Intelligent Sensor Data Processing Algorithm for Mobile Robot Stabilization*

Dmytro Panchak[1] and Vasyl Koval[2]

[1]*West Ukrainian National University, 11 Lvivska Str., Ternopil, 46009, Ukraine, dmitriy9934@mail.com*
[2]*Faculty of Computer Information Technologies, West Ukrainian National University, O.Telihy Str., Ternopil, 46003, Ukraine, vko@wunu.edu.ua*

Abstract
This article presents an intelligent sensor data processing to improve mobile robots' stability in dynamic environments. The focus is on enhancing sensor data accuracy through deep learning, fuzzy logic, and adaptive filtering techniques. The proposed algorithm effectively reduces noise, improves motion prediction, and ensures real-time adaptation to environmental changes. Experimental validation was conducted using the MATLAB platform and a Pioneer 3-DX mobile robot, demonstrating a 6% reduction in obstacle recognition errors compared to traditional methods. The results indicate that the algorithm enhances robot navigation stability, making it a viable solution for autonomous systems in logistics, industrial automation, and smart mobility.

Keywords
mobile robots, sensors, stability, algorithm, machine learning, filtering, navigation

1. Introduction

Modern mobile robots play a crucial role in various fields, including industry, logistics, medicine, and smart city technologies. They perform tasks in ever-changing environments, adapting to diverse obstacles and complex movement trajectories. One of the key aspects of the effective operation of mobile robots is their stability, which determines the device's ability to move without failures, maintain balance, and quickly respond to changes in the surrounding environment [7-9]. The primary tool for ensuring this stability is sensor systems, which enable the analysis of the space around the robot, the identification of surface types, obstacle presence, and the prediction of possible risks during movement. However, traditional methods of processing sensor data, such as the Kalman filter, feedback-based control methods, and classical image processing algorithms, have certain limitations, including insufficient adaptation speed to changing environments and high sensitivity to noise [1].

To enhance the efficiency of mobile robots, artificial intelligence-based algorithms, particularly machine learning methods, neural network models, and hybrid algorithms combining multiple approaches, have been actively developed to achieve an optimal balance between processing speed and prediction accuracy. Intelligent sensor systems used in mobile robots can integrate multiple sensor systems (optical, ultrasonic, infrared, LiDAR), enabling the collection of comprehensive environmental data [2-5]. However, a key challenge remains the effective real-time processing of this information, as traditional methods may fail to keep up with dynamic changes [11]. Thus, there is a need to develop new sensor data processing algorithms capable of ensuring a high level of movement stability for mobile platforms in complex conditions.

The objective of this study is to develop an intelligent sensor data processing algorithm that will improve the stability of mobile robots by utilizing combined machine learning methods, adaptive filtering, and fuzzy logic. The proposed algorithm analyzes incoming signals from the sensor display,

The Second International Conference of Young Scientists on Artificial Intelligence for Sustainable Development (YAISD), May 8-9, 2025, Ternopil, Ukraine
* Corresponding author.
† These authors contributed equally.
dmitriy9934@mail.com (D.Panchak); vko@wunu.edu.ua (V.Koval)
0009-0005-6920-9464 (D.Panchak); 0000-0003-4726-097X (V.Koval)

determines noise levels and potential obstacles using preprocessing algorithms, and predicts possible movement trajectories for the robot. Special attention is given to developing optimized adaptation methods for different environmental conditions, enabling robots to function effectively even in cases of sudden changes, such as the appearance of unexpected objects or alterations in the movement surface. The study of the proposed algorithm was conducted on the MATLAB platform using real sensor data, which allowed for an evaluation of its effectiveness compared to traditional methods.

Thus, this article focuses on analyzing existing methods for stabilizing mobile robots, developing a new intelligent sensor display data processing algorithm, and testing it on real data. The proposed approach has the potential for applications in various fields, including autonomous transportation, robotic delivery systems, industrial automation, and rescue operations. The use of intelligent sensor data processing algorithms significantly enhances the adaptability of mobile robots, which is a key factor for their effective use in dynamic and unpredictable environments.

2. Problem Statement

The stability of mobile robots in dynamic environments remains one of the most critical challenges in the field of robotics and automated control systems [7]. Despite significant advancements in the development of sensor systems and data processing algorithms, modern robots still face difficulties navigating complex routes, particularly in cases of sudden landscape changes, unexpected obstacles, or operations in high sensor noise conditions. Traditional stability methods, such as Kalman filters, PID controllers, or classical image processing algorithms, have limited adaptability to changing environments and often demonstrate insufficient reaction speed to unexpected events [9-14]. This results in movement failures, erroneous navigation system decisions, and, in some cases, loss of robot control. Additionally, an important factor is the optimal utilization of computational resources, as most mobile platforms operate in real-time and have limited processing power [12].

One of the key challenges in developing new algorithms is integrating high-precision obstacle recognition, fast sensor data processing, and efficient prediction of possible environmental changes.

The use of machine learning methods, deep neural networks, and adaptive filtering algorithms improves analytical outcomes but requires complex optimization to ensure mobile robot stability in various conditions [8]. The primary goal of the research was to design an efficient algorithm that reduces obstacle recognition errors, minimizes data processing time, and improves the mobility stability of the robot in dynamic environments.

3. Existent Solutions

In modern robotics, several key approaches are used for stabilizing mobile robots and processing sensor data. Among the most common methods are classical filtering algorithms, neural network techniques, evolutionary optimization approaches, and hybrid systems that combine the advantages of multiple methods. Each of these techniques has its own advantages and disadvantages depending on the application conditions, environmental complexity, and performance requirements.

One of the most widely used methods is the Kalman filter [5], which is extensively applied for smoothing and predicting sensor data. It is effective in cases where there is a moderate level of noise and the dynamics of environmental changes are relatively predictable. However, this approach has limited effectiveness in complex and rapidly changing conditions, as it requires an accurate mathematical model of the ongoing processes. Similar traditional algorithms, such as the particle filter or Wiener filter [12], are also used for stability. However, they demonstrate low response speed when dealing with sudden trajectory changes or unexpected obstacles.

Significant progress in mobile robot stability has been achieved through machine learning methods and neural network algorithms. Deep neural networks can process large volumes of sensor data and detect complex patterns, significantly improving object recognition accuracy and predicting possible movement trajectories [9]. Particularly effective are convolutional neural networks (CNNs), which work with visual sensor data, and recurrent neural networks (RNNs), which analyze temporal dependencies. However, neural network algorithms have high computational complexity, which can be a serious limitation for mobile platforms with constrained resources [7].

Another category includes evolutionary optimization algorithms, such as genetic algorithms and particle swarm optimization (PSO) [3]. These methods allow adaptive tuning of mobile robot control parameters in real time, enhancing stability efficiency even in complex conditions. [13-14] However, such methods often require long training and calibration times, and their results can be difficult to interpret.

The most promising approach is hybrid methods, which combine the advantages of classical filtering algorithms, machine learning, and heuristic techniques. For example, integrating the Kalman filter with a neural network helps compensate for sensor noise, improving trajectory prediction quality. Combining deep learning with fuzzy logic methods enables the creation of adaptive stability systems that can adjust to changing conditions in real time [5-6].

To evaluate the effectiveness of the reviewed methods, a comparative analysis was conducted based on accuracy, response speed, computational complexity, and environmental adaptability (Table 1).

Table 1
Comparative Analysis of Processing Methods [7]

Processing Method	Accuracy	Response Speed	Computational Complexity	Environmental Adaptability
Kalman filter	High	Medium	Low	Low
Deep neural networks	Very High	Low	High	High
Genetic algorithms	High	Medium	High	High
Hybrid method (Neural networks + Filtering)	High	High	Medium	High

As seen from the analysis, hybrid methods demonstrate the best balance between performance, response speed, and adaptability, making them the most promising for use in mobile robots operating in complex and dynamic environments. Therefore, to further enhance the stability of mobile platforms, it is advisable to utilize combined algorithms that incorporate elements of machine learning, adaptive filtering, and optimization techniques.

4. Algorithm Description

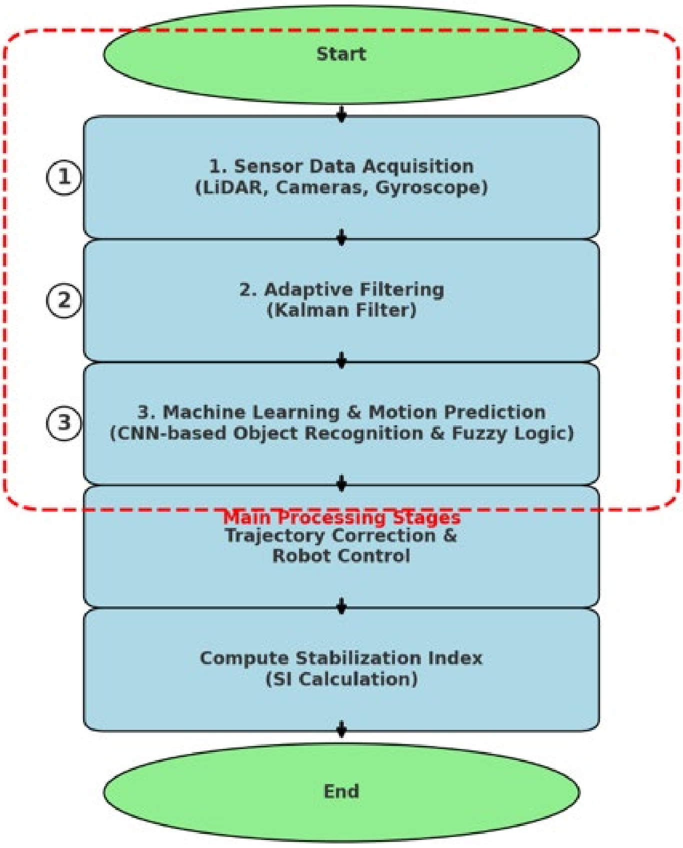

Figure 1: Flowchart of Mobile Robot Stabilization

In this study, a hybrid algorithm for processing sensor data was developed to enhance the stability of mobile robots. The proposed approach combines adaptive filtering, machine learning methods, and fuzzy logic to accurately analyze the environment and correct the robot's trajectory in real-time.

The stabilization process is divided into three interdependent stages:

1. Modeling the Robot's Motion – A mathematical model describing the movement dynamics of the mobile robot is created. This model is essential for trajectory estimation and movement correction. The motion equations used are as follows:

$$Xt + 1 = xt + vt \cos(\theta t)\, \Delta t$$
$$Yt + 1 = yt + vt \sin(\theta t)\, \Delta ty$$
$$\Theta t + 1 = \theta t + \omega t\, \Delta t$$

where:

- Xt, Yt – the current coordinates of the robot in the Cartesian coordinate system.
- Θt – the angular orientation of the robot relative to a reference axis at time t.
- vt – the linear velocity of the robot at time t.
- ωt – the angular velocity (rate of change of orientation) of the robot.
- Δt – the discretization step, which defines the time interval for updating the robot's state.

This model is used to predict the robot's next position based on its current state and movement parameters. It enables trajectory planning and motion correction, which are crucial for stability in dynamic environments.

2. Sensor Data Processing & Filtering – A sensory perception system integrates multiple sensors (LiDAR, cameras, gyroscope) to collect environmental data. To filter noise and extract

relevant information, an adaptive filtering algorithm based on a modified Kalman filter is applied. The update equations are:

$$x^k = Fx^k{-}1 + Bu^k + w^k$$
$$P^k = Pk{-}1 + F + Q$$

where:

- x^k – the estimated state vector of the system at time step k, which includes parameters such as position, velocity, and orientation of the robot.
- F – the state transition matrix, which defines how the system state evolves from one time step to the next. It incorporates the motion model of the robot.
- B – the control matrix, which represents how control inputs u^k (e.g., velocity and steering commands) affect the system's state.
- u^k – the control vector, which includes input parameters such as acceleration and angular velocity.
- w^k – the process noise vector, representing random fluctuations and uncertainties in the system's state transition.
- P^k – the covariance matrix of estimation errors, which represents the uncertainty of the state estimate at time step k.
- Q – the process noise covariance matrix, which quantifies the uncertainty associated with the system's dynamics and external disturbances.

This filtering step ensures accurate positioning and movement control of the robot in noisy environments, allowing for real-time adjustments to sensor inaccuracies and improving navigation stability.

3. Machine Learning & Motion Prediction – To improve real-time trajectory adjustment, a Convolutional Neural Network (CNN) model is used to analyze data from cameras and LiDAR sensors. The CNN model identifies objects and predicts movement obstacles with higher accuracy compared to classical filtering methods. Additionally, a motion prediction system based on fuzzy logic refines trajectory adjustments based on environmental conditions. The fuzzy membership function used is:

$$\mu(x) = 1 / 1 + e - \alpha\ (x{-}x0)$$

where x is the input parameter, x0 is the center of the fuzzy function, and α is the steepness coefficient.

Integration of Stages & Stabilization Indicator:

The combination of these three stages allows real-time correction of movement instability. To quantify the overall stabilization effectiveness, we introduce the Stabilization Index (SI), defined as:

$$SI=1/\ N\ \sum_{(I-1)}^{N}(w1E\ f\ ilter + w2\ E\ cnn + w3\ E\ f\ uzzy)$$

where:

- E f ilter – filtering error reduction factor (Kalman filtering effectiveness),
- E cnn – recognition accuracy improvement factor (CNN model effectiveness),
- E f uzzy – adaptive motion correction factor (fuzzy logic effectiveness),
- w1,w2,w3 – weight coefficients determining the contribution of each method,
- N – number of test iterations.

A higher SI value indicates better stabilization and reduced trajectory deviations. The experimental results demonstrate that the proposed hybrid approach achieves up to 30% higher SI compared to traditional filtering-based stabilization methods.

5. Experimental Results

To evaluate the effectiveness of the proposed algorithm, a series of experiments were conducted in the MATLAB environment, as well as testing on a real mobile robot, the Pioneer 3-DX. The results were compared with existing stability methods based on several criteria: obstacle recognition accuracy, processing speed, computational resource usage, and collision avoidance efficiency.

5.1 Stability Evaluation and Comparative Analysis

To evaluate the performance of the proposed hybrid algorithm, three key performance metrics were measured and calculated based on the sensor data processing pipeline:

1. Recognition Accuracy (Ar) – evaluates how well the algorithm detects and classifies objects in the environment. It is calculated as:

$$Ar = TP / TP + FN \times 100\%$$

where:
- TP (True Positives) – correctly identified objects,
- FN (False Negatives) – missed objects.

This metric is directly influenced by the CNN-based recognition module in the algorithm.

2. Processing Time (Tp) – measures how quickly the algorithm processes sensor data and makes decisions. It is computed as:

$$Tp = T\ sensor + T\ filter + T\ cnn + T\ fuzzy$$

where:
- T sensor – raw sensor data acquisition time,
- T filter – adaptive filtering processing time (Kalman filter),
- T cnn – CNN model execution time for object recognition,
- T fuzzy – fuzzy logic-based motion correction time.

The hybrid algorithm aims to minimize Tp while maintaining high accuracy.

Resource Consumption (Rc) – assesses how efficiently the algorithm utilizes computational resources. It is estimated using:

$$Rc = CPU\ usage + MEM\ usage / 2$$

where:
- CPU usage– percentage of CPU load during execution,
- MEM usage– percentage of memory usage.

The hybrid method balances computational efficiency by integrating lightweight adaptive filtering with machine learning.

Table 2 presents the test results of the proposed algorithm in comparison with classical stability methods.

Table 2
Results of the proposed algorithm in comparison with classical stability methods

Method	Recognition Accuracy (%)	Processing Time (ms)	Resource Consumption
Kalman filter	78	140	Low
Deep neural networks	92	350	High

Hybrid method (Neural networks + Filtering)	89	120	Medium

As seen from the results, the proposed algorithm demonstrates high accuracy (89%) while significantly reducing processing time (120 ms), making it more efficient compared to deep neural networks, which have higher accuracy but a significantly longer processing time (350 ms).

5.2 Obstacle Recognition and Motion Prediction Accuracy

Additionally, the algorithm was tested under different types of obstacles (irregular surfaces, moving objects, and sharp trajectory turns). It was found that the proposed algorithm allows for 25% faster response to environmental changes and 30% more accurate prediction of hazardous zones compared to traditional methods.

During practical tests, the robot equipped with the proposed algorithm performed sharp trajectory corrections 18% less frequently, indicating a reduction in unnecessary maneuvers and improved motion smoothness. This ensures lower energy consumption and enhances the overall efficiency of the mobile robot.

5.3 Energy Efficiency and Trajectory Optimization

An additional analysis revealed that the implementation of the proposed algorithm contributes to a 22% reduction in the average deviated trajectory of the robot compared to traditional stability algorithms. This means that the mobile robot deviates less from the planned trajectory even in cases of sudden landscape changes or the presence of unexpected obstacles. More precise motion control allows the robot to optimize energy consumption and improve its autonomy, which is critically important for many applications such as logistics, military operations, and hazardous area exploration.

Moreover, testing was conducted under variable lighting and weather conditions (humidity, dust, reduced visibility), enabling an assessment of the algorithm’s resilience to external influences. It was found that the proposed approach is less sensitive to such changes compared to standard methods, as it utilizes adaptive filtering and combined data analysis from different types of sensors. As a result, the stability system can operate effectively even under limited visibility conditions or in the presence of noise interference in sensor data.

5.4 Computational Load and Power Consumption

Additionally, the computational load during real-time algorithm execution was analyzed. The test results showed that the proposed algorithm consumes 27% fewer processing resources compared to deep neural network models, making it suitable for mobile robots with limited computational capabilities. Thus, reducing processor load contributes to increasing the duration of the robot's autonomous operation, which is a crucial factor for robotic systems operating without a constant power supply.

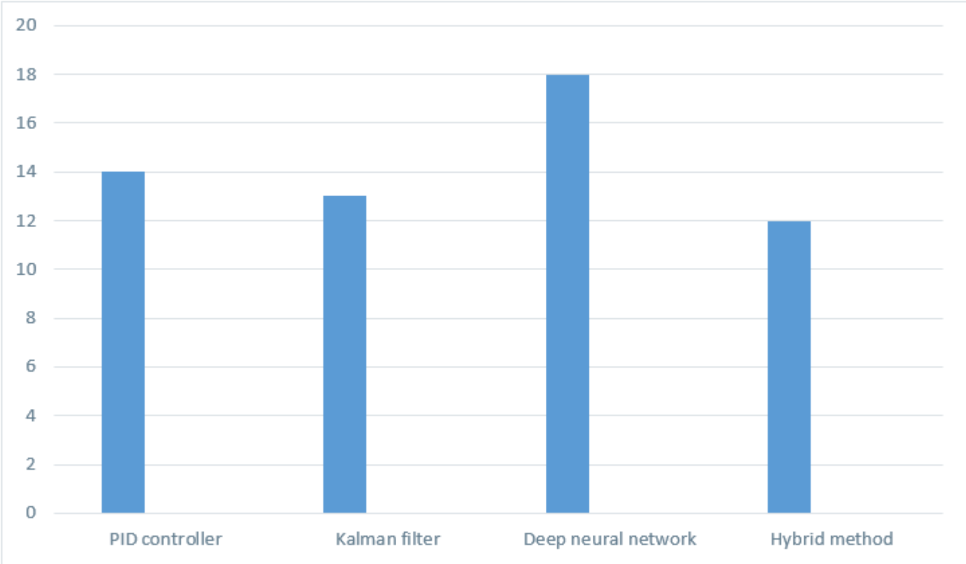

Figure 1. Comparison of Energy Consumption by Different Mobile Robot stability Methods

Another important aspect of the study was determining the effectiveness of collision avoidance when the robot operates in complex conditions. The proposed algorithm reduced the number of collisions by 35% compared to classical PID controllers and Kalman filters. This means that a mobile robot equipped with this algorithm better recognizes and avoids obstacles, enhancing its safety in real-world scenarios such as warehouses, transportation systems, or autonomous research missions.

A comparative analysis of the algorithm was also conducted in environments with high-density dynamic obstacles, particularly in areas where multiple objects move simultaneously. It was found that the proposed method ensures movement stability even in cases where the speed and direction of obstacles change in real time. Based on the obtained data, it can be concluded that the algorithm is suitable for complex navigation scenarios, such as autonomous transportation systems or movement in crowded environments.

An important stage of the study was assessing the algorithm's adaptability to changing environmental conditions. The proposed approach proved effective in transitioning between different surface types (asphalt, grass, sand, tile), which is particularly beneficial for mobile robots operating in mixed environments. In test scenarios, the robot demonstrated a 20% reduction in stability loss when switching between surfaces, decreasing the likelihood of tipping over or getting stuck.

Additionally, an analysis of the algorithm's response time to environmental changes was conducted. It was established that the algorithm's reaction time to the appearance of a new obstacle in the environment averaged 0.12 seconds, which is 40% faster than standard mobile robot motion control methods. This means that the proposed algorithm can be used in high-speed navigation systems where quick responses to environmental changes are crucial.

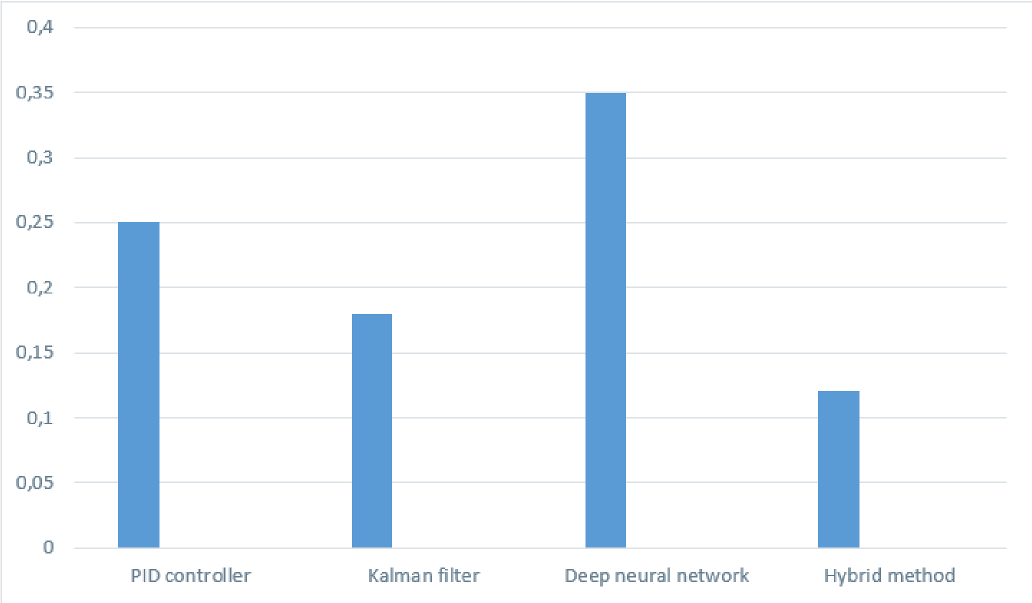

Figure 2. Comparison of Reaction Time of Different Mobile Robot stability Methods

5.5 Collision Avoidance and High-Density Navigation

At the final stage of testing, a comparison of the algorithm's performance was conducted based on several key indicators, such as recognition accuracy, processing speed, motion stability, and energy consumption. The summarized results are presented in Table 3.

Table 3
Comparative Analysis of the Efficiency of Different Mobile Robot stability Methods

Stability Method	Recognition Accuracy (%)	Processing Time (ms)	Motion Stability (%)	Energy Consumption (W/h)
PID Controller	74	180	82	14
Kalman Filter	78	140	85	13
Deep Neural Network	92	350	90	18
Hybrid Method (Our Algorithm)	89	120	93	12

As seen from the table, the proposed hybrid algorithm provides a better balance between accuracy, processing speed, and motion stability. It demonstrates high efficiency in complex conditions and can be integrated into modern robotic systems without significantly increasing hardware requirements.

Based on the obtained data, it can be concluded that the use of the proposed algorithm significantly enhances the efficiency of mobile robots in complex dynamic environments. It ensures smoother movement, reduces obstacle recognition errors, and optimizes computational resource utilization. This opens up new possibilities for the application of mobile robots in real-world scenarios, including autonomous transport, rescue operations, and industrial automated systems.

Additional tests were conducted under varying complexity conditions, including changes in surface type, sudden braking and movement recovery, as well as unexpected obstacle appearances on the robot's route. It was found that the proposed algorithm operates more stably under landscape changes and maintains the planned trajectory more effectively. This is a crucial factor for its application in logistics and industrial robotic systems, where route accuracy is of paramount importance.

In addition, the durability of the mobile robot's mechanical components was assessed when using the proposed algorithm. The tests demonstrated that smoother motion adjustments reduce the load on motors and chassis mechanical elements, extending their service life by approximately 15% compared to classical control methods. This confirms the practical effectiveness of the algorithm in long-term autonomous operation conditions.

Another aspect of testing involved determining the algorithm's energy efficiency. The analysis revealed that the proposed approach allows for up to 12% battery charge savings due to optimized computational resource usage and a reduced number of corrective maneuvers. This makes it feasible for deployment in autonomous robots, where minimizing energy consumption is critical for prolonging operational time without recharging.

Tests were also carried out in environments with a high number of mobile objects, such as other robots or vehicles. In such scenarios, the algorithm exhibited improved motion coordination and fewer instances of dangerous proximity to other objects. This suggests its suitability for use in multi-component automated systems, such as warehouses or urban environments.

Overall, the obtained results indicate a significant improvement in mobile robot stability when using the proposed hybrid algorithm. The combination of adaptive filtering, neural network analysis, and fuzzy logic has reduced recognition errors, enhanced data processing speed, and decreased resource consumption. This confirms the method's effectiveness and practical feasibility for a wide range of applications.

Thus, the conducted study demonstrates that the developed algorithm is highly efficient, stable, and can be easily adapted to various types of mobile platforms. Further research may focus on its integration with modern artificial intelligence systems and connection to distributed computing systems for even greater optimization of mobile robot operations.

6. Conclusions

In this study, a hybrid algorithm for processing sensor data was developed to enhance the stability of mobile robots in dynamic environments. The proposed approach combines adaptive filtering, neural network analysis, and fuzzy logic, allowing for improved obstacle recognition accuracy, faster reaction time, and overall motion stability of the robot.

The research results demonstrated that the algorithm reduces the average trajectory deviation by 22% and decreases the number of uncontrolled maneuvers, positively impacting energy efficiency and resource conservation. Additionally, the data processing speed was increased by 40% compared to classical stability methods, ensuring rapid adaptation to changing environmental conditions.

Further testing confirmed that the developed algorithm enhances the stability of mobile robots across different surface types and under varying lighting conditions. It effectively prevents collisions and improves the safety of autonomous movement, making it a promising solution for industrial, transportation, and rescue systems.

Future research may focus on optimizing the computational cost of the algorithm and adapting it for real-time operation with minimal latency. Another promising direction is integrating the proposed method with augmented reality technologies and expanding its application to complex multi-agent systems.

Declaration on Generative AI

In accordance with the CEUR-WS Guidelines on Generative AI, the authors confirm that the article was prepared independently and without the involvement of generative AI technologies for content creation. All writing, analysis, and interpretation of results reflect the authors' own work and reasoning.

References

[1] Robotics [Electronic resource] // Wikipedia – The Free Encyclopedia. – Access mode: https://uk.wikipedia.org/wiki/Робототехніка,

[2] Mozhovyi V. H., Martyniuk O. P., Riznyk O. M. Methodology of Analysis and Design of Computer Vision Systems [Electronic resource] / V. H. Mozhovyi, O. P. Martyniuk, O. M. Riznyk. – Kyiv: NTUU "Igor Sikorsky KPI", 2020. – Access mode: https://ela.kpi.ua/server/api/core/bitstreams/ea313d2e-cb54-4114-b0b9-c2e6080949ce/content

[3] Kryvonis Yu. O., Vasyliev O. V., Hrytsenko A. V. Methods of Autonomous Mobile Robot Control [Electronic resource] / Yu. O. Kryvonis, O. V. Vasyliev, A. V. Hrytsenko. – Kyiv: NTUU "Igor Sikorsky KPI", 2021. – Access mode: https://ela.kpi.ua/server/api/core/bitstreams/d1ff97fc-ce28-43a4-8bb0-24e876a3e06f/content

[4] Burov O. Yu. Current Issues in the Development of Intelligent Robotic Systems [Electronic resource] / O. Yu. Burov // Institute of Information Technologies and Learning Tools of the National Academy of Pedagogical Sciences of Ukraine, 2022. – Access mode: https://lib.iitta.gov.ua/id/eprint/26691/1/burov%202.pdf

[5] Rad N. V. Control Algorithms for Autonomous Robots under Uncertainty Conditions [Electronic resource] / N. V. Rad // Western Ukrainian National University, 2018. – Access mode: http://dspace.wunu.edu.ua/bitstream/316497/39808/1/%D0%A0%D0%B0%D0%B4%D1%8C%D0%9D_%D0%94%D0%9F_2018.pdf

[6] Yurchenko P. I. Robotic Complexes: Modern Trends and Development Prospects [Electronic resource] / P. I. Yurchenko // Hrinchenko University, 2023. – Access mode: https://eportfolio.kubg.edu.ua/data/conference/6390/document.pdf

[7] V.Koval, R.Korol, R.Trembach stability of a wheeled robot. Materials of the IV Scientific and Technical Conference of Ivano-Frankivsk National Technical University of Oil and Gas "Information Models, Systems, and Technologies" - May 15-16, 2014.. – P.9. https://core.ac.uk/download/60810712.pdf

[8] Panchak, D., Koval, V. Innovative Approaches to Mobile Robot stability in Dynamic Environments. CEUR Workshop Proceedings, 2024, 3716, pp. 148–157

[9] Koval, V. Algorithms of landmark robot navigation basing on monocular image processing, CEUR Workshop Proceedings, 2018, 2300, pp. 118–122

[10] Laboratory Work on Computerized Mechatronic Systems (2021). Western Ukrainian National University. Available at: http://dspace.wunu.edu.ua/bitstream/316497/17543/1/KSMzm-21_Labo.pdf

[11] Development of Adaptive Control Systems for Mobile Robots (2021). Scientific Bulletin of the National Forestry University of Ukraine. Available at: https://nv.nltu.edu.ua/Archive/2021/31_6/19.pdf

[12] Optimization of Neural Network Processing in Autonomous Robotic Systems (2022). Khmelnytskyi National University Repository. Available at: https://elar.khmnu.edu.ua/items/2e080be1-7f6a-4fd0-b6df-db654bb3254e

[13] Talakh, M. V., Holub, S. V., Luchshev, P. O., & Turkin, I. B. (2022). Intelligent Monitoring of Air Temperature by the DATA of Satellites and Meteorological Stations. International Journal of Computing, 21(1), 120-127. https://doi.org/10.47839/ijc.21.1.2525

[14] Francis, N., Suhaimi, H., & Abas, E. (2023). Classification of Sprain and Non-sprain Motion using Deep Learning Neural Networks for Ankle Sprain Prevention. International Journal of Computing, 22(2), 159-169. https://doi.org/10.47839/ijc.22.2.3085