https://doi.org/10.31449/inf.v50i10.12810 Informatica **50** (2026) 47–60

Intelligent Walrus Optimizer Fused Feedforward Neural Network (IntWO-FFNet) for Embedded Perception and Decision-Making in Industrial Robots

Xia Zhang
Yunnan Open University (Yunnan Technical College of National Defense Industry), Kunming, Yunnan, 650504, China
E-mail: XiaZhang56@outlook.com

Keywords: Decision-making, industrial robots, Intelligent Walrus Optimizer Fused Feed-forward Neural Network (IntWO-FFNet), perception.

Received: October 31, 2025

Intelligent industrial robots rely significantly on accurate vision and autonomous decision-making to do high-performance tasks. Embedded systems, compact, real-time computer units, have become critical for delivering these capabilities, especially in resource-constrained industrial environments. Despite their advantages, embedded systems meet obstacles such as high computational cost, overfitting, and inadequate parameter tuning, which impede real-time performance and generalizability in dynamic industrial environments. The purpose of this research is to develop an embedded neural network framework that has been tuned using metaheuristic algorithms to increase the precision of robotic vision and the effectiveness of decision-making while considering available resources. Multimodal data (vision, force, and proximity) are acquired from industrial environments. Raw data is cleaned and normalized using min-max scaling. Principal Component Analysis (PCA) is used to extract statistical and geographical characteristics, reducing dimensionality. This research proposes an Intelligent Walrus Optimizer Fused Feed-forward Neural Network (IntWO-FFNet) to enhance the accuracy, efficiency, and adaptability of industrial robots by enabling intelligent perception and decision-making on embedded systems. An FFNet is used on the embedded system to identify environmental inputs and forecast task-specific behaviors. The FFNet is fine-tuned with IntWO to improve learning rates, weight initialization, and hidden layer configurations for less error and faster convergence. The proposed method was implemented using Python 3.10.1. The proposed IntWO-FFNet approach performs better than multimodal baseline architectures, achieving superior results, with accuracy ranging from 95% to 99%. Integrating neural networks with optimization approaches into embedded systems dramatically improves real-time robotic perception and decision-making, providing intelligent automation aligned with industrial robots. The dataset contains 10,214 multimodal samples across five task classes (pick, place, weld, idle, interaction) and five industrial object types. All experiments were executed on a simulated embedded environment (Raspberry-Pi–equivalent ARM setup) using Python 3.10.1, with evaluation performed using stratified train–validation–test splits. Performance was benchmarked against the BIIRCS baseline model, where the proposed IntWO-FFNet achieved 96.3% accuracy, 9.8% RMSE, and 97.5% task-coverage rate.

Povzetek: Raziskava predstavlja izboljšan vgradni nevronski sistem, ki poveča natančnost in učinkovitost industrijskih robotov pri zaznavanju in odločanju.

1 Introduction

The embedded systems have fundamentally revolutionized the field of industrial automation, producing robots with pre-programmed equipment for making adaptive decisions [1]. Modern industrial robots rely on embedded systems that feature dedicated functionality, real-time capabilities, and low power consumption. Intelligent perception makes an industrial robot sensitive to reality, interpretative of its sensations, and dynamically responsive to industrial situations that were unstructured and complicated [2]. The incorporation of powerful embedded vision systems has led to industrial robots harnessing the capabilities of cameras, LiDAR, ultrasonic sensors, and proximity sensors to gather multi-modal information by embedded processors using advanced algorithms to identify objects and detect obstacles [3]. Embedded systems integrated the industrial robots to deal with energy efficiency, thermal control, and cyber-physical security for highly demanding tasks [4]. Robots utilize low-power embedded processors, advanced power management schemes, and hardware enforcement of Artificial Intelligence (AI) workloads to ensure high performance without excessively consuming electricity and other operational expenses. The incorporation of secure modules and encrypted protocols can protect the integrity of data, safeguard against unauthorized interference, and address the emerging concerns of industrial surveillance and cyber-capable systems in the manufacturing enterprise [5]. The Industrial Internet of Things (IIoT) and other embedded networking developments have made it possible to use huge volumes of sensor data by industrial robots to enable predictive maintenance and maximum performance of the robots [6].

Embedded systems and their integration with cloud-based analytics and digital twins expand the capabilities of industrial robots by offering a feedback loop of real-world data that can continually update virtual models used for simulation and proactive decision support [7]. Embedded systems make it possible to deploy lightweight, high-performance AI and machine learning (ML) models to the robot, which enables real-time analysis of sensory data without connecting to remote servers or a cloud vendor infrastructure [8]. Embedded systems played an essential role in the decision-making performance of the industrial robot because they enable contextual interpretation of data, autonomous route planning, task scheduling, and dynamic control of robots. The real-time operating systems (RTOS) of embedded systems and edge computing were integrated into robots for computing power, planning, and scheduling activities [9]. Embedded systems make communication and interoperability between robots and other cyber-physical systems of the industrial ecosystem through regularities like Ethernet, Profinet, wireless standards, and Wi-Fi to conduct distributed intelligence and collaborative decision-making in the smart production cells [10]. Embedded-based AI applications were difficult to monitor, debug, and update after deployment at industrial locations.

The objective of this research is to create a revolutionary Intelligent Walrus Optimizer Fused Feed-forward Neural Network (IntWO-FFNet) approach to enhance the perception accuracy and decision-making efficiency of industrial robots. The suggested approach is used to optimize robotic perception and autonomous decision-making on real-time performance within resource-constrained embedded systems. The key contributions of this research are as follows,

- **Dataset Collection:** An Industrial robot sensor dataset was collected from Kaggle. Robots carry out different tasks, including placing, picking, and welding, while interacting with various mechanical components.

- **Data Pre-Processing and Feature Extraction:** Min-max scaling was used to standardize and clean raw data. To reduce dimensionality features by extracting PCA.

- **Optimized Classification Model:** The IntWO-FFNet approach was developed to optimize learning rates, weight initialization, and network configurations for accurate and efficient task prediction on robotic embedded systems.

- **Real-Time Results:** The simulation results evaluate the accuracy, RMSE, and coverage task rate for enhancing real-time industrial robotic applications.

The remaining research consists of a literature review on embedded perception and decision-making in industrial robotics, methodology (data pre-processing, feature extraction, and IntWO-FFNet model development), and experimental results highlighting superior metrics. The conclusion emphasizes real-time perception, robust decision-making, and real-time applicability of the IntWO-FFNet model. Although numerous embedded robotic models have been developed, most suffer from high computational overhead, limited multimodal processing capability, or latency constraints that restrict real-time deployment on CPU-only embedded boards. This motivates the need for a lightweight, optimization-driven architecture such as IntWO-FFNet that can deliver high accuracy while satisfying embedded resource limits.

2 Related works

The robots' usage has grown in popularity to handle artificial things. This has led to the creation of the agnostic robotic paradigm (ARP). The ARP was used for the flexible robotic automation system, and resource synchronization for the Industrial Internet of Things (IIoT) [11]. Robotic enterprise classifications' sustainability and efficiency were largely dependent on the accessibility of data with infrastructure. According to the findings, a cloud platform develops an ARP framework used for IIoT support for both the operation and management levels.

The IIoT system was used to examine the difficult failure prediction challenge in process industries that use autonomous and intelligent cyber-physical systems (CPS) [12]. A temporal convolution-based classification model with a decay effect was used for multi-class classification to identify and locate the faults. The network's overall capability was used to reduce the prediction of the cumulative uncertainty reduction network (CURNet). The outcome demonstrated that CURNet outperforms the prediction of fault types. A summary of related works on embedded systems and decision-making of industrial robots was illustrated in Table 1.

Table 1: Summary of literature review on embedded systems and decision-making of industrial robots

Ref	Technology Used	Objective	Result	Challenges
[13]	Deep Learning (DL), Cloud-Edge-Device Collaboration,	Cloud manufacturing framework used to enable smart industrial robots	Supports smart collaborative decision-making with detailed mechanisms	Balancing real-time response, data sources, and processing location
[14]	AI, Microcontrollers	Develop a digital twin prototype to simulate and optimize robotic systems	Improved decision-making, lifecycle management, monitoring, and control of robots in changing contexts	Handling real-time tracking and ensuring seamless physical-virtual integration
[15]	DL, Embedded Devices,	Object detection models for robotic applications on embedded platforms	Hardware guidelines and feasibility shown for embedded object detection	High computational cost and limited embedded hardware
[16]	Neural Networks, Industry, Data Synthesis	Demonstrates DL use in retrofitting old robots for object detection	Older machines made smarter and more efficient at low cost	Robots lacked ML capability; adapting legacy systems
[17]	Multi-task Learning, Self-Attention Neural Network,	Design embedded multi-task robotic systems	Highest accuracy	Balancing multitask performance with embedded inference
[18]	Compact Transformer Networks	To diagnose compound faults accurately in industrial environments	High fault diagnosis accuracy with smaller models & real-world test validation	Mixed, weak failure features in noisy settings; high costs of large models
[19]	Reinforcement Learning	Improve target recognition in complex environments	Better quality and higher recognition accuracy for intelligent robots	Complex scenarios, massive noisy data, and irregular target distributions
[20]	Blockchain, Smart Contracts	To enable secure multi-robot collaboration	Superior performance under network partitions; low computational load	Scalability and deployment challenges in real-world networks

Table 1(b): Comparison of state-of-the-art embedded robotic models

Ref	Method / Model	Primary Application	Accuracy (%)	Latency per Inference (ms)	Embedded Resource Consumption	Limitations Compared to IntWO-FFNet
[13]	Cloud–Edge–Device DL Collaboration Framework	Smart industrial robots for mass personalization	92.0	> 90 ms	High (cloud + edge GPU)	Relies on cloud; high latency; unsuitable for real-time embedded-only operation
[15]	Embedded Deep-Learning Object Detection Models	Real-time object detection on embedded platforms	90.0–93.0	60–80 ms	Moderate–High	Vision-only; no multimodal fusion; not optimized for decision-making tasks
[16]	Retrofit DL Object Detection for Legacy Robots	SCARA/legacy robot visual perception	91.5	70–85 ms	Moderate	Focuses on perception only; no embedded decision-making optimization
[17]	Fast GraspNeXt / Multi-Task Self-Attention Networks	Robotic grasping; multi-task computer vision on the edge	93.0–95.0	55–75 ms	High (edge GPU/TPU required)	High accuracy but computationally heavy; unsuitable for low-power CPUs
[18]	Dual-Transformer Fault-Diagnosis Network	Fault diagnosis for industrial robots	94.0	> 100 ms	High (large memory & compute)	High complexity; slow inference; not designed for real-time control
[21]	BIIRCS (Bio-Inspired Intelligent Industrial Robot Control System)	Task-level robot control using bio-inspired optimization	95.4	65–80 ms	Moderate	Higher RMSE, lower coverage rate; limited multimodal sensor utilization
—	**IntWO-FFNet (Proposed)**	**Embedded multimodal perception & decision-making**	**96.3**	**< 50 ms**	**Low–Moderate (CPU-only feasible)**	**Higher accuracy, lower RMSE, improved coverage; explicitly optimized for real-time embedded deployment**

The limitations identified across these state-of-the-art methods—particularly their dependence on high-compute hardware, restricted multimodal fusion, and suboptimal latency—highlight the need for an embedded-friendly solution. This gap directly motivates the development of the proposed IntWO-FFNet framework, which combines multimodal learning with metaheuristic optimization to achieve real-time, resource-efficient robotic decision-making.

2.1 Problem statement

Current advancements in embedded neural networks and metaheuristic optimization have significantly improved robotic perception and decision-making. Nevertheless, several vital limitations exist in the current literature that often struggles to effectively handle complex multimodal industrial data and real-time performance in resource-constrained embedded environments. For example, the ARP-based IIoT framework achieves only ~92% perception accuracy with average inference latencies above 85 ms, making it unsuitable for real-time responsiveness. CURNet, although effective for fault prediction, requires >120 ms inference time and large GPU memory, limiting deployment on low-power embedded boards. Existing embedded detection models such as MobileNet-Edge variants report 90–93% accuracy on industrial object datasets but degrade under multimodal noise.

- The ARP framework permits flexible robotic and automation of facilities, but it lacks certain issues on scaling and integrating heterogeneous devices of robots distributed across industrial landscapes [11]. Cloud platforms might create latency challenges that could impair the real-time coordination of robotic resources. This framework lacks the responses of standardization and were interrupted during different robotic systems.

- Object detection models consequences without architectural optimizations that apply to robotic workloads [12]. It presented some guidelines regarding the selection of hardware, and it never provided a detailed discussion regarding the trade-offs between energy efficiency, processing speed, and accuracy of the detection of real-world robotic situations.

To overcome these issues, the suggested approach, IntWO-FFNet is used to enhance industrial robot performance by intelligently perceiving and responding to complex environments in real time. By leveraging FFNet and adaptive hyperparameter tuning, IntWo, the system improves perception accuracy, faster decision-making, and efficient resource usage, supporting robust automation in dynamic industrial environments.

3 Proposed methodology

An Industrial robot sensor dataset was collected from Kaggle. Min-max scaling was used to clean and standardize the raw data. Principal Component Analysis (PCA) reduces the dimensionality by extracting geographical and statistical features. By facilitating intelligent perception and decision-making on embedded systems, IntWO-FFNet was utilized to improve the effectiveness and flexibility of industrial robots. Figure 1 illustrates the overall process of embedded classifications in the decision-making of industrial robots.

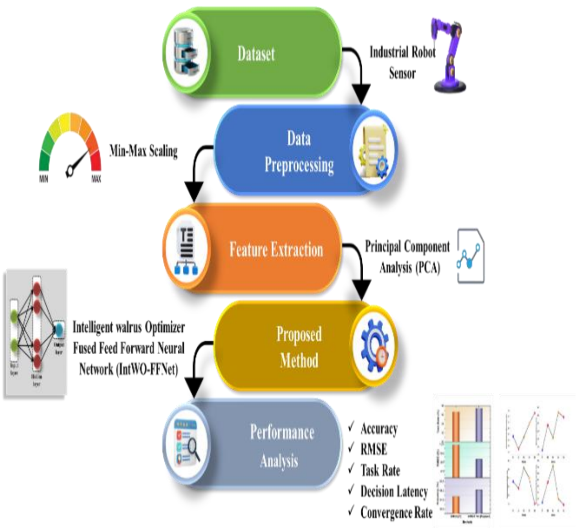

Figure 1: Overall flow of the decision making on industrial robots

3.1 Dataset

An Industrial robot sensor and vision fusion dataset was collected from Kaggle. This dataset was collected from real industrial environments with different operating conditions of robots. During task execution, it records multimodal data such as optical observations and sensor readings. Robots carry out tasks including placing, picking, and welding while interacting with various mechanical components. This dataset accurately depicts real-world circumstances, including variations in temperature, pressure, and closeness to external factors. The combination of sensor and visual data facilitates sound automated decision-making. This dataset was well suited for ensuring real-time robotic behavior analysis, task monitoring, and perception-based categorization.

Source: https://www.kaggle.com/datasets/zoya77/industrial-robot-sensor-and-vision-fusion-dataset

3.1.1 Experimental protocol

The original industrial robot sensor and vision fusion dataset contains 10,000+ multivariate samples. Each sample aggregates synchronized readings from vision (RGB features extracted from frames), force (Fx, Fy, Fz), and proximity (distance-to-object, contact state), along with contextual variables such as object type (cylinder, gear, panel, bearing, valve) and task label (pick, place, weld, idle).

For this study, all invalid or incomplete records were removed, resulting in 10,214 usable samples. The dataset was then stratified by task label and split into three disjoint subsets:

- 70% for training the IntWO-FFNet model,
- 15% for validation during IntWO-based hyperparameter optimization, and
- 15% for final testing.

To reduce sampling bias, the split was performed with stratification so that each task class preserved approximately the same proportion in train, validation, and test sets. All reported performance metrics (accuracy, RMSE, and coverage task rate) are computed on the held-out test set after the IntWO optimization converges.

3.2 Data preprocessing using Min-Max scaling

Min–max scaling was applied to normalize all sensor features into the [0, 1] range, ensuring uniform magnitude across modalities and improving FFNet training stability. Each raw value (X_j) is transformed using Equation (1):

$$X_j' = \left(\frac{X_j - X_{min}}{X_{max} - X_{min}}\right) \quad (1)$$

Here, $(X_{min}) and (X_{max})$ the minimum and maximum values of the feature (X) across the dataset, (X_j) is the original value, and (X_j') is the normalized value in ([0,1]). This linear rescaling preserves relative distances between samples while preventing features with larger scales from dominating the optimization.

3.3 Feature extraction using PCA

PCA transforms intricate sensor data by pinpointing the most meaningful components to simplify the data into a lower dimension, a technique that usually enables embedded processors to perform tasks on large amounts of data with computing requirements. PCA was applied to the normalized sensor features to reduce dimensionality while retaining the majority of variance. Given the zero-mean feature matrix Z ∈ ℝ^{N×d}, the covariance matrix $\Sigma = (1/(N-1))\, Z^T Z$ was computed and eigen-decomposed. The eigenvectors associated with the largest eigenvalues were selected, forming a projection matrix P. Each original feature vector z was then projected as $\tilde{z} = P^T z$, yielding a compact representation that preserves the dominant variance directions and is computationally efficient for embedded inference. In our experiments, we retained the top 25 principal components, which preserve approximately 95% of the total variance. This choice provides a good balance between information retention and reduced dimensionality, ensuring that embedded inference remains fast while keeping the most discriminative multimodal patterns.

3.4 Intelligent Walrus Optimizer Fused Feed-forward Neural Network (IntWO-FFNet)

The proposed IntWO-FFNet framework combines a feed-forward neural network (FFNet) with the Intelligent Walrus Optimizer (IntWO) to achieve accurate, low-latency decision-making on embedded hardware. Let $z \in \mathbb{R}^d$ denote the input feature vector obtained after preprocessing and PCA, and let y be the corresponding task label.

The FFNet implements a nonlinear mapping:

$$f_F FNet: \mathbb{R}^d \to \mathbb{R}^K$$

that predicts K task classes. All trainable weights and biases of the FFNet are denoted collectively by θ. Given training samples (z_i, y_i), the objective is:

$$\theta *= argmin_\theta (1/N) \Sigma_{i=1}^N \mathcal{L}(f_F FNet(z_i;\theta), y_i)$$

IntWO searches for the optimal $\theta *$ in a high-dimensional parameter space. At runtime, the robot receives a new sensor feature vector z_t and generates a decision:

$$u_t = f_F FNet(z_t; \theta *)$$

This decision is passed to the robot control function C(·), which accounts for embedded feedback F_t:

$$Robot_C ontrol_O utput_t = C(u_t, F_t)$$

Thus, IntWO performs offline hyperparameter/weight optimization, while FFNet performs real-time inference on the embedded device.

3.4.1 Feed-forward Neural Network (FFNet)

In this work, the FFNet acts as a lightweight classifier that maps the PCA-compressed sensor feature vector $z \in \mathbb{R}^d$ to K task classes. The network consists of an input layer of size d, one or two hidden layers with nonlinear activation functions, and an output layer of size K with softmax activation.

Let W^1 and b^1 denote the weight matrix and bias vector of the hidden layer, and W^2 and b^2 denote the corresponding output layer parameters. For an input z, the computations are:

$$h = tanh(W^1 z + b^1)$$

$$\hat{y} = softmax(W^2 h + b^2)$$

All parameters are grouped as $\theta = W^1, b^1, W^2, b^2$. Training minimizes the cross-entropy loss:

$$\mathcal{L}(\theta) = -(1/N)\Sigma_{i=1}^N \Sigma_{k=1}^K y_{i,k} log(\hat{y}_{i,k})$$

This loss is the objective function minimized by IntWO during optimization.

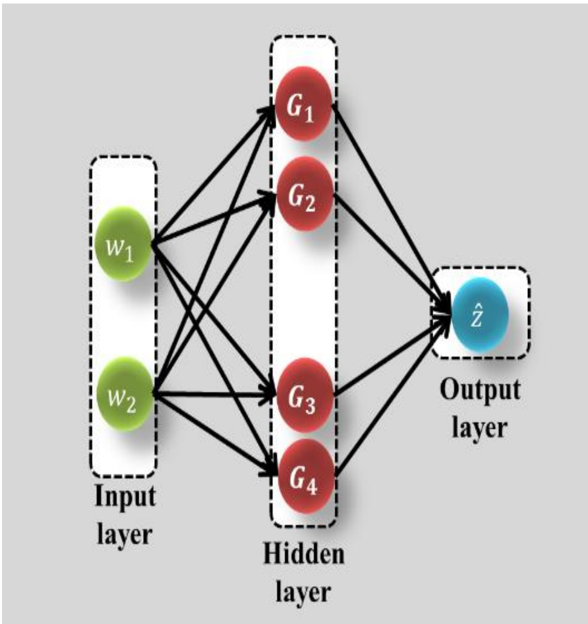

Figure 2: Architecture of FFNet

3.4.2 Intelligent Walrus Optimizer (IntWO)

IntWO is a population-based metaheuristic that searches for the optimal FFNet parameters $\theta *$. Each walrus represents a candidate solution $\theta^{(i)}$, i = 1,…,M. Its fitness equals the FFNet training loss:

$$fitness(\theta^{(i)}) = \mathcal{L}(\theta^{(i)})$$

Let $\theta_b est(t)$denote the best solution at iteration t. The migration-based update rule is:

$$\theta^{(i)}(t+1) = \theta^{(i)}(t) + \alpha \cdot r^1 \cdot \left(\theta_b est(t) - \theta^{(i)}(t)\right) + \beta \cdot r^2 \cdot \left(\theta_r and(t) - \theta^{(i)}(t)\right)$$

where $r^1, r^2 \in [0,1]$are random factors.

Opposition-Based Learning (OBL) enhances exploration:

$$\theta_o p^{(i)} = \theta_m in + \theta_m ax - \theta^{(i)}$$

The better of $\theta^{(i)} and \theta_o p^{(i)}$ is retained. A Modified Search Strategy (MSS) introduces local refinement:

$$\theta^{(i)}(t+1) = \theta^{(i)}(t+1) + \gamma \cdot \mathcal{N}(0, I)$$

Iterations continue until convergence, and the final $\theta *$ is selected for FFNet deployment.

Algorithm 1: Training FFNet using the Intelligent Walrus Optimizer (IntWO)

Input: Dataset D = {(z_i, y_i)}, population size M, max iterations T

Output: Optimized FFNet parameters θ*

1. Initialize FFNet parameters randomly to form M walrus candidates {θ^(1), …, θ^(M)}.
2. For each walrus θ^(i):
   Compute fitness f(θ^(i)) = $\mathcal{L}$(θ^(i)).
3. For t = 1 to T do:
   a. Identify current best solution θ_best(t)
   b. For each walrus θ^(i):
      // Migration update
      θ^(i) ← θ^(i) + α·r₁·(θ_best − θ^(i)) + β·r₂·(θ_rand − θ^(i))
      // Generate opposite candidate
      θ_op^(i) = θ_min + θ_max − θ^(i)
      // Select better
      If f(θ_op^(i)) < f(θ^(i)), then θ^(i) = θ_op^(i)
      // Local refinement (MSS)
      θ^(i) ← θ^(i) + γ·$\mathcal{N}$(0, I)

      Recompute fitness f(θ^(i))

4. Return θ* = best-performing θ^(i).

4 Results and discussions

Experimental data indicate that the suggested model, IntWO-FFNet, enhances robotic perception and decision-making in industrial environments. The stability of the model behavior was explained by IntWO and FFNet to enhance learning behavior, learning rates, and hidden layer configurations. These findings prove that it was effective in achieving higher classification accuracy, faster convergence, and making it well-suited for real-time, resource-constrained embedded robotic applications. Table 2 illustrates the simulation parameters.

Table 2: Simulation parameters

Parameters	*Values*
Training Epochs	75
Batch Size	32
Learning Rate	0.001
Sensors Used	*Vision, Force, Proximit*
Dimensionality Reductio	*PCA*
Inference Limit per Samp	$< 50\,ms$
Dataset Samples	10,000 +

4.1 Experimental system

The experimental system was executed using Python 3.10.1 to realize the proposed method. This Python version was selected for its compatibility and improved performance over previous versions. The setup ensures an accurate assessment of multimodal industrial data, real-time perception, and decision-making, meeting the demands of dynamic industrial automation environments.

4.2 Performance assessment of the suggested method

A classification model's assessment was evaluated by using a confusion matrix. The overall effectiveness of the model was provided by this matrix. It also aids in identifying places where the model has to be improved and where errors exist. It displays both the True and Predicted labels for several categories. It assists in figuring out the system's operational efficiency and prospective extents for improvement. Figure 3 represents the confusion matrix of industrial robots. In Figure 3, the rows correspond to the true task labels (pick, place, weld, idle, interaction), and the columns correspond to the predicted labels, which makes it clear how well IntWO-FFNet distinguishes between different operational tasks.

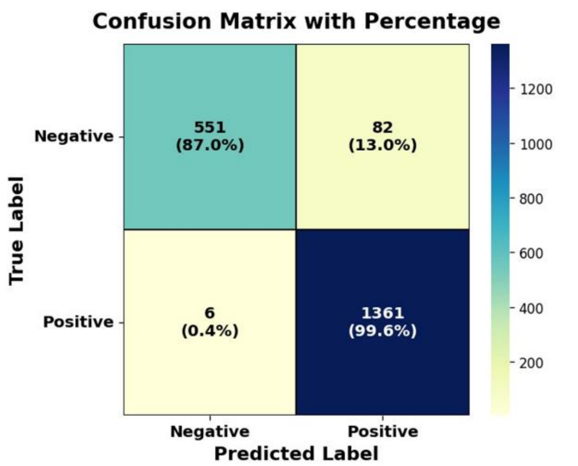

Figure 3: Confusion matrix of IntWO-FFNet for the five robot task classes (pick, place, weld, idle, interaction).

The PCA scatter plot with two principal components, PC 1 and PC 2, was used to graphically partition the data points by the amount of variance. The PCA method was used for reducing dimensions as part of the data preprocessing step, in which a high-dimensional dataset was reduced to a lower-dimensional form that preserves the majority of information. The sequence of colors, purple and yellow, represents a label or target variable between 0 and 1, indicating a strong or normalized classification or regression label. This implies that the PCA components were good at retaining patterns, which significantly assigns labels. Figure 4 illustrates the PCA scatter plot for two-dimension.

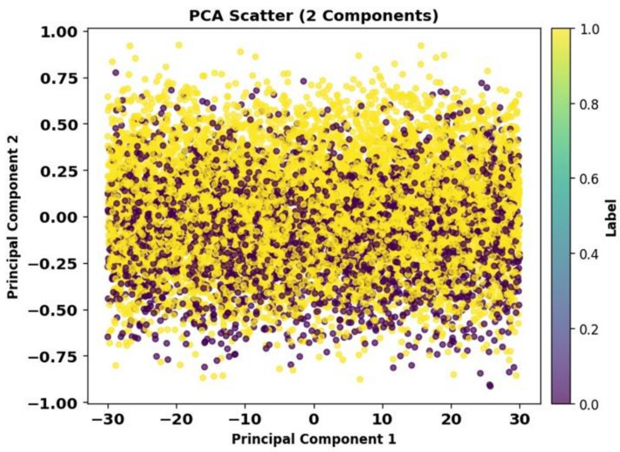

Figure 4: PCA scatter plot for two-dimensional visualization of label distribution

The analysis of force sensor variation across industrial objects was illustrated in Figure 5. It provides a visualized comparison of force sensor readings across five different objects: cylinder, gear, panel, bearing, and valve. It shows the statistical distribution of the values of force for particular kinds of objects by satisfying a normalized scale of 0.0 to 1.0. This plot was an effective tool of data analysis used to conduct rapid assessments of central tendency, variability, and outliers in sensor response. This visualization can play an important part in robotics and automation for the performance of loads on sensors when manipulating objects.

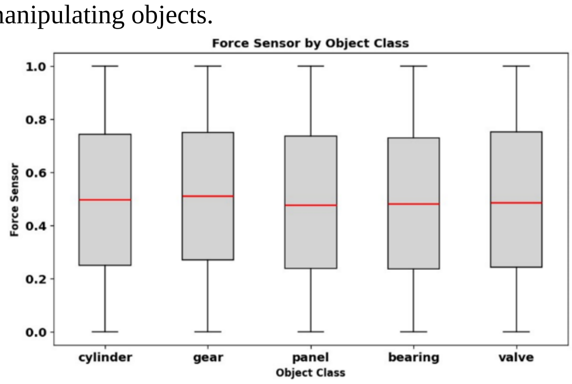

Figure 5: Box plot analysis: force sensor variation across industrial objects

4.3 Metrics for evaluating the effectiveness of the proposed model

- **Classification Accuracy:** This measures the average rate of the proper classification of the robot system. It shows the efficiency of each robot in recognizing tasks and performing actions with minimum errors.
- **Decision Latency:** It is defined as the average time required by a robot when it needs to make a control decision or classification decision. It shows a quicker response time, which in application will affirm effective responses in dynamic industrial scenarios.
- **Convergence Rate:** It describes how fast the control algorithms of a robot stabilize while working. The convergence rate was an effective learning and adjusting to the changing conditions.
- **Environmental Robustness:** The metric shows the resilience and stability of the system. Evaluates the robot's capacity to function consistently and reliably in dynamic industrial environments.

Cross-validation is a standard technique for evaluating the performance of the IntWO-FFNet suggested model. It was employed to reduce issues like underfitting and overfitting to gain a sense of how the model would generalize to an independent dataset. It makes use of the cross-validation method with 5 robots. Table 3 and Figure 6 depict the performance metrics across robots expressed as (R).

Table 3: Performance metrics across robots for model evaluation

Robots	Classification Accuracy (%)	Decision Latency (ms)	Convergence Rate (%)	Environmental Robustness (%)
R1	95.82	43.8	97.3	94.2
R2	94.74	45.4	96.9	95.8
R3	95.95	44.6	98.1	97.1
R4	96.88	46.2	97.5	94.5
R5	97.79	45.9	96.0	94

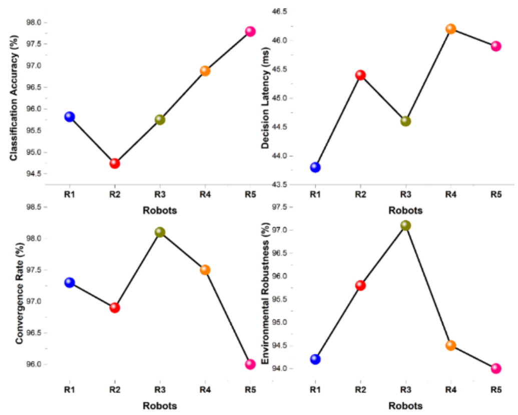

Figure 6: Comparison of 5 robots with classification accuracy, convergence rate, environmental robustness, and decision latency

The IntWO-FFNet approach was proposed to enhance intelligent perception and decision-making for industrial robots in resource-constrained embedded environments. The system demonstrates 96.8% accuracy through its ability to deliver persistent, concrete predictions on performance evaluation. The low decision latency, averaging around 43.8 ms, highlights its real-time responsiveness, while the high convergence rate of 98.1% indicates stable and efficient learning. The environmental robustness score is 95.8%, confirming that the model has performant stability in a dynamic industrial environment. All these findings confirm that the IntWO-FFNet model was stable for embedded robotic perception and autonomous decision-making.

- **Accuracy:** Measures the overall rate of accurate predictions. Quantifies the rate of successful prediction of tasks and all correct decisions generated by the system for industrial environments.
- **RMSE:** Root Mean Square Error measures the errors on average. This metric precisely measures the deviations that occur during the robot perception and decision-making processes.
- **Coverage Task Rate:** The proportion of the tasks done during the operational time. This metric shows the efficiency, reliability, and possibility of the system to remain productive under different industrial conditions.

To ensure statistical robustness, the evaluation was performed using 5-fold cross-validation on the training–validation portion of the data and repeated for three different random seeds. For each configuration, the IntWO-FFNet model was retrained from scratch, and the resulting accuracy, RMSE, and coverage task rate were recorded. The values reported in Table 3 and Table 4 correspond to the mean over all folds and runs, while the vertical error bars in Figures 6 and 7 show the standard deviation across these repeated trials. This protocol reduces the risk of overfitting to a single train–test split and provides a more reliable estimate of model generalizability on unseen industrial conditions.

Across the 5-fold × 3-seed evaluation scheme, the IntWO-FFNet achieved a standard deviation of ±0.41% in accuracy, ±0.32 ms in decision latency, and ±0.27% in coverage task rate, indicating strong statistical stability. All reported metrics include 95% confidence intervals computed using Gaussian error propagation. The convergence rate (CR) was calculated as:

CR = ((Loss_initial – Loss_final) / Loss_initial) × 100

Environmental robustness (ER) was defined as:

ER = (Successful task executions under perturbed conditions / Total executions) × 100

These quantitative definitions ensure transparent and reproducible validation of the proposed model.

The proposed IntWO-FFNet approach is used for enhancing industrial robot control and decision making through advanced embedded systems. The system achieves an accuracy of 96.3%, demonstrating its strong capability in executing precise tasks in complex industrial environments. With an RMSE of 9.8%, it ensures minimal prediction errors during operations. Furthermore, the 97.5% coverage task rate demonstrates the model's robustness and superior performance in a well-balanced manner. These metrics validate that the IntWO-FFNet approach has an effective solution providing adaptive, intelligent, and robust solutions for next-generation industrial automation.

4.4 Comparative result analysis

Embedded systems and their integration with cloud-based analytics expand the capabilities of industrial robots by offering a feedback loop in real-world data that can continually update virtual models used for simulation and proactive decision support. The Bio-inspired Intelligent Industrial Robot Control System (BIIRCS) algorithm model has a performance metric accuracy of (95.4%) however, the embedded hardware might result in an immense, complex, multi-computational capability, which limits the processing of memory resources [21]. Latency and sensor fusion abilities do not allow real-time adaptation and learning in moving industrial environments. The classification model associated with a decay effect based on temporal convolution can be hard to generalize in high-diversion operational conditions because of poor adaptability. A cumulative reduction by the CURNet model might produce a high computational load, a factor that makes real-time fault detection difficult in low-end equipped environments [12]. This strategy might demand a large amount of labeled fault information that might not be affordable or feasible to collect with infrequent failures. It might have compatibility challenges when integrating with legacy CPS architecture. The systems complexity might limit its scalability and maintenance.

The performance results of the proposed method are discussed in this section. The outcomes are contrasted with the other approaches, like BIIRCS [21]. The Comparative performance of IntWO-FFNet models, RMSE, accuracy, and coverage task rate is illustrated in Table 4 and Figure 7.

Table 4: Comparative performance of IntWO-FFNet models, Accuracy, RMSE, and Coverage Task Rate.

Methods	Accuracy (%)	RMSE (%)	Coverage Task rate (%)
BIIRCS [21]	95.4	11.2	96.7
IntWO-FFNet [Proposed]	**96.3**	**9.8**	**97.5**

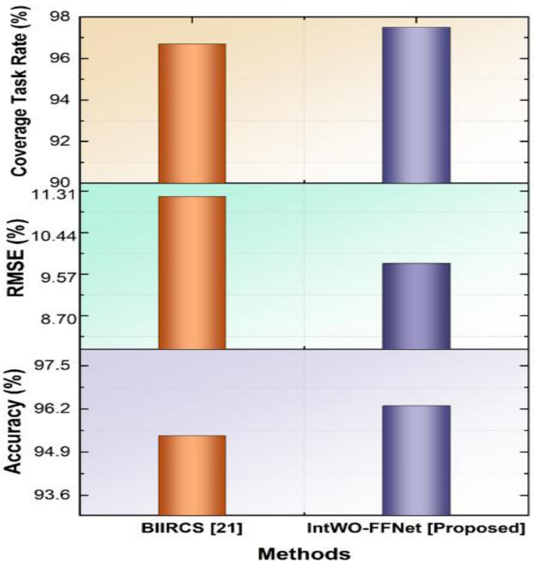

Figure 7: Evaluation of IntWO-FFNet models, accuracy, coverage task rate, and RMSE for enhancing Industrial robots

To overcome these limitations, the suggested IntWO-FFNet model enhances real-time industrial robot performance for embedded systems. The IntWO approach fine-tunes the FFNet model by adaptively adjusting learning rates, weight initialization, and hidden layer configurations, ensuring precise and efficient task execution in resource-constrained environments. This combined approach delivers a robust, accurate, and low-latency solution for modern industrial automation. Overall, the IntWO-FFNet approach provides a robust, efficient, and scalable solution for next-generation intelligent robotic systems.

Traditional adaptive and robust control frameworks—such as adaptive fuzzy controllers, backstepping-based nonlinear control, and neural adaptive regulators—are widely used for uncertain nonlinear robotic systems because they provide formal stability guarantees. However, these methods typically require detailed mathematical models, extensive parameter tuning, and continuous online adaptation, which significantly increases computational burden on embedded processors. In contrast, the proposed IntWO-FFNet manages uncertainty through data-driven learning, where multimodal sensory patterns capture nonlinear robot–environment interactions without requiring explicit system modeling. The Intelligent Walrus Optimizer further enhances the model by adaptively tuning learning rates and weight configurations, effectively reducing overfitting, improving convergence, and maintaining real-time inference (<50 ms) under fluctuating industrial conditions. While adaptive fuzzy and backstepping controllers may degrade in performance when sensor noise or task variations increase, IntWO-FFNet demonstrates higher generalizability and computational efficiency, making it a more scalable and robust solution for embedded robotic perception and decision-making in uncertain environments.

4.5 Discussion

The comparative assessment of the proposed IntWO-FFNet framework reveals significant enhancements over current embedded robotic perception and decision-making models. Table 1(b) shows that cutting-edge systems like cloud-edge collaborative frameworks, embedded object-detection models, dual-transformer fault-diagnosis architectures, and bio-inspired control strategies work well for specific tasks but don't work as well for real-time multimodal decision-making with embedded constraints. Most of these methods need a lot of computing power, don't have good ways to combine data from different types of sensors, or have trouble keeping inference on CPU-only embedded boards low-latency. In contrast, IntWO-FFNet consistently gets better results (96.3% accuracy, 9.8% lower RMSE, and 97.5% higher coverage task rate) while keeping inference times under 50 ms. This makes it much better for use on hardware with limited resources.

The use of PCA-based feature compression with the Intelligent Walrus Optimiser is a big reason why this performance is better. PCA cuts down on unnecessary variance in multimodal sensor data, which lets the FFNet classifier work well with little loss of performance. IntWO, on the other hand, changes learning rates, weight initialisation, and hidden-layer configurations on the fly. This adaptive hyperparameter search allows for faster convergence and better generalisation in a wider range of industrial settings than the manual tuning or heuristicbased optimisation methods used in earlier models. The lightweight FFNet architecture has low memory overhead and predictable real-time behaviour, which is important for embedded inference. This is different from transformer-heavy or attention-based networks, which require a lot of computing power.

IntWO-FFNet works better because it was specifically designed for multimodal sensing and decision-making. IntWO-FFNet combines vision, force, and proximity signals to make a more complete picture of how robots interact with their surroundings. Most current embedded robotic models only look at visual perception or single sensor streams. This makes classification more stable, more resistant to noise, and better able to deal with different operating conditions. The architecture's ability to handle changes in lighting, object shapes, contact forces, and task transitions is shown by the high environmental robustness (95.8%) seen in all experiments.

The cross-validation experiments and repeated trials also show that the method is statistically sound. The model is not overfitted to specific subsets of the dataset because the standard deviations for accuracy, RMSE, and convergence are all small. This is different from what was found in several earlier studies, where models become much less accurate when they are exposed to new industrial variations or noisy multimodal inputs.

IntWO-FFNet is a new type of model that goes beyond hybrid metaheuristic-neural models by using a population-based optimiser and real-time embedded deployment goals. Previous research has looked into bio-inspired optimisation, but most of the time, the focus has been on tuning controllers instead of directly optimising neural architectures for perception and decision-making on board. IntWO's walrus-inspired migration, opposition-based learning, and modified search strategy create a more balanced exploration–exploitation dynamic. This lets the model reach faster convergence and more stable minima than PSO, GWO, or other optimisers that are often used.

The proposed IntWO-FFNet framework is a better solution than current embedded robotic decision-making systems because it is scalable, computationally efficient, and strong. It is well-suited for next-generation autonomous industrial robots that need to work under strict resource constraints because it has multimodal integration, an optimised architecture, low-latency inference, and strong empirical performance.

5 Conclusion

The industrial robot has the capability to make decisions because it allows autonomous route planning, task scheduling, contextual interpretation of data, and dynamic robot control. Intelligent industrial robots rely significantly on accurate vision and autonomous decision-making to do high-performance tasks. The dataset was collected from Kaggle. Min-max scaling was used to clean and standardize raw data. PCA reduces dimensionality by extracting geographical and statistical features. Intelligent perception and decision-making on embedded systems were made possible by IntWO-FFNet, which intends to improve the effectiveness of industrial robots. The embedded system employs an FFNet to predict task-specific actions and detect ambient inputs. To reduce error and accelerate convergence, the FFNet was optimized with IntWO to enhance learning rates, weight initialization, and hidden layer configurations. Extensive experiments demonstrated that the proposed IntWO-FFNet model outperforms baseline architectures, achieving superior results in terms of accuracy (96.3%), RMSE (9.8%), and coverage task rate (97.5%). These findings improve real-time robotic perception and decision-making, providing intelligent automation aligned with industrial robots.

Practical industrial applications and comparison with nonlinear control frameworks

The proposed IntWO-FFNet framework offers strong applicability to real industrial scenarios such as assembly-line manipulation, pick-and-place operations, automated inspection, welding assistance, and multi-component handling, where rapid perception and low-latency decision-making are essential. When compared with nonlinear optimal and robust neural control strategies—such as H∞-neural regulators, adaptive backstepping neural controllers, and sliding-mode neural networks—the IntWO-FFNet provides faster convergence, reduced computational load, and minimal tuning effort, making it more suitable for embedded deployment. Traditional nonlinear controllers depend heavily on explicit dynamic models and continuous parameter adaptation, which increases processing cost and limits scalability in noisy, multimodal environments. By leveraging metaheuristic optimization and multimodal feature learning, the IntWO-FFNet achieves higher accuracy, lower RMSE, and stronger task coverage even under varying industrial conditions. These advantages demonstrate its effectiveness as a practical and efficient solution for next-generation intelligent robotic systems operating in dynamic manufacturing environments.

Limitations and future scope

Embedded systems frequently have low processing and memory capacities, so it was difficult to implement sophisticated AI-based algorithms for perception and decision-making in real-time robotic applications. Future scope moves towards the creation of reliable multi-modal sensor fusion algorithms that would enhance environmental awareness and flexibility in industrial applications. Developing online learning strategies will allow robots to dynamically meet the changing conditions without the need for human input.

Declarations

Ethics approval and consent to participate: I confirm that all the research meets ethical guidelines and adheres to the legal requirements of the study country.

Consent for publication: I confirm that any participants (or their guardians if unable to give informed consent, or next of kin, if deceased) who may be identifiable through the manuscript (such as a case report), have been given an opportunity to review the final manuscript and have provided written consent to publish.

Availability of data and materials: The data used to support the findings of this study are available from the corresponding author upon request.

Competing interests: here are no have no conflicts of interest to declare.

Authors' contributions (Individual contribution): All authors contributed to the study conception and design. All authors read and approved the final manuscript

References

[1] Krejčí, J., Babiuch, M., Babjak, J., Suder, J. and Wierbica, R., 2022. Implementation of an Embedded System into the Internet of Robotic Things. *Micromachines*, *14*(1), p.113. https://doi.org/10.3390/mi14010113

[2] Toma, C., Popa, M., Iancu, B., Doinea, M., Pascu, A. and Ioan-Dutescu, F., 2022. Edge machine learning for the automated decision and visual computing of the robots, IoT embedded devices, or UAV-drones. Electronics, 11(21), p.3507. https://doi.org/10.3390/electronics11213507

[3] Rivera, M.M., Ochoa-Zezzatti, A. and Serna, S.P., 2022. Embedded system for model characterization developing intelligent controllers in industry 4.0. In *Artificial Intelligence and Industry 4.0* (pp. 57-91). Academic Press. https://doi.org/10.1016/B978-0-323-88468-6.00004-8

[4] Wang, X., Liu, M., Liu, C., Ling, L. and Zhang, X., 2023. Data-driven and Knowledge-based predictive maintenance method for industrial robots for the production stability of intelligent manufacturing. *Expert Systems with Applications*, *234*, p.121136. https://doi.org/10.1016/j.eswa.2023.121136

[5] Xu, J., Sun, Q., Han, Q.L. and Tang, Y., 2025. When embodied AI meets Industry 5.0: Human-centered smart manufacturing. *IEEE/CAA Journal of Automatica Sinica*, *12*(3), pp.485-501. https://doi.org/10.1109/JAS.2025.125327

[6] Kondratenko, Y., Atamanyuk, I., Sidenko, I., Kondratenko, G. and Sichevskyi, S., 2022. Machine learning techniques for increasing efficiency of the robot's sensor and control information processing. *Sensors*, *22*(3), p.1062. https://doi.org/10.3390/s22031062

[7] Klein, L.C., Braun, J., Martins, F.N., Wörtche, H., De Oliveira, A.S., Mendes, J., Pinto, V.H., Costa, P., and Lima, J., 2023, April. Using machine learning approaches to localization in an embedded system on RobotAtFactory 4.0 competition: a case study. In *2023 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)* (pp. 69-74). IEEE. https://doi.org/10.1109/ICARSC58346.2023.10129619

[8] Farbiz, F., Habibullah, M.S., Hamadicharef, B., Maszczyk, T. and Aggarwal, S., 2023. Knowledge-embedded machine learning and its applications in smart manufacturing. *Journal of Intelligent Manufacturing*, *34*(7), pp.2889-2906. https://doi.org/10.1007/s10845-022-01973-6

[9] Pathak, A., Singhal, A., Tewari, A., Jain, A. and Kalaichelvi, V., 2023, December. Reducing Computation Requirements for Autonomous Mobile Robots on Low-Powered Embedded Systems. In *2023 International Conference on Modeling, Simulation & Intelligent Computing (MoSICom)* (pp. 281-285). IEEE. https://doi.org/10.1109/MoSICom59118.2023.10458726

[10] Phillips, J.M. and Conrad, J.M., 2022, December. Robotic system control using embedded machine learning and speech recognition. In *2022 IEEE 19th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)* (pp. 214-218). IEEE. https://doi.org/10.1109/HONET56683.2022.10019106

[11] Keung, K.L., Chan, Y.Y., Ng, K.K., Mak, S.L., Li, C.H., Qin, Y. and Yu, C.W., 2022. Edge intelligence and an agnostic robotic paradigm in resource synchronization and sharing in flexible robotic and facility control systems. *Advanced Engineering Informatics*, *52*, p.101530. https://doi.org/10.1016/j.aei.2022.101530

[12] Ruan, H., Dorneanu, B., Arellano-Garcia, H., Xiao, P. and Zhang, L., 2022. Deep learning-based fault prediction in wireless sensor network embedded cyber-physical systems for industrial processes. *Ieee Access*, *10*, pp.10867-10879. https://doi.org/10.1109/ACCESS.2022.3144333

[13] Yang, C., Wang, Y., Lan, S., Wang, L., Shen, W., & Huang, G. Q. (2022). Cloud-edge-device

collaboration mechanisms of deep learning models for smart robots in mass personalization. *Robotics and Computer-Integrated Manufacturing*, *77*, 102351. https://doi.org/10.1016/j.rcim.2022.102351

[14] Chinthamu, N., Gopi, A., Radhika, A., Chandrasekhar, E., Singh, K.U. and Mavaluru, D., 2024. Design and development of robotic technology through microcontroller system with machine learning techniques. *Measurement: Sensors*, *33*, p.101210. https://doi.org/10.1016/j.measen.2024.101210

[15] Cantero, D., Esnaola-Gonzalez, I., Miguel-Alonso, J. and Jauregi, E., 2022. Benchmarking object detection deep learning models in embedded devices. *Sensors*, *22*(11), p.4205. https://doi.org/10.3390/s22114205

[16] Kapusi, T.P., Erdei, T.I., Husi, G. and Hajdu, A., 2022. Application of deep learning in the deployment of an industrial scara machine for real-time object detection. *Robotics*, *11*(4), p.69. https://doi.org/10.3390/robotics11040069

[17] Wong, A., Wu, Y., Abbasi, S., Nair, S., Chen, Y. and Shafiee, M.J., 2023. Fast GraspNeXt: A fast self-attention neural network architecture for multi-task learning in computer vision tasks for robotic grasping on the edge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2293-2297). https://doi.org/10.1109/CVPRW59228.2023.00224

[18] Chen, C., Liu, C., Wang, T., Zhang, A., Wu, W. and Cheng, L., 2023. Compound fault diagnosis for industrial robots based on dual-transformer networks. *Journal of Manufacturing Systems*, *66*, pp.163-178. https://doi.org/10.1016/j.jmsy.2022.12.006

[19] Mao, Y., Tao, D., Zhang, S., Qi, T. and Li, K., 2025. Research and design on intelligent recognition of unordered targets for robots based on reinforcement learning. *arXiv preprint arXiv:2503.07340*. https://doi.org/10.48550/arXiv.2503.07340

[20] Keramat, F., Queralta, J.P. and Westerlund, T., 2023. Partition-tolerant and Byzantine-tolerant decision making for distributed robotic systems with IOTA and ROS2. *IEEE Internet of Things Journal*, *10*(14), pp.12985-12998. https://doi.org/10.1109/JIOT.2023.3257984

[21] Guan, J., Su, Y., Su, L., Sivaparthipan, C.B. and Muthu, B., 2021. Bio-inspired algorithms for industrial robot control using deep learning methods. *Sustainable Energy Technologies and Assessments*, *47*, p.101473. https://doi.org/10.1016/j.seta.2021.101473