevance (MRMR), and ReliefF to optimize model performance.

This research has several main objectives: to conduct experimental analysis on the WDBC dataset from the UCI machine learning repository and examine the correlation between features and the observed target class, to implement various current machine learning classification models on the dataset with various feature selection methods, and to conduct comparative analysis of the results to identify the optimal combination between feature selection methods and classification algorithms. Based on the research background and objectives, this research attempts to answer two main research questions (primary research questions):

1) Among the three feature selection techniques (Chi-square (χ^2), MRMR, and ReliefF), which method demonstrates the highest effectiveness in enhancing the predictive accuracy of breast cancer classification?
2) Among the five classification algorithms (SVM, Neural Networks, DT, Bagged Tree, and Boosted Tree), which model exhibits superior performance based on comprehensive evaluation metrics, including accuracy, precision, specificity, recall, and F1-score?

The WDBC dataset has been used in many studies, but most of them test only one feature selection method or one classifier at a time, which makes it hard to tell which combination works best. This study differs by comparing several feature selection methods and classifiers together under the same settings, so that the most accurate classifier and the most efficient feature selection method can be identified, along with the trade-off between the number of features and accuracy.

II. Literature Review

Previous studies have applied various machine learning techniques to breast cancer detection using the WDBC dataset and its extensions. As summarized in Table I, recent literature highlights a clear progression in optimizing these models through feature selection and algorithm tuning. In 2021, Ara et al. [14] evaluated 30 features from the WDBC dataset, showing that the SVM achieved 96.5% accuracy while DT reached 95.1%. Similarly, Naji et al. [18] demonstrated that reducing the dataset to 11 features still yielded high accuracies of 98.4% for SVM and 98.8% for DT.

In 2022, studies began emphasizing advanced ensembles and optimizing pipelines. Aamir et al. [16] reported that supervised learning models, particularly multilayer perceptron (MLP), could achieve a very high diagnostic accuracy of 99.12% on the WDBC dataset. Furthermore, Rasool et al. [20] proved that optimized machine learning pipelines significantly improve breast cancer prediction, with SVM reaching 99.3% accuracy when combined with appropriate feature selection and Preprocessing.

TABLE I. Summary of Recent Work

Authors	Dataset	Algorithm	Best Accuracy
Ara et al. (2021) [14]	WDBC, 357 benign and 212 malignant, 30 features	SVM, DT	Acc: 96.5% (SVM), 95.1% (DT)
Naji et al. (2021) [18]	WDBC, 357 benign and 212 malignant, 11 features	SVM, DT	Acc: 98.4% (SVM), 98.8% (DT)
Aamir et al (2022) [16]	WDBC, 569 instances, 30 features	RF, GB, SVM, ANN, MLP	Acc: 99.12% (MLP)

Authors	Dataset	Algorithm	Best Accuracy
Rasool et al (2022) [20]	WDBC, 569 instances, 30 features	SVM, LR, KNN, EC	Acc: 99.3% (SVM)
Uddin et al. (2023) [13]	WDBC, 357 benign and 212 malignant, 30 Features	SVM, DT	Acc: 98.07% (SVM), 94.20% (DT)
Khalid et al. (2023) [15]	WDBC, 11 features	DT	Acc: 93.8% (DT)
Strelcenia and Prakoonwit. (2023) [17]	Wisconsin Breast Cancer Diagnosis Dataset	LR, RF, DT, KNN, MLP, XGBoost	Acc: 98.64% (DT)
Al-Imran et al. (2024) [19]	WDBC, 699 instances - NM	SVM	Acc: 98.5% (SVM)
Arifin et al. (2024) [22]	WDBC from Metabric, 15 features	SVM, DT	Acc: 72.9% (SVM), 90.6% (DT)
Okundalaye and Özdemir (2026) [21]	WDBC	SVM, RF, k-NN	Acc: 98.0% (SVM)

Subsequent research in 2023 continued to underscore the critical role of feature engineering and algorithm selection. Uddin et al. [13] and Khalid et al. [15] consistently affirmed the reliability of SVM and DT algorithms, obtaining Accuracy ranging from 93.8% to 98.07% depending on the chosen feature subsets. Moreover, Strelcenia and Prakoonwit [17] showed that feature engineering remains highly effective, with DT reaching 98.64% accuracy in a comparative evaluation of six classifiers.

More recently, studies by Al-Imran et al. [19] and Arifin et al. [22] in 2024 expanded evaluations across different configurations, maintaining SVM's position as a robust classifier with accuracies up to 98.5%. Finally, Okundalaye and Özdemir [21] further confirmed that optimized machine learning workflows based on WDBC can maintain strong predictive performance, with SVM outperforming RF and k-NN under rigorous feature selection and cross-validation.

Taken together, these studies indicate that breast cancer classification performance depends not only on the choice of classifier, but also heavily on feature engineering, dimensionality reduction, and validation strategies.

III. Methodology

The proposed approach to enhance predictive accuracy in breast cancer diagnosis involves the implementation of multiple machine learning algorithms, including DT, SVM, Neural Networks, Bagged Tree, and Boosted Tree, to generate classification predictions. With 5 parameters, 10 parameters, 15 parameters, and 30 parameters tested, resulting in TP, TN, FP, and FN. Figure 1 presents a summary of the research stages in the form of a flow diagram.

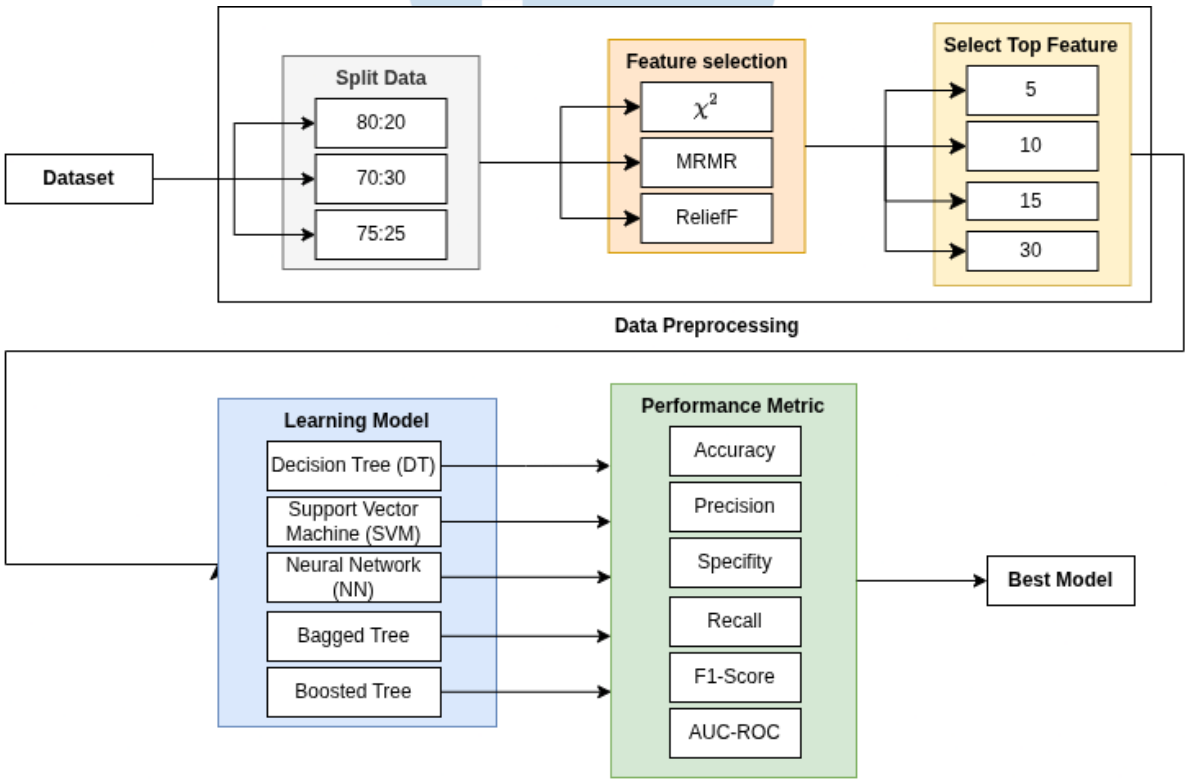


Fig. 1. Methodology of the research

A. Breast Cancer Dataset

This research uses a public dataset available on Kaggle named Wisconsin Diagnostic Breast Cancer (WDBC). This dataset contains parameters used to distinguish between benign (non-tumorous) and malignant (cancer-related) conditions. The dataset consists of 569 data instances and 30 numeric features

obtained from the UCI ML Repository, and contains no missing values. These features are not raw images, they are numeric values computed from digitized images of fine-needle aspirate (FNA) samples of breast masses, where each image was processed to measure characteristics of the cell nuclei (such as radius, texture, perimeter, and area). The classification in this study is therefore performed on these 30 computed features rather than on image pixels. From the total data, the dataset includes 357 benign cases and 212 malignant cases. The description of the 30 features is presented in Table II.

TABLE II. DESCRIPTIONS OF FEATURES

Feature Name	Feature Description
Radius	Mean distance from the centroid to points located along the lesion boundary.
Texture	Variation in grayscale intensity values across the region of interest.
Perimeter	Total length of the boundary contour surrounding the lesion.
Area	Total number of pixels enclosed within the lesion boundary, including a weighted contribution from boundary pixels.
Smoothness	Measure of local variation in radial distances, computed from differences between adjacent radial lengths.
Compactness	Ratio of the squared perimeter to the area, reflecting the degree of shape compactness.
Concavity	Quantitative assessment of the depth or prominence of concave regions along the contour.
Concave points	Number of discrete concave segments identified along the lesion boundary.
Symmetry	Degree of similarity between two halves of the lesion when divided along the major axis, measured by differences in perpendicular distances to the boundary.
Fractal Dimension	Estimated fractal dimension of the contour, representing boundary complexity; higher values indicate increased irregularity, which may correlate with greater malignancy risk.

B. Data Splitting

To ensure robust model evaluation, the dataset was partitioned using stratified random sampling with three training-to-testing ratios: 80:20, 75:25, and 70:30. The use of multiple train–test ratios was intended to assess the sensitivity of classification performance to the amount of training data available. Stratified sampling was applied to preserve the original class distribution of benign and malignant cases in both the training and testing subsets.

For each train–test configuration, a 5-fold cross-validation procedure was performed on the training dataset. The training data were divided into five approximately equal folds. In each iteration, four folds were used to train the model, while the remaining fold was used for validation. This process was repeated five times so that each fold served once as the validation set. The final cross-validation performance was obtained by averaging the results across all five folds. After model validation, the optimized model was evaluated on the independent test set corresponding to the selected train–test ratio.

C. Feature Selection

In this section, the dataset undergoes ranking iterations considering the top 5, 10, 15, and 30 features, where this ranking uses several feature selection algorithms: χ^2, MRMR, ReliefF.

1) Chi-square (χ^2)

Chi-square is a statistical approach that assesses how categorical variables are associated with the target class [23]. The χ^2 method is utilized in this research, as multi-class data is supported and mixed attributes (numeric and discrete) can be effectively handled. The algorithm evaluates the dependency between class labels and intervals using the χ^2 statistic. Intervals with similar class distributions are merged based on the following formula [23]:

$$\chi^2 = \sum_{i=1}^{2}\sum_{j=1}^{k}\frac{\left(A_{ij}-E_{ij}\right)^2}{E_{ij}} \quad (1)$$

k = count of classes
A_{ij} = quantity of patterns in the ith interval, j_{th} class
R_i = total patterns in the i_{th} interval = $\Sigma_{j=1}{}^{k}\, A_{ij}$
C_j = total patterns belonging to the j_{th} class = $\Sigma_{i=1}{}^{2}\, A_{ij}$
N = overall patterns = $\Sigma_{i=1}{}^{2}\, R_i$
E_{ij} = expected value of $A_{ij} = R_i \times \frac{C_j}{N}$

Equation (1) measures the dependency between each feature interval and the class label using the χ^2 statistic. A higher χ^2 value indicates a stronger association between the feature and the target class, meaning that the feature contributes more discriminative information for classification. In this study, the χ^2 criterion was used to rank all candidate features, and the top-ranked 5, 10, 15, and 30 features were retained for subsequent classifier evaluation [23].

2) MRMR

MRMR uses a more sophisticated approach by taking into account two criteria at once, which are maximizing feature relevance to the target class and minimizing redundancy among chosen features [24]. This algorithm is chosen because MRMR is highly effective for high-dimensional datasets as it can identify optimal feature subsets without excessive redundancy. The formula for MRMR is as follows [25]:

$$max X_i \in S\left[I(X_i;c) - \frac{1}{|S|}\sum_{X_j} I(X_j;X_i)\right] \quad (2)$$

Where:

Xi = the *i-th* candidate feature that has not been selected yet (not in the set *S*)
S = The set of features which has been previously selected.
c = the target class/label
Xj = the *j-th* feature contained in *S* (previously chosen)
|S| = the count of features in *S*

Equation (2) shows that MRMR selects features by maximizing their mutual information with the target class while minimizing redundancy among the selected features [25]. In other words, the method favors features that are both highly relevant for distinguishing benign and malignant cases and minimally overlapping in the information they provide. In this study, the MRMR criterion was applied to rank all candidate features, after which the top 5, 10, 15, and 30 features were used as input subsets for the classification models [24].

3) ReliefF

ReliefF is a development of the Relief algorithm designed to handle multi-class data and missing values. This algorithm operates by assessing feature quality, determining how effectively a specific feature can discriminate between instances across different classes [26]. This algorithm is used because ReliefF is capable of detecting interactions between features and can work with various data types without specific distribution assumptions. The core mechanism for estimating feature weights (W) is mathematically formalized as follows [27]:

$$\mathrm{W[A]} = \mathrm{W[A]} - \frac{\mathrm{diff(A, R_i, H)}}{\mathrm{m}} + \frac{\mathrm{diff(A, R_i, M)}}{\mathrm{m}} \quad (3)$$

$$diff(A, X, Y) = \frac{|value(A, X) - value(A, Y)|}{\max(A) - \min(A)} \quad (4)$$

Equation (3) updates the weight of each feature according to its ability to distinguish between neighboring instances from different classes while remaining consistent for instances belonging to the same class. Equation (4) defines the normalized difference function used in the weight update, allowing feature values to be compared on a common scale. Therefore, features that consistently separate malignant from benign samples receive higher weights and higher ranking positions. In this study, the resulting ReliefF weights were used to construct the top 5, 10, 15, and 30 feature subsets for model evaluation [26], [27].

4) Feature Ranking Strategy

Each feature selection algorithm will produce feature rankings based on their respective scores. From these rankings, 4 subsets will be selected: top 5, 10, 15, and 30 features to evaluate their performance. The number of selected features is determined to observe the impact of feature ranking on the computational accuracy of the machine learning model.

D. Machine Learning Classifiers

This section will discuss the working procedures of using DT, SVM, Neural Networks, Gradient Boosted Tree, Gradient Bagged Tree.

1) Decision Trees (DT)

DT is a classification model that computes the output of a function $f(x)$ through a sequential testing process on input x. In this structure, the result of each specific test dictates the subsequent step, proceeding iteratively until the precise value of $f(x)$ is determined [28]. The DT works by selecting the optimal split at each node based on a purity criterion. This model uses the following entropy formula [29]:

$$H(S) = -\Sigma i\ pi\ log2(pi) \quad (5)$$

where pi = the proportion of samples belonging to class *i*. The value of entropy ranges from 0 (pure) to $\log_2$(c), where *c* is the number of classes.

Equation (5) is used to measure the impurity of a node in the decision tree. A lower entropy value indicates that the samples within a node belong predominantly to a single class, whereas a higher entropy value reflects greater class heterogeneity. During training, the decision tree selects splits that maximize entropy reduction, thereby producing more homogeneous child nodes and improving classification performance.

2) Support Vector Machine (SVM)

SVM is a supervised machine learning algorithm used primarily for classification. By analyzing labeled training datasets, the model learns to accurately predict outcomes for new, previously unseen data. SVM operates by identifying a separating hyperplane that establishes the widest possible gap between the closest data points of different categories. This boundary effectively acts as a decision threshold, allowing the model to classify data based on its position relative to the line [30]. Where the equation of the hyperplane is defined as follows [31]:

$$f(x) = w^T x + b \quad (6)$$

where the weight vector is denoted by w, and the bias is represented by b.

Equation (6) defines the separating hyperplane used by SVM to distinguish between benign and malignant classes. The optimal hyperplane is determined by maximizing the margin between support vectors from different classes, thereby improving generalization performance.

3) Neural Networks (NN)

Neural Networks is a computational architecture consisting of interconnected layers of neurons. In this research, the Neural Network model was implemented using the Narrow Neural Network architecture available in MATLAB Classification Learner. This architecture consists of a feedforward neural network with a single hidden layer containing 10 neurons. The

number of input neurons depends on the selected feature subset (5, 10, 15, or 30 features), while the output layer contains two neurons representing benign and malignant classes. The selection of Neural Networks is based on its ability to learn complex non-linear patterns in data and its proven effectiveness in various classification tasks [32].

The learning process is performed using the backpropagation algorithm. The main formula used in weight updating is given as follows [33]:

$$\frac{\partial E}{\partial w} = \frac{\partial E}{\partial o} \times \frac{\partial o}{\partial net} \times \frac{\partial net}{\partial w} \quad (7)$$

where:
E = the error function
w = the weight
o = the output
net = the weighted sum of inputs

Equation (7) represents the weight adjustment process in the backpropagation learning algorithm. During training, the network minimizes the error function by iteratively updating connection weights, enabling the model to learn complex nonlinear relationships within the dataset.

4) Boosted Tree

Boosted Tree is an ensemble method that combines multiple decision trees sequentially for classification tasks. Each new tree is trained to correct prediction errors from the combination of previous trees by minimizing the loss function through gradient descent [34]. The main formula of this model is as follows [35]:

$$F_{m(x)} = F_{\{m-1\}(x)} + \gamma_m h_{m(x)} \quad (8)$$

where:
$F_{\mathrm{m}}(x)$ = the ensemble model at the m-th iteration
γ_{m} is the learning rate
$h_{\mathrm{m}}(x)$ = the weak learner (decision tree) trained to predict the negative gradient of the loss function

Equation (8) describes the ensemble prediction generated during each boosting iteration. The model progressively reduces classification errors by adding weak learners that focus on previously misclassified samples.

5) Bagged Tree

Bagged Tree is an ensemble method that uses bootstrap aggregating techniques to improve classification accuracy. In this ensemble, multiple decision trees are trained on resampled subsets of the dataset, with final predictions determined through majority voting [34]. The main formula of this model is as follows [36]:

$$\varphi_{A(x)} = \left(\frac{1}{B}\right) * \Sigma(b = 1 \; to \; B)\varphi(x, L_b) \quad (9)$$

where:
$\varphi_{A(x)}$ = the aggregated predictor
B = the number of bootstrap samples
φ (x, L_b) = the predictor trained on the b-th bootstrap sample
L_b = the bootstrap sample drawn from the original dataset

Equation (9) represents the aggregation mechanism of bagging, where predictions from multiple bootstrap-trained trees are combined through majority voting to improve stability and reduce overfitting.

E. Hyperparameter Configuration

All classification models were implemented using MATLAB R2024a Classification Learner using the built-in classifier configurations. The Support Vector Machine (SVM) classifier employed a Gaussian kernel function with standardized predictors. The Decision Tree classifier used the default Gini diversity criterion for node splitting. The Neural Network classifier utilized MATLAB's Narrow Neural Network architecture, while Bagged Tree and Boosted Tree classifiers were implemented using the ensemble learning methods available in MATLAB Classification Learner. To ensure a fair comparison, all classifiers were trained and evaluated under identical data partition settings, feature selection methods, and performance evaluation metrics. All model hyperparameters were kept at the default settings provided by MATLAB R2024a Classification Learner unless otherwise specified. For SVM, a Gaussian kernel with standardized predictors was used. The Neural Network model employed MATLAB's Narrow Neural Network architecture with one hidden layer of 10 neurons. The Bagged Tree and Boosted Tree classifiers were implemented using MATLAB's default ensemble settings. Model selection within each training set was based on the average validation performance obtained from 5-fold cross-validation, and the selected model was then evaluated on the corresponding held-out test set.

F. Model Evaluation

Each classification algorithm's performance will be assessed based on the following metrics [36]:

1) Accuracy (A): the ratio of correct predictions to the total number of predictions made. Accuracy formula:

$$\mathrm{A} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \quad (10)$$

2) Precision (PPV): proportion of true positive from total positive predictions. Precision formula:

$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \quad (11)$$

3) Specificity (TNR): proportion of true negative from total negative predictions. Specificity formula:

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \quad (12)$$

4) Recall (TPR): proportion of true positive from total actual positive cases. Recall formula:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \quad (13)$$

5) F1-score (F): harmonic mean of precision and recall. F1-score formula:

$$\mathrm{F} = \frac{2 \times \mathrm{PPV} \times \mathrm{TPR}}{\mathrm{PPV} + \mathrm{TPR}} \quad (14)$$

6) AUC-ROC: total area situated beneath the Receiver Operating Characteristic curve.

Equations (10) - (14) define the evaluation metrics used to compare the performance of all classification models. Accuracy in Equation (10) measures the overall proportion of correct predictions, precision in Equation (11) reflects the reliability of positive predictions, specificity in Equation (12) quantifies the correct identification of benign cases, recall in Equation (13) measures the ability to detect malignant cases, and F1-score in Equation (14) summarizes the balance between precision and recall. These metrics were selected to provide a comprehensive assessment of model performance, particularly because breast cancer classification requires not only high overall accuracy but also strong sensitivity and specificity [36].

IV. Results and Discussions

This part offers an in-depth examination of the results from experiments focused on applying machine learning methods for identifying breast cancer in the fields of healthcare studies and data analytics.. A total of 180 experimental scenarios were systematically designed by varying multiple parameters, including data partition ratios (80:20, 75:25, and 70:30), feature selection techniques (χ^2, MRMR, and ReliefF), the number of chosen features (5, 10, 15, and 30), and classification algorithms (DT, SVM, Neural Networks, Bagged Tree, and Boosted Tree). Model performance was rigorously assessed using several evaluation metrics, namely accuracy, precision (PPV), specificity (TNR), recall (TPR), and F1-score. All experiments were implemented using MATLAB R2024a to identify the most optimal combination of feature selection strategies and classification models for breast cancer diagnosis.

A. Algorithm Performance Comparison

Table III presents the best performance achieved by each algorithm across all experimental configurations. SVM achieved the highest accuracy of 97.81% using MRMR feature selection with 15 features and 80:20 data split, followed by Neural Networks with 97.42% accuracy.

TABLE III. Performance Comparison of Machine Learning Algorithms

Algorithm	FS	No. Features	Train: Test	Acc	PPV	TNR	F1-score	Recall
SVM	MRMR	15	80:20	97.81	99.38	99.65	96.99	94.71
Neural Network	ReliefF	30	75:25	97.42	98.05	98.88	96.49	94.97
Bagged Tree	MRMR	30	80:20	95.83	96.32	97.90	94.29	92.35
Decision Tree	χ^2	5	80:20	94.08	91.81	95.10	92.08	92.35
Boosted Tree	MRMR	5	75:25	75.18	93.44	98.51	51.82	35.85

SVM consistently demonstrated superior performance across different configurations. The algorithm achieved high precision (99.38%) and specificity (99.65%), indicating excellent ability to correctly identify both malignant and benign cases. Neural Networks showed competitive performance with balanced metrics across all evaluation criteria.

B. Performance Comparison Based on Feature Selection Methods

Table IV compares the effectiveness of different feature selection methods under the 80:20 data split in achieving optimal performance of each algorithm.

TABLE IV. Accuracy Comparison of Feature Selection Methods Across Algorithm.

Algorithm	χ^2(%)	MRMR(%)	ReliefF(%)	Best Feature Selection
SVM	97.81(30)	**97.81(15)**	97.81(30)	MRMR with 15 parameters
Neural Networks	**96.49(30)**	96.05(30)	96.27(15)	χ^2 with 30 parameters
Bagged Tree	94.52(30)	**95.83(30)**	95.39(5)	MRMR with 30 parameters
DT	**94.08(5)**	93.64(30)	93.42(5)	χ^2 with 5 parameters
Boosted	69.08(15)	69.30(5)	**69.52(30)**	ReliefF with 30 parameters

MRMR demonstrated superior efficiency by achieving peak accuracy with only 15 features, representing a 50% reduction compared to the full feature set. All three feature selection methods achieved the same peak accuracy (97.81%) when combined with SVM, indicating the robustness of the algorithm.

Note that Table IV is restricted to the 80:20 split; the overall best Neural Networks result (97.42% with ReliefF) in Table III occurs at the 75:25 split.

C. Algorithm Performance Analysis

1) Support Vector Machine (SVM)

SVM demonstrated superior and consistent performance compared to other algorithms. SVM's highest accuracy reached 97.81% with various feature selection configurations. Since SVM separates the two classes by searching for the widest margin between them, it worked well on the standardized features of this dataset. Its precision (99.38%) and specificity (99.65%) were high because very few benign cases were predicted as malignant. The accuracy of SVM also changed little across the three train–test ratios, showing that it was not very sensitive to the amount of training data.

2) Neural Networks (NN)

Neural Networks showed very good performance with the highest accuracy of 97.42% in the configuration of top 30 features with ReliefF. Neural Networks is capable of capturing complex patterns in data, but requires more careful parameter tuning compared to SVM.

3) Decision Tree (DT)

DT provided good performance with average accuracy above 90%. Although it did not achieve accuracy as high as SVM or Neural Networks, DT still provided acceptable results with the highest accuracy of 94.08% on the top 5 features with χ^2

4) Bagged Tree

Bagged Tree showed improved performance compared to single DT with the best accuracy of 95.83% in several configurations. This ensemble method is effective in reducing overfitting and improving model generalization.

5) Boosted Tree

Boosted Tree showed suboptimal performance with very low accuracy (around 62-75%). The confusion matrices explain why. In most configurations the model predicted almost everything as benign, producing very few true positives and a recall near zero. An accuracy of 62% here mainly reflects the share of benign cases in the data rather than any real ability to detect malignant tumours. Even the best run, 75.18% with MRMR and five features, reached a recall of only 35.85%, meaning many malignant cases were missed. Such a result is not acceptable for screening. Since the other four classifiers handled the same data well, this failure is specific to the boosting configuration rather than a property of the dataset. The most likely cause is the default ensemble setting in MATLAB, which was not suited to this case, and not a weakness of boosting as a method, so the result should not be generalized. Tuning the number of weak learners and the learning rate would be expected to improve it.

D. Feature Selection Performance Analysis

1) Chi-square Feature Selection

Evaluation results using the Chi-square method showed that SVM had the best performance with accuracy up to 97.81% on the top 30 features (80:20 data split). In this configuration, SVM achieved precision of 98.78%, specificity of 99.30%, F1-score of 97.01%, and recall of 95.29%. Neural Networks also showed good performance with 96.49% accuracy on the top 30 features, while DT achieved maximum accuracy of 94.08% on the top 5 features.

2) MRMR Feature Selection

The MRMR approach delivered excellent performance, with the SVM model reaching its highest accuracy of 97.81% when using the top 15 features under the 80:20 data split. These results demonstrate MRMR's efficiency in selecting relevant features with minimum redundancy. Under this setup, the SVM model recorded strong metric performance, attaining a precision of 99.38%, a specificity of 99.65%, an F1-score of 96.99%, and a recall rate of 94.71%. Neural Networks with MRMR achieved 96.96% accuracy at the 75:25 data split with the top 30 features.

3) ReliefF Feature Selection

ReliefF showed competitive performance with SVM achieving 97.81% accuracy on the top 30 features (80:20 data split). Neural Networks with ReliefF demonstrated very good performance with 97.42% accuracy at the 75:25 data split with the top 30 features, achieving precision of 98.05%, specificity of 98.88%, F1-score of 96.49%, and recall of 94.97%.

V. Conclusion and Future Works

Recent advances in computational intelligence have highlighted the significant role of machine learning cancer diagnosis. In this study, the WDBC was utilized to conduct a comprehensive evaluation of several classification algorithms, namely SVM, DT, Neural Networks, Boosted Tree, and Bagged Tree. These classifiers were systematically integrated with three feature selection approaches, MRMR, χ^2, and ReliefF, across multiple feature subset configurations (5, 10, 15, and 30 features).

The empirical findings demonstrate that SVM consistently showed better performance compared to the other models for every evaluation measure, achieving a maximum accuracy of 97.81% when 15 and 30 features were employed. Notably, this peak accuracy was reached by MRMR using only 15 of the 30 features, halving the dimensionality without any loss in accuracy, which is useful for building lighter computer-aided diagnosis systems. In contrast, The Bagged Tree model also showed strong performance, reaching its highest accuracy of 95.83% with MRMR using 30 selected features. In a reduced-feature setting, the model remained competitive, achieving 95.39% with ReliefF using only five features. These outcomes underscore the critical importance of aligning appropriate feature selection strategies with suitable classification algorithms to optimize predictive performance in breast cancer diagnosis. Within the scope of this investigation, SVM, Neural Networks, and Bagged Tree models were identified as the most effective configurations.

The results further provide methodological implications for subsequent research. While a conventional Neural Network architecture was implemented, more sophisticated deep learning models, such as Convolutional Neural Networks (CNNs), may yield enhanced predictive capability, particularly when complemented by data augmentation techniques. Despite the high performance achieved by SVM, further refinement remains feasible. Enhanced model performance may be realized through more rigorous hyperparameter optimization, including systematic tuning of kernel functions (e.g., linear, radial basis function, or polynomial), the regularization parameter

(C), and the gamma parameter. Additionally, the incorporation of dimensionality reduction techniques such as PCA, as well as imbalance-handling methods like SMOTE, may further strengthen generalization performance and model robustness.

ACKNOWLEDGMENT

The authors gratefully acknowledge the Department of Medical Technology, Institut Teknologi Sepuluh Nopember, for providing the facilities and institutional support that contributed to the completion of this research.

REFERENCES

[1] [1 Y. Zhang, Y. Ji, S. Liu, J. Li, J. Wu, Q. Jin, X. Liu, H. Duan, Z. Feng, Y. Liu, Y. Zhang, Z. Lyu, F. Song, F. Song, L. Yang, H. Liu, and Y. Huang, "Global burden of female breast cancer: new estimates in 2022, temporal trend and future projections up to 2050 based on the latest release from GLOBOCAN," J. Natl. Cancer Cent., vol. 5, no. 3, pp. 287–296, 2025, doi: https://doi.org/10.1016/j.jncc.2025.02.002.

[2] A. C. Society, "Breast Cancer Facts & Figures 2023-2024," American Cancer Society, Atlanta, 2023. [Online]. Available: https://www.cancer.org/research/cancer-facts-statistics/breast-cancer-facts-figures.html

[3] R. L. Siegel, K. D. Miller, N. S. Wagle, and A. Jemal, "Cancer statistics, 2023.," CA. Cancer J. Clin., vol. 73, no. 1, pp. 17–48, Jan. 2023, doi: 10.3322/caac.21763.

[4] H. Sung, J. Ferlay, R. L. Siegel, M. Laversanne, I. Soerjomataram, A. Jemal, and F. Bray, "Global Cancer Statistics 2020: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries.," CA. Cancer J. Clin., vol. 71, no. 3, pp. 209–249, May 2021, doi: 10.3322/caac.21660.

[5] W. H. Organization, "Breast cancer," World Health Organization, Mar. 2024. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/breast-cancer

[6] P. H. Tan, I. Ellis, K. Allison, E. Brogi, S. B. Fox, S. Lakhani, A. J. Lazar, E. A. Morris, A. Sahin, R. Salgado, A. Sapino, H. Sasano, S. Schnitt, C. Sotiriou, P. van Diest, V. A. White, D. Lokuhetty, and I. A. Cree, "The 2019 World Health Organization classification of tumours of the breast," Histopathology, vol. 77, no. 2, pp. 181–185, 2020, doi: 10.1111/his.14091.

[7] W.-C. Shia, L.-S. Lin, and D.-R. Chen, "Classification of malignant tumours in breast ultrasound using unsupervised machine learning approaches," Sci. Rep., vol. 11, no. 1, p. 1418, 2021, doi: 10.1038/s41598-021-81008-x.

[8] M. Arnold, E. Morgan, H. Rumgay, A. Mafra, D. Singh, M. Laversanne, J. Vignat, J. R. Gralow, F. Cardoso, S. Siesling, and I. Soerjomataram, "Current and future burden of breast cancer: Global statistics for 2020 and 2040.," Breast, vol. 66, pp. 15–23, Dec. 2022, doi: 10.1016/j.breast.2022.08.010.

[9] J. T. Henderson, E. M. Webber, M. S. Weyrich, M. Miller, and J. Melnikow, "Screening for Breast Cancer: Evidence Report and Systematic Review for the US Preventive Services Task Force.," JAMA, vol. 331, no. 22, pp. 1931–1946, Jun. 2024, doi: 10.1001/jama.2023.25844.

[10] F. Cucchiara, R. Scarpitta, S. Crucitta, C. Scatena, R. Arici, A. G. Naccarato, S. Fogli, R. Danesi, and M. Del Re, "Diagnosis and treatment monitoring in breast cancer: how liquid biopsy can support patient management.," Pharmacogenomics, vol. 23, no. 2, pp. 119–134, Jan. 2022, doi: 10.2217/pgs-2021-0099.

[11] K. Loizidou, R. Elia, and C. Pitris, "Computer-aided breast cancer detection and classification in mammography: A comprehensive review," Comput. Biol. Med., vol. 153, p. 106554, 2023, doi: 10.1016/j.compbiomed.2023.106554.

[12] H. Asri, H. Mousannif, H. Al Moatassime, and T. Noel, "Using Machine Learning Algorithms for Breast Cancer Risk Prediction and Diagnosis," Procedia Comput. Sci., vol. 83, pp. 1064–1069, 2016, doi: https://doi.org/10.1016/j.procs.2016.04.224.

[13] K. M. M. Uddin, N. Biswas, S. T. Rikta, and S. K. Dey, "Machine learning-based diagnosis of breast cancer utilizing feature optimization technique," Comput. Methods Programs Biomed. Updat., vol. 3, p. 100098, 2023, doi: https://doi.org/10.1016/j.cmpbup.2023.100098.

[14] S. Ara, A. Das, and A. Dey, "Malignant and Benign Breast Cancer Classification using Machine Learning Algorithms," in 2021 International Conference on Artificial Intelligence (ICAI), 2021, pp. 97–101. doi: 10.1109/ICAI52203.2021.9445249.

[15] A. Khalid, A. Mehmood, A. Alabrah, B. F. Alkhamees, F. Amin, H. AlSalman, and G. S. Choi, "Breast Cancer Detection and Prevention Using Machine Learning.," Diagnostics (Basel, Switzerland), vol. 13, no. 19, Oct. 2023, doi: 10.3390/diagnostics13193113.

[16] S. Aamir, A. Rahim, Z. Aamir, S. F. Abbasi, M. S. Khan, M. Alhaisoni, M. A. Khan, K. Khan, and J. Ahmad, "Predicting Breast Cancer Leveraging Supervised Machine Learning Techniques," Computational and Mathematical Methods in Medicine, vol. 2022, Article ID 5869529, 2022, doi: 10.1155/2022/5869529.

[17] E. Strelcenia and S. Prakoonwit, "Effective Feature Engineering and Classification of Breast Cancer Diagnosis: A Comparative Study," BioMedInformatics, vol. 3, no. 3, pp. 616-631, 2023, doi: 10.3390/biomedinformatics3030042.

[18] M. A. Naji, S. El Filali, K. Aarika, E. L. H. Benlahmar, R. A. Abdelouhahid, and O. Debauche, "Machine Learning Algorithms For Breast Cancer Prediction And Diagnosis," Procedia Comput. Sci., vol. 191, pp. 487-492, 2021, doi: https://doi.org/10.1016/j.procs.2021.07.062.

[18] Al-Imran, S. Akter, M. A. S. Mozumder, R. J. Bhuiyan, T. Rahman, M. J. Ahmmed, M. N. H. Mir, M. A. Hasan, A. C. Das, and M. E. Hossen, "Evaluating Machine Learning Algorithms for Breast Cancer Detection: A Study on Accuracy and Predictive Performance," Am. J. Eng. Technol., vol. 6, no. 09, pp. 22-33, 2024, doi: 10.37547/tajet/Volume06Issue09-04.

[19] A. Rasool, C. Bunterngchit, L. Tiejian, M. R. Islam, Q. Qu, and Q. Jiang, "Improved Machine Learning-Based Predictive Models for Breast Cancer Diagnosis," International Journal of Environmental Research and Public Health, vol. 19, no. 6, art. 3211, 2022, doi: 10.3390/ijerph19063211.

[20] O. O. Okundalaye and N. Özdemir, "Early breast cancer prediction using optimized machine learning and tumor-immune modeling," Journal of Computational and Applied Mathematics, vol. 473, art. 116875, 2026, doi: 10.1016/j.cam.2025.116875.

[21] T. Arifin, I. W. P. Agung, E. Junianto, R. Rachman, I. R. Wibowo, and D. D. Agustin, "Breast cancer identification using machine learning and hyperparameter optimization," Indones. J. Electr. Eng. Comput. Sci., vol. 36, no. 3, pp. 1620–1630, 2024, doi: 10.11591/ijeecs.v36.i3.pp1620-1630.

[22] R. Setiono and H. Liu, "Chi2: Feature Selection and Discretization of Numeric Attributes," in 2012 IEEE 24th International Conference on Tools with Artificial Intelligence, 1995, p. 388. doi: 10.1109/TAI.1995.479783.

[23] I. Jo, S. Lee, and S. Oh, "Improved Measures of Redundancy and Relevance for mRMR Feature Selection," 2019. doi: 10.3390/computers8020042.

[24] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy," IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 8, pp. 1226–1238, 2005, doi: 10.1109/TPAMI.2005.159.

[25] O. Reyes, C. Morell, and S. Ventura, "Scalable extensions of the ReliefF algorithm for weighting and selecting features on the multi-label learning context," Neurocomputing, vol. 161, pp. 168–182, 2015, doi: https://doi.org/10.1016/j.neucom.2015.02.045.

[26] M. Robnik-Šikonja and I. Kononenko, "Theoretical and Empirical Analysis of ReliefF and RReliefF," Mach. Learn., vol. 53, no. 1, pp. 23–69, 2003, doi: 10.1023/A:1025667309714.

[27] H. Blockeel, L. Devos, B. Frénay, G. Nanfack, and S. Nijssen, "Decision trees: from efficient prediction to responsible AI," Front. Artif. Intell., vol. Volume 6-, 2023, doi: 10.3389/frai.2023.1124553.

[28] C. E. Shannon, "A Mathematical Theory of Communication," Bell Syst. Tech. J., vol. XXVII, no. 3, pp. 379–423, 1948.

[29] I. Bomze, F. D'Onofrio, L. Palagi, and B. Peng, "Feature selection in linear support vector machines via a hard cardinality constraint: A scalable conic decomposition approach," Eur. J. Oper. Res., 2025, doi: https://doi.org/10.1016/j.ejor.2025.03.017.

[30] C. Cortes and V. Vapnik, "Support-vector networks," Mach. Learn., vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.

[31] E. Bollt, "How neural networks work: Unraveling the mystery of randomized neural networks for functions and chaotic dynamical systems," Chaos An Interdiscip. J. Nonlinear Sci., vol. 34, no. 12, p. 123164, Dec. 2024, doi: 10.1063/5.0206840.

[32] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," Nature, vol. 323, no. 6088, pp. 533–536, 1986, doi: 10.1038/323533a0.

[33] A. Plaia, S. Buscemi, J. Fürnkranz, and E. L. Mencía, "Comparing Boosting and Bagging for Decision Trees of Rankings," J. Classif., vol. 39, no. 1, pp. 78–99, 2022, doi: 10.1007/s00357-021-09397-2.

[34] J. H. Friedman, "Greedy function approximation: A gradient boosting machine.," Ann. Stat., vol. 29, no. 5, pp. 1189–1232, Oct. 2001, doi: 10.1214/aos/1013203451.

[35] L. Breiman, "Bagging predictors," Mach. Learn., vol. 24, no. 2, pp. 123–140, 1996, doi: 10.1007/BF00058655.

[36] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," Information Processing & Management, vol. 45, no. 4, pp. 427–437, 2009, doi: 10.1016/j.ipm.2009.03.002.

[37] A. Elaraby, A. Saad, H. Elmannai, M. Alabdulhafith, M. Hadjouni, and M. Hamdi, "An approach for classification of breast cancer using lightweight deep convolution neural network," Heliyon, vol. 10, no. 20, p. e38524, Oct. 2024, doi: 10.1016/j.heliyon.2024.e38524.

ISSN 2355-3286

Design and Development of an Automated Gymnastics Movement Evaluation Device

Ari Bawono Putranto[1*], Naurah Elfrida Anargya[2], Ahmad Ridlo Hanifudin Tahier[3], Fakhruddin Mangkusasmito[4]

[1,2,3]Automation Engineering Technology, Diponegoro University, Semarang, Indonesia
[4] Industrial Electrical Engineering, Diponegoro University, Semarang, Indonesia
[1]aribawonoputranto@lecturer.undip.ac.id , [2]naurahelfrida@gmail.com, [3]ridlohanifudin@lecturer.undip.ac.id, [4]fakhm17@lecturer.undip.ac.id
* Corresponding Author

Accepted on February 20, 2026
Approved on June 28, 2026

***Abstract*— The assessment of rhythmic gymnastics need precision, balance, and harmonious body coordination. It is often carried out with visual observation. This research aims to develop a sensor-based assessment system to process, classify, and score measurement data automatically. An ADXL335 accelerometer sensor and an HMC5883L magnetometer sensor are integrated with an ESP32-C3 mini transmitter to acquire three-axis acceleration and heading data in real time. The receiver sends data to a PC application that performs downsampling with average pooling to reduce signal density, min–max normalisation to harmonise scale across axes, and compares the processed test sequence against reference data. Deviations are quantified through per-parameter error computation, which is converted to accuracy and generate an overall performance score by combined using predefined weights that is displayed on an LCD/P10 panel. Experimental trials on SKJ 2012 warm-up movements show that the proposed method can classify sensor data and generate scoring outputs. The findings indicate that sequences closely matching the reference pattern consistently yield high scores about 80–85, while deliberately incorrect movement produce much lower scores, around. This research provides an objective, repeatable, and real-time scoring mechanism that approach for transforming sensor measurements into interpretable assessment results.**

***Index Terms*— ADXL335 Accelerometer; HMC5883L Magnetometer; Average Pooling; Min-Max Normalisation; Rhythmic Gymnastics Performance Evaluation**

I. INTRODUCTION

Rhythmic gymnastics combines repeated body movement, timing, balance, and coordination. The objective of gymnastics is to enhance health and physical fitness [1] [2]. The intensity of gymnastic movements varies according to the rhythm of the selected music [3]. In order to determine the quality of performance and competence of participants in performing movements in accordance with the movement references of each gymnastics routine, the evaluation process in sports activities must be considered. Currently, rhythmic gymnastics students are predominantly assessed manually that rely on visual observation. Nevertheless, this approach is tends to inconsistency and bias due to factors such as assessor fatigue, environmental conditions and time constraints that are difficult to avoid [5]. Wearable inertial sensors provide a practical route to reduce this subjectivity because they can record body motion directly during training [4]. Automatic scoring has also been explored in other sports contexts, such as electronic scoring for taekwondo, and statistical approaches have been discussed for improving fairness in gymnastics judging [6]. Research undertaken hitherto has demonstrated that the implementation of accelerometer and magnetometer sensors can be utilised to facilitate the monitoring of rhythmic gymnastics activities in conjunction with an Android application [7]. However, the present research has not been equipped with an automatic assessment feature that can directly generate a final score for gymnastics activities. Moreover, concomitant research employing the HMC5883L magnetometer sensor for the development of shooting practice and research on tools for monitoring sit-ups with accelerometer and gyroscope sensors demonstrates that the application of sensor technology can enhance the accuracy of movement phase determination [8] [9].

Accordingly, this research developed an automatic gymnastics movement assessment system as a pioneering solution that minimizes subjectivity and inconsistency in the assessment process [10]. The system comprises an ESP32-C3 Mini as a transmitter and a Wemos D1 Mini ESP32 as a receiver, operating in an integrated manner. The transmitter module is employed to collect data from the ADXL335 accelerometer and HMC5883L magnetometer sensors, thereby enabling the real-time recording of SKJ 2012 [11] [12] [13]. Meanwhile, the receiver module, comprising the power supply and LCD components, is tasked with receiving data, processing the data sent to the application, and sending the score to the display as a form of data visualisation.

II. METHOD

In this research, an automatic gymnastics movement assessment tool was designed using the downsampling method with average pooling and min-max normalisation. Integrating key components such as the ADXL335 accelerometer sensor, HMC5883L magnetometer sensor, ESP32 C3-Mini, and Wemos D1 Mini ESP32 using client-server communication [14] [15].

A. System Design

The device's design requires the utilisation of a block diagram to facilitate comprehension of its operational workflow. The design of this device consist of three modules: the transmitter module, the receiver module, and the display panel module. As illustrated in Figure 1a, the transmitter module facilitates the transmission of sensor data to the receiver module, a process that necessitates the utilisation of several components, including the ESP32-C3 mini, a magnetometer sensor, an accelerometer sensor, a 18650 Li-Ion battery, an on-off switch, an LED, and a step-down voltage regulator. As illustrated in Figure 1b, the receiver module is responsible for the processing of data and the transmission of values to the display panel module, which subsequently displays the values visually. The components required for the construction of the device are as follows: an on-off switch, a DF player mini, a 12V SMPS, a Wemos D1 mini ESP32, a 20x4 LCD, a push button, an LED, an EEPROM, a PAM 8610, a step-down voltage regulator, an HC-05 Bluetooth module, and a PC. Meanwhile, the components in Figure 1c require a 5V SMPS, HC-05 Bluetooth module, Arduino Nano, and P10 display panel.

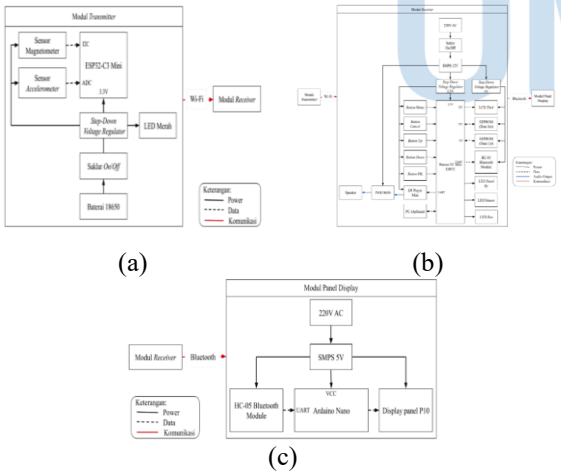


Figure 1. Block Diagram of the Module System (a) Transmitter and (b) Receiver (c) Display Modul panel

In the process of designing an automatic gymnastics movement assessment tool, the flowchart or flow diagram illustrating the overall working concept can be found in Figure 2. The preliminary stage in the system design process is to initiate the system, which involves configuring the hardware and software to enable connection. This encompasses the configuration of the transmitter and receiver modules, the initialisation of the microcontroller, the initialisation of the accelerometer and magnetometer sensors for data collection, and the activation of the LCD and button inputs. Subsequent to the establishment of a connection, the user is able to select the type of rhythmic gymnastics to be used. The sensors will immediately record body orientation and movement data in real time, then transmit it to the server via a Wi-Fi connection and forward it to the PC/laptop application [16]. The test data will then be compared with the master data from the application. The participant's performance score is calculated based on various predetermined parameters. This score will then be transmitted back to the server via Bluetooth connection and forwarded to the display panel module. The retesting process can be repeated by executing the same procedure once more.

The complete workflow is shown in Figure 2. After the system is initialized, the user selects the gymnastics routine to be evaluated. The transmitter then records acceleration and orientation data while the participant performs the movement. The receiver sends this data to the PC application, where the test sequence is processed and compared with the master sequence. The calculated score is returned to the receiver and then sent to the P10 display.

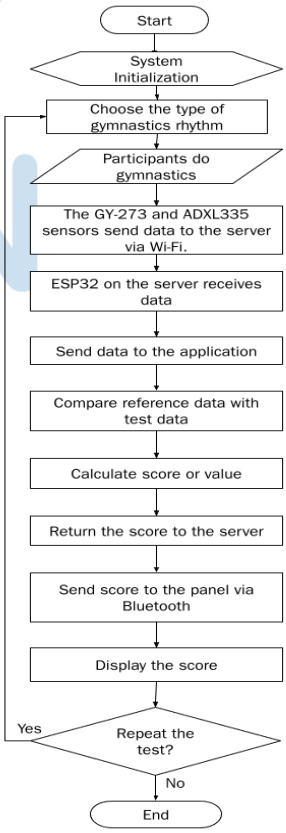


Figure 2. System Flowchart

B. 3D Design

The 3D design consists of three constituent parts: a transmitter module measuring 10 x 7.5 x 3.5 cm, a receiver module measuring 23.5 x 19.5 x 10 cm, and a display panel module measuring 69 x 37 x 5 cm. The transmitter module is composed of 1 mm-thick plastic, with a relatively diminutive size. It is equipped with an on/off switch and LED indicator to signal that the module is active. The receiver module has been equipped with ventilation on the side and top to prevent overheating, and is equipped with a 20 x 4 LCD, 5 buttons for menu navigation and system settings, 3 LEDs for indicators, and a front panel with an on/off switch. The display panel module is equipped with four panels, each with a black casing to protect the components.

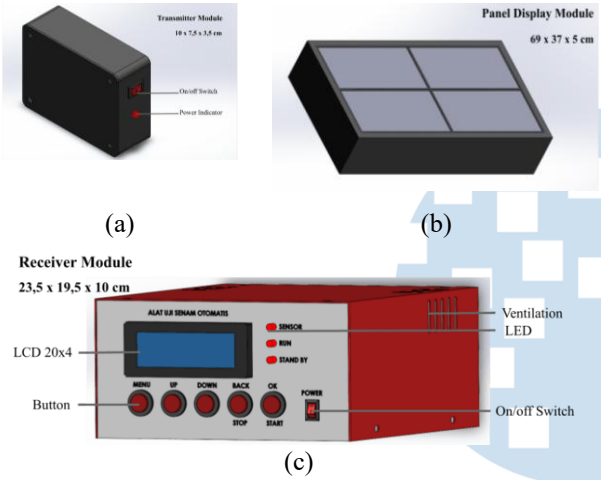


Figure 3. 3D Design of Modules (a) Transmitter, (b) Display Panel, (c) Receiver

C. Specifications and Features

The specifications and features required for an automatic gymnastics assessment tool consist of a system with accelerometer and magnetometer sensors, wireless sensor data via Wi-Fi protocol with IP address configuration, a receiver module with settings and selection of gymnastics accompaniment songs, movement assessment results, a transmitter module with compact and lightweight LED indicators, and a 120,000-byte sensor data storage system.

D. Software Design

The design of software is initiated with the acquisition of sensor data on the transmitter module, where the ADXL335 accelerometer sensor is converted from analogue to digital signals via a 12-bit ADC (with a sampling rate of 25 Hz), while the HMC5883L magnetometer sensor transmits data in digital format via the I2C communication protocol [17] [18]. The data are sent to the receiver module is facilitated by means of Wi-Fi UDP communication. It is evident that the measurements obtained from the two types of sensors result in the generation of four parameters, namely the X, Y, Z axes, and direction [19]. The subsequent stage of the process is the sensor data processing stage, which includes downsampling using the average pooling method and min-max normalisation. The evaluation of test data is achieved through a comparison with master data, with the error value per parameter being calculated using equations 1 and 2. Subsequently, the accuracy is calculated based on equation 3, then multiplied by the weighting of each parameter according to equation 4 to obtain a final score, which is then displayed on the display panel.

$$Total\ error_i = \sum |\text{reference value}_i - \text{measured value}_i| \quad (1)$$

$$\text{Relative error}_i = \frac{Total\ Error_i}{\sum|\text{Total Reference error}_i|} \times 100\% \quad (2)$$

$$\text{Accuracy}_i = 100 - \text{Relative error}_i \quad (3)$$

$$\text{Score} = \sum Evaluation \times \text{Accuracy}_i \quad (4)$$

E. Hardware Design

The hardware assembly is began with the selection and testing of elements that are deemed essential to the functionality of the device. These components encompass the power supply, the display panel, the microcontroller, the ADXL335 accelerometer sensor, and the HMC5883L magnetometer sensor. Subsequently, the direction and placement of the transmitter module in the participant's back pocket are determined in order to obtain data with sensors that are adjusted according to the direction of the body. The subsequent stage involves the assembly and integration of components such as PCB components, cable connections, and displays for the gymnastics movement assessment device, as well as testing the hardware system, which includes sensor response, checking communication between modules, LED and LCD panel data display, MP3 player, and power supply stability.

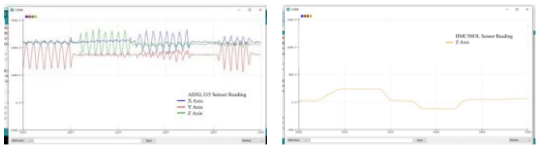
III. Result and Discussion

This research comprised a series of tests, including the evaluation of the ADXL335 and HMC5883L sensors, the assessment of the functionality of the DFPlayer Mini player, the analysis of WiFi UNDIP and Bluetooth HC-05 communication, the testing of EEPROM data storage, the investigation of gymnastics sample processing with downsampling, average pooling and data normalisation, and the evaluation of the overall movement scoring system for optimisation of the automatic rhythmic gymnastics apparatus prototype.

A. Sensor Testing

A programme of sensor testing was initiated in order to ensure the stability of sensor performance. The ADXL335 accelerometer was tested by moving the module forwards and backwards (X-axis), up and down (Y-axis), and left and right (Z-axis), while the HMC5883L magnetometer sensor was tested by

rotating the module left and right for heading calibration. The results of the ADXL335 accelerometer test are visualised in Figure 4a, which shows sharp oscillations on the dominant axis, while Figure 4b for the HMC5883L magnetometer shows variations in heading, enabling it to capture vibrations and changes in the rotational orientation of gymnastic movements. The dynamic response of the sensors to changes in linear acceleration and relative magnetic field changes is illustrated by this graphical representation. These changes are used to distinguish between low-impact and high-impact phases, and changes in body orientation during rhythmic gymnastics movements.

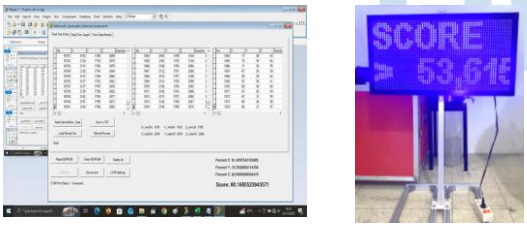


(a) (b)

Figure 4. Graphical Visualisation of Sensor Readings (a) ADXL335 Sensor and (b) HMC5883L Sensor

B. Data Communication Testing

A series of tests were conducted to ensure the reliability of data transfer within the system, with a particular focus on inter-device communication. The Bluetooth master-slave communication testing using the HC-05 module at a baud rate of 38400 was conducted to ensure that the gymnastics assessment parameters were displayed consistently on the application and display panel. As illustrated in Figure 5a, the Delphi7 application displays the real-time data, while Figure 5b shows the P10 display panel synchronously displaying the gymnastics movement scores.



(a) (b)

Figure 5. Bluetooth Master-Slave Communication Testing (a) Application Display and (b) Display Panel

The testing of client-server communication utilised RSSI (Received Signal Strength Indicator) measurements as an indicator of wireless signal strength, with RSSI values close to 0 dBm reflecting stronger signal intensity. As illustrated in Table 1, the outcomes of signal stability measurements at distances ranging from 1 to 6.5 metres are presented. The range of -1 to -75 dBm is indicative of a stable connection, characterised by the absence of delay or packet loss.

Table 1. Wi-Fi Signal Strength Test Results for *Client-Server* Communication

No	Distance (cm)	gnal Strength (dBm)	Remarks
1	0	-1	ıooth connection
2	10	-17	ıooth connection
3	20	-26	ıooth connection
4	30	-32	ıooth connection
5	40	-49	ıooth connection
6	50	-56	ıooth connection
7	100	-59	ıooth connection
8	150	-61	ıooth connection
9	200	-63	ıooth connection
10	250	-69	ıooth connection
11	300	-72	ıooth connection
12	350	-75	ıooth connection
13	400	-77	ata transmission delay occurred
14	450	-	ata transmission delay occurred
15	500	-83	ata transmission delay occurred
16	550	-85	ata frequently disconnects and reconnects
17	600	-85	ata frequently disconnects and reconnects
18	650	ata not sent	ata completely disconnected

C. Data Storage Mechanism

The data storage mechanism has been designed to process and manage gymnastics movement sensor data in real time. The ESP32's internal storage mechanisms facilitate the retention of data from the transmitter module through a series of processes including parsing and temporary buffering in RAM. As illustrated in Figure 6, the ESP32's internal memory is allocated as follows: flash memory stores 952,512 bytes of program firmware, leaving 2,193,216 bytes of space, and 320KB of RAM, of which 167,532 bytes are used, including 47,532 bytes for global variables for system configuration and 120,000 bytes for sensor buffer arrays. The device under consideration provides storage options on 128KB external EEPROM. It is possible for users to elect to undertake non-volatile data logging when required. The EEPROM capacity has been

demonstrated to be capable of accommodating up to 120,000 bytes of sensor data.

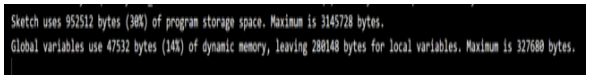
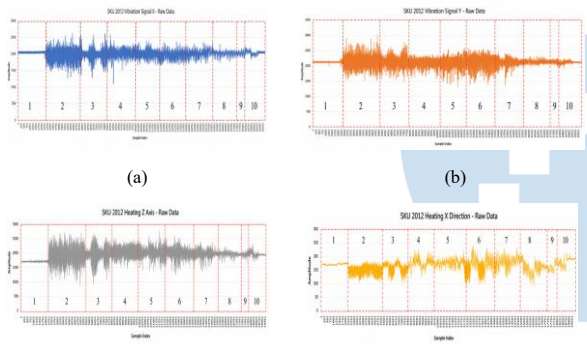


Figure 6. ESP32 Internal Storage

D. Analysis and Processing of Gymnastics Data Samples

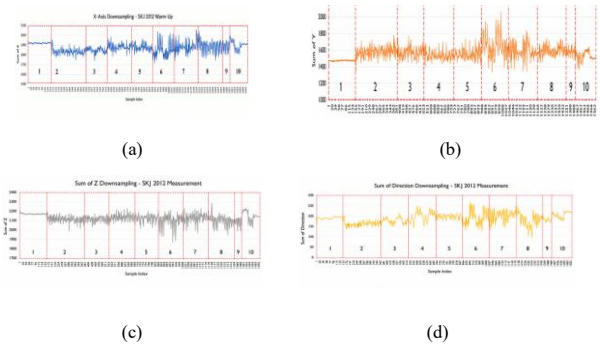
The acquisition and analysis of raw data samples were conducted to identify the initial response characteristics of the sensors during the execution of gymnastics movements. The data was obtained from four parameters: the X-axis, Y-axis, Z-axis, and orientation direction as illustrated on Figure 7 (a), (b), (c) and (d), respectively. The 2012 SKJ warm-up movements yielded 7,210 data samples per parameter from 10 movement phases, which presents the low-impact acceleration pattern and variations in the direction of gymnastics warm-ups.



(a) (b) (c) (d)

Figure 7. Raw Data from the 2012 SKJ Warm-up ADXL335 Sensor Output Signal (a) X-axis Acceleration, (b) Y-axis Acceleration, (c) Z-axis Acceleration, and (d) Direction Reading from the HMC5883L Sensor

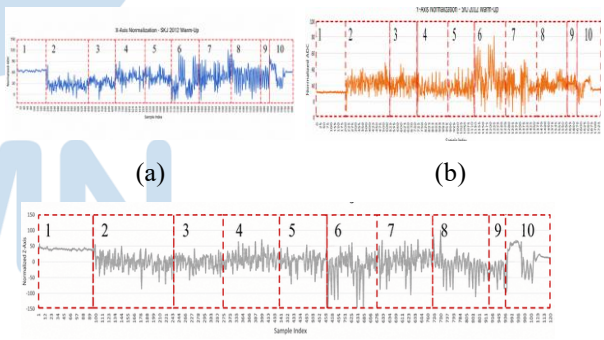
The downsampling and average pooling processes were carried out by dividing the data index series into fixed blocks of 10 samples, where each block was reduced to a single average value, thereby reducing the total data by 90%. The data was obtained from four parameters: the X-axis, Y-axis, Z-axis, and orientation direction as illustrated on Figure 8 (a), (b), (c) and (d), respectively. As demonstrated in Figure 8, the raw signal graph appears dense, with the details of the movement phases being difficult to distinguish due to the large number of samples in each parameter. Following the implementation of the downsampling and average pooling processes, the 2012 SKJ heating data was compressed into 721 samples, thereby yielding a more lucid wave representation, as illustrated in Figure 8.



(a) (b) (c) (d)

Figure 8. Downsampling Data with Average Pooling SKJ 2012 Heating ADXL335 Sensor Output Signal (a) X-axis Acceleration, (b) Y-axis Acceleration, (c) Z-axis Acceleration, and (d) Direction Reading from the HMC5883L Sensor

The data normalisation stage was carried out to linearly remap the downsampled data based on the minimum and maximum values per axis so that the amplitude of all parameters was projected to a range of 0-100. The direction parameter was not normalised to this range because the HMC5883L sensor reading values already directly represented the body's orientation. The graph in Figure 9 is still expressed in ADC output units with different ranges between axes, while Figure 9 displays the normalised data. The data was obtained from three parameters: the X-axis, Y-axis and Z-axis orientation direction as illustrated on Figure 9 (a), (b) and (c), respectively.



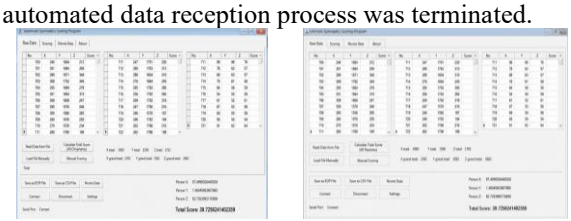
(a) (b) (c)

Figure 9. Normalized SKJ 2012 Data for Heating the ADXL335 Sensor Output Signal (a) X-axis acceleration, (b) Y-axis acceleration, and (c) Z-axis acceleration

E. Implementation and Evaluation of the Automatic Assessment System

The present research examined the implementation and evaluation of an automatic scoring system, with the objective of evaluating the system's performance in acquiring and interpreting body movements. Prior to the commencement of the testing phase, the participants were tasked with ensuring that both the transmitter and receiver modules were synchronised and connected in a correct manner. Thereafter, they proceeded to select the specific type of rhythmic gymnastics to be evaluated within the system.

Subsequent to the initiation of the session, the receiver module perpetually acquired data transmitted by the transmitter module until the gymnastics rhythm sequence reached its conclusion, at which juncture the automated data reception process was terminated.



(a) (b)

Figure 10. Testing of SKJ 2012 Warm-up Movements (a) Correct Movements and (b) Incorrect Movement

As shown in Figure 10, the system application incorporates a data processing flow involving a comparison with master data. The application displays the accuracy value per axis as X%, Y% and Z% with a weighting of 0.3 for each axis and 0.1 for the orientation direction. Figure 10a shows that the 2012 SKJ warm-up test, which incorporates movements that align with the master data, yielded a score of 85. Figure 10b, on the other hand, shows that diverging movements result in a significantly lower score about 39.72.

The research demonstrated that the automatic gymnastics assessment tool was capable of successfully acquiring data on movement dynamics and body orientation from the ADXL335 accelerometer and HMC5883L magnetometer sensors. The system has been demonstrated to possess the capacity to process large-scale sensor data with dense, fluctuating signals and difficult-to-read movement noise patterns into a more concise data representation, thereby reducing the computational load. Furthermore, the system is capable of distinguishing movement data patterns through the process of master data comparison, thereby producing quantitative scores that reflect the level of conformity. It has been demonstrated that significant differences between test data and master data trigger high errors and a decrease in movement validation accuracy.

IV. Conclusion

The research results indicate that the gymnastics movement assessment system installed in the back pocket of the participant's trousers, integrating the ESP32 microcontroller, the ADXL335 sensor, and the HMC5883L sensor, successfully acquired three-axis acceleration and body orientation data. The data was transmitted to the receiver module in real time for analysis by the application, thereby producing a numerical score for each gymnastics routine. The methodology employed to reduce data density by 90%, minimise noise and preserve movement trends involved implementing a downsampling technique. The data then consolidated into a single sample using an average pooling method, summarising 10 raw data samples into one. Data normalisation is integral to standardising the scale range of all parameters. Subsequent comparison of the results allows the accuracy and error rate of the test to be calculated against the master data. The outcome of this process is a performance score that evaluates the degree to which the test movement aligns with the reference movement.

References

[1] R. Roji and Y. Yulianti, "Physical Education and Health", vol. 8. Indonesia: Centre for Curriculum and Textbooks, Research and Development Agency, Ministry of Education and Culture, 2017.

[2] A. A. Ulfah, D. Dimyati, and A. J. A. Putra, "Analysis of the Application of Rhythmic Gymnastics in Improving the Gross Motor Skills of Early Childhood," J. Obs. J. Educ. Early Childhood, vol. 5(2), pp. 1844–1852, 2021, https://doi.org/10.31004/obsesi.v5i2.993.

[3] S. Kartini, I. Sari, and P. Nasution, "Efforts to Improve Student Interest and Achievement in Gymnastics through the Demonstration Method at Tanah Jawa Private High School," vol. 4(2), pp. 145–159, 2024. https://doi.org/10.47006/pendalas.v4i2.499s.

[4] V. Camomilla, E. Bergamini, S. Fantozzi, and G. Vannozzi, "Trends Supporting the In-Field Use of Wearable Inertial Sensors for Sport Performance Evaluation: A Systematic Review," Sensors, vol. 18, no. 3, p. 873, 2018, doi: 10.3390/s18030873.

[5] A. R. Hasibuan, A. Z. Hasibuan, and M. S. Asih, "Designing an Arduino-Based Automatic Scoring System for Taekwondo," J. Inf. Syst. Res., vol. 4(1), pp. 150–157, 2022, https://doi.org/10.47065/josh.v4i1.2278.

[6] Olympic.com, "Judging the judges – how statistical analysis evaluates fairness and accuracy in gymnastics scoring.", (19 October 2023) Available: https://www.olympics.com/en/news/how-statistical-analysis-evaluates-fairness-accuracy-gymnastics [Accessed 08 May 2025].

[7] P. Astuti, "Design and Development of a Gymnastics Activity Monitor Using ADXL335 Vibration Sensors and GY-273 Direction Sensors Based on an Android Application to Support Sports Learning Evaluation," Diponegoro University, 2021.

[8] O. D. Lara and M. A. Labrador, "A Survey on Human Activity Recognition using Wearable Sensors," IEEE Communications Surveys & Tutorials, vol. 15, no. 3, pp. 1192-1209, 2013, doi: 10.1109/SURV.2012.110112.00192.

[9] R. A. Pratama, D. Syauqy, and A. S. Budi, "Sit-up Movement Monitoring System based on Accelerometer and Gyroscope Sensor Data using the K-Nearest Neighbour Algorithm on NodeMCU ESP8266," J. Inf. Information Technology and Computer Science, vol. 7(4), pp. 1758–1766, 2023, [Online]. Available: http://j-ptiik.ub.ac.id.

[10] C. C. Yang and Y. L. Hsu, "A Review of Accelerometry-Based Wearable Motion Detectors for Physical Activity Monitoring," Sensors, vol. 10, no. 8, pp. 7772-7788, 2010, doi: 10.3390/s100807772.

[11] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat, "Physical Human Activity Recognition Using Wearable Sensors," Sensors, vol. 15, no. 12, pp. 31314-31338, 2015, doi: 10.3390/s151229858.

[12] I. Sayekti et al., "RANCANG BANGUN VIDEOTRON BERBASIS RASPBERRY Pi SEBAGAI MEDIA INFORMASI DI LABORATORIUM ELEKTRONIKA," vol. 19, no. 1, pp. 1–8, 2023.

[13] F. Demrozi, G. Pravadelli, A. Bihorac, and P. Rashidi, "Human Activity Recognition using Inertial, Physiological and Environmental Sensors: A Comprehensive Survey," IEEE Access, vol. 8, pp. 210816-210836, 2020, doi: 10.1109/ACCESS.2020.3037715.

[14] D. Program, E. Engineering, and F. Cycle, "Development and Evaluation of a Modular Multi-Sensor System for Comprehensive Water Quality Analysis," 2024.

[15] Suryono, Supriyati, Dadi, S. Kusumastuti, and Sasongko, "Rancang Bangun Sensor Gesture Sebagai Pengganti," Orbith, vol. 18, no. 1, p. 53, 2022.

[16] A. P. Launuru, G. Manu, H. K. Tupan, and R. Hutagalung, "Rancang Bangun Sistem Kontrol Nirkabel on – Off Peralatan Listrik Dengan Perintah Suara Menggunakan Smartphone Android," J. Simetrik, vol. 11, no. 1, pp. 388– 397, 2021.

[17] W. Lee and K. Seo, "Downsampling for Binary Classification with a Highly Imbalanced Dataset Using Active Learning," Big Data Res., vol. 28, p. 100314, 2022.

[18] D. A. Nasution, H. H. Khotimah, and N. Chamidah, "Perbandingan Normalisasi Data untuk Klasifikasi Wine Menggunakan Algoritma K-NN," Comput. Eng. Sci. Syst. J., vol. 4, no. 1, p. 78, 2019, doi: 10.24114/cess.v4i1.11458.

[19] A. M. Argina, "Penerapan Metode Klasifikasi K-Nearest Neigbor pada Dataset Penderita Penyakit Diabetes," Indones. J. Data Sci., vol.no. 2, pp. 29–33, 2020, doi: 10.33096/ijodas.v1i2.11.

ISSN 2355-3286

Enhancing Energy Visibility: A Real-Time Power Monitoring System via Web Scraping and NoSQL

Moeljono Widjaja[1], Rahmi Andarini[2]
[1] Department of Informatics, Universitas Multimedia Nusantara, Tangerang, Indonesia
[2] Department of Engineering Physics, Tangerang, Indonesia
moeljono.widjaja@umn.ac.id
rahmi.andarini@lecturer.umn.ac.id

Accepted March 11, 2026
Approved April 09, 2026

***Abstract*— Modern industrial power meters are often equipped with Ethernet ports, providing web-based interfaces for real-time monitoring. However, automating data extraction from these interfaces for long-term analysis can be challenging without costly proprietary software. This paper presents a systematic design for an energy logging system that utilizes a web scraping approach to systematically retrieve data from power meter web APIs. The system is implemented using a Python-based scraping script deployed on a Raspberry Pi and utilizes a MongoDB database for scalable data storage and subsequent load profile visualization. Experimental results from an installation at the Universitas Multimedia Nusantara campus demonstrate the system's ability to effectively monitor high-load Mechanical Ventilation and Air Conditioning (MVAC) systems, providing critical insights into peak consumption periods and operational efficiency. This low-cost, automated solution facilitates data-driven energy management and supports institutional energy-saving initiatives.**

***Index Terms*— Energy Monitoring, Load Profiles, TCP/IP Protocol, MongoDB, Web Scraping**

I. Introduction

The global energy crisis has become a crucial issue with significant economic and political consequences. Driven by rapid population growth and industrial expansion, energy demand continues to outpace fossil fuel reserves. In Indonesia, this issue is intensified by a heavy reliance on subsidized fuels, which often encourages inefficient consumption. Consequently, prioritizing energy efficiency—particularly in the commercial and construction sectors, where savings potential is estimated at 20% to 35% [1]—is critical.

Universitas Multimedia Nusantara (UMN) has actively committed to campus-wide efficiency, focusing specifically on its Mechanical, Ventilation, and Air Conditioning (MVAC) system, the university's significant energy user [2], [3]. Currently, Buildings C and D share a unified MVAC infrastructure, which prevents facility managers from isolating individual building performance. While separate meters exist for lighting, cooling consumption is recorded only via two shared chiller meters. This lack of granular data makes it difficult to identify waste or quantify the impact of specific conservation efforts [4].

The technical complexity of UMN's water-based chiller system further complicates monitoring. Water enters the chillers at 12°C and is cooled to 5–7°C before distribution to Fan Coil Units (FCUs) that serve clusters of three to four classrooms. While air at the FCU coil begins at 15°C and is initially distributed at 19°C, the integration of Fresh Air (FA) piping results in a final ambient classroom temperature of 25°C. This cooling process often runs at full capacity regardless of need; UMN's room occupancy averages only 53% to 64.3%, meaning chillers frequently cool unoccupied spaces. As noted by Zhang [5], the primary barrier to effective campus energy management is the combination of outdated monitoring technology and insufficient data granularity, which diminishes the sense of accountability among academic units [6]–[8].

The Internet of Things (IoT) offers a viable solution to these institutional challenges by providing a network of devices capable of remote monitoring and control. It has seen rapid adoption across multiple domains—notably in smart cities [9], [10], homes [11], and security [12]. IoT technology is increasingly attractive for smart campus management due to its ease of implementation [13], [14], though the field still lacks standardized application references for widespread methodological use [15].

To bridge this visibility gap, this paper proposes an automated energy logging system designed for real-time MVAC monitoring at UMN. The system utilizes

Python-based web scraping [16], [17] to extract data from industrial power meters, effectively bypassing the limitations of proprietary interfaces. By centralizing this data in a scalable MongoDB database, the framework establishes a technical foundation for data-driven management. The entire pipeline—from acquisition and processing to visualization—is integrated within a cohesive Python environment to ensure long-term scalability and efficiency.

II. METHODOLOGY

The research method involves several stages, namely: setting up the network configuration on the power meter and scraping data from the website

A. Setup Network Configuration and Web-Page Access

As explained in [18], the PowerLogic PM5560 series is equipped with an Ethernet port with the following default configuration:

- IP method = Stored
- IP address = 169.254.0.10
- Subnet mask = 255.255.0.0
- Gateway = 0.0.0.0
- HTTP server = Enabled
- DPWS = Enabled
- EtherNet/IP = Enabled
- DNP3 = Disabled
- Device name = PM55-#xxxxxxxxxx

The display of default network configuration is shown in **Error! Reference source not found.**.

For meters equipped with a display, basic Ethernet settings can be configured using the display. For meters without a display, it requires the following steps to configure basic Ethernet settings before connecting the meter to the local network.

1) Disconnect the computer from the network. If the computer has wireless communication, be sure to disable the wireless network connection.

NOTE: After you disconnect your computer from the network, its IP address should automatically update to a default IP address of 169.254.###.### (where ### equals a number from 0 to 255) and a subnet mask of 255.255.0.0.

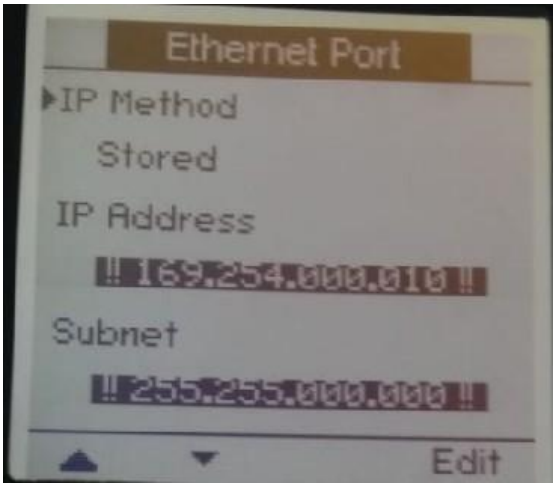


Fig. 1. Default Ethernet port configuration on the PowerLogic PM5560 series

TABLE I. DEFAULT LOGIN CREDENTIALS

Username	Password
user1	pass1
user2	pass2

2) Use an Ethernet cable to connect the computer to one of the meter's Ethernet ports.

3) Open a web browser and enter 169.254.0.10 in the address bar.

4) Log in to the meter webpage. The default login credentials can be found in Default Ethernet port configuration on the PowerLogic PM5560 series

5) .

The MAC address of the PowerLogic PM5560 installed on Chiller 2 is MAC: 00-80-67-AA-C4-45, as shown in **Error! Reference source not found.**.

An attempt was made to access the PowerLogic PM5560 installed in Chiller 2 in the Utility Building on the UMN campus via the web. Logging in to the web server was done using the default username and password. The logging process was successful; however, the webpage only displayed static data. Fig. 3 shows the web-based configuration interface for a Schneider Electric PowerLogic PM5560, which is a high-end power and energy meter used in industrial and commercial electrical systems. The page appears to be missing its CSS (styling). Instead of a professional dashboard with buttons and a sidebar, it is displaying as a plain list of blue hyperlinks on a white background. This often happens if:

- The connection is slow or unstable.
- The browser blocked the loading of scripts/styles for security reasons (common with "Not Secure" HTTP connections).
- The meter's internal web server is under high load

Fig. 2. MAC address on PowerLogic PM5560 installed on Chiller 2

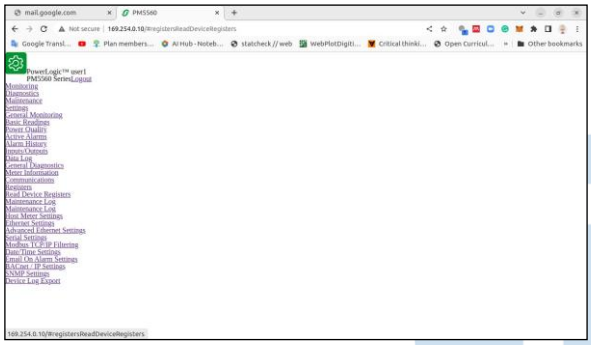


Fig. 3. Webpage on PowerLogic PM5560 Chiller 2

However, after restarting the Power Meter unit, the web server can display a complete dashboard as shown in **Error! Reference source not found.**. Now, it shows the fully rendered web interface for a Schneider Electric PowerLogic PM5560 Series power meter. Unlike the previous image, the styling (CSS) is loading correctly, providing a graphical dashboard for real-time electrical monitoring.

Here is an overview of the dashboard as shown in **Error! Reference source not found.**.

- Device Identity: The meter is a PM5560 Series, accessed via IP 192.168.11.254.
- User: Logged in as teknisi1.
- Active View: The user is in the Monitoring tab under General Monitoring - Basic Readings.

The dashboard is split into two primary visual areas:

1) Gauges (Visual Representation): This section provides a quick look at the Load Current for the three phases:

a) Phase A (Ia): 280.39 A.
b) Phase B (Ib): 322.11 A.
c) Phase C (Ic): 267.51 A.
d) Users can toggle the radio buttons to switch these gauges to display Power, Voltage LL (Line-to-Line), or Voltage LN (Line-to-Neutral).

2) Basic Readings Table (Detailed Data): This table provides a comprehensive snapshot of electrical performance, including Minimum, Present, and Maximum recorded values:

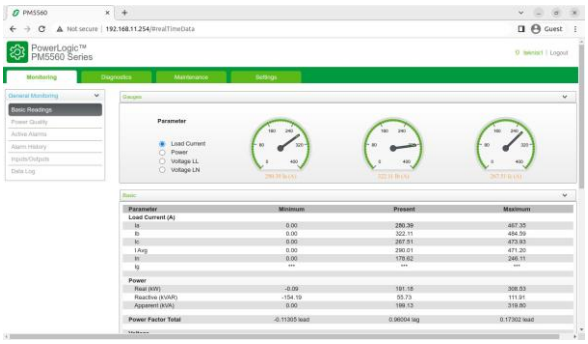


Fig. 4. Webpage on PowerLogic PM5560 Chiller 2 after restarting the Power Meter unit

a) Load Current (A): Shows the current for each phase, the average (I_{Avg}), and the neutral current (I_n).
b) Power (kW, kVAR, kVA):
- Real Power: Currently 191.18 kW.
- Reactive Power: Currently 55.73 kVAR.
- Apparent Power: Currently 199.13 kVA

c) Power Factor Total: Currently 0.96004 lag. A "lagging" power factor usually indicates an inductive load, such as motors or transformers

B. Web-Scraping Data PowerLogic PM5560

After successfully accessing the website from the PowerLogic PM5560, the next step is to automatically read the required parameters by performing web-scraping.

To perform web scraping from the PowerLogic PM5560 website, do the following:

1) Open the PM5560 website on your local network.
2) Log in to the website using username = user1 and password = pass1.
3) Inspect the webpage by right-clicking and selecting "Inspect".
4) Click the "Network" tab, then refresh by pressing the "Clear" button.
5) Select an element on the webpage to inspect by pressing "ctrl-shift-C."
6) Copy the section of the element to be extracted as "Copy as cURL" (see **Error! Reference source not found.**).
7) Paste the "cURL command" into the website https: //curlconverter.com/python/
8) Copy the converted file to Python by clicking "Copy to clipboard."

By combining these steps, a technician or engineer can "sniff" the communication between the browser

and the PM5560 meter. The resulting cURL command can then be pasted into a terminal or a script to programmatically retrieve the meter's current, power, and voltage readings for external logging or custom dashboards.

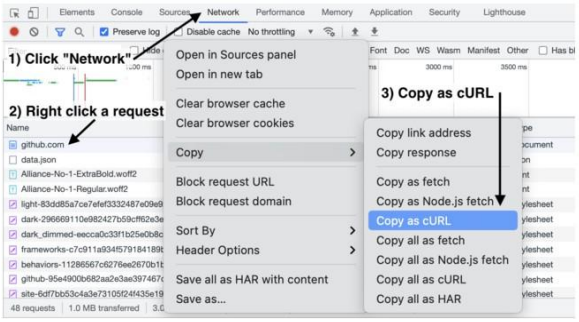


Fig. 5. Illustration of the web-scraping process

Based on the web inspection, identify key features to fetch useful data from the website. An example of a Python script obtained from the web-scraping can be seen in Listing 1.

This Python script is a practical application of the "Copy as cURL" technique shown in **Error! Reference source not found.**. It automates the process of requesting real-time data directly from the Schneider Electric PM5560 power meter.

Listing 1. Python script for web-scraping

```
  import requests
  from bs4 import BeautifulSoup

  headers = {
      'Accept': 'text/plain, /; q=0.01',
      'Accept-Language':'en-
US,en;q=0.9,id;q=0.8',
      'Authorization':'Basic
***************',
      'Connection': 'keep-alive',
      'Content-Type':'application/x-www-form-
urlencoded; charset=UTF-8',
      'Origin': 'http://192.168.11.254',
      'Referer': 'http://192.168.11.254/',
      'User-Agent':    'Mozilla/5.0    (Linux;
Android 6.0; Nexus 5 Build/MRA58N)
      AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/107.0.0.0 Mobile
      Safari/537.36',
      'X-Requested-With': 'XMLHttpRequest',
  }

  data=
'PL_*^H3000[86]PL,PL*^H3110[2]PL,PL*^H3200[48
]PL,PL*^H3706[94]PL,
  PL*^H3802[54]PL,PL*^H27218[96]PL,PL*^H27616
[2]PL,PL*^H27694[96]PL,
  PL*^H28092[2]_PL'

  response=
requests.post('http://192.168.11.254/UE/Post_
PL_Data',
  headers=headers, data=data, verify=False)
```

The detailed explanation of Listing 1 is given in the following:

1) Script Components

- Target Device: The script sends a request to 192.168.11.254, which matches the IP address of the PM5560 meter seen in the web interface.
- Authentication: The 'Authorization': 'Basic **********' header handles the login credentials for the user (e.g., teknisi1) so the script can access the data without a manual login.
- Request Method: It uses requests.post to hit the /UE/Post_PL_Data endpoint, which is the internal "mailbox" the meter uses to serve data to the web dashboard.

2) The Data Payload (Modbus Registers) The data string contains the core instructions for what information to retrieve:

- Register Groups: Codes like H3000, H3110, and H3200 correspond to specific Modbus registers within the meter.
- Requested Parameters: These registers contain the values you see on the dashboard, such as Load Current, Power (kW), and Voltage.
- Syntax: The PL_*^...PL format is a proprietary Schneider Electric wrapper used to batch multiple register requests into a single network call for efficiency.

3) Purpose and Outcome

- Bypassing the GUI: Instead of looking at the visual gauges and tables, this script pulls the raw numbers directly into Python.
- Data Integration: Once the response is received, a developer can use BeautifulSoup to parse the text and save it to a database, an Excel file, or a custom monitoring system.
- Scalability: This allows for continuous logging of parameters like the 0.96004 lag Power Factor or 322.11 A Load Current at high frequencies (e.g., every second).

In order to streamline the process of logging, the parsing process is modularized as a function as described in Listing 2. To parse the response from the Schneider Electric PM5560, you can process the raw text returned by the meter into a structured Python dictionary. Since the meter returns data in a proprietary string format (delimited by PL_ and _PL markers), we can use a helper function to clean and map these values to the electrical parameters seen on the dashboard.

This method is very effective due to the following reasons:

- Automation: This script allows logging the 322.11 A current or 191.18 kW power readings every few seconds without manually refreshing a browser.

- Data Integrity: By targeting the specific registers (like H3000), for pulling the exact numerical value stored in the meter's memory.
- Scalability: easy expansion of the payload string to include more registers, such as Voltage LL or Harmonics, by adding their respective Modbus addresses.

Listing 2. Parsing raw data

```
def requestData():
    data = dict()
    data_raw=
'PL__*^H3000[86]__PL,PL__*^H3110[2]__PL,PL__*
^H3200[48]__PL,PL__*^H3706[94]__PL,PL__*^H380
2[54]__PL,PL__*^H27218[96]__PL,PL__*^H27616[2
]__PL,PL__*^H27694[96]__PL,PL__*^H28092[2]__P
L'
    response=
requests.post('http://192.168.24.20/UE/Post__
PL__Data',  headers=headers,  data=data_raw,
verify=False)
    data_text = response.text
    data_lst = data_text.split(',')
    data_int = list(map(int,data_lst))
    data['Ia'] = toFL32(data_int[0:2])
    data['Ib'] = toFL32(data_int[2:4])
    data['Ic'] = toFL32(data_int[4:6])
    data['Vab'] = toFL32(data_int[20:22])
    data['Vbc'] = toFL32(data_int[22:24])
    data['Vca'] = toFL32(data_int[24:26])
    data['P'] = toFL32(data_int[60:62])
    data['pf'] = toFL32(data_int[84:86])
    data['freq'] = toFL32(data_int[86:88])
    A = data_int[92]
    B = data_int[93]
    C = data_int[94]
    D = data_int[95]
    data['E']=
(A*281474976710656+B*4294967296+C*65536+D)/10
00
    return data
```

In the payload string 'PL_*^H3000 [86]PL. . . ' , the number in brackets (e.g., [86]) tells the meter to read 86 consecutive registers starting from address 3000.

To extract a specific value, the position must be specified in the response string such as:

- H3000 will contain I_a.
- H3002 will contain I_b.
- H3004 will contain I_c.

This Python script is a complete Internet of Things (IoT) Data Logger. It bridges the gap between the physical hardware (the PM5560 Power Meter) and a cloud database (MongoDB).

Listing 3. Setting connection to MongoDB database

```
WIB = tz.gettz('WIB')
MONGODB_SRV                                    =
'mongodb+srv://username:***********@energy.gs
1cq.mongodb.net/test?retryWrites=true&w=major
ity'
MONGODB_DB = 'SmartEnergy'
MONGODB_COLLECTION = 'PowerMeter'

loggerID = 'CHILLER_2'
```

The data obtained through web scraping is register data of type Integer. The current, voltage, power, power factor, and frequency data are stored in two registers that must be converted to Float32 format. The conversion process from two Integer numbers to Float32 is performed using the function shown in Listing 4.

Listing 4. Converting to Float32

```
def toFL32(val):
    a = val[0]
    b = val[1]
    c=struct.unpack('>f',
bytes.fromhex(f"{a:0>4x}" + f"{b:0>4x}"))[0]
    return c
```

Power meters store data in 16-bit registers, but values like "Amps" or "kW" are usually 32-bit floating-point numbers. Therefore, they must be converted using a conversion function (toFL32) as given in Listing 4. Here is the explanation of the conversion function:

- The Problem: The meter sends two separate 16 -bit integers for one reading.
- The Solution: The toFL32 function takes those two integers (val[0] and val[1]), converts them to Hex, joins them together, and uses struct. unpack to turn them back into a human-readable decimal (Float 32).

3) Database Integration (logData) This function acts as the "courier":
   - Internet Check: It pings Google/MongoDB to ensure the connection is live.
   - MongoDB Upload: It connects to a cluster named SmartEnergy and inserts a new document into the PowerMeter collection.
   - Metadata: Each entry is tagged with loggerID: 'CHILLER_2', allowing you to distinguish this meter's data from others in the same database.

The complete source code is shown in Listing 5.

Listing 5. Logging Data to MongoDB Database

```
def logData():
    url = "https://cloud.mongodb.com/"
    timeout = 5
    my_dict = {'loggerID' : loggerID}
    my_dict.update(readTimeStamp())
    my_dict.update(requestData())
    print(my_dict)
    try:
        request      =      requests.get(url,
timeout=timeout)
        print("Connected to the Internet")

        try:
            client = False
            client                             =
pymongo.MongoClient(MONGODB_SRV)
            print("MongoDB:   connection   is
opened")
            db=client[MONGODB_DB]
            collection                         =
db[MONGODB_COLLECTION]
            try:
```

```
                result                          =
collection.insert_one(my_dict)
                print ("MongoDB: successfully
insert data.")
            except
pymongo.errors.NetworkTimeout as error:
                print("MongoDB    error:    ",
error)
            except
pymongo.errors.ServerSelectionTimeoutError as
error:
                print("MongoDB    error:    ",
error)

        except
pymongo.errors.ConfigurationError as error:
            print("MongoDB error: ", error)
        finally:
            if client:
                client.close()
                print("MongoDB: connection is
closed")
    except          (requests.ConnectionError,
requests.Timeout) as exception:
        print("No internet connection.")
```

The sent energy data consists of four registers and must be combined into one value with a calculation that can be seen in Listing 6. The energy *E* is stored across four 16-bit registers: *A, B, C, D*. The script reconstructs the 64-bit value and scales it to kilowatt-hours (kWh).

The mathematical expression used in the code is:

$$E = \frac{(A \cdot 2^{48}) + (B \cdot 2^{32}) + (C \cdot 2^{16}) + D}{1000}$$

Note: the code uses 281,474,976,710,656 which is 2^{48} and 4,294,967,296 which is 2^{32}.

Listing 6. Converting four registers into one value

```
A = data_int[92]
B = data_int[93]
C = data_int[94]
D = data_int[95]
data['E']                                   =
(A*281474976710656+B*4294967296+C*65536+D)/10
00
```

Instead of running once and stopping, the script uses the *timeloop* library in order to regularly logging data from the power meter. Listing 7 implements the periodic task. The @tl.job decorator is set to run the logData() function every 5 minutes.

Listing 7. Executing periodic task

```
from timeloop import Timeloop

tl = Timeloop()

@tl.job(interval=timedelta(minutes=5))
def sample_job_every_5m():
    logData()
    print  ("Finish  5m  job  current  time  :
{}".format(time.ctime()))
```

After successfully web-scraping data from the PowerLogic PM5560 and converting the register data into data that matches the measured parameters, the next step is to periodically send the data to the MongoDB Atlas database via the Internet.

III. Results and Discussion

This section will explain two folds: the results of study the electrical system and logging the power meter via web scraping, and the discussion on the visualization of the logged data.

A. The Electrical System Study on the UMN Campus

This section will explain the electrical data for Buildings C and D on the Multimedia Nusantara University campus. The UMN campus has four main buildings, as shown in **Error! Reference source not found.**.

Fig. 6. Main building on yhe UMN Campus
Source: https://360.umn.ac.id/

The distribution of the kWh meter panels can be seen in **Error! Reference source not found.**. The electricity panel Chiller 1 kWh meter is used to measure all electricity consumption in the Utility Building and Chiller 1. Electricity consumption in the Utility Building consists of Chiller plant I, including pumps, fans, and cooling towers, as well as all lights in the Utility Building. The kWh Chiller 1 also reads the electricity usage of the Building C transfer pump, which is the pump that functions to move water from the ground tank to the roof tank. The Building C transfer pump usually runs 2 to 3 times a day. Chiller 1 is only used 1 to 2 times a week, namely on weekends or in the afternoon when the cooling load is not too large.

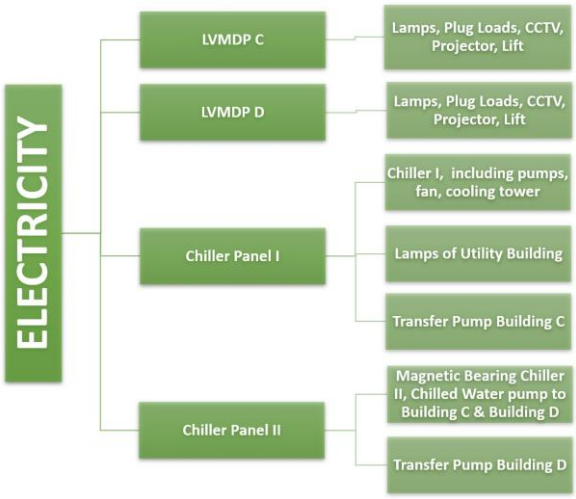


Fig. 7. Division of kWh meter panels in Buildings C and D

The electricity Panel Chiller 2 kWh meter reads the electrical energy consumption of Chiller Plant 2, including the pumps that distribute cold water to Building C and Building D. Chiller 2 in question is a magnetic bearing type with a high level of efficiency used routinely in Buildings C and D during working days and working hours. In addition, the electricity consumption of the transfer pump in Building D is also recorded on the electricity kWh meter in Panel Chiller 2.

Given the combined use of the Chiller between Buildings C and D, the development of this monitoring system is expected to determine the amount of energy consumed by Building C, particularly for the use of the Chiller.

B. Data Visualisaion

Installation and testing of the Modbus logger have been successfully carried out for the PowerLogic PM800 Power Meter, while readings from the PowerLogic PM5560 were done by web-scraping on the website available on the unit.

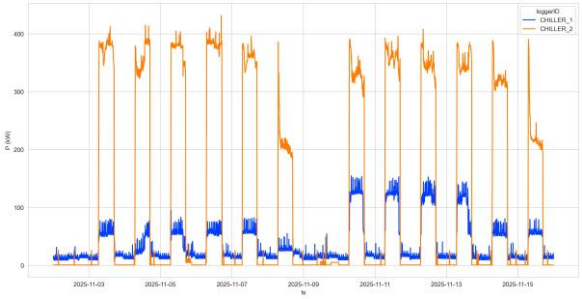


Fig. 8. Profiles of power consumptions from Chiller 1 and 2 including LVMDP building C and D

Fig. 8 shows a plot of the Power Meter data reading results for Chiller units 1 and 2 and LVMDP buildings C and D. During peak load (morning to afternoon) the power consumption of Chiller 2 is highest and the lowest is Chiller 1, while the power consumption in Buildings C and D is relatively the same. Based on information from the building management, Chiller 2 is newer and more efficient so that the operation of Chiller 2 is prioritized over Chiller 1.

1) *Daily Electrical Energy Load Profile Analysis:* The electrical energy load profile of the chiller for a single weekday can be seen in Fig. 9. Peak load occurs between 7 a.m. and 5 p.m., with the largest energy load on Chiller 2. This incident is in accordance with the MVAC system schedule at the university that Chiller 2 is the main Chiller operated during weekdays. There is a sudden increase in load at the start of the day, reaching 350 kW, as the chillers are turned on at approximately 6 a.m. This schedule is implemented because it requires some time to cool the classrooms before the first-class schedule starts at 8 a.m. Most classes end at 4 p.m. There is an opportunity to shape the load between 7 a.m. and 4 p.m. by adjusting the chiller set point to match the load of classes conducted during the day. In addition, since it is known that the average Chiller's power approximately at 310 kW, then reducing its operating hours by 2 hours per day will reduce energy consumption by 620 kWh, or about 18 percent.

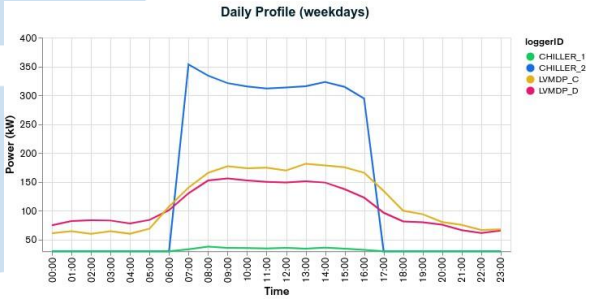


Fig. 9. Daily profiles (Monday-Friday) of load consumption from Chiller 1 and 2 including LVMDP building C and D

Fig. 10. Weekly profile of power consumption Chiller 1 and 2 and LVMDP of buildings C and D

2) *Weekly Electrical Energy Load Profile Analysis:* The electrical energy load profile for one week can be seen in Fig. 10. Peak load occurs during weekdays (Monday - Friday). The largest energy load is on Chiller 2. Most activities on the campus are conducted during weekdays, while there are a few classes also conducted on Saturday. The load reflects the activities on campus from Monday to Sunday. There is also an opportunity to optimize the load by

adjusting the chiller proportional to the number of classes held on specific days.

IV. Conclusion

This paper has presented the systematic design and successful implementation of an automated energy logging system that utilizes a web scraping approach to monitor industrial power meters. By leveraging the built-in web API of the Schneider Electric PowerLogic PM5560, the system provides a systematic method to bypass the limitations of manual data recording and expensive proprietary software, enabling continuous, high-frequency data acquisition. The integration of Python-based scraping with a MongoDB Atlas cloud database offers a scalable, low-cost, and robust IoT solution for institutional energy management.

The deployment at the Universitas Multimedia Nusantara campus provided critical granular visibility into the load profiles of the university's MVAC systems. Data analysis confirmed that peak electrical loads consistently occur on weekdays between 7:00 a.m. and 5:00 p.m.. Furthermore, the findings validated the campus operational strategy of prioritizing the more efficient magnetic-bearing Chiller 2 over Chiller 1. Crucially, the system identified significant opportunities for optimization, such as a potential 18% reduction in energy consumption through the adjustment of chiller operating hours. This level of transparency is a vital prerequisite for achieving the 20–35% energy savings potential identified in the commercial and construction sectors.

Future work will expand the system's scope by integrating additional meters across the campus and developing a real-time web dashboard for live visualization of energy performance indicators. Additionally, the accumulated historical data will serve as the foundation for developing predictive models for cooling load demand and automated anomaly detection, further enhancing the efficiency of building operations.

ACKNOWLEDGMENT

This research was supported by Universitas Multimedia Nusantara under contract no 058/PI/LPPM-UMN/III/2022.

References

[1] Menteri ESDM RI, "Handbook of Energy & Economy Statistics of Indonesia 2020," Book, pp. 1–111,2021. [Online]. Available:https://www.esdm.go.id/en/publication/ handbook-of-energy-economic-statistics-of-indonesia-heesi

[2] M. Zhou, A. M. Abdulghani, M. A. Imran, and Q. H. Abbasi, "Internet of things (iot) enabled smart indoor air quality monitoring system," ACM International Conference Proceeding Series, pp. 89–93, 2020. doi: 10.1145/3398329.3398342.

[3] R. Andarini, M. Widjaja, and A. M. Sutopo, "Penilaian kualitas udara dalam ruangan di kampus universitas multimedia nusantara," Technical Report, 2021.

[4] R. Andarini, M. Salehuddin, and C. O. Harahap, "Pengembangan Sistem Manajemen Energi di Kampus Universitas Multimedia Nusantara," Universitas Multimedia Nusantara, Tangerang, Tech. Rep., 12 2019.

[5] J.-P. Zhang, J. Zhou, Q.-M. Luo, and G.-B. Fan, "Research on Campus Energy Consumption Monitoring System Based on IPV6," DEStech Transactions on Social Science, Education and Human Science, no. icaem, 3 2018. doi: 10.12783/DTSSEHS/ICAEM2017/19145.

[6] K. A. Barfi, "Internet of things applications for smart environments," Intelligent Systems Reference Library, vol. 121, pp. 93–103, 2022. doi: 10.1007/978-3-030-97516-6 5.

[7] J. Saini, M. Dutta, and G. Marques, "Sensors for indoor air quality monitoring and assessment through Internet of Things: a systematic review," Environmental Monitoring and Assessment, vol. 193, no. 2, 2 2021. doi: 10.1007/S10661-020-08781-6.

[8] M. Widjaja, D. K. Halim, and R. Andarini, "The development of an iot-based indoor air monitoring system towards smart energy efficient classroom," Ultima Computing : Jurnal Sistem Komputer, vol. 14, pp. 28–35, 2022. doi: 10.31937/sk.v14i1.2565.

[9] J. Shah and B. Mishra, "IoT enabled environmental monitoring system for smart cities," 2016 International Conference on Internet of Things and Applications, IOTA 2016, pp. 383–388, 2016. doi: 10.1109/IOTA.2016.7562757.

[10] B. Hammi, R. Khatoun, S. Zeadally, A. Fayad, and L. Khoukhi, "IoT technologies for smart cities," IET Networks, vol. 7, no. 1, pp. 1–13, 1 2018. doi: 10.1049/IET-NET.2017.0163.

[11] L. Smirek, G. Zimmermann, and M. Beigl, "Just a Smart Home or Your Smart Home - A Framework for Personalized User Interfaces Based on Eclipse Smart Home and Universal Remote Console," Procedia Computer Science, vol. 58, pp. 107–116, 2016. doi: 10.1016/J.PROCS.2016.09.018.

[12] S. S. Vedaei, A. Fotovvat, M. R. Mohebbian, G. M. Rahman, K. A. Wahid, P. Babyn, H. R. Marateb, M. Mansourian, and R. Sami, "COVIDSAFE: An IoT-based system for automated health monitoring and surveillance in post-pandemic life," IEEE Access, vol. 8, pp. 188538– 188551, 2020. doi: 10.1109/ACCESS.2020.3030194.

[13] A. Zhamanov, Z. Sakhiyeva, R. Suliyev, and Z. Kaldykulova, "IoT smart campus review and implementation of IoT applications into education process of university," 2017 13th International Conference on Electronics, Computer and Computation (ICECCO), Abuja, Nigeria, 2017, pp. 1–4. doi: 10.1109/ICECCO.2017.8333334.

[14] G. Lilis, G. Conus, N. Asadi, and M. Kayal, "Towards the next generation of intelligent building: An assessment study of current automation and future IoT based systems with a proposal for transitional design," Sustainable Cities and Society, vol. 28, pp. 473–481, 1 2017. doi:10.1016/J.SCS.2016.08.019.

[15] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787–2805, 10 2010. doi: 10.1016/J.COMNET.2010.05.010.

[16] C. Bhatt, A. Bisht, R. Chauhan, A. Vishvakarma, M. Kumar, and S. Sharma, "Web Scraping Techniques and Its Applications: A Review," Proceedings - 2023 3rd International Conference on Innovative Sustainable Computational Technologies, CISCT 2023, 2023. doi: 10.1109/CISCT57197.2023.10351298.

[17] M. A. Wahed, M. S. Alzboon, M. Alqaraleh, J. Ayman, M. AlBatah, and A. F. Bader, "Automating Web Data Collection: Challenges, Solutions, and Python-Based Strategies for Effective Web Scraping," 2024 7th International Conference on Internet Applications, Protocols, and Services, NETAPPS 2024, 2024. doi: 10.1109/NETAPPS63333.2024.10823528.

[18] E. Schneider, "PowerLogic PM5500 series user manual," 7 2020.[Online].Available:https://www.se.com/ww/en/download/document/ HRB1684301/

ISSN 2355-3286

Design and Development of a Water Filling System Using An Arduino Mega 2560 Based on Flowmeter Sensor

[1*]Ahmad Ridlo Hanifudin Tahier, [2]Wahyu Riandhika, [3]Dhani Nur Indra Syamputra
[1] Automation Engineering, Diponegoro University, Semarang, Central Java, Indonesia, 50275
[2,3] Physics Department, Diponegoro University, Semarang, Central Java, Indonesia, 50275
[1]ridlohanifudin@lecturer.undip.ac.id , [2]wahyuriandhika86@gmail.com , [3]dsyamputra@fisika.fsm.undip.ac.id

Accepted March 02, 2026
Approved June 28, 2026

***Abstract*— The utilisation of automation technology is undergoing rapid proliferation in the industrial sector, including in the bottled drinking water industry. The objective of this study is to design and implement an automation system for filling bottled drinking water with a range of 30–330 millilitres using an Arduino Mega 2560 as the controller. The system has been designed to optimise the use of the YF-S201 pulse-based flowmeter sensor as a setpoint. The setpoint is determined by calibrating the pulse value obtained based on the time delay in filling volumes in multiples of 30 at values ranging from 30 to 330 millilitres. The accuracy value for volumes of 30 and 60 millilitres was found to be 99.3%, while volumes of 60-330 millilitres demonstrated an accuracy level of >99.6%. Subsequently, the system was integrated with a conveyor and E18-D80NK proximity sensor to detect bottles during the filling process, and a counter to enumerate the bottles post-water filling. The process was initiated upon the pressing of the interface button and the subsequent detection of the bottle by the proximity sensor. Thereafter, the system activated the filling pump until the pulse attained the setpoint. The status of the operating system was displayed on the LCD and could be controlled via push buttons. Preliminary testing indicated that the system demonstrated a capacity for precise bottle filling, with an error margin of less than 1%. This finding suggests that the system exhibits a low level of inaccuracy and responsiveness.**

***Index Terms*— Automatic filling system; Arduino Mega 2560; YF-S201 sensor; Proximity sensor; Conveyor.**

I. INTRODUCTION

Technological advancements have precipitated a marked improvement in the bottled water industry, particularly with regard to the automation of water filling. The implementation of an automatic water filling system has been demonstrated to enhance production efficiency, reduce labour requirements and minimise human error. It is evident that small industries and home industries frequently utilise conventional methods that are less efficient. A prime example of this is AMDK filling. The rise in the number of small industries in the AMDK sector is proportional to the rise in the AMDK population. Consequently, many industries require a filling system that is both affordable and accurate, and which is also simple to implement in small and medium-sized industries [1].

The advent of automated mechanical systems for the purpose of filling bottles with liquid (liquid filling) represents a significant outcome of technological advancements within the industrial sector. The system has been designed to engender efficiency, which is expected to have a positive impact, especially on industries that produce liquid products. However, bottle-filling systems in small or home-based industries, such as those producing water, syrup, sauce, milk, and honey, are still inefficient because they generally still use manual methods [2].

It has been determined that the human body is composed of approximately 75% water, and that no individual can survive for a period exceeding four to five days without the consumption of this vital fluid [3]. AMDK is a fundamental component in the contemporary era, demonstrating persistent market expansion. As demonstrated by data provided by the Indonesian Bottled Water Association (ASPADIN), there is an evident increase in the consumption of AMDK, with a growth rate of 8-10% on an annual basis. This necessitates an efficient production system, particularly during the filling stage. It is imperative that such high demand is balanced with an efficient production system, particularly during the filling stage, which is a bottleneck in many small industries. Manual systems, which are still widely utilised, are not only slow but also prone to quality variations due to human error. [4].

The utilisation of time delay as a reference in extant research demonstrates that the system's accuracy remains relatively high at 6%. It is evident that, in light of the observed discrepancy, the present study undertook sensor calibration with the objective of implementing a setpoint that utilised pulses in the filling

process. The research design involved the creation of a 330-ml mineral water bottle filling system, utilising the Arduino Mega 2560 as the central control unit. The system incorporates YF-S201 water flow sensors, proximity sensors, and DC gearbox motors to ensure volume and time efficiency [5].

Another automated system was developed using a Twido Programmable Logic Controller (PLC) from Schneider Electric [6]. The system was designed to automatically fill and seal bottles, and uses infrared sensors to detect the presence of bottles on the filling line. The process of filling is facilitated by a direct current (DC) motor, while the movement of the bottles is achieved through the utilisation of a conveyor motor [16] [17]. Subsequent to the completion of the filling process, the bottle caps are automatically affixed by means of a solenoid. As demonstrated by the test results, the system has been shown to be capable of filling 250-ml bottles in approximately 14 seconds while maintaining stability. As demonstrated in previous research, there has been an emergence of low-cost automatic filling equipment intended for utilisation by Micro, Small and Medium-sized Enterprises (MSMEs). The system has been demonstrated to have the capacity to fill 500-ml bottles of Aloe Vera syrup by means of a time-step method [7].

In this research, a switching mode power supply (SMPS) was utilised to provide voltage to the system. In an SMPS, the AC input is subjected to direct rectification, after which it is filtered to yield the desired DC voltage. The DC voltage produced is then switched on and off at a high frequency by an electronic switching circuit, thereby generating AC current. In the event of effective isolation of the high-frequency transformer, the transformer's output will be electrically isolated by relay [8].

The present study proposes a design for an automatic filling system for bottles that utilises an Arduino Mega 2560 [9] [10] [11]. The system incorporates a YF-S201 water flow sensor to ensure volume precision and an IR proximity sensor to detect the position of the bottle [12] [13]. The selection of the Arduino Mega 2560 was made on the basis of an assessment of its superior input/output capacity and enhanced computational capabilities in comparison with the earlier model, the Arduino Uno. LCDs are utilised extensively due to their multifaceted functionality and straightforward programming. In order to establish a connection between an LCD and a microcontroller, it is necessary to connect the LCD's PORT to the microcontroller's corresponding PORT. It is imperative to acknowledge that this specific port on the microcontroller is exclusively dedicated to LCD functions. It is not possible to utilise this port for other functions, such as I/O functions [15].

II. METHODS

The primary instruments utilised in this research comprise a YF-S201 sensor for pulse input, an Arduino Mega 2560 as the primary controller, infrared proximity detection for bottle recognition, and a direct current (DC) water pump as the actuator. The data processing procedure entails the collection of water measurements ranging from 30 to 330 millilitres, which displayed on a 16x2 Liquid Crystal Display (LCD). The testing procedure is initiated by actuating the system by pressing the designated start or reset button, displayed the number of bottles and the volume LCD. The data processing was conducted through the collection of water volume measurements (ml) ranging from 30 ml to 330 ml.

Upon actuation of the start button, the system will display a start message on the LCD and activate the blue indicator light, indicating that the system is in a steady state and make the conveyor transport the bottle to the filling point. When the proximity detect the bottle, the conveyor will automatically halt and initiate the water pump. The volume of water flowing then displayed on the LCD using a predetermined calibration factor (pulse/mL). Upon attaining the designated setpoint, the water pump will cease operation, and the system will initiate the conveyor process for the subsequent bottle. The system will continue to run until the user initiates a cessation of this process by pressing the designated stop button. Upon actuation of the stop button, the system will initiate a sequence of events that will result in the deactivation of the system.

A. System Design

To facilitate understanding of how this automatic bottle filling machine works, please refer to the flowchart provided. The design process and water filling system at AMDK are explained in flowchart figure 1.

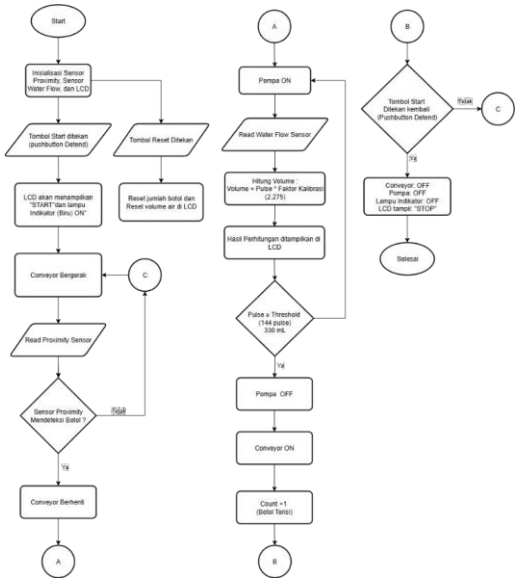


Fig. 1. System Flowchart

As illustrated in Figure 1, the process commences with the activation of the system via the designated start push button. In its initial state, the Arduino will initialise the components that are utilised. Upon completion of the system's readiness, the user is permitted to initiate the process by pressing the designated start button. Alternatively, the reset button can be utilised to restore the number of bottles and the water volume displayed on the liquid crystal display (LCD) to their default settings. Upon actuation of the start button, the LCD will display the start status and the blue indicator light will be activated. Subsequently, the conveyor belt will be programmed to transport the bottle to the proximity sensor.

B. Hardware Design

A block diagram was representated a visual view to elucidate concepts. The purpose of this diagram is to facilitate comprehension of the relationship between inputs and outputs in a system.

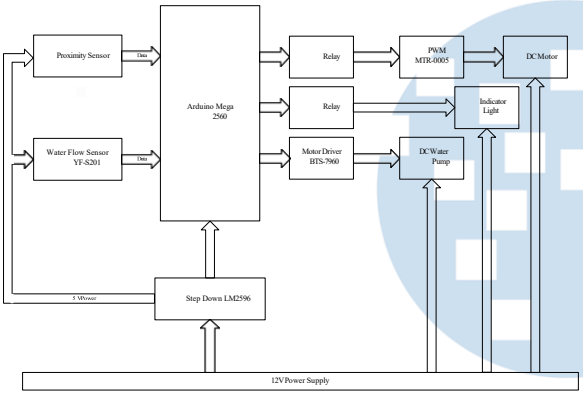


Fig. 2. Hardware Block Diagram

As illustrated in Figure 2, the components of the Arduino Mega 2560-based automatic bottle filling system function collectively within an open-loop system. The proximity sensor fulfils the function of initially detecting the bottle's presence on the conveyor, thus activating the water pump to initiate the filling process from the tank into the bottle. The filling volume is regulated by the YF-S201 water flow sensor based on the calculation of water flow pulses. The Arduino Mega 2560 processed the data from the proximity and water flow sensors and transmitting control signals to relays to activate actuator (conveyor). The primary function of the water pump is to facilitate the transportation of water from the storage tank to the bottles. The step-down module, on the other hand, plays a crucial role in ensuring that the voltage supplied to the system components meets the requisite specifications. In this open-loop system, the process as a whole is dependent on pre-set programming logic, with no provision for a feedback mechanism to dynamically adjust filling based on actual conditions. The proximity sensor is the initial trigger for the process, while the water flow sensor is used to estimate the volume of liquid being added, based on pulse calculations.

(a) (b)

Fig. 3. 3D concept image (a) front view, (b) top view

As demonstrated in Figures 3(a) and (b), the system comprises a control box panel that incorporates electrical components. Within the mechanical system under consideration, the movement of the tool to the filling position is facilitated by a conveyor driven by a 12 V DC motor. The speed of the conveyor can be adjusted using the MTR-0005 driver.

The power system utilises a 12-volt power supply as its primary energy source, with an LM2596 step-down module incorporated to facilitate the distribution of power to the sensor, Arduino, and associated electronic components. The user interface employed is a 16x2 LCD integrated with a push button and indicator light.

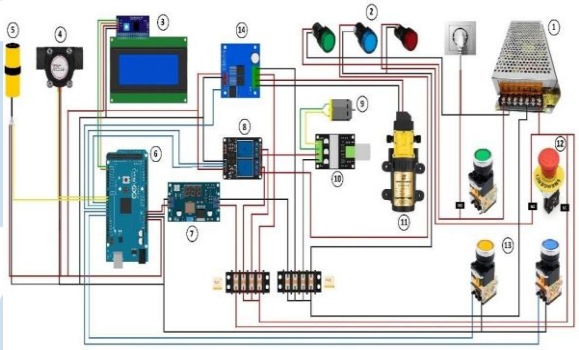


Fig. 4. Hardware Circuit Schematic

Figure 4 presents the electrical circuit of the system. The initiation of the filling process is facilitated by a proximity sensor connected to pin 3 of the Arduino digital board, which detects the presence of a bottle and subsequently activates the DC pump, that connect to BTS7960 driver on pin 6 of the Arduino. The volume of water is measured using a YF-S201 sensor on digital pin 2. The start push button is integrated with an MTR-0005 driver based on the PWM output from digital pin 5. There are three indicator lights are connected via a relay that is controlled by the Arduino digital pin. Furthermore, the availability of facilities on pins, such as I2C communication, PWM pins, and non-PWM digital pins, is taken into consideration. The circuit has been designed as an open-loop system that relies on initial sensor calibration.

C. Open Loop System Design

The system diagram for this device utilises an open-loop approach, incorporating an Arduino Mega 2560, a conveyor belt, an infrared proximity sensor, a DC pump, and a YF-S201 water flow sensor.

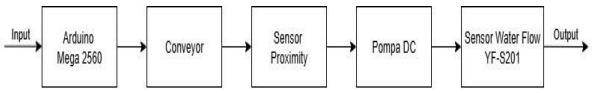


Fig. 5. Open Loop System Diagram

As demonstrated in Figure 5, the system diagram was an open-loop approach to regulate the filling process. In this system, the proximity sensor functions as the initial trigger for the process, detecting the presence of the bottle at the filling position. The water flow sensor is used to calculate the volume based on number of water pulses reading.

D. Experimental Calibration Setup

The experimental setup was designed to evaluate the performance and calibration characteristics of the flowmeter sensor. This procedure includes key components such as: pulse generation in the YF-S201 flowmeter sensor based on the volume measured from the discharge, average pulse data collection, error measurement, and determination of the calibration factor (mL/pulse).

1) Principles of measurement

The measurements are derived from the hall effect sensor mechanism. The measured volume value is derived by multiplying the discharge by the elapsed time.

$$Q = V.t \quad \text{....(2.1)}$$

Where Q = Debit (mL/s)
V = Volume (mL)
T = time (s)

The value of V is proportional to N.

$$V \propto N$$

Therefore, the linear equation can be written as

$$V = K\,.N \quad \text{...(2.2)}$$

Where K is the calibration factor (mL/pulse) and N is the number of pulses counted (pulse)

2) Callibration procedure

Calibration was performed by calculating the measured pulse value for each volume to obtain the relationship between volume and pulse. Subsequently, the mean value of the test results was calculated on five occasions. The K value was then obtained using the following equation (2.2) : The following alterations were made:

$$K = \frac{\sum_i^N V_i}{\sum_i^N N_i} \quad \text{...(2.3)}$$

$$K_{avg} = \frac{\bar{V}}{N} \quad \text{... (2.4)}$$

3) Error Analysis

Error testing was conducted using standard error and standard deviation measurements with the equation:

$$\%Error = \frac{(V_{reference} - V_{measured})}{V_{reference}} \times 100\% \text{ ... (2.5)}$$

For standar deviation measurement

$$SD = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2} \quad (2.6)$$

For 5 measurements the standard deviation is given by:

$$SD = \sqrt{\frac{1}{4}((x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_4 - \bar{x})^2 + (x_5 - \bar{x})^2}$$

where:
s = sample standard deviation
n= number of observations
x_i= individual measurement
$\bar{x}$= sample mean

III. Results and Discussion

A series of tests and analyses were conducted to ensure that each component functioned in accordance with the specified specifications. The objective of this test was to evaluate the overall performance of the device and to ascertain the pulse value that was utilised as the setpoint in the water filling process.

A. Measurement and Testing

The device was subjected to a rigorous testing process, encompassing the evaluation of the performance of each electronic component employed. This included the power supply, the E18-D80NK proximity sensor, the keypad, and the DC motor.

1) Power Supply Testing

The power supply sensor was tested by comparing the output and input voltage readings between the power supply specifications and the multimeter. The testing procedure was repeated on three occasions.

TABLE I. Power Supply Testing

Test	Input Voltage (VAC)	Output Voltage (VDC)	V Specification (VDC)
1.	225	12	12
2.	225	12	12
3.	225	12	12

As shown in Table 1, the experimental findings indicate that the power supply has the capacity to deliver a consistent voltage of 12 V DC. Voltage does not fluctuate to a significant degree that could be concluded that this power supply is suitable for supporting the entire system.

2) Proximity Sensor E18-D80NK Testing

The E18-D80NK proximity sensor is utilised for the purpose of detecting the presence of bottles at the filling position. Testing was conducted by moving an object closer to and further away from the sensor at distances ranging from 3 cm to 7 cm. The sensor produces a logic

signal of LOW when an object is detected and HIGH when no object is present. The most accurate and stable detection was observed to occur at a proximity of approximately 5 centimetres as shown on Table 2.

TABLE II. PROXIMITY SENSOR E18-D80NK TESTING

Test	Distance (cm)	Proximity indicator	Conditions
1.	3	On	Object Detected
2.	4	On	Object Detected
3.	5	On	Object Detected
4.	6	On	Object Detected
5.	7	Off	No Object

3) DC Motor Testing

A series of experiments was conducted with the objective of ascertaining the optimal operating voltage capable of producing stable motor performance for conveyor. The testing process entailed an evaluation of the sensor's impact on the final position accuracy of the bottle with various voltage input.

TABLE III. TESTING OF CONVEYOR WITH 12V INPUT VOLTAGE

Input Voltage (V)	Proximity Object Detection	Nozzle Position
12	ON	Incorrect
12	ON	Incorrect
12	ON	Incorrect
12	ON	Incorrect
12	ON	Incorrect

Tests are conducted to ascertain the optimal working voltage capable of engendering stable motor performance. These tests also include an evaluation of the sensors and their effect on the accuracy of the final position of the bottles under various input voltage variations.

TABLE IV. TESTING OF CONVEYOR WITH 8V INPUT VOLTAGE

Input Voltage (V)	Proximity Object Detection	Nozzle Position
8	ON	Correct
8	ON	Correct
8	ON	Correct
8	ON	Correct
8	ON	Correct

The test was conducted on five occasions to ensure that the conveyor functioned at the correct speed. This was necessary to ensure that when the proximity sensor detected a bottle, the water would be filled correctly into the bottle. The significance of this test lies in its role in minimising errors that could potentially result in production losses during the filling process.

TABLE V. TESTING OF CONVEYOR WITH 12V INPUT VOLTAGE

Input Voltage (V)	Proximity Object Detection	Nozzle Position
12	ON	Incorrect
12	ON	Incorrect
12	ON	Correct
12	ON	Incorrect
12	ON	Correct

As demonstrated in the Table 3, Table 4 and Table 5, the motor was tested with a voltage supply ranging from 6 volts to 12 volts. Test results demonstrate that the motor is capable of propelling the conveyor at a voltage of 6 volts; however, the movement of the bottles is characterized by an absence of smoothness and an inherent instability. It has been demonstrated that the system functions most stably at a voltage of 8 volts, at which point the bottles are able to move smoothly and stop precisely at the expected filling position. It has been demonstrated that the application of a voltage in excess of 12 volts has the effect of accelerating the conveyor. This has been demonstrated to impede the process of bottle control, resulting in inaccurate bottle positioning during filling. This has been identified as a factor that is liable to compromise the accuracy of the filling process. Consequently, the operating voltage selected for the system implementation is 8 volts because the motor is able to consistently move the bottles to the filling position.

B. Testing of the water filling system and equipment

The objective of the system testing process, which involved the utilization of the waterflow sensor YF-S201, was to validate the accuracy of volume measurement at various set point volumes. The experiment was conducted by comparing the volume of water measured by the system with the actual volume and subsequently calculating the percentage error. The range of volumes that were examined was from 30 mL to 330 mL

1) Callibration Testing

During the calibration test of the YF-S201 water flow sensor, each measurement point was repeated five times for each specified set point delay. Each repetition was carried out using the same procedure. From the five repetitions, the system recorded the number of pulses read and the volume of water successfully discharged. Subsequently, the mean and standard deviation of the measured volume were calculated in order to ascertain the stability of the measurement results.

Furthermore, the mean pulse from each delay set point is employed as the setpoint pulse value, thereby

representing a specific target volume. Consequently, the target volume is assigned the most suitable setpoint pulse value, calculated based on the mean outcomes of five replicates. Moreover, the comparison between the target volume and the average measured volume is employed to calculate the error and accuracy percentages. This enables the determination of the level of system precision for each volume variation.

TABLE VI. DELAY SETPOINT MEASUREMENT

Setpoint (s)	Volume (ml)	St dev	Error (%)	Accuracy (%)
0.41	30.20	0.20	0.67	99.33
0.85	59.60	0.24	0.67	99.33
1.30	90.20	0.20	0.22	99.78
1.75	120.40	0.24	0.33	99.67
2.19	149.80	0.20	0.13	99.87
2.62	180.40	0.40	0.22	99.78
3.07	209.80	0.20	0.10	99.90
3.53	240.40	0.24	0.17	99.83
4.00	270.40	0.24	0.15	99.85
4.25	298.80	0.73	0.40	99.60
4.80	329.40	0.40	0.18	99.82

2) Testing Using Pulse Setpoint

After conducting volume testing and obtaining the pulse setpoint, each setpoint was tested to measure its accuracy. The test was conducted five times to detect valid accuracy. At the pulse calibration setpoint, the pulse value was rounded because the system uses integer values to provide pulse input to the flowmeter. The rounded values are as follows: 30 mL = 11 pulses, 60 mL = 24 pulses, 90 mL = 38 pulses, 120 mL = 52 pulses, 150 mL = 66 pulses, 180 mL = 79 pulses, 210 mL = 93 pulses, 240 mL = 108 pulses, 270 mL = 121 pulses, 300 mL = 130 pulses, and 330 mL = 144 pulses. The test values are shown in Table 7.

TABLE VII. MEASUREMENT USING PULSE SETPOINT

Setpoint (Pulse)	Volume (ml)	Volume Measurement (ml)	Stdev	ERROR	Accuracy
11	30	30.40	0.55	1.33%	98.67%
24	60	59.40	0.55	1.00%	99.00%
38	90	89.40	0.55	0.67%	99.33%
52	120	119.40	0.89	0.50%	99.50%
66	150	150.00	1.41	0.00%	100.00%
79	180	178.80	1.30	0.67%	99.33%
93	210	210.80	2.17	0.38%	99.62%
108	240	238.40	2.07	0.67%	99.33%
121	270	269.60	1.14	0.15%	99.85%
130	300	298.80	2.59	0.40%	99.60%
144	330	329.40	0.89	0.18%	99.82%

Following the testing of the pulse set point, a range of values was obtained for each filling volume. The accuracy of the results obtained was generally less than 1%. The lowest values of accuracy were observed at volumes of 30 and 60 mL. This is due to the YF-S201 flowmeter sensor exhibiting superior measurement accuracy when the pulse rate exceeds 35 beats per minute. When a volume of 330 millilitres of water was added, an accuracy value of 99.82% was obtained, indicating that the set point of 144 has a satisfactory level of accuracy. This value is significant given that, in the context of the overall system testing, bottles with a capacity of 330 millilitres were utilised.

3) System Validation

After conducting the pulse test, a comprehensive system test was carried out using 330 mL AMDK bottles. The 330 mL volume was chosen for the system test because the majority of AMDK bottles have a volume of 330 mL. In addition, the 330 mL bottle design has an ideal height, making it easier to control on the conveyor. As demonstrated in Table 6, the system functions at an optimal level, exhibiting a minimal error margin of 0.18%, as indicated by the accuracy of 99.82% recorded in Table 7. Furthermore, the information displayed on the user interface is found to be comprehensive.

TABLE VIII. OVERALL SYSTEM TESTING

No	Condition	Sensor Input	System Action (Actuator)	Stop/Move Criteria	Output/Indicator
1	System turned on (standby)	—	Conveyor OFF/ON according to initial mode, pump OFF	Waiting for START button / auto mode to activate	LCD: "STOP/READY", standby indicator light
2	START button pressed / auto mode active	—	Conveyor ON to bring the bottle to the filling position	Bottle reaches proximity sensor area	LCD: "START", process indicator active
3	Bottle enters detection area	Proximity detect bottle	Conveyor OFF (stops precisely at the nozzle), pump still OFF	Bottle is in filling position (stable)	LCD: status " ready"
4	Filling process begins	Flow sensor starts counting pulses	Pump ON, conveyor OFF	Pulse begins to increase toward setpoint	LCD displays running volume
5	Filling in progress	Pulse continues measured	Pump ON	If pulse ≥ setpoint	LCD display volume approaching the target
6	Setpoint reached	Flowmeter reached setpoint	Pump OFF	After the pump stops	LCD: "filling done"
7	Bottle removed from filling station	Proximity sensor starts "not detected"	Conveyor ON	Bottle move exits proximity area	Bottle counter increases, LCD updates bottle count
8	Cycle repeats	Proximity detects the next bottle	Return to step 3	STOP button pressed / system stopped	System returns to standby

In comparison with previous research, the present system focuses on low-cost implementation and calibrated pulse-based volume control. The comparison is summarised in Table 9.

TABLE IX. TESTING OF CONVEYOR WITH 12V INPUT VOLTAGE

Previous research	Comparison and contribution of this study
Microcontroller-based automatic water filling [5]	Previous work showed that microcontroller filling can be implemented, but the present work adds calibrated YF-S201 pulse setpoints for 30-330 mL volume control.
PLC-based automatic filling and capping [6]	PLC implementation provides stable industrial operation, while this study offers a lower-cost Arduino Mega 2560 alternative with conveyor, proximity detection, pump control, LCD interface, and bottle counting.
Low-budget Aloe Vera syrup filling [7]	The previous system used a time-step method for filling, while this research uses pulse counting from the flowmeter to reduce dependence on pump running time only.
IoT drinking-water usage monitoring [14]	The cited IoT system focuses on monitoring water usage in dispensers. In contrast, this study controls the filling process and evaluates volume accuracy directly.
This research	The system achieved 98.67%-100.00% accuracy for 30-330 mL, with 99.82% accuracy at the 330 mL AMDK target volume and the highest accuracy of 100.00% at 150 mL.

IV. CONCLUSION

A series of tests were conducted on a volume range of 30 mL to 330 mL in order to ascertain the functionality of the water filling system that uses the YF-S201 water flow sensor. The conclusion of these tests is that the system operates correctly and with sufficient accuracy. The consistent volume and pulse measurement results indicate the stability of the sensor and the effectiveness of the system. The calibration factor calculated for each volume also tends to be stable, with an average value of 2.352 mL/pulse. It has been demonstrated that the system is capable of measuring and distributing water volume with adequate precision within the range of 30-330 mL. In tests involving larger volumes (210–330 mL), the system demonstrated stability with a low error rate. The mean error was 0.38% at a volume of 210 mL, and 0.15% at a volume of 270 mL. For the commonly used AMDK volumes of 240 and 330 mL, the errors were found to be 0.67% and 0.184%, respectively. The system has been demonstrated to demonstrate capacity for filling a range of volumes, with a minimum of 30 mL and a maximum of 330 mL. This capacity has been determined through the conduction of a series of tests.

REFERENCES

[1] A. F. Seno Prasetyo, "Rancang Bangun Automatic Liquid Filling Machine Berbasis Internet of Thing Meggunakan NodeMCU dan Telegram," Jurnal Ilmiah Pendidikan Teknik Elektro, pp. 3-4, 2023.

[2] P. Machudor Yusman, "Prototipe Sistem Otomasi Pada Pengisian Depot Air Minum Isi Ulang Berbasis Arduino Uno," Jurnal Teknologi dan Informatika (JEDA), vol. vol.2, no. 2745 - 8911, pp. 74 - 84, 2021.

[3] Ilham Sakti Wibowo, "Mesin Pengisian Botol Minuman Bir Pletok Secara Otomatis Berbasis Internet of Things (IoT)," Prosiding Seminar Nasional Teknik Elektro , vol. vol. 4, pp. 376 - 381, 2019.

[4] T. S. Bambang Riyanta, "Pembuatan Mesin Pengisian Botol Sirup Aloe Vera Low Budget di UKM SMK Muhammadiyah 2 Turi Yogyakarta," Jurnal Ilmiah Multi Disiplin Indonesia, vol. 1, no. 2809 - 1620, pp. 885 - 889, 2022.

[5] S. Sulbiyah, K. Sari, dan H. Wijayanti, "Rancang Bangun Alat Pengisian Air Otomatis Berbasis Mikrokontroler," Jurnal Ilmiah, vol. 5, no. 2, 2016.

[6] R. Faulianur dan Z. Fachr, "Prototype Pengisi Air dan Penutup Botol Otomatis Berbasis Programmable Logic Controller," Jurnal Teknik Elektro, vol. 9, no. 1, 2021.

[7] B. Riyanta, A. Budiarto, dan F. Nugroho, "Pembuatan Mesin Pengisian Botol Sirup Aloe Vera Low Budget di UKM SMK Muhammadiyah 2 Turi Yogyakarta," Seminar Nasional Teknik Elektro, 2017.

[8] D. Lestari, "Trainer Troubleshooting Switching Mode Power Supply (Smps) Sebagai Media Pembelajaran Pada Mata Kuliah Teknik Audio," 2019.

[9] I. Oktariawan, "Pembuatan Sistem Otomasi Dispenser Menggunakan Arduino Mega 2560," Jurnal Ilmu Teknologi Mesin, vol. 1, p. 2, 2013.

[10] D. L, Arduino dan Sensor Pada Project Arduino DIY, Semarang: Yayasan Prima, 2018.

[11] D. N. Asyifa, "Simulasi Penentuan Lokasi Gangguan Satu Fasa Pada Jaringan Tegangan Menengah 20 Kv Berbasis Arduino Mega 2560 Dilengkapi Dengan Monitoring Melalui Vtscada," Gema Teknologi , vol. 20 No.2, p. 54, 2019.

[12] A. M. Jannah, Rancang Bangun Mesin Pencetak Bakso Dengan Variasi Ukuran Berbasis Mikrokontroler Atmega328p, Semarang: Universitas Diponegoro, 2023.

[13] K. A, Arduino& Sensor, Yogyakarta: Andi, 2018.

[14] Singgeta R. L, "Implementasi Sistem Monitoring Penggunaan Air Minum Pada Multiple Dispenser Berbasis IoT," Rang Teknik Journal, vol. 4 No. 1, pp. 127 - 133, 2021.

[15] M. A. Mazidi, The Microcontroller and Embedded. System: Using Assembly and C. Pearson Education, New Jersey: Pearson Education, 2011.

[16] K. S. Kiangala and Z. Wang, "An Industry 4.0 approach to develop auto parameter configuration of a bottling process in a small to medium scale industry using PLC and SCADA," in Procedia Manufacturing, vol. 35, pp. 725–730, 2019, doi: 10.1016/j.promfg.2019.06.015.

[17] Hartanto, "Tegangan Motor DC Terhadap Berat Barang Pada Ban Berjalan," Jurnal Elektro, vol. 10 No.2, pp. 174 - 181, 2022.

AUTHOR GUIDELINES

1. Manuscript criteria

- The article has never been published or in the submission process on other publications.
- Submitted articles could be original research articles or technical notes.
- The similarity score from plagiarism checker software such as Turnitin is 20% maximum.
- For December 2021 publication onwards, Ultima Computing : Jurnal Sistem Komputer will be receiving and publishing manuscripts written in English only.

2. Manuscript format

- Article been type in Microsoft Word version 2007 or later.
- Article been typed with 1 line spacing on an A4 paper size (21 cm x 29,7 cm), top-left margin are 3 cm and bottom-right margin are 2 cm, and Times New Roman's font type.
- Article should be prepared according to the following author guidelines in this template. Article contain of minimum 3500 words.
- References contain of minimum 15 references (primary references) from reputable journals/conferences

3. Organization of submitted article

The organization of the submitted article consists of Title, Abstract, Index Terms, Introduction, Method, Result and Discussion, Conclusion, Appendix (if any), Acknowledgment (if any), and References.

- Title
  The maximum words count on the title is 12 words (including the subtitle if available)
- Abstract
  Abstract consists of 150-250 words. The abstract should contain logical argumentation of the research taken, problem-solving methodology, research results, and a brief conclusion.
- Index terms
  A list in alphabetical order in between 4 to 6 words or short phrases separated by a semicolon (;), excluding words used in the title and chosen carefully to reflect the precise content of the paper.
- Introduction
  Introduction commonly contains the background, purpose of the research, problem identification, research methodology, and state of the art conducted by the authors which describe implicitly.
- Method
  Include sufficient details for the work to be repeated. Where specific equipment and materials are named, the manufacturer's details (name, city and country) should be given so that readers can trace specifications by contacting the manufacturer. Where commercially available software has been used, details of the supplier should be given in brackets or the reference given in full in the reference list.
- Results and Discussion
  State the results of experimental or modeling work, drawing attention to important details in tables and figures, and discuss them intensively by comparing and/or citing other references.
- Conclusion
  Explicitly describes the research's results been taken. Future works or suggestion could be explained after it
- Appendix and acknowledgment, if available, could be placed after Conclusion.
- All citations in the article should be written on References consecutively based on its' appearance order in the article using Mendeley (recommendation). The typing format will be in the same format as the IEEE journals and transaction format.

4. Reviewing of Manuscripts

Every submitted paper is independently and blindly reviewed by at least two peer-reviewers. The decision for publication, amendment, or rejection is based upon their reports/recommendations. If two or more reviewers consider a manuscript unsuitable for publication in this journal, a statement explaining the basis for the decision will be sent to the authors within six months of the submission date.

5. Revision of Manuscripts

Manuscripts sent back to the authors for revision should be returned to the editor without delay (maximum of two weeks). Revised manuscripts can be sent to the editorial office through the same online system. Revised manuscripts returned later than one month will be considered as new submissions.

6. Editing References

- **Periodicals**
  J.K. Author, “Name of paper,” Abbrev. Title of Periodical, vol. x, no. x, pp. xxx-xxx, Sept. 2013.

- **Book**
  J.K. Author, “Title of chapter in the book,” in Title of His Published Book, xth ed. City of Publisher, Country or Nation: Abbrev. Of Publisher, year, ch. x, sec. x, pp xxx-xxx.

- **Report**
  J.K. Author, “Title of report,” Abbrev. Name of Co., City of Co., Abbrev. State, Rep. xxx, year.

- **Handbook**
  Name of Manual/ Handbook, x ed., Abbrev. Name of Co., City of Co., Abbrev. State, year, pp. xxx-xxx.

- **Published Conference Proceedings**
  J.K. Author, “Title of paper,” in Unabbreviated Name of Conf., City of Conf., Abbrev. State (if given), year, pp. xxx-xxx.

- **Papers Presented at Conferences**
  J.K. Author, “Title of paper,” presented at the Unabbrev. Name of Conf., City of Conf., Abbrev. State, year.

- **Patents**
  J.K. Author, “Title of patent,” US. Patent xxxxxxx, Abbrev. 01 January 2014.

- **Theses and Dissertations**
  J.K. Author, “Title of thesis,” M.Sc. thesis, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year. J.K. Author, “Title of dissertation,” Ph.D. dissertation, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.

- **Unpublished**
  J.K. Author, “Title of paper,” unpublished. J.K. Author, “Title of paper,” Abbrev. Title of Journal, in press.

- **On-line Sources**
  J.K. Author. (year, month day). Title (edition) [Type of medium]. Available: http:// www.(URL) J.K. Author. (year, month). Title. Journal [Type of medium]. volume(issue), pp. if given. Available: http://www.(URL) Note: type of medium could be online media, CD-ROM, USB, etc.

7. Editorial Adress

Jl. Scientia Boulevard, Gading Serpong
Tangerang, Banten, 15811
Email: **ultimacomputing@umn.ac.id**

Paper Title

Subtitle (if needed)

Author 1 Name[1], Author 2 Name[2], Author 3 Name[2]
[1] Line 1 (of affiliation): dept. name of organization, organization name, City, Country
Line 2: e-mail address if desired
[2] Line 1 (of affiliation): dept. name of organization, organization name, City, Country
Line 2: e-mail address if desired

Accepted on mmmmm dd, yyyy
Approved on mmmmm dd, yyyy

***Abstract*—This electronic document is a "live" template which you can use on preparing your Ultima Computing paper. Use this document as a template if you are using Microsoft Word 2007 or later. Otherwise, use this document as an instruction set. Do not use symbol, special characters, or Math in Paper Title and Abstract. Do not cite references in the abstract.**

***Index Terms*—enter key words or phrases in alphabetical order, separated by semicolon (;)**

I. INTRODUCTION

This template, modified in MS Word 2007 and saved as a Word 97-2003 document, provides authors with most of the formatting specifications needed for preparing electronic versions of their papers. Margins, column widths, line spacing, and type styles are built-in here. The authors must make sure that their paper has fulfilled all the formatting stated here.

Introduction commonly contains the background, purpose of the research, problem identification, and research methodology conducted by the authors which been describe implicitly. Except for Introduction and Conclusion, other chapter's title must be explicitly represent the content of the chapter.

II. EASE OF USE

A. Selecting a Template

First, confirm that you have the correct template for your paper size. This template is for Ultima Computing. It has been tailored for output on the A4 paper size.

B. Maintaining the Integrity of the Specifications

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them.

III. PREPARE YOUR PAPER BEFORE STYLING

Before you begin to format your paper, first write and save the content as a separate text file. Keep your text and graphic files separate until after the text has been formatted and styled. Do not add any kind of pagination anywhere in the paper. Please take note of the following items when proofreading spelling and grammar.

A. Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Abbreviations that incorporate periods should not have spaces: write "C.N.R.S.," not "C. N. R. S." Do not use abbreviations in the title or heads unless they are unavoidable.

B. Units

- Use either SI (MKS) or CGS as primary units (SI units are encouraged).
- Do not mix complete spellings and abbreviations of units: "Wb/m2" or "webers per square meter," not "webers/m2." Spell units when they appear in text: "...a few henries," not "...a few H."
- Use a zero before decimal points: "0.25," not ".25."

C. Equations

The equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.

Number the equations consecutively. Equation numbers, within parentheses, are to position flush right, as in (1), using a right tab stop.

$$\int_0^{r_2} F(r,\phi)\, dr\, d\phi = [\sigma r_2 / (2\mu_0)] \qquad (1)$$

Note that the equation is centered using a center tab stop. Be sure that the symbols in your equation have been defined before or immediately following the

equation. Use "(1)," not "Eq. (1)" or "equation (1)," except at the beginning of a sentence: "Equation (1) is ..."

D. Some Common Mistakes

- The word "data" is plural, not singular.
- The subscript for the permeability of vacuum μ_0, and other common scientific constants, is zero with subscript formatting, not a lowercase letter "o."
- In American English, commas, semi-/colons, periods, question and exclamation marks are located within quotation marks only when a complete thought or name is cited, such as a title or full quotation. When quotation marks are used, instead of a bold or italic typeface, to highlight a word or phrase, punctuation should appear outside of the quotation marks. A parenthetical phrase or statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.)
- A graph within a graph is an "inset," not an "insert." The word alternatively is preferred to the word "alternately" (unless you really mean something that alternates).
- Do not use the word "essentially" to mean "approximately" or "effectively."
- In your paper title, if the words "that uses" can accurately replace the word using, capitalize the "u"; if not, keep using lower-cased.
- Be aware of the different meanings of the homophones "affect" and "effect," "complement" and "compliment," "discreet" and "discrete," "principal" and "principle."
- Do not confuse "imply" and "infer."
- The prefix "non" is not a word; it should be joined to the word it modifies, usually without a hyphen.
- There is no period after the "et" in the Latin abbreviation "et al."
- The abbreviation "i.e." means "that is," and the abbreviation "e.g." means "for example."

IV. Using the Template

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention as below

ULTIMATICS_firstAuthorName_paperTitle.

In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper. Please take note on the following items.

A. Authors and Affiliations

The template is designed so that author affiliations are not repeated each time for multiple authors of the same affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization).

B. Identify the Headings

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include ACKNOWLEDGMENTS and REFERENCES, and for these, the correct style to use is "Heading 5."

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles, named "Heading 1," "Heading 2," "Heading 3," and "Heading 4", are prescribed.

C. Figures and Tables

Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation "Fig. 1," even at the beginning of a sentence.

TABLE I. TABLE STYLES

Table Head	**Table Column Head**		
	Table column subhead	***Subhead***	***Subhead***
copy	More table copy		

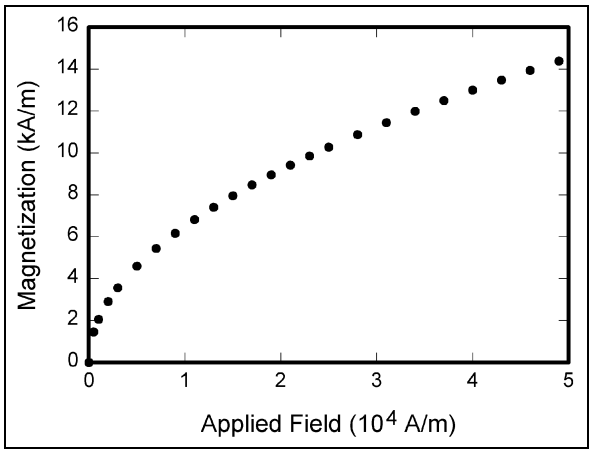


Fig. 1. Example of a figure caption

V. Conclusion

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions.

Appendix

Appendixes, if needed, appear before the acknowledgment.

ACKNOWLEDGMENT

The preferred spelling of the word "acknowledgment" in American English is without an "e" after the "g." Use the singular heading even if you have many acknowledgments. Avoid expressions such as "One of us (S.B.A.) would like to thank ... ." Instead, write "F. A. Author thanks ... ." You could also state the sponsor and financial support acknowledgments here.

References

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use "Ref. [3]" or "reference [3]" except at the beginning of a sentence: "Reference [3] was the first ..."

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors' names; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [4]. Papers that have been accepted for publication should be cited as "in press" [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

[1] G. Eason, B. Noble, and I.N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955. (*references*)

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[3] I.S. Jacobs and C.P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.

[4] K. Elissa, "Title of paper if known," unpublished.

[5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740-741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

UMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA

ISSN 2355-3286

Universitas Multimedia Nusantara
Scientia Garden Jl. Boulevard Gading Serpong, Tangerang
Telp. (021) 5422 0808 | Fax. (021) 5422 0800