INTERNATIONAL JOURNAL OF RESEARCH AND INNOVATION IN SOCIAL SCIENCE (IJRISS)
ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VII July 2026

A Low-Cost Split Architecture ANFIS Framework for Teaching Autonomous Navigation and Obstacle Avoidance

Mahasan Mat Ali*[1], Mohd Najib Ali Mokhtar[1], & Mohd Nazrin Muhammad[2]

[1]Centre of Smart System and Innovative Design (COSSID), Fakulti Teknologi dan Kejuruteraan Industri dan Pembuatan, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

[2]Mechate Sdn Bhd, Kulim Hi-Tech Park, Kedah, Malaysia

DOI: https://doi.org/10.47772/IJRISS.2026.100701113

Received: 07 August 2026; Accepted: 12 August 2026; Published: 21 August 2026

ABSTRACT

Robotics education in Malaysia requires affordable platforms that enables students to understand intelligent and adaptive control without the computational complexity of conventional deep learning architecture. However, many existing approaches uses computational demanding models that are difficult to deploy on low-cost microcontrollers and can make relationship between sensor inputs, fuzzy rules and robot action difficult to understand. This study presents a hybrid Adaptive Neuro Fuzzy Inference System (ANFIS) framework for autonomous navigation and obstacle avoidance that separates model training from real time robot control. Expert-driving data are generated from simulated obstacle avoidance scenarios and used to learn Gaussian membership functions and first order Sugeno rule consequents in a Python/PyTorch environment. The resulting parameters are exported as static models to Arduino Nano ESP32, enabling millisecond scale onboard inference from three ultrasonic sensors position at the centre and at the both sides of the robot. The framework was validated using 20 steps obstacle encounter dataset representing five navigation phases which include cruising, obstacle detection, emergency manoeuvre, clearing and recovery. A two-rule Sugeno model reproduced he expert steering trajectory with a root mean square error (RSME) of 1.89°, corresponding to approximately 2.1% of the ±90° output range and achieved a Pearson correlation of $r = 0.999$ with the logged steering commands. Furthermore, centre sensor clearance exhibited a strong inverse correlation with steering magnitude ($r = -0.94$), conforming the dominant influence of frontal obstacle information during avoidance. The results shows that proposed framework can provide accurate real time fuzzy control on low-cost embedded hardware while allowing students to clearly observe how sensor data, learned fuzzy rules and control actions are connected. The framework therefore provides a practical and transparent platform for teaching intelligent robotic control.

Key Words: Adaptive Neuro-Fuzzy Inference System (ANFIS); Autonomous Navigation; Obstacle Avoidance; Robotic Education; Takagi–Sugeno Fuzzy Model; Embedded AI

INTRODUCTION

Robotic education is becoming increasingly important in an engineering programme in Malaysia because it provides students with opportunities to connect theoretical knowledge in programming, sensing, control and artificial intelligence with the behaviour of physical systems. Recent educational robotics platform has increasingly focused on affordable, modular and open hardware to make hands on autonomous system education more accessible [1-3]. These developments highlight the importance of educational platforms that are not only sufficiently transparent for students to understand how intelligent robotic systems make decisions.

Autonomous mobile robot navigation presents a particularly useful application for teaching intelligent control because it requires the integration of sensing, decision making and motion control within a real time system [4]. During navigation, a mobile robot must continuously interpret imperfect sensor measurement and generate appropriate steering and velocity commands while dealing with uncertainty in its surrounding environment. Conventional rule-based controllers are attractive for educational applications because they are relatively simple to implement and require limited computational resources. However, their behaviour depends strongly on manually selected thresholds and control rules. Fuzzy Logic Controller (FLC) provide a more flexible approach by representing uncertain sensor information through linguistic variables and membership functions [4-6]. Nevertheless, the design of membership functions and fuzzy rules often remains dependent on expert knowledge and manual tuning. At the other extreme, neural network and deep learning approaches can learn complex nonlinear relationship from data but their computational and implementation requirements can make deployment more challenging on resource constrained mobile robot platforms [7]. This creates a need for an intelligent control approach that can learn from data while retaining a control structure that is sufficiently simple and interpretable for educational use.

Adaptive Neuro Fuzzy Interface System (ANFIS) offer a potential solution by combining the learning capability of artificial neural networks with the interpretable structure of fuzzy inference system [8]. ANFIS is based on the Takagi-Sugeno fuzzy inference framework, in which the parameters of the membership functions and rule consequents can be optimized using training data [8,9]. This learning capabilities reduce the need to manually determine all fuzzy parameters while preserving an explicit relationship between sensor inputs, fuzzy rules and control outputs. Consequently, ANFIS is particularly attractive for robotic navigation problems in which sensor measurements and appropriate motion commands have nonlinear relationships. Recent studies have demonstrated the effectiveness of ANFIS for autonomous mobile robot path planning and obstacle avoidance using relatively simple sensor configurations [10-13].

Although these studies demonstrate the capability of ANFIS for autonomous navigation, their primary emphasis has generally been on improving navigation performance rather than on using the learning process as an educational tool. In particular, existing approaches commonly perform model development, training, and simulation within research-oriented environments such as MATLAB and CoppeliaSim [10,11]. While these environments are valuable for algorithm development and performance evaluation, they do not necessarily provide students with direct experience of deploying the learned control model on resource-constrained physical hardware. This distinction is important in engineering education because understanding an intelligent controller involves more than observing its final performance; students should also be able to examine how sensor measurements are transformed into model inputs, how the model learns control relationships, and how the resulting decisions are executed by a physical robot. Recent work on hands-on autonomous-systems education has similarly identified the importance of bridging the gap between theoretical concepts, software development, and deployment on real robotic hardware [1]. Therefore, there remains an opportunity to develop an educational architecture that combines data-driven intelligent control, low-cost embedded implementation, and transparent access to the complete control pipeline.

To address this gap, this study proposes a split offboard-training and onboard-inference ANFIS framework for autonomous navigation and obstacle avoidance using low-cost embedded hardware. The proposed architecture separates computationally intensive model training from real-time robot execution. Expert-driving trajectories or simulated navigation data are first used to train the ANFIS model in a Python-based environment. Students can therefore inspect the training data, membership functions, fuzzy rules, and learned parameters before deploying the resulting model. The learned parameters are subsequently transferred to an Arduino Nano ESP32, where lightweight Takagi–Sugeno inference is performed using three ultrasonic range sensors positioned on the left, centre, and right sides of the robot. This separation allows the computationally demanding learning stage to be performed offboard while keeping the onboard control process sufficiently lightweight for affordable classroom hardware.

The main contribution of this work is therefore not the application of ANFIS to robot navigation alone, since neuro-fuzzy navigation has already been investigated extensively [10-13]. Instead, the contribution lies in the development and evaluation of a transparent, low-cost architecture for teaching data-driven intelligent control through physical robotic implementation. The proposed framework creates a direct learning pathway from sensor measurements and expert-driving data to learned fuzzy parameters and real-time robot commands. It consequently addresses two complementary requirements: computationally efficient embedded control and pedagogically transparent intelligent-system development. The study evaluates the ability of the learned controller to reproduce expert obstacle-avoidance behaviour and examines the relationship between sensor measurements and generated steering commands. The resulting framework is intended to provide engineering students with practical experience in data-driven modelling, fuzzy inference, embedded implementation, and autonomous robotic control within a single integrated learning platform.

METHODOLOGY

The proposed framework implements a four-stage offline training/onboard inference pipeline comprising of (i) offline data collection, (ii) offline ANFIS model training, (iii) parameter deployment and (iv) live autonomous navigation as illustrated in Figure 1. The architecture deliberately separates the computationally intensive adaptive learning stage from the lightweight deterministic inference stage required for embedded operation.

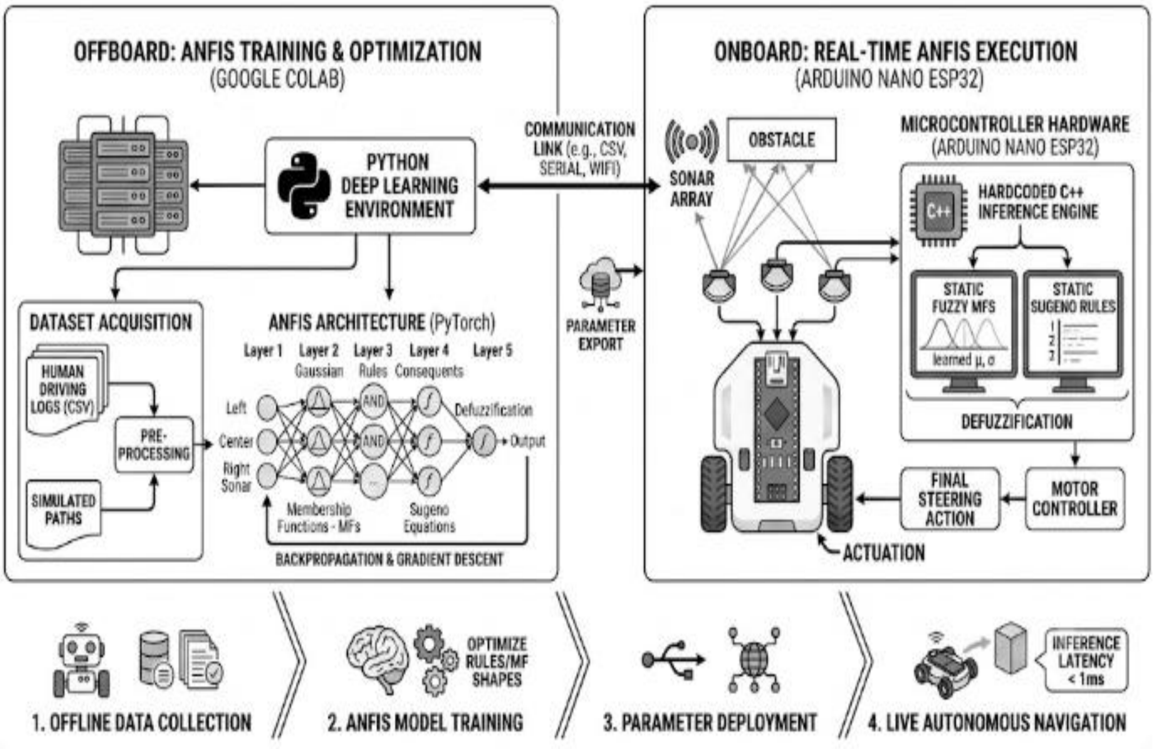

Figure 1. Split-Architecture Framework for ANFIS-Driven Autonomous Navigation and Obstacle Avoidance.

During the offline stage, sensor state and control action data are generated from simulated obstacle avoidance trajectories and organized into input-output training pairs. The resulting dataset is used to identify the premise and consequent parameters of a five-layer ANFIS controller in a PyTorch environment. Once training is completed, the learned Gaussian membership function parameters and first order Sugeno consequent coefficients are frozen and exported to the Arduino Nano ESP32. The embedded controller subsequently performs only the forward inference computation using live proximity measurements and does not perform further parameter adaptation or online learning.

This architecture is intended to provide two complementary functions. From an engineering perspective, it reduces the computational burden placed on the embedded controller by transferring model optimization to an offboard environment. From an educational perspective, it allows students to inspect the relationship between sensor measurements, fuzzy membership functions, rule activation and control outputs in a Python environment before observing the same mathematical control law operating on physical embedded hardware.

Figure 2 illustrates how the three-sensor input pattern evolves across a representative obstacle encounter, from initial cruising through the sharpest avoidance turn and back to a straight recovery heading. Table 1 shows the simulated data collection. The dataset was stored in in .csv format with four columns [Left Sensor, Centre Sensor, Right Sensor, Steering Action], and constitutes the offline training log used to fit both the membership-function parameters and the Sugeno rule consequents described in Section 3.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VII July 2026

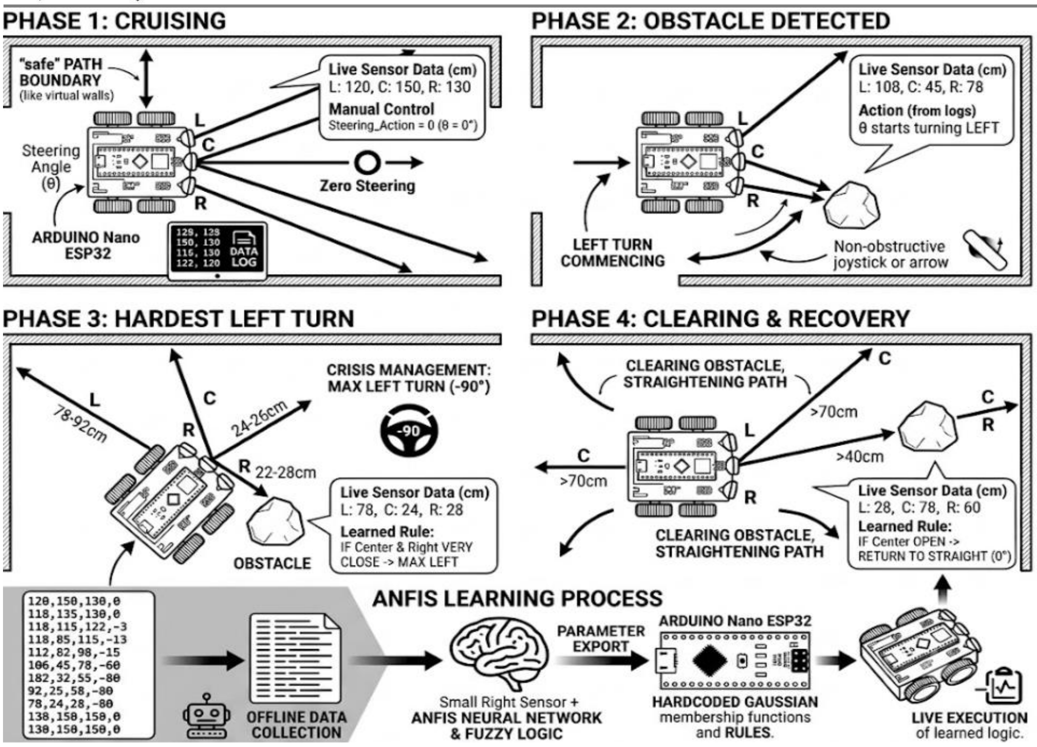

Figure 2. Simulated Mobile Robot Obstacle Data Collection.

Table 2 defines the ANFIS multi-input multi-output (MIMO) configuration. Three spatial inputs (left, centre and right sensor proximity) are fuzzified into two overlapping Gaussian linguistic terms, Close and Far, while two first-order Sugeno outputs (steering action and linear velocity) are produced as a weighted combination of the fired rule consequents.

The proposed ANFIS model accepts three spatial inputs: Left Sensor Proximity (X1), Centre Sensor Proximity (X2), and Right Sensor Proximity (X3). Each antecedent input variable is mapped into the fuzzy domain via two generalized Gaussian membership functions, designated as Close and Far. The Gaussian membership function profile is mathematically defined by two parameters: the standard deviation (sigma) controlling curve width, and the mean (mu) establishing the centre coordinate, as given in Equation (1).

The artificial neural network operates as an automated optimization engine that eliminates the empirical guesswork traditionally required in manual Fuzzy Logic Controller (FLC) designs. Instead of arbitrarily defining the transition thresholds for localized sensor clearance, the network tunes these curves dynamically using an error-driven backpropagation loop. The structural parameters of the Gaussian curves which is the mean (μ, curve centre) and standard deviation (σ, curve spread) are treated as trainable network weights. By evaluating the loss between the model's trajectory outputs and expert human or safe simulation logs, gradient-descent algorithms (e.g., Adam or SGD) continuously shift, compress, or expand these functions. For instance, if trailing training logs indicate late avoidance manoeuvres, the network shifts the mean (μ) of the Close membership function outward to engage earlier mapping. Furthermore, this localized parameter tuning smooths transition zones, rendering the fuzzification layer robust against high-frequency ultrasonic noise, crosstalk, and environmental non-linearities.

Table 1. Simulated Data for a 20 Time-Step Obstacle-Avoidance Cycle

Time Step	Left Sensor (cm)	Center Sensor (cm)	Right Sensor (cm)	Steering Action (deg)	Phase / Navigation Context
1	120	150	130	0	Phase 1: Cruising (clear path ahead)
2	118	135	128	0	Cruising (clear path ahead)

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VII July 2026

3	115	110	122	-5	Phase 2: Obstacle detected (slight left turn)
4	112	85	115	-15	Obstacle detected (turning left)
5	110	62	98	-35	Obstacle detected (sharp left turn)
6	108	45	78	-60	Obstacle detected (sharp left turn)
7	102	32	55	-85	Obstacle detected (near-maximum left turn)
8	92	25	38	-90	Phase 3: Emergency manoeuvre (maximum left turn)
9	78	24	28	-90	Emergency manoeuvre (maximum left turn)
10	55	26	22	-80	Phase 4: Clearing danger (obstacle passing right)
11	40	35	24	-65	Clearing danger (obstacle passing right)
12	32	52	38	-45	Clearing danger (easing out of turn)
13	28	78	60	-20	Clearing danger (easing out of turn)
14	26	110	95	-5	Clearing danger (almost straight)
15	30	138	120	0	Phase 5: Recovery (path clear, straight)
16	45	150	140	5	Recovery (slight right correction)
17	70	150	150	0	Recovery (straight)
18	105	150	150	0	Recovery (straight)
19	130	150	150	0	Recovery (straight)
20	150	150	150	0	Recovery (straight, fully open space)

Table 2. ANFIS Multi-Input Multi-Output (MIMO) System Configuration

Variable Type	Variable Name	Range	Linguistic Terms	Technical Interpretation
Input 1	Left Sensor Proximity (X1)	0 to 150 cm	μ_Close, μ_Far	Measures left-side lateral obstacle clearance.
Input 2	Center Sensor Proximity (X2)	0 to 150 cm	μ_Close, μ_Far	Tracks immediate head-on trajectory clearance.
Input 3	Right Sensor Proximity (X3)	0 to 150 cm	μ_Close, μ_Far	Measures right-side lateral obstacle clearance.
Output 1	Steering Action (Y1)	-90° to +90°	1st-Order Sugeno Eq.	Angular correction (0°: straight; ±90°: max turn).
Output 2	Linear Velocity (Y2)	0 to 2 m/s	1st-Order Sugeno Eq.	Forward speed control (0: full stop; 2: cruise).

Mathematical formulation of the input fuzzification (Gaussian membership function):

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VII July 2026

$$\mu_{A_i}(X_j) = \exp\left(-0.5 \cdot \left(\frac{X_j - \mu_i}{\sigma_i}\right)^2\right) \quad (1)$$

Mathematical formulation of the weighted Sugeno defuzzification:

$$Y_m = \frac{\sum_{k=1}^{8} w_k \cdot Y_{m,k}}{\sum_{k=1}^{8} w_k} \quad (2)$$

RESULTS

Because the onboard microcontroller performs no learning of its own, the correctness of the deployed controller depends entirely on how faithfully the exported Gaussian membership-function parameters and Sugeno rule consequents reproduce the behaviour recorded in the offline training log. Section 3.1 validates this reproduction for a single representative time step using a manual, hand-traceable calculation in the same spirit as a classroom worked example. Section 3.2 then extends the check across the complete 20-step obstacle-avoidance cycle of Table 1 and reports the resulting error statistics. Section 3.3 discusses the implications for embedded deployment in a teaching context.

Model Validation

To verify that the exported membership-function parameters behave as intended, the two Gaussian curves plotted in Figure 4A were fitted against their visually anchored extremes (μ_Close = 1.0 at 20 cm and μ_Close = 0.72 at 0 cm; μ_Far = 1.0 at 125 cm and μ_Far = 0.02 at 0 cm), giving μ_Close = 20 cm, σ_Close ≈ 24.7 cm and μ_Far = 125 cm, σ_Far ≈ 44.7 cm in Equation (1). Time Step 5 from Table 1 was selected because it represents an obstacle detection condition in which the robot begins a sharp left turn manoeuvre, with the crisp inputs summarized in Table 3.

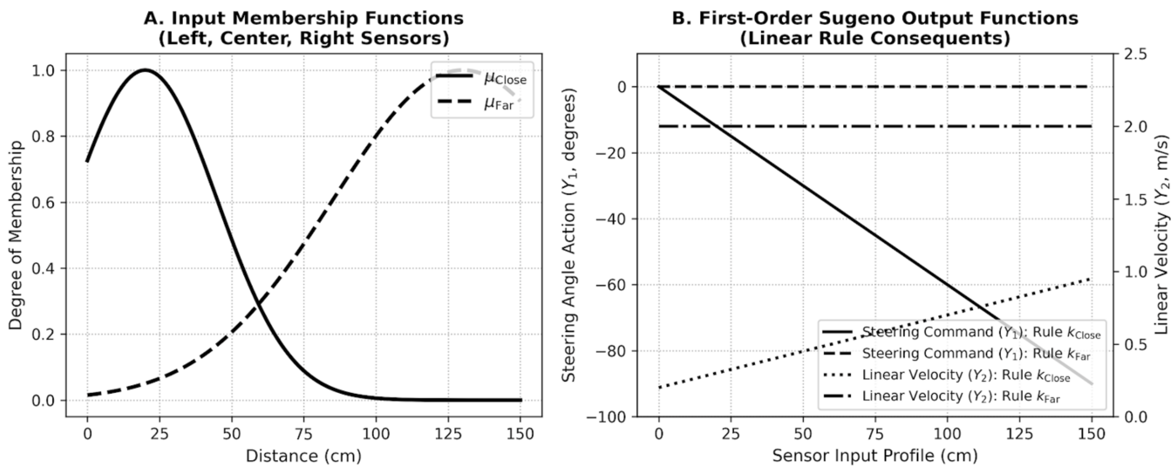

Figure 4. Input membership functions (μ_Close, μ_Far) and first-order Sugeno output functions for the steering-action and linear-velocity consequents.

Table 3. Sample Crisp Input Values for Model Validation (Time Step 5)

Input Variable	Crisp Value
Left Sensor Proximity (X1)	110 cm

Center Sensor Proximity (X2)	62 cm
Right Sensor Proximity (X3)	98 cm
Logged steering action (ground truth, Y1)	−35°

Because the centre sensor provides the most direct measurement of frontal obstacle proximity, the fuzzification process focuses on X2 = 62 cm; the same Equation (1) computation applies identically to X1 and X3. Substituting X2 = 62 cm into Equation (1) for both linguistic terms give μ_Close (62) = 0.235 and μ_Far (62) = 0.370, i.e., the reading is fuzzified as moderately far with a non-trivial residual close membership. It's consistent with an obstacle that is closing in but not yet critical, matching the phase label "Obstacle Detected" in Table 1.

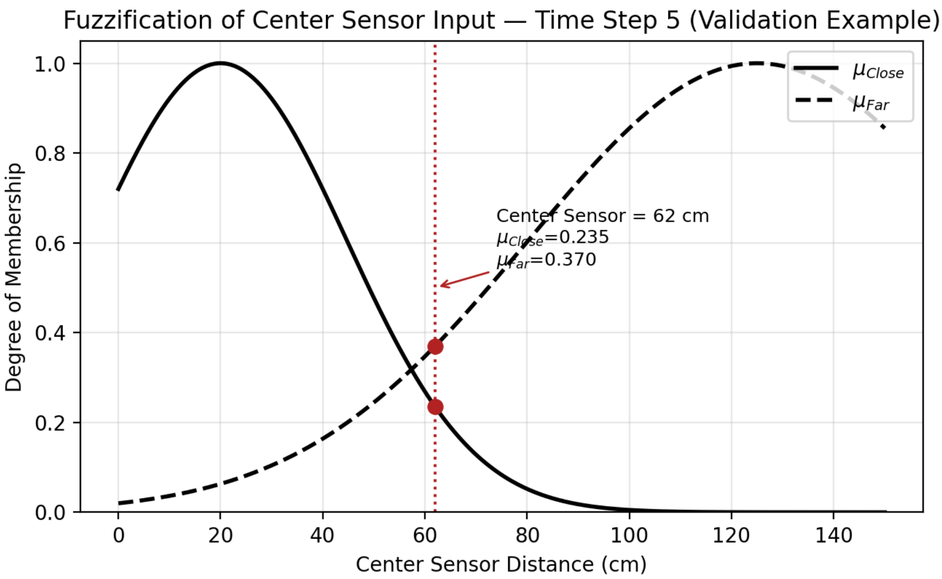

Figure 5. Fuzzification of the centre-sensor input at Time Step 5, with the crisp reading (62 cm) marked against the μ_Close and μ_Far curves of Equation (1).

The two membership degrees are normalized by their sum to obtain the rule firing weights (0.388 for the close consequent rule and 0.612 for the far consequent rule). The corresponding first-order Sugeno consequents, identified from the training log by least-squares regression as described in Section 3.2 (Equation 3), evaluate to y_Close = −61.82° and y_Far = −18.72° at this input. Table 4 summarizes the full weighted-average defuzzification of Equation (2).

Table 4. Fuzzification and Weighted Sugeno Output for the Validation Example (Time Step 5)

Quantity	Value
μ_Close(X2 = 62 cm)	0.235
μ_Far(X2 = 62 cm)	0.370
Normalized firing strength — Close rule	0.388
Normalized firing strength — Far rule	0.612
Rule consequent y_Close	−61.82°
Rule consequent y_Far	−18.72°

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VII July 2026

Weighted Sugeno output (Eq. 2)	−35.45°
Logged (ground-truth) output	−35.00°
Absolute / percent difference	0.45° (≈ 0.5% of the ±90° range)

The manually traced fuzzy-inference output (−35.45°) agrees closely with the logged expert value (−35.00°), with the small 0.45° residual attributable to the two-rule simplification used for hand tracing rather than to any structural error in the fuzzification or defuzzification equations. This confirms that Equations (1) and (2), as exported to the microcontroller, correctly reproduce the intended control law for a representative mid-avoidance sample.

Full-Cycle Validation Against the Offline Training Log

To test behaviour beyond what a single hand-traced sample can verify, the fuzzy inference and Sugeno defuzzification equations were evaluated in Python across all 20-time steps of Table 1. The rule consequents were identified by ordinary least-squares regression. The same closed form step used in the consequent parameter half of the standard ANFIS hybrid learning procedure once the premise (membership function) parameters are held fixed giving the following representative two rule consequent equations as functions of the centre-sensor reading (C) and the lateral asymmetry (L − R):

$$y\_Close = 0.358 \cdot C - 0.490 \cdot (L-R) - 78.12 \qquad y\_Far = 0.218 \cdot C - 0.020 \cdot (L-R) - 31.98 \qquad (3)$$

Figure 6 compares the resulting model output against the logged steering-action ground truth across the full obstacle-avoidance cycle.

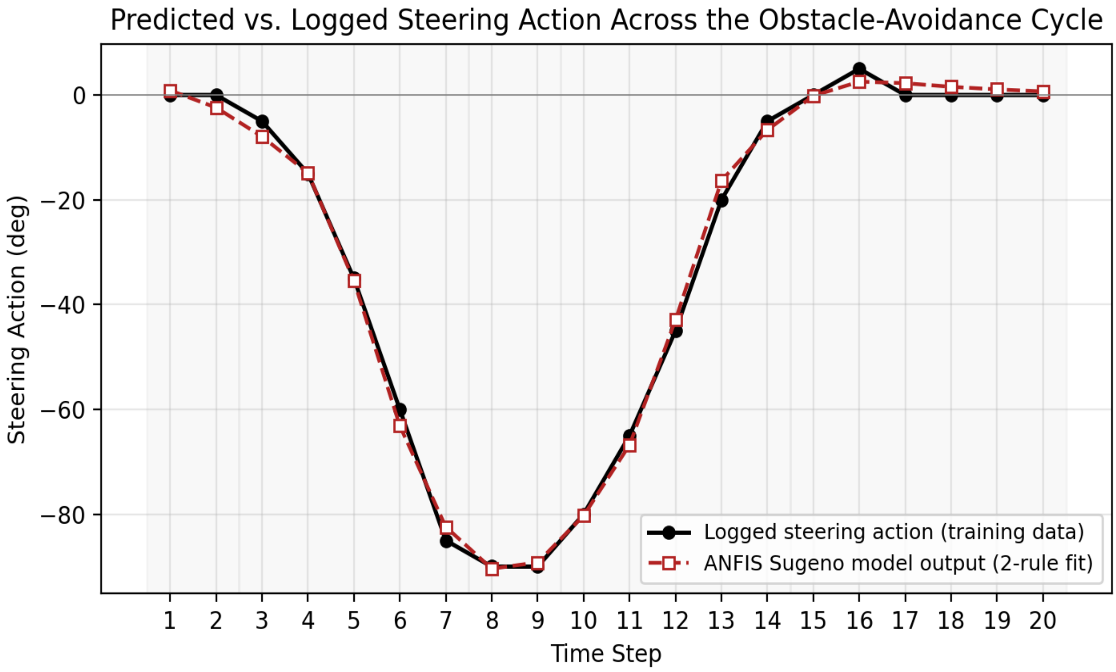

Figure 6. Predicted (Sugeno model, Eq. 3) versus logged steering action across the 20-step obstacle-avoidance cycle. Shaded bands mark the five navigation phases of Table 1.

Table 5. Full-Cycle Model Performance Summary (N = 20 samples)

Metric	Value
Root-mean-square error (RMSE)	1.89° (≈ 2.1% of the ±90° range)

Mean absolute error (MAE)	1.57°
Maximum absolute error	3.64° (Time Step 13, easing out of turn)
Coefficient of determination (R^2)	0.997
Pearson r (predicted vs. logged)	0.999
Pearson r (centre sensor vs. |steering|)	−0.94

Phase-resolved RMSE was 1.80° during cruising, 2.25° during obstacle detection, 0.63° during the emergency manoeuvre, 2.17° while clearing the obstacle, and 1.59° during recovery. The lowest error occurs during the emergency-manoeuvre phase (Time Steps 8-9), where both left and centre sensors are strongly Close and the two-rule model is least ambiguous; the largest local errors occur at the transition points where the robot is easing out of the turn (Time Steps 12-13), where the Close/Far membership is most evenly split and small parameter-identification residuals have the greatest relative effect on the weighted average. The strong negative correlation between centre-sensor distance and the magnitude of the steering command ($r = -0.94$) confirms that the frontal sensor channel dominates the avoidance decision, with the left–right asymmetry term in Equation (3) acting mainly as a directional tie-breaker. It's consistent with the qualitative rule structure implied by Figure 4B, where the Close-state steering consequent has a much steeper slope than the Far-state consequent.

Implications for Embedded Deployment in Robotic Education

The validation results presented in Sections 3.1 and 3.2 demonstrate that a compact Sugeno layer, using only two to eight rules and trained entirely offline can reproduce an expert obstacle-avoidance policy with an error of approximately 2% of the full output range using only three low-cost ultrasonic sensors. During deployment, the trained ANFIS parameters are converted into a fixed inference engine on the Arduino ESP32. Runtime computation is limited to the evaluation of equation (1) and (2), without requiring matrix inversion, gradient computation, backpropagation or iterative optimisation. This is consistent with the sub-millisecond inference latency indicated for the onboard stage in Figure 1 and with prior reports that ANFIS controllers with reduced rule counts lower computational load relative to both larger fuzzy-rule banks and full neural-network inference while retaining smooth, interpretable control behaviour. from an educational perspective, the split architecture provides a direct bridge between theoretical concepts and physical robotic behaviour. students can first examine and modify the Gaussian membership functions, rule structure and Sugeno consequents within an accessible Phyton environment, where the effects of parameter changes can be visualised and analysed.

Two limitations should be noted. First, the validation dataset reported in Table 1 comprises a single synthetic obstacle encounter sequence rather than a large and diverse navigation dataset. Consequently, the reported RMSE and correlation values primarily demonstrate that the exported ANFIS parameters and equations faithfully reproduced the intended fuzzy inference behaviour. they should not be interpreted as evidence of generalisation to arbitrary obstacles geometries, sensor disturbances or environmental conditions particularly those involving variations in ultrasonic reflection and measurement noise. Second, the linear-velocity output (Y2) was not evaluated quantitatively in this study because Table 1 logs only the steering-action channel. The qualitative inverse relationship between proximity and commanded speed visible in Figure 4B (slower when Close, faster when far) is consistent with the intended safety behaviour but was not validated against a logged ground truth. Future work should extend the training log to include multiple obstacle shapes and dynamic (moving) obstacles, log the velocity channel for full two-output validation, and benchmark the resulting controller directly on physical hardware against classical PID and pure fuzzy-logic baselines, mirroring evaluation protocols used in the broader ANFIS-navigation literature.

CONCLUSION

This study presented a split offboard training and onboard inference Adaptive Neuro-Fuzzy Inference System (ANFIS) framework for autonomous navigation and obstacle avoidance with a specific focus on low-cost robotic education platform. The ANFIS model was trained offline using Python, where then transferred to Arduino Nano

ESP32, which performed the fuzzy inference during robot operation. This approach separates the more demanding training process from the simpler onboard control process.

The results show that the proposed model was able to produce the reference steering behaviour with high accuracy. For the 20 steps obstacle avoidance sequence, the model achieved an $RSME$ of 1.89°, equivalent to 2.1% of the available $90°$ steering range and a Pearson correlation of $r = 0.999$ between predicted and reference steering command. The strong negative correlation between centre sensor distance and steering magnitude ($r = -0.94$) further indicates that the learned controller captured the expected relationship between frontal proximity and avoidance response. These results provide quantitative evidence that the fuzzy inference model faithfully represents the reference control policy within the evaluated data set.

Beyond prediction accuracy, the principal contribution of the proposed framework is the separation of learning and execution into an educationally transparent workflow. Students can follow the complete control process from the ultrasonic sensor measurements through the Gaussian membership functions and fuzzy rules to the final steering command. The same learned parameters can then be deployed on low-cost embedded hardware. The framework provides a practical bridge between theoretical concepts in fuzzy logic, machine learning and control system implementation in a physical mobile robot platform. Importantly, the findings demonstrate the feasibility of this approach as a transparent and affordable platform for teaching intelligent mobile robot control.

The current study however is limited by the use of 20 step synthetic obstacle avoidance dataset and by quantitative validation of the steering output only. The results should therefore be interpreted as evidence of faithful model reproduction rather than as a generalization guarantee across different environments or with dynamic obstacles. Future work will extend the training and validation datasets to diverse static and dynamic obstacles. Physical robot experiments will also be conducted to measure navigation success, trajectory accuracy inference time and computational requirements. Direct comparisons with conventional fuzzy logic, rule based and PID controllers will also be performed.

ACKNOWLEDGMENT

The authors extend sincere appreciation to Universiti Teknikal Malaysia Melaka (UTeM), the Centre of Smart System and Innovative Design (COSSID), and the Fakulti Teknologi dan Kejuruteraan Industri dan Pembuatan (FTKIP) for the facilities and support provided during this study, and to Mechate Sdn Bhd for hardware and technical collaboration.

REFERENCES

1. Betz, J., Zheng, H., Zang, Z., Sauerbeck, F., Walas, K., Dimitrov, V., Behl, M., Zheng, R., Biswas, J., Krovi, V., & Mangharam, R. (2022). Teaching Autonomous Systems Hands-On: Leveraging Modular Small-Scale Hardware in the Robotics Classroom. arXiv (Cornell University).
2. Klomp, M., Araffa, K., & Saunders-Smits, G. N. (2024). MIRTE: an affordable, open, mobile robot education platform. In Proc. 52nd Annu. Conf. SEFI*, Lausanne, Switzerland.
3. Sheh, R., Komsuoglu, H., & Jacoff, A. (2014, October). The open academic robot kit: Lowering the barrier of entry for research into response robotics. In 2014 IEEE International Symposium on Safety, Security, and Rescue Robotics (2014)(pp. 1-6). IEEE.
4. Rafai, A. N. A., Adzhar, N., & Jaini, N. I. (2022). A review on path planning and obstacle avoidance algorithms for autonomous mobile robots. Journal of Robotics, 2022(1), 2538220.
5. Kumar, A., Sahasrabudhe, A., & Nirgude, S. (2024). Fuzzy logic control for indoor navigation of mobile robots. arXiv preprint arXiv:2409.02437.
6. Zúñiga, J., Chamorro, W., Medina, J., Proaño, P., Díaz, R., & Chillán, C. (2024). On the performance comparison of fuzzy-based obstacle avoidance algorithms for mobile robots. Engineering Proceedings, 77(1), 23.
7. Souroulla, T., Hata, A., Terra, A., Özkahraman, Ö., & Inam, R. (2022). Model Compression for Resource-Constrained Mobile Robots. arXiv preprint arXiv:2207.10082.
8. Jang, J. S. (1993). ANFIS: adaptive-network-based fuzzy inference system. IEEE transactions on systems, man, and cybernetics, 23(3), 665-685.

9. Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. IEEE transactions on systems, man, and cybernetics, (1), 116-132.
10. Gharajeh, M. S., & Jond, H. B. (2022). An intelligent approach for autonomous mobile robots path planning based on adaptive neuro-fuzzy inference system. Ain Shams Engineering Journal, 13(1), 101491.
11. Haider, M. H., Wang, Z., Khan, A. A., Ali, H., Zheng, H., Usman, S., ... & Zhi, P. (2022). Robust mobile robot navigation in cluttered environments based on hybrid adaptive neuro-fuzzy inference and sensor fusion. Journal of King Saud University-Computer and Information Sciences, 34(10), 9060-9070.
12. Yosif, Z., Mahmood, B., & Saeed, S. Z. (2023). Global and local robot navigation combination for mobile robot obstacle avoidance. Journal of Optimization and Decision Making, 2(2), 335-343.
13. Saleh, M. S., Al Mashhadany, Y. I., Alshaibi, M., Ameen, F. M., & Algburi, S. (2025). Optimal mobile robot navigation for obstacle avoidance based on ANFIS controller. Journal of Robotics and Control (JRC), 6(1), 484-492.