

A new system for underwater vehicle balancing control based on weightless neural network and fuzzy logic methods

Ahmad Zarkasi¹, Hadipurnawan Satria², Anggina Primanita², Abdurahman¹, Nurul Afifah¹, Sutarno¹

¹Department of Computer Engineering, Faculty of Computer Science, Sriwijaya University, Palembang, Indonesia

²Department of Informatic Engineering, Faculty of Computer Science, Sriwijaya University, Palembang, Indonesia

Article Info

Article history:

Received Nov 30, 2023

Revised Mar 11, 2024

Accepted Mar 25, 2024

Keywords:

Balance control system

Fuzzy logic

Underwater vehicle

Unmanned aerial vehicles

Weightless neural network

ABSTRACT

The utilization of humans to be in the water for short time, resulting in limited area underwater that can be explored, so the information obtained is very limited, plus the influence of irregular water movements, changes in waves, and changes in water pressure, indirectly also constitutes obstacle to this problem. One of the best solutions is to develop underwater vessel that can travel either autonomously or by giving control of movement and navigation systems. New system for underwater vehicle balance control through weightless neural network (WNN) and fuzzy logic methods was proposed in this study. The aim was to simplify complicated data source on stability system using WNN algorithm and determine depth level of autonomous underwater vehicle (AUV) through fuzzy logic method. Moreover, speed control of underwater vehicle was determined using fuzzy rule-based design and inference. The tests were conducted by showing convergence performance of system in the form of AUV simulator. The results showed that proposed system could produce real-time motion balance performance, faster execution time, and good level of accuracy. This study was expected to produce real-time motion balance system with better performance, faster execution time, and good level of accuracy which could be subsequently used to design simple, cheap, and efficient hardware prototype.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Ahmad Zarkasi

Department of Computer Engineering, Faculty of Computer Science, Sriwijaya University

Al Gazali Mosque Road, Bukit Lama, Ilir Bar. I, Palembang City, South Sumatra 30128, Indonesia

Email: zarkasi98@gmail.com

1. INTRODUCTION

The utilization of water is very difficult for humans due to the limitations associated with staying underwater for a very long period. This has led to attempts to construct vehicles capable of long-term survival and underwater exploration. Some of the underwater vehicles constructed could be driven automatically or with the assistance of motion and navigation systems. The design process required the consideration of weight, dynamic motion, and gravitational pressure [1], [2]. Moreover, the delicate motion and location of these vehicles could also be indirectly influenced by irregular water motion, wave variations, and changes in water pressure. This showed the need to have a steady motion condition from expected or unpredicted disturbances in order to tackle the challenge [3], [4].

Several fundamental controllers have been developed for autonomous underwater vehicle (AUV) depth control such as the proportional integral derivative (PID), fuzzy logic control (FLC), artificial neural network (ANN), and sliding mode control (SMC) [5], [6]. The controls designed based on PID are widely utilized due to the ease of implementation in real-time systems and have been used in different underwater

vehicles such as remotely operated vehicle (ROV). However, the PID control could not be dynamically correct for unmodeled vehicle hydrodynamic forces or unknown sudden-onset disturbances [7], [8]. There was also the possibility of paradoxical parameter combinations such as those between the rising time and overshoot variables which led to the attempt to design another control using the fuzzy logic method.

Several studies have used the fuzzy logic method in designing and applying controls successfully in different underwater vehicles [8], [9]. This is based on the ability of the logic to provide a computational solution for control when a mathematical model cannot be calculated correctly or understood repeatedly [10], [11]. The segmentation applied through the process can be used in modelling the complex hydrodynamics of underwater vehicles. Meanwhile, the disadvantage of the method is associated with the ability to increase the complexity of analyzing parameters. It is also difficult to tune the fuzzy rules and the overshoot prediction time is not smooth [12], [13]. Attempts have been made to overcome the problems through the development of underwater vehicle control system using the ANN method [14], [15]. The preference for the method is due to its ability to automatically adjust to changes in the inputs used for the underwater vehicle system. Moreover, ANN can learn, adapt, and evolve like the human brain [16], [17]. It has been used to anticipate the performance of the vehicle system and minimize the weight function value of each layer to determine the best input. The method was observed to have outperformed other controllers [18], [19], applied to tune the PID, and adapted to the changing depth of the underwater vehicle [20], [21]. However, the disadvantage of ANN is the long computation time which often leads to lagging despite the fast and too complex responses required to satisfy in real-time system applications. The need to overcome these problems led to the proposition of a hybrid weightless neural network (WNN) and fuzzy logic method.

The aim of developing the hybrid system is to solve complex data sources using WNN and simple data through fuzzy logic. The adoption of the WNN method was due to its ability to recognize all input patterns in binary or weightless data [22], [23] in order to allow the processing of all input variables into balance control in binary data format. This is because the method works in binary levels in memory [24]–[26], has speed, is considered reliable in processing sensor data, and can be processed on limited resources or with embedded platforms. Furthermore, the data is normally grouped in memory cells to be analyzed to produce quality information [27] needed in the decision-making process related to the balancing of motion for underwater vehicles. This study was expected to produce a real-time motion balance system with better performance, faster execution time, and a good level of accuracy which can be subsequently used to design a simple, cheap, and efficient hardware prototype.

2. MATERIAL AND CONTROLLER DESIGN

2.1. Autonomous underwater vehicle design

Underwater vehicle system was developed through two main steps including the hardware and software design. The hardware was based on two systems which included the vehicle in the form of a simulator and AUV. The focus was to design the AUV simulator controller on a laboratory scale using several basic components without any implementation in the real environment. This was achieved using Mega256 with a processor clock frequency of 16 MHz as the controller. Moreover, the actuator system had four stepper motors with 10/step specifications which were used to test the simulation of the unmanned aerial vehicles (UAV) tilt position in order to display x, y, and z data from the MPU6050 inertial measurement unit (IMU) sensor. The location of each stepper motor as well as the components used in the AUV simulator are presented in the Figure 1.

The stepper motors were mounted on the upper frame of the AUV simulator with steppers 1 and 2 placed on the front and rear, respectively, to collect data on the angles in the x orientation. Meanwhile, steppers 3 and 4 were positioned on the right and left, respectively, to obtain the y value. The movement of the four stepper motors was combined to produce the z value as presented in Figure 1. The simulator frame was approximately 100 cm tall with a width of 40 cm and a length of 60 cm. Moreover, a belt conveyor with a length of 50 cm and a width of 20 cm was located at the bottom of the frame and powered by a motor to transport several things such as boxes and balls. The conveyor was believed to be the fundamental aesthetic representation of a river, lake, or sea. Furthermore, the AUV simulator was 30 cm in length and 25 cm in breadth and was designed with IMU and sonar sensors as presented in Figure 2. The tilt angle of the AUV was detected through the MPU6050 IMU sensor which consisted of accelerometer and gyroscope sensors [28], [29]. The two signals had distinct functions depending on the usage in the system. The accelerometer sensor was used to determine the acceleration in the surroundings [30], [31] and measured the tilt angle of an object through a comparison with the detected gravitational pull of the earth. The reading of the accelerometer against gravity is presented in Figure 2.

The accelerometer sensor in Figure 2 had three axes, including the x-axis, y-axis, and z-axis, with an orientation towards gravity g. The angular shift of the starting position with respect to the x-axis is, y-axis is, and z-axis is while the values regarding the x' axis is, y' axis is, and z' axis is. The variable g was used to

represent the direction of the gravity with the value at rest recorded to be 0 while the moving value was 1. The initial location of the three axes of movement for the accelerometer sensor is presented in Figure 2(a). The position was neutral with the x and y axes horizontally orientated while the z-axis was directed vertically to the gravitational force *g* of the earth. Meanwhile, Figure 2(b) shows the displacement of the two axes, x and z, with respect to the y-axis line. The location of the x' and z' axes was used to show the amount of the angular shift. Furthermore, Figure 2(c) depicts a y and z-axis shift with regard to the x-axis orientation at a magnitude of ψ and ϕ respectively. The 3-axis displacement of the accelerometer axis is presented in Figure 2(d) while (1), (2), and (3) showed the amount of the shift in the angle of gravity.

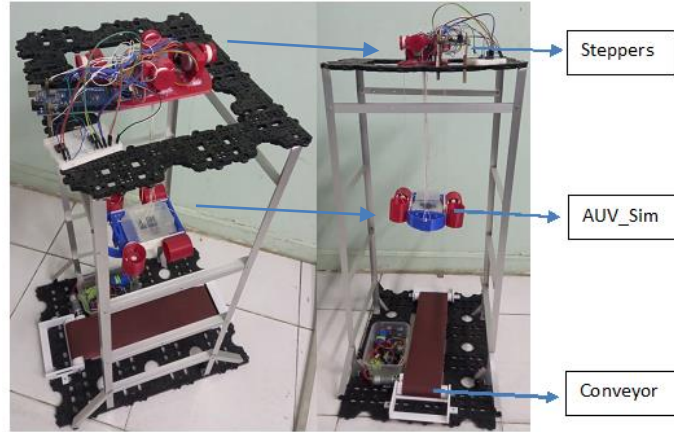


Figure 1. AUV simulator and components

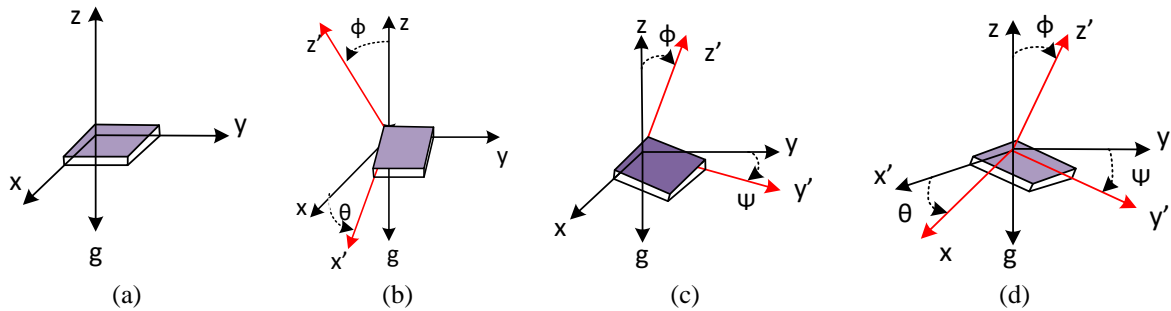


Figure 2. Changes in accelerometer angle with respect to gravity, (a) neutral position, (b) the two axes, x and z, with respect to the y-axis line, (c) the two axes, y and z, with respect to the y-axis line, (d) the 3-axis displacement of the accelerometer axis

$$\theta = \tan^{-1} \left(\frac{Ax'}{\sqrt{Ay'^2 + A^2z'^2}} \right) \tag{1}$$

$$\Psi = \tan^{-1} \left(\frac{Ay'}{\sqrt{Ax'^2 + A^2z'^2}} \right) \tag{2}$$

$$\phi = \tan^{-1} \left(\frac{\sqrt{A^2x'^2 + A^2y'^2}}{A^2z'} \right) \tag{3}$$

A gyroscope is a micro-machined electro-mechanical systems (MEMS) device that uses mechanical forces to detect angular velocity or sustain rotational motion by exploiting the Coriois effect which states rotating with angular velocity, mass (*m*), and rotation speed (*v*). The operation of a gyroscope and its angular orientation is presented in Figure 3. The three-axis gyroscope is capable of measuring rotation along three axes, including x, y, and z, as well as detects angular rotation displacement by measuring the changes in capacitance. Furthermore, the angular velocity of the IMU was calculated by integrating the data obtained. This is necessary because a gyroscope is essentially a spinning mass that rotates around its axis to stabilize and maintain the rotational orientation. The detachment of a gyroscope from a three-axis gimbal allows the mass to travel in the same axis orientation but the direction changes.

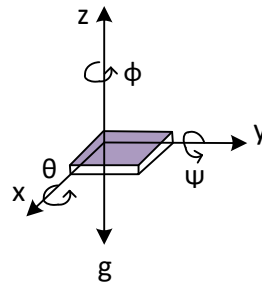


Figure 3. The basic principle of gyroscope and rotation angles θ , Ψ , ϕ

2.2. Controller design

A hybrid system combining WNN and fuzzy logic methods was recommended in this study using the Mega256 microcontroller. The main system block diagram is presented in the following Figure 4 with the data from the accelerometer and sonar sensors used as input response. Both were integrated into the process block with the accelerometer sensor based on WNN while sonar was associated with the fuzzy logic, and the results obtained were used as input in the AUV simulator model.

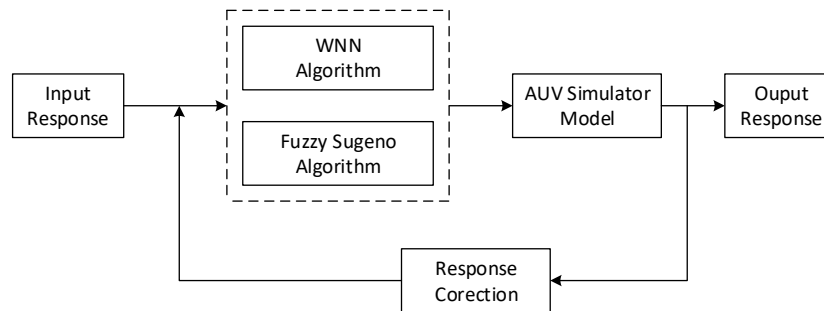


Figure 4. System block diagram

It was observed that the model was used to simulate the motion and distance based on every circumstance of the AUV environment. The focus was particularly on the level of tilt towards the x, y, and z axes in order to determine the balancing of the system during movement while the fuzzy logic method was used to check the depth level. The output from the simulator movement was sent back into the response correction block in order to keep the error value below the threshold level set. The WNN approach acts as the Accelerometer data processor, which processes data on the inclination state of the AUV simulator. Both sets of data will be supplied to the response output block, which will drive the movement system, and the response correction block, which will constantly verify the bending level and depth of the AUV simulator.

2.3. Weightless neural network controller

RAM intelligence was used in the WNN algorithm [27], [32], [33] with a focus on binary data which had high computational speed and efficiency, as well as considered ideally suited for use in systems with embedded platforms. The architectural idea underlying WNN is presented in Figure 5 where the IMU sensor provides input data in the form of three separate groups correlating to the sensor output as x, y, and z axes degrees which are to be further converted to binary integers. It is important to state that WNN is a conceptually simple artificial network model based on a neural network model, with input and output binary integers (0 or 1). The neural network functions are saved in a look-up table which is implemented in RAM. Moreover, the algorithm learns by changing the contents of the data in the look-up table of RAM to produce an extremely flexible and quick training method.

The 24 data bits (input data n_0 until n_{23}), are placed and divided into 3 RAM nodes designated as RAM_node1, RAM_node2, and RAM_node3. In principle, the RAM node is a collection of several cells that form a RAM with a certain capacity based on the pattern of digital number data rules such as 1 bit, 2 bits, and 3 bits. The RAM node is different from digital data by being the location or memory where the data is placed or processed. This showed that the sensor data were grouped in the input pattern with a data width of n_0 to

n₂₃ based on 8 bits provided to each of the x, y, and z axes. The data were later processed in the RAM node through a direct comparison to the reference data using the minority decision sub-algorithm. This was conducted to speed up the process of calculating the findings based on the assumption that the sensor data input was simply 1-axis motion. The aim was to ensure a direct comparison of the data to the reference in order to determine the winner class output value. In other words, a vector input with only one active RAM node such as RAM_{node1} can be directly compared to the pattern reference data using the comparator.

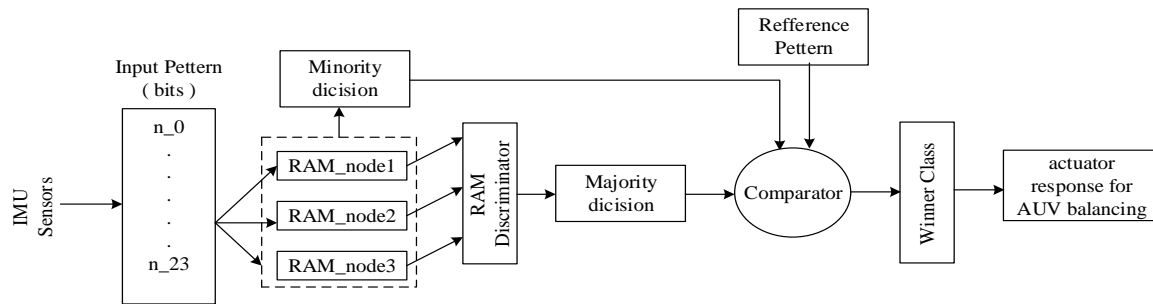


Figure 5. WNN controller architecture

Meanwhile, when the input sensor data contains two or three axes, the data needs to be first processed in the RAM discriminator. This shows that the existence of more than one RAM_{node} active requires processing the data in the discriminator to determine the data majority decision. The discriminator is essentially a series of word X bits RAM with n input vectors and a summing device. It is important to state that each discriminator can receive a binary pattern of (Xn) bits as input while the summing device sums and divides the output of the RAM nodes by the number of input vectors. The data obtained are usually sorted based on the highest importance and compared to the reference pattern data to acquire the best value and determine the winning class. The final result can be used to indicate the vulnerable position of AUV to be promptly addressed on the propulsion systems in order to ensure a safe level.

2.4. Fuzzy logic controller

The fuzzy logic method was used to handle sonar sensor data due to its ability to transform a quantity represented using linguistic language [34]–[36]. The method has several advantages, including the ability to produce results in the form of response values based on ambiguous and qualitative data. The stages used in designing the FLC included fuzzification, interference engine, rule basis, and defuzzification as presented in Figure 6. The process required using two sonar sensors, d₁ and d₂, to produce binary distance data with a conversion of 10 bits each, leading to the determination of 20 bits of RAM which ranged from n₂₄ to n₄₃. Moreover, the set point value for each sensor was calculated and the fuzzification stage was used to convert crisp data into fuzzy input in the form of linguistic values ranging from 0 to 1.

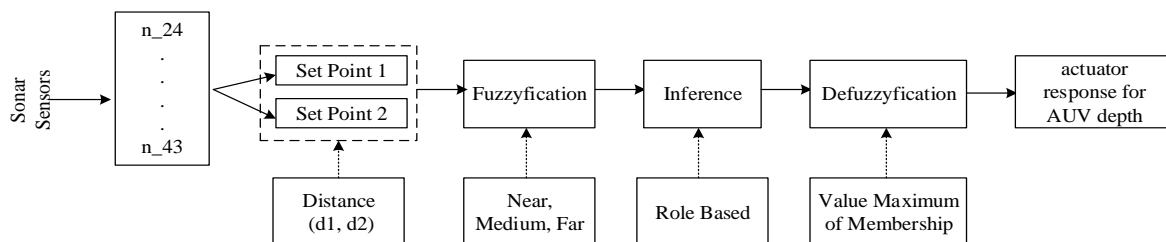


Figure 6. Block diagram of FLC

In theory, the fuzzification stage focused on processing the incoming data into the control system based on the distance of the sensor to the item. Moreover, the speed control of the AUV depth location was determined using fuzzy rule-based design and inference through a procedure known as defuzzification. It is

important to state that all the fuzzy logic procedures are used to produce the location value and form of the undersea surface.

3. RESULTS AND DISCUSSION

3.1. Weightless neural network algorithm controller

The driving system such as the stepper motor used in this study was the first step in the test analysis. The stepper motor used was a 28BYJ-48 with four coils and a rotation degree of 5.625° per step which showed the need for 64 steps to obtain one complete revolution of 360°. The four stepper motors applied were required to be tested in order to determine the accuracy level. It was important to state that the devices were mounted on the four sides of the AUV simulator to ensure counterbalance and vertical motion. Each was evaluated in the motion test through a vertical spin at a specified distance ranging from 5 cm to 25 cm and compared to the one-centimeter reference distance data. The results showed that Stepper1 had the highest distance and accuracy at 15 cm by 15.45 cm and the lowest at 25 cm by 25.8 cm while the values were recorded for Stepper4 at 15 cm by 15.55 cm and 25 cm by 25.73 cm respectively as presented in Figure 7. Furthermore, the least total error data of 2.63 cm was reported in Stepper2 while the maximum was 3.1 cm in Stepper4. It was also observed that Stepper2 had the least average error with 0.53 cm while Stepper4 had the highest with 0.62 cm.

The next step was to test the IMU sensor and this was achieved by first determining the magnitude of the reference angle to the end orientation axes, with the balanced point of sensor detection set at 90° position. A clockwise change was observed to have led to an inclination of the angle towards 0°, thereby showing the position of the UAV was tilted to the left. Meanwhile, an anti-clockwise change led to an inclination of the angle towards 180°, showing the position of the UAV was tilted to the right. It was discovered that the change in angle from the balanced point of 90° used as a reference could lead to a decrease in multiples of 5 to the left tilt angle of 0° or an increase in multiples of 5 to the maximum right tilt angle of 180°. The results of the tilt angle detection test conducted using 10 times per 5° as a sample are presented in the following Figure 8.

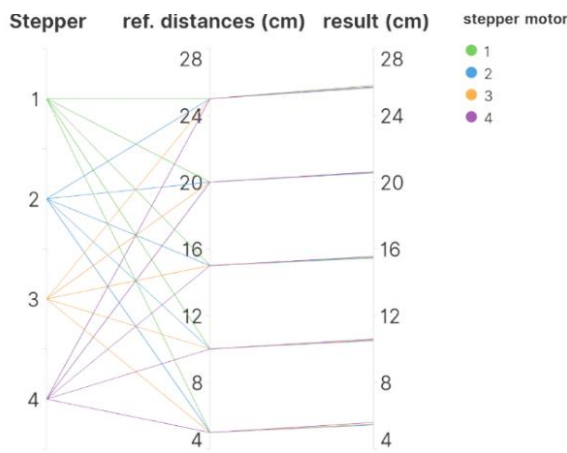


Figure 7. Results of the movement test for all the steppers (cm)

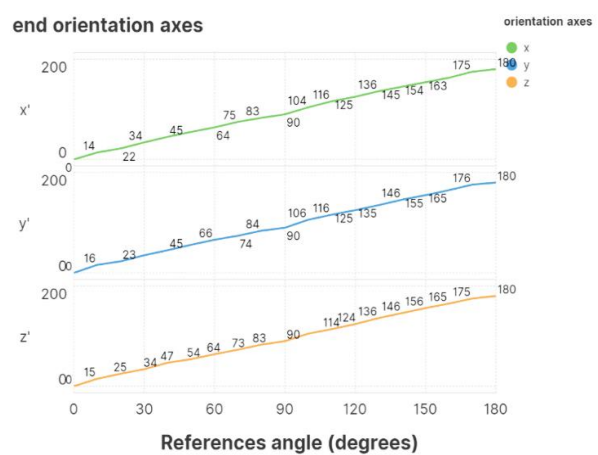


Figure 8. The result of the tilt angle detection test at 10 times

The highest angle difference between the x-axis and x'-axis was found to be 6° between 116° and 136° while the lowest was 2° and identified at 22°. Moreover, the highest between the y-axis and y' axis was recorded to be 6° at angular points of 50°, 60°, 100°, 110°, and 176° while the smallest was 3° at 23°. It was also discovered that the highest between the z-axis and z' axis was 7° at 47° while the smallest was 30° between 73° and 83°. Furthermore, the RAM node test results are presented in the Table 1.

The determination of the data related to the change in degree for each axis, x, y, and z, was followed by the calculation of the ADC data converted from 0° to 90°. The results showed that the conversion data was 8 bits long or 255 decimals, thereby indicating every 10-angle variation caused a 1.4-bit change in the input vector. The data were subsequently stored in the RAM address of the node with RAM node 1 focused on the x-axis, RAM node 2 for y-axis data, and RAM node 3 for z-axis data. Reference patterns were also

used as the comparison basis for the input data to decide the output to be generated. Moreover, the movement of the stepper motor was adjusted to the data for the output patterns determined in the previous step.

The data on the RAM nodes 1, RAM node 2, and RAM node 3 are presented in Table 1 and Figure 8. It was observed that the values presented in Table 1 for each RAM node were different but those for 90°, 0°, and 180° were the same for each RAM node based on the status as the degree of calibration. Furthermore, the data for the reference patterns were provided to be ideal based on the increase and decrease in the detecting angle of 5° per step, as shown in Figure 8. The data in the output pattern column were later determined based on the odd increment in RAM for angle detection from 0° to 180° which ranged from 0000 0001b (1 decimal) to 0010 0101b (37 decimal). The smallest PWM data was recorded to be 100 and found in output pattern 19 at an angle of 90° while the highest was 210 at 0° and 180°. Moreover, PWM data 140 was used for four different output patterns, 15, 17, 21, and 23 with a similar trend observed for PWM data 200 which was applied for 3, 5, 33, and 37. It was important to state that the data for the output patterns were converted from the binary form in the RAM.

Table 1. RAM node data

RAM_node1	RAM_node2	RAM_node3	Reference Patterns	Output Patterns	PWM
0000 0000	0000 0000	0000 0000	0000 0000	0000 0001	210
0001 0011	0001 0110	0001 0101	0000 1110	0000 0011	200
0001 1111	0010 0000	0010 0011	0001 1100	0000 0101	200
0011 0000	0011 0001	0011 0000	0010 1010	0000 0111	180
0011 1111	0011 1111	0100 0010	0011 1000	0000 1001	180
0100 1101	0100 1111	0100 1100	0100 0110	0000 1011	160
0101 1010	0101 1101	0101 1010	0101 0101	0000 1101	160
0110 1010	0110 1000	0110 0111	0110 0011	0000 1111	140
0111 0101	0111 0111	0111 0101	0111 0001	0001 0001	140
0111 1111	0111 1111	0111 1111	0111 1111	0001 0011	100
1001 0011	1001 0110	1001 0100	1000 1101	0001 0101	140
1010 0100	1010 0100	1010 0001	1001 1011	0001 0111	140
1011 0001	1011 0001	1010 1111	1010 1010	0001 1001	160
1100 0000	1011 1111	1100 0000	1011 1000	0001 1011	160
1100 1101	1100 1110	1100 1110	1100 0110	0001 1101	180
1101 1010	1101 1011	1101 1101	1101 0100	0001 1111	180
1110 0110	1110 1001	1110 1001	1110 0010	0010 0001	200
1111 0111	1111 1001	1111 0111	1111 0000	0010 0011	200
1111 1111	1111 1111	1111 1111	1111 1111	0010 0101	210

3.2. Fuzzy logic algorithm controller

The focus of this section is to provide information on the procedures associated with the development and application of FLC. Data were received from two sonar sensors and used to calculate the set point. The two main purposes served by the sonar sensors were to provide input on the distance and underwater object pattern to the controller. The first stage of the test conducted was to collect distance measurement data from each sonar up to ten times for the level ranging from 5 cm to 60 cm to ensure validation against error. The results were later compiled and the average was calculated. The next step was to determine the membership function output to calculate the defuzzification and the data obtained after the completion of the test are presented in Table 2.

Table 2. Sonar movement test and PWM

Reference distances (cm)	Sonar measurement results (cm)		PWM	
	sonar1 (d1)	sonar2 (d2)	Membership function output	Scrip
5	5.3	5.5	Fast (F)	180
10	10.4	10.2	Fast (F)	180
15	15.2	15.4	Fast (F)	180
20	20.4	20.2	Fast (F)	180
25	25.5	25.4	Medium (M)	150
30	30.5	30.5	Medium (M)	150
35	35.7	35.5	Medium (M)	150
40	40.5	40.7	Medium (M)	150
45	45.3	45.5	Slow (S)	100
50	50.5	50.2	Slow (S)	100
55	55.5	55.5	Slow (S)	100
60	60.3	60.5	Slow (S)	100

The test was conducted at a 60cm reference distance presented in Table 3 by dropping the UAV simulator vertically. Data were collected starting from a distance of 5 cm to 60 cm with the sonar1 (d1) sensor placed on the front of the UAV simulator while the sonar2 (d2) sensor was at the back. The smallest error recorded for sonar1 was 0.2 cm at a distance of 15 cm while the highest was 0.7 cm at 35 cm and the average was found to be 0.5 cm. The smallest error on sonar2 (d2) was 0.2 cm at 10 cm and 50 cm distances while the highest was 0.7 cm at 40 cm as presented in Figure 9. Furthermore, the PWM column showed that the output membership function had three variables, including fast (F), medium (M), and slow (S), used to provide the response speed of the stepper motor in the script value. It was observed that the PWM script values for each variable output were 180, 150, and 100.

The error data were represented in Figure 9 using three vertical lines made up of reference distances. The measurement gap between sonar1 and the reference was represented by error_1 while the difference between the reference data and sonar2 was error_2. The results showed that the error ranged from 0.2 cm to 0.7 cm with sonar1 having five values of 0.2, 0.3, 0.4, 0.5, and 0.7. The most recurring error was found to be 0.5 cm which was observed five times for sonar1 and six times for sonar2. Meanwhile, the lowest was 0.7 cm which was found once with sonar1 and sonar2, and the average error for each was 0.43 cm. The evaluation of the sensor data was followed by the determination of the set point value required for the fuzzification process. The results showed that the set point value for each sonar sensor was the same which was indicated by the 0 cm used for the lowest and up to 60 cm for the furthest distance. Moreover, the distance was further divided into three rule-based areas, including near for 0 cm up to 30 cm, medium for 10 cm to 50 cm, and far for 30 cm to 60cm. This was followed by the establishment of the membership degree of each sensor which ranged from a set point of 10 cm to 50 cm.

Table 3. Defuzzification test results

Set Point (cm)	Membership function	Rule base			Defuzzification ($Z = \frac{\sum(uxi) \cdot (x_0)}{\sum(uxi)}$)		
		N	M	F	$\sum(uxi)(x_0)$	$\sum(uxi)$	Z
10	$\alpha_1 = 1$ and $\alpha_2 = 0$	α_1	α_2	0	900	5	180
15	$\alpha_1 = 0.75$ and $\alpha_2 = 0.25$	α_1	α_2	0	795	4.5	176.6
20	$\alpha_1 = 0.50$ and $\alpha_2 = 0.50$	α_1	α_2	0	690	4	172.5
25	$\alpha_1 = 0.25$ and $\alpha_2 = 0.75$	α_1	α_2	0	765	4.5	170
30	$\alpha_1 = 0$ and $\alpha_2 = 1$	0	α_2	0	840	5	168
35	$\alpha_1 = 0.25$ and $\alpha_2 = 0.75$	0	α_1	α_2	737.5	4.5	163.88
40	$\alpha_1 = 0.50$ and $\alpha_2 = 0.50$	0	α_1	α_2	635	4	158.75
45	$\alpha_1 = 0.75$ and $\alpha_2 = 0.25$	0	α_1	α_2	682.5	4.5	151.66
50	$\alpha_1 = 1$ and $\alpha_2 = 0$	0	α_1	α_2	730	5	146

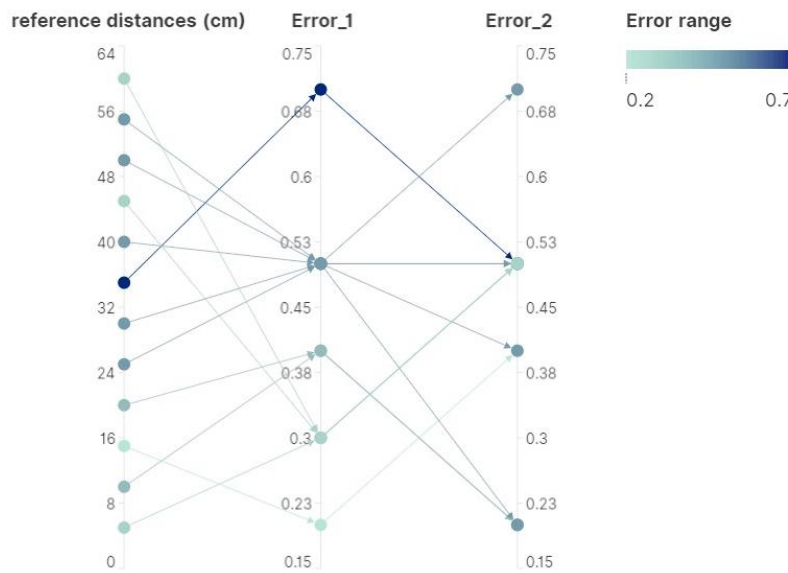


Figure 9. Results of errors at different distances

Table 3 shows that each membership function input designated as near (N), medium (M), and far (F) has two degrees, α_1 and α_2 . Moreover, the maximum value for the membership degree was used for the

defuzzification equation (Z). It was observed that the accuracy for Near and Medium functions at a distance of 10 cm to 25 cm were α_1 and α_2 while far had zero. Meanwhile, only medium and far were active at 35 cm to 50 cm with an accurate value of α_2 and α_1 . The results also showed that all membership functions had affective chances at a distance of 30 cm because the position was in the middle of the computation. For example, the first test was conducted by setting the initial set point at a distance of 10 cm between sonar sensors d1 and d2. The membership degree value for d1 was found to be $\alpha_1=1$ and $\alpha_2=0$ followed by $\alpha_1=1$ and $\alpha_2=0$ for d2. The calculations also showed that the number of membership functions (uxi) on the rule-based was 5 and the membership function output (xo) with the greatest PWM value of 180 was fast. Furthermore, when the xo value was 900, the resultant number of membership functions (uxi) was also recorded to be 900 while the application of the defuzzification equation in Table 3 led to the production of 180. The distance of the observed item was found to be very close and this led to the quick movement of the stepper motor.

3.3. System performance test

The focus of this section is on testing and analyzing the system developed in the form of an AUV simulator based on the method proposed by showing the convergence performance. The simulator was produced through the combination of controllers developed using both the WNN algorithm and fuzzy logic methods. The orientation of the UAV movement based on the angle determined through the IMU sensor was summarized in three dimensions, including towards the angle x of the front view at (X), the angle y of the side view (Y), and the angle z of the top view (Z). Moreover, the movement angle was further displayed in larger sizes, X, Y, and Z, through the degree of freedom (DOF) using 90° as the starting point as discussed in the previous section. The concepts associated with WNN controllers such as input vector, RAM node, RAM discriminator, reference patterns, winner class, and PWM have also been discussed. It was observed that the binary data in the WNN controller table was set as the initial values when the system was in normal condition.

Meanwhile, the base rule determined for the FLC simulator included the membership degrees S1.0 and S1.1 which were found to be α_1 and α_2 for sensor1 and S2.0 and S2.1 recorded as α_2 and α_1 for sensor2. The PWM single-tone output part on the PWM (rad) and the sensor value was also presented as distance (cm). The last defuzzification process was conducted in the Z display and the initial distance sensor data were captured by the controller. The motion performance was used to determine the movement of the AUV simulator developed in three angular axes, X, Y, and Z, facing opposite directions. The aim was to determine the response of the simulator to angular changes in the three axes and the subsequent effect on the value displayed in the RAM data.

The highest X-axis angle was detected to be 101° in Figure 10 while the Y-angle was 77° and the Z axis was 96°. These three angles were used as the vector input on the WNN controller with the 101° angle stored in RAM_node1 as 10001111, 77° in RAM_node2 as 01101101, and 96° in RAM_node3 as 10001000, to be processed using the discriminator to determine the best. The process focused on comparing the values of each of the RAM data to determine the highest which was found to be RAM_node2 data and designated as the winner class. The difference in the Y-axis angle and the neutral axis was recorded to be 130 and the comparison of 01101101 in RAM_node2 against the reference pattern showed the need for a motor PWM of 140 to return to normal position. As the AUV moved towards the neutral position along the Y-axis orientation, data checks were performed on each RAM node sequentially up to the moment the system achieved a balanced position.

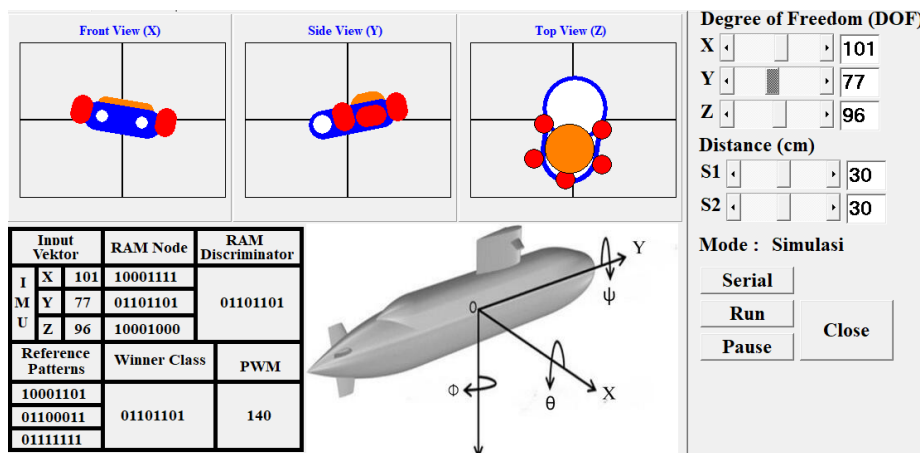


Figure 10. WNN controller balance performance test

The next step was to use the FLC method to measure and test the performance of the depth level. The results presented in Figure 11 showed that the distance for S1 was 47 cm and S2 was 43. The membership function chart was used to analyze the sensor data and S1 sensor α_1 was found to be 0.85 while α_2 had 0.15. It was also observed that the α_1 for the S2 sensor was 0.65 while α_2 was 0.35. The active basic rules were reported to be M-N, F-N, N-M, M-M, F-M, N- F, M-F, and F- F while the inactive was N – N. According to the following equation, the total maximum number of membership functions ($\sum(ux_i)$) based on the base rule data was 4.7.

$$\begin{aligned} \sum(ux_i) &= ux_1 + ux_2 + ux_3 + ux_4 + ux_5 + ux_6 + ux_7 + ux_8 + ux_9 \\ &= 0 + 0.35 + 0.65 + 0.15 + 0.35 + 0.65 + 0.85 + 0.85 + 0.85 \\ &= 4.7 \end{aligned}$$

The x_0 for each successively active base rule was found to be 210, 180, 180, 180, 150, 150, 150, and 100. This shows that the total number of input function membership degrees with output membership functions ($\sum(ux_i) * (x_0)$) is as follows:

$$\begin{aligned} \sum(ux_i) * (x_0) &= ux_1 * x_{03} + ux_2 * x_{02} + ux_3 * x_{02} + ux_4 * x_{02} + ux_5 * x_{02} + ux_6 * x_{01} + ux_7 * x_{01} + ux_8 * x_{01} + ux_9 * x_{00} \\ &= 0 * 210 + 0.35 * 180 + 0.65 * 180 + 0.15 * 180 + 0.35 * 180 + 0.65 * 150 + 0.85 * 150 + 0.85 * 150 + 0.85 * 100 \\ &= 0 + 63 + 117 + 27 + 63 + 97.5 + 127.5 + 127.5 + 85 \\ &= 707.5 \end{aligned}$$

where x_03 is very fast, x_02 is fast, x_01 is medium, and x_00 is slow.

The collection of all the data was followed by the determination of the defuzzification value ($Z = \sum(ux_i) * (x_0) / \sum(ux_i)$). This was achieved using the following relationship:

$$\begin{aligned} Z &= \sum(ux_i) * (x_0) / \sum(ux_i) \\ &= (707.5) / (4.7) \\ &= 150.5 \end{aligned}$$

The PWM data of the motor was found to be 150.5 when the AUV system moved towards the far for the sensor S2 and achieved the optimal location based on the defuzzification data. The state of the two sensors was further examined to determine the next movement of the system.

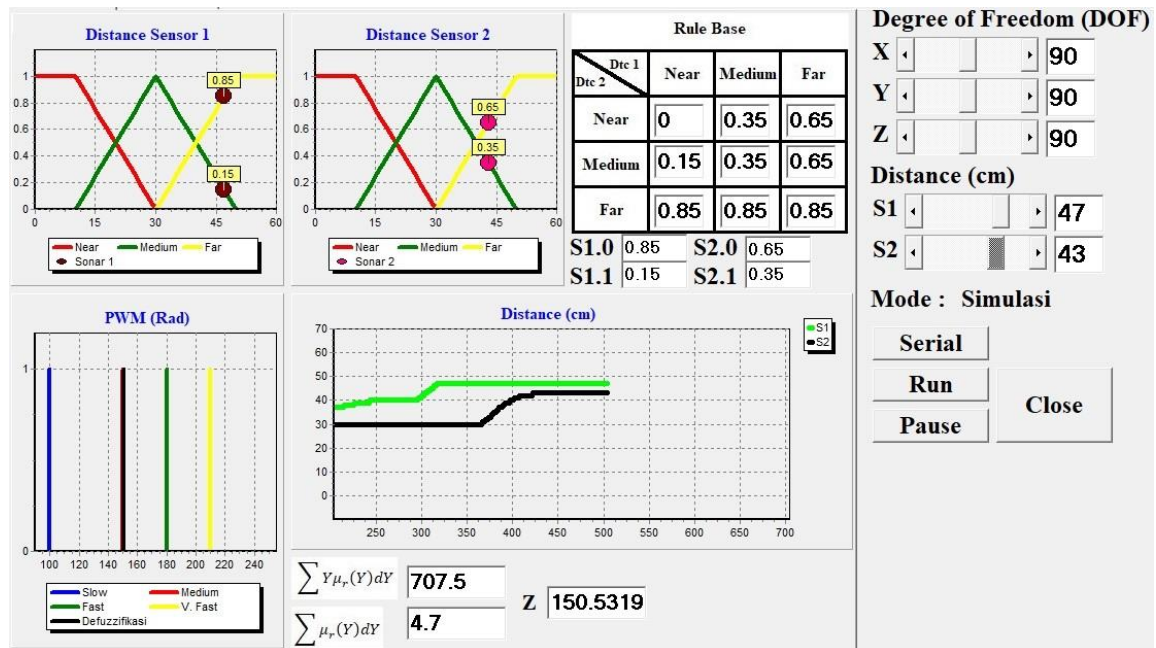


Figure 11. FLC depth performance test

4. CONCLUSION

The study objectives were fulfilled by developing techniques for AUV stability and depth sensing systems utilizing hybrid approaches combining WNN and fuzzy logic. The WNN algorithm performed well in simplifying the complicated data source on the stability system. The RAM inputs are separated into three groups, each representing the axis degree data x, y, and z. The degree information will be converted to binary integers. As a result of the test, every 1° angle change causes a 1.4 bit change in the data on the vector input on the RAM node. The fuzzy logic technique has effectively processed and interpreted water depth data. There are nine rule base inputs and four membership function outputs. According to the test findings, the average distance test error for each sonar was 0.43 cm. The results obtained from the process of fuzzification to defuzzification on the measurement of system performance are the same between the results of the calculation and the simulation. The approaches used in this study may be extended to multisensors and underwater image processing, yielding more diverse and complicated findings.

ACKNOWLEDGEMENTS

The research publication of this article was funded by DIPA of Public Service Agency of Universitas Sriwijaya 2022. Nomor SP DIPA-023.17.2.677515/2022, On November 17, 2021. In accordance with the Dean's Decree Number: 2670/UN9.1.9/LT/2022, On August 2, 2022," was supported by Universitas Sriwijaya. The first author would like to thank the Faculty of Computer Science, Strategic Pervasive Computing and Intelligent Embedded Systems Research Group (SPCIES-RG), and COMNETS RG, as well as Universitas Sriwijaya's Embedded Systems, Control System, and Robotic Laboratory.





REFERENCES

- [1] F. A. Azis, M. S. M. Aras, M. Z. A. Rashid, M. N. Othman, and S. S. Abdullah, "Problem identification for underwater remotely operated vehicle (ROV): a case study," *Procedia Engineering*, vol. 41, pp. 554–560, 2012, doi: 10.1016/j.proeng.2012.07.211.
- [2] Z. Luo, X. Xiang, and Q. Zhang, "Autopilot system of remotely operated vehicle based on ardupilot," *Intelligent Robotics and Applications*, Springer International Publishing, pp. 206–217, 2019, doi: 10.1007/978-3-030-27535-8_19.
- [3] A. Sabanovic, "SMC framework in motion control systems," *International Journal of Adaptive Control and Signal Processing*, vol. 21, pp. 731–744, 2007, doi: 10.1002/acs.
- [4] M. W. N. Azmi *et al.*, "Comparison of controllers design performance for underwater remotely operated vehicle (ROV) depth control," *International Journal of Engineering and Technology (IAE)*, vol. 7, no. 3, pp. 419–423, 2018.
- [5] L. C. L. Hansen *et al.*, "A low-cost remotely operated vehicle (ROV) with an optical positioning system for under-ice measurements and sampling," *Cold Regions Science and Technology*, vol. 151, pp. 148–155, 2018, doi: 10.1016/j.coldregions.2018.03.017.
- [6] Y. R. Petillot, G. Antonelli, G. Casalino, and F. Ferreira, "Underwater robots: from remotely operated vehicles to intervention-autonomous underwater vehicles," *IEEE Robotics and Automation Magazine*, vol. 26, no. 2, pp. 94–101, 2019, doi: 10.1109/MRA.2019.2908063.
- [7] A. M. Abdullah *et al.*, "Review of the control system for an unmanned underwater remotely operated vehicle," *Advanced Structured Materials*, vol. 85, pp. 609–631, 2018, doi: 10.1007/978-3-319-72697-7_49.
- [8] A. Sarhan and S. Qin, "Adaptive PID control of UAV altitude dynamics based on parameter optimization with fuzzy inference," *International Journal of Modeling and Optimization*, vol. 6, no. 4, pp. 246–251, 2016, doi: 10.7763/ijmo.2016.v6.534.
- [9] C. Fu, M. A. O. Mendez, R. S. Fernandez, and P. Campoy, "Monocular visual-inertial SLAM-based collision avoidance strategy for fail-safe UAV using fuzzy logic controllers: comparison of two cross-entropy optimization approaches," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 73, no. 1–4, pp. 513–533, 2014, doi: 10.1007/s10846-013-9918-3.
- [10] M. Panda, B. Das, B. Subudhi, and B. B. Pati, "A comprehensive review of path planning algorithms for autonomous underwater vehicles," *International Journal of Automation and Computing*, vol. 17, no. 3, pp. 321–352, 2020, doi: 10.1007/s11633-019-1204-9.
- [11] Ş. Ulus and İ. Eski, "Neural network and fuzzy logic-based hybrid attitude controller designs of a fixed-wing UAV," *Neural Computing and Applications*, vol. 33, no. 14, pp. 8821–8843, 2021, doi: 10.1007/s00521-020-05629-5.
- [12] S. Kundu and D. R. Parhi, "Reactive navigation of underwater mobile robot using ANFIS approach in a manifold manner," *International Journal of Automation and Computing*, vol. 14, no. 3, pp. 307–320, 2017, doi: 10.1007/s11633-016-0983-5.
- [13] D. Erdos and S. E. Watkins, "UAV autopilot integration and testing," *2008 IEEE Region 5 Conference*, Kansas City, MO, USA, 2008, pp. 1–6, doi: 10.1109/TPSD.2008.4562731.
- [14] H. A. D. Oliveira and P. F. F. Rosa, "Adaptive genetic neuro-fuzzy attitude control for a fixed wing UAV," *Proceedings of the IEEE International Conference on Industrial Technology*, pp. 726–731, 2017, doi: 10.1109/ICIT.2017.7915449.
- [15] B. Selma, S. Chouraqui, and H. Abouaïssa, "Optimal trajectory tracking control of unmanned aerial vehicle using ANFIS-IPSO system," *International Journal of Information Technology*, vol. 12, no. 2, pp. 383–395, Mar. 2020, doi: 10.1007/s41870-020-00436-6.
- [16] Q. Chen, D. Zhu, and Z. Liu, "Attitude control of aerial and underwater vehicles using single-input FUZZY P+ID controller," *Applied Ocean Research*, vol. 107, 2021, doi: 10.1016/j.apor.2020.102460.
- [17] Y. Jiang, C. Feng, B. He, J. Guo, D. R. Wang, and P. F. LV, "Actuator fault diagnosis in autonomous underwater vehicle based on neural network," *Sensors and Actuators, A: Physical*, vol. 324, 2021, doi: 10.1016/j.sna.2021.112668.
- [18] J. Gao, A. Proctor, and C. Bradley, "Adaptive neural network visual servo control for dynamic positioning of underwater vehicles," *Neurocomputing*, vol. 167, pp. 604–613, 2015, doi: 10.1016/j.neucom.2015.04.028.
- [19] A. Bagheri, T. Karimi, and N. Amanifard, "Tracking performance control of a cable communicated underwater vehicle using adaptive neural network controllers," *Applied Soft Computing Journal*, vol. 10, no. 3, pp. 908–918, 2010, doi: 10.1016/j.asoc.2009.10.008.





- [20] C. Yu, X. Xiang, F. Maurelli, Q. Zhang, R. Zhao, and G. Xu, "Onboard system of hybrid underwater robotic vehicles: integrated software architecture and control algorithm," *Ocean Engineering*, vol. 187, 2019, doi: 10.1016/j.oceaneng.2019.106121.
- [21] A. Bagheri and J. J. Moghaddam, "Simulation and tracking control based on neural-network strategy and sliding-mode control for underwater remotely operated vehicle," *Neurocomputing*, vol. 72, no. 7–9, pp. 1934–1950, 2009, doi: 10.1016/j.neucom.2008.06.008.
- [22] S. Nurmaini and A. Zarkasi, "Simple pyramid ram-based neural network architecture for localization of swarm robots," *Journal of Information Processing Systems*, vol. 11, no. 3, pp. 370–388, 2015, doi: 10.3745/JIPS.01.0008.
- [23] A. Zarkasi, S. Nurmaini, D. Stiawan, and B. Y. Suprpto, "Weightless neural networks face recognition learning process for binary facial pattern," *Indonesian Journal of Electrical Engineering and Informatics (IJEI)*, vol. 10, no. 4, pp. 955–969, 2022, doi: 10.52549/ijeei.v10i4.3957.
- [24] S. Nurmaini, A. Zarkasi, D. Stiawan, B. Y. Suprpto, S. D. Siswanti, and H. Ubaya, "Robot movement controller based on dynamic facial pattern recognition," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22, no. 2, pp. 733–743, 2021, doi: 10.11591/ijeecs.v22.i2.pp733-743.
- [25] I. Aleksander, M. D. Gregorio, F. M. G. França, P. M. V. Lima, and H. Morton, "A brief introduction to weightless neural systems," *ESANN 2009 Proceedings, 17th European Symposium on Artificial Neural Networks - Advances in Computational Intelligence and Learning*, pp. 299–305, 2009.
- [26] L. Santiago *et al.*, "Weightless neural networks as memory segmented bloom filters," *Neurocomputing*, vol. 416, pp. 292–304, 2020, doi: 10.1016/j.neucom.2020.01.115.
- [27] N. Nedjah, F. M. G. França, M. D. Gregorio, and L. D. M. Mourelle, "Weightless neural systems," *Neurocomputing*, vol. 183, pp. 1–2, 2016, doi: 10.1016/j.neucom.2015.11.064.
- [28] H. A. Hashim, M. Abouheaf, and M. A. Abido, "Geometric stochastic filter with guaranteed performance for autonomous navigation based on IMU and feature sensor fusion," *Control Engineering Practice*, vol. 116, 2021, doi: 10.1016/j.conengprac.2021.104926.
- [29] L. Tang, Y. Fan, F. Ye, and W. Lu, "Estimation of IMU orientation using nesterov's accelerated gradient improved by fuzzy control rule," *Sensors and Actuators A: Physical*, vol. 332, 2021, doi: 10.1016/j.sna.2021.113062.
- [30] A. Prato, F. Mazzoleni, G. D'Emilia, A. Gaspari, E. Natale, and A. Schiavi, "Metrological traceability of a digital 3-axis MEMS accelerometers sensor network," *Measurement: Journal of the International Measurement Confederation*, vol. 184, 2021, doi: 10.1016/j.measurement.2021.109925.
- [31] A. Umek and A. Kos, "Validation of MEMS accelerometer for rapid hand movement measurement," *Procedia Computer Science*, vol. 187, pp. 530–537, 2021, doi: 10.1016/j.procs.2021.04.095.
- [32] S. Nurmaini, S. Z. M. Hashim, and D. N. A. Jawawi, "Modular weightless neural network architecture for intelligent navigation," *International Journal of Advances in Soft Computing and its Applications*, vol. 1, no. 1, pp. 1–18, 2009.
- [33] R. Barbosa, D. O. Cardoso, D. Carvalho, and F. M. G. França, "Weightless neuro-symbolic GPS trajectory classification," *Neurocomputing*, vol. 298, pp. 100–108, 2018, doi: 10.1016/j.neucom.2017.11.075.
- [34] J. S. -Guerrero, F. P. Romero, and J. A. Olivas, "Fuzzy logic applied to opinion mining: a review," *Knowledge-Based Systems*, vol. 222, 2021, doi: 10.1016/j.knosys.2021.107018.
- [35] J. Zhang, X. Xiang, L. Lapierre, Q. Zhang, and W. Li, "Approach-angle-based three-dimensional indirect adaptive fuzzy path following of under-actuated AUV with input saturation," *Applied Ocean Research*, vol. 107, 2021, doi: 10.1016/j.apor.2020.102486.
- [36] Husnawati, G. F. Fitriana, and S. Nurmaini, "The development of hybrid methods in simple swarm robots for gas leak localization," *Proceedings - International Conference on Signals and Systems, ICSigSys 2017*, pp. 197–202, 2017, doi: 10.1109/ICSIGSYS.2017.7967040.

BIOGRAPHIES OF AUTHORS







Dr. Ahmad Zarkasi, M.T.     was born in Palembang on August 25, 1979. In 2023, he obtained his Doctor in Informatics Engineering from the University of Sriwijaya, Faculty of Engineering. His research interests include microprocessors, system-on-chip (SoC), embedded systems, and robotics. They include subjects such as WNNs in robotic systems and pattern recognition for robot vision. Currently, he is a lecturer in Department of Computer Engineering, Faculty of Computer Science, University of Sriwijaya, Indonesia. He can be contacted at email: zarkasi98@gmail.com.







Hadipurnawan Satria, Ph.D.     was born in Palembang on April 18, 1980. Currently, he is a lecturer in Department of Informatics engineering, Faculty of Computer Science, University of Sriwijaya, Indonesia. He received her Ph.D. in Computer from Sun Moon University's Faculty of Computer Science in 2010. His research interests include A Virtual Development Environment for Embedded Software (VDEES). He can be contacted at email: hadipurnawan.satria@gmail.com.







Anggina Pramanita, Ph.D.     born in Bekasi, August 6, 1989. Currently, she is a lecturer in Department of Informatics engineering, Faculty of Computer Science, University of Sriwijaya, Indonesia She earned a Ph.D. from the Japan Advanced Institute of Science and Technology at the Faculty of Information Science in 2021. Her research interests are in computing games. She can be contacted at email: anggina.primanita@gmail.com.







Abdurahman     was born in Palembang. Oct. 22, 1994. In 2020, he graduated with a Master's degree from the University of Defense's Faculty of Defence Technology. His scientific interests are in remote sensing. He is currently a computer systems lecturer in the Faculty of Computer Science at the University of Sriwijaya in Indonesia. He can be contacted at email: abdurahmanfaqod@gmail.com.



Nurul Afifah, M. Kom.     was born in Palembang on November 10, 1992. She received his Master of Science in Computer Science from the University of Sriwijaya in 2019. Her research interests include computer networking and artificial intelligence. She is currently a computer systems lecturer in the Faculty of Computer Science at the University of Sriwijaya in Indonesia. She can be contacted at email: nurulafifah@ilkom.unsri.ac.id.



Sutarno     was born in Gunung Raya. Nov. 01, 1978. In 2005, he graduated with a Master's degree from the University of Gadjah Mada of Technical Faculty, Department of Electrical Engineering. His scientific interests are in artificial intelligence and image processing. He is currently a computer systems lecturer in the Faculty of Computer Science at the University of Sriwijaya in Indonesia. He can be contacted at email: sutarno@unsri.ac.id.