



Article

Controlling of Unmanned Underwater Vehicles Using the Dynamic Planning of Symmetric Trajectory Based on Machine Learning for Marine Resources Exploration

Yuriy Kozhubaev ^{1,*} , Victor Belyaev ¹, Yuriy Murashov ²  and Oleg Prokofev ²

¹ Department of Informatics and Computer Technologies, St. Petersburg Mining University, 2, 21st Line, St. Petersburg 199106, Russia; belyaev_vv@pers.spmi.ru

² Institute of Computer Science and Technology, Higher School of Cyberphysical Systems & Control, Peter the Great St. Petersburg Polytechnic University, St. Petersburg 195251, Russia; murashov_yuv@spbstu.ru (Y.M.); prokofiev_ov@spbstu.ru (O.P.)

* Correspondence: kozhubaev_yun@pers.spmi.ru

Abstract: Unmanned underwater vehicles (UUV) are widely used tools in ocean development, which can be applied in areas such as marine scientific research, ocean resources exploration, and ocean security. However, as ocean exploration advances, UUVs face increasingly challenging operational environments with weaker communication signals. Consequently, autonomous obstacle avoidance planning for UUVs becomes increasingly important. With the deepening of ocean exploration, the operational environment of UUVs has become increasingly difficult to access, and the communication signals in the environment have become weaker. Therefore, autonomous obstacle avoidance planning of UUVs has become increasingly important. Traditional dynamic programming methods face challenges in terms of accuracy and real-time performance, requiring the design of auxiliary strategies to achieve ideal avoidance and requiring cumbersome perception equipment to support them. Therefore, exploring an efficient and easy-to-implement dynamic programming method has significant theoretical and practical value. In this study, an LSTM-RNN network structure suitable for UUVs was designed to learn the dynamic programming mode of UUVs in an unknown environment. The research was divided into three main aspects: collecting the required sample dataset for training deep networks, designing the LSTM-RNN network structure, and utilizing LSTM-RNN to achieve dynamic programming. Experimental results demonstrated that LSTM-RNN can learn planning patterns in unknown environments without the need for constructing an environment model or complex perception devices, thus providing significant theoretical and practical value. Consequently, this approach offers an effective solution for autonomous obstacle avoidance planning for UUVs.

Keywords: deep learning; dynamic planning; LSTM-RNN; ant colony algorithm; UUV



Citation: Kozhubaev, Y.; Belyaev, V.; Murashov, Y.; Prokofev, O. Controlling of Unmanned Underwater Vehicles Using the Dynamic Planning of Symmetric Trajectory Based on Machine Learning for Marine Resources Exploration. *Symmetry* **2023**, *15*, 1783. <https://doi.org/10.3390/sym15091783>

Academic Editors: Beloglazov Ilya and Rudolf Kawalla

Received: 31 May 2023

Revised: 18 July 2023

Accepted: 19 July 2023

Published: 18 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Today's unmanned underwater vehicles (UUVs) face major challenges, such as complex marine environments, weakened communication signals, limited energy and endurance, limitations in perception and sensing capabilities, and the complexity of autonomous obstacle avoidance and path planning in difficult operating environments. Solving these challenges is essential to improve the performance and effectiveness of UUV in difficult operating environments [1].

The weakening of communication signals in the environment has multiple effects on the performance of unmanned underwater vehicles (UUV). First of all, the communication range is limited, which limits the effective communication distance between the UUV and the ground control station or other equipment. Secondly, the communication rate drops, causing the data transmission speed to slow down, which may delay the operation response time. In addition, weakened communication signals increase the difficulty of task planning

and coordination, affecting the efficiency of team collaboration and task allocation [2]. Most importantly, the weakening of the communication signal also increases the safety risk of UUV in the marine environment. Due to the inability to receive instructions in time or perform emergency shutdown operations, it may lead to potential collisions or other dangerous situations. In order to deal with these effects, it is necessary to improve the performance of communication equipment, improve communication protocols, and optimize signal transmission technology, while strengthening the autonomy and intelligence capabilities of UUV to reduce dependence on ground control and explore reliable and robust communication solutions to improve the performance and reliability of UUV in difficult operating environments. In this article, artificial intelligence technology is used to enable navigators to have autonomous planning functions and reduce dependence on ground communication.

The unknown obstacle movements in complex operational environments and the uncertainty of underwater detection are constraining the development of UUV autonomous obstacle avoidance technology [3]. The UUV autonomous obstacle avoidance planning system is a dynamic nonlinear system, and deep learning technology can establish a dynamic nonlinear description between the input space and the target space through continuous training, without artificially establishing a state space describing the problem. It also has strong adaptability to disturbances in the input space. Therefore, this study conducted in-depth research on UUV autonomous obstacle avoidance planning methods based on deep learning technology.

Path planning refers to finding a collision-free and safe path from the starting point to the endpoint, and optimizing specific planning indicators (such as the shortest path, minimum risk, maximum task completion, etc.) [4]. Path planning algorithms have been extensively studied domestically and internationally, with commonly used ones including A* algorithm, Dijkstra algorithm, and the swarm intelligence and genetic algorithm. Dynamic obstacle avoidance is essentially real-time path planning, which puts forward requirements for the planning speed of the algorithm. The reason why dynamic obstacle avoidance is singled out here is that in complex environments, simply calling the path planning algorithm in real-time may not achieve ideal results, especially when there are moving obstacles. Therefore, other auxiliary avoidance strategies often need to be designed to meet the requirements [5].

Traditional dynamic programming systems, as shown in Figure 1, prioritize real-time performance. However, the construction of an environment model and the iterative process of intelligent algorithms hinder real-time performance the most. Furthermore, an accurate environment model depends on precise sensing equipment, and planning auxiliary strategies often require accurate auxiliary information (such as velocity and direction), which can result in high implementation costs. Therefore, exploring a more cost-effective, simple, and reliable dynamic programming method is of great significance for reducing algorithm investment, lowering costs, and improving system versatility [6,7].

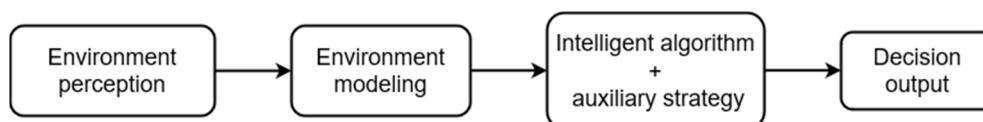


Figure 1. Traditional dynamic programming system.

The limitations of traditional dynamic planning methods in UUV obstacle avoidance include dependence on accurate environmental models, difficulties in dealing with complex obstacles and dynamic conditions, real-time limitations, and limitations of hard-coded strategies. In order to overcome these limitations, in recent years, researchers used artificial intelligence technologies such as deep learning and reinforcement learning to propose some new methods and algorithms to improve UUV obstacle avoidance performance and autonomy.

Deep learning (DL) is undoubtedly the most promising algorithm for achieving this goal. Any planning system is nothing more than a mapping from perception to decision [8]. As shown in Figure 1, the environment model, auxiliary strategy, and intelligent algorithm are a series of mapping functions. The deep learning network has powerful nonlinear expression ability due to its special structure. It only requires very simple perception devices, can learn the mapping relationship between input and output through sufficient training, and store the learned weights on the connections between neurons. Therefore, the dynamic programming system based on deep learning will become the structure shown in Figure 2. The trained learning algorithm only needs several matrix multiplication and addition calculations in the planning process, without looping iteration, and has good real-time performance.

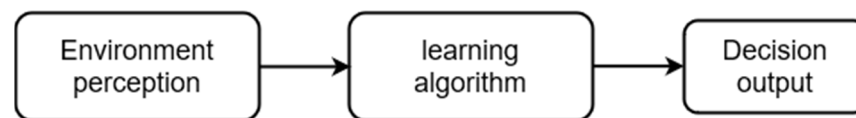


Figure 2. Learning-based dynamic programming system.

In summary, compared with traditional dynamic planning methods, artificial intelligence dynamic planning methods have the advantages of stronger self-learning and adaptability in UUV obstacle avoidance, no need for accurate environmental models, strong adaptability and high flexibility, and improved accuracy and real-time. These advantages make the DL dynamic planning method an effective means to improve the obstacle avoidance performance of UUV [9,10].

Sustainability requires us to rationally manage and protect marine resources to ensure their long-term sustainable use. Traditional UUV operations may face the risk of environmental damage and pollution. The LSTM-RNN method can help UUV avoid obstacles more intelligently during operations, reduce conflicts and collisions with the environment, and, thereby, reduce its negative impact on the marine ecosystem. At the same time, it would plan paths more efficiently and avoid conflicts, thereby reducing unnecessary energy consumption. This helps to improve the energy efficiency of UUV, reduce the demand for limited energy resources, and is in line with the principle of sustainable development.

In this paper, the research on symmetric dynamic path planning of unmanned underwater vehicles based on deep learning had the following innovations:

1. Introduction of deep learning methods: This study applied deep learning methods to the field of dynamic path planning for underwater unmanned vehicles (UUVs). Traditional dynamic programming approaches face challenges in complex environments, while deep learning methods have the ability to handle complex data and learn environmental patterns. By introducing the LSTM-RNN network structure, this research combined deep learning with path planning, enabling UUVs to learn and apply dynamic planning patterns for autonomous obstacle avoidance in unknown environments.
2. Design without models or perception devices: Unlike traditional methods, this research did not rely on complex environment models or cumbersome perception devices to support the decision-making process. By training the deep learning network, UUVs can directly learn environmental patterns from actual observation data and autonomously plan obstacle avoidance in real-time environments. This model-free, perception-device-free design is significant for improving the practicality and reliability of the system and is innovative in its application in underwater environments.
3. Applicability to unknown environments: Another scientific novelty of this research lies in its applicability to unknown environments. Traditional approaches often require prior knowledge of environmental characteristics and parameters, while in practical applications, environmental changes and uncertainties are common. By training the deep learning network to learn planning patterns in unknown environments, UUVs can autonomously plan obstacle avoidance in real-time environments, whether the

environment has been previously explored or modeled. This capability is important for expanding the application domains of UUVs and enhancing their adaptability.

During the research process, we gained the following new knowledge:

1. Application of LSTM-RNN networks in dynamic path planning for UUVs: We verified the effectiveness of the LSTM-RNN network structure in the task of dynamic path planning for UUVs. By training the network, we enabled UUVs to learn and infer patterns in the environment, and make path planning and obstacle avoidance decisions based on the learned patterns.
2. Model-free, perception-device-free autonomous obstacle avoidance method: We developed a deep-learning-based method that achieved autonomous obstacle avoidance for UUVs in unknown environments without the need for prior environment modeling or reliance on perception devices. This approach simplified the complexity of the system and improved its practicality.
3. Performance validation in practical scenarios: Through experimental results, we validated the performance of the method in real-world applications of UUVs. The experimental results demonstrated that using LSTM-RNN networks can effectively learn planning patterns in unknown environments and achieve autonomous obstacle avoidance in practical applications, providing strong support for the actual application of UUVs in complex environments.

In summary, this research offers scientific novelty by introducing deep learning methods and designing a model-free, perception-device-free autonomous obstacle avoidance method for UUVs. The new knowledge gained during the research not only expands the theoretical foundation of the field but also provides important insights for enhancing the performance and intelligence development of UUVs in practical applications.

2. Materials and Methods

This chapter mainly implements the basic static planning algorithm of the teacher system, while dynamic planning is a reasonable appeal to the algorithm using some strategies. This chapter section three specific parts. The static path planning algorithm was based on the ant colony algorithm to achieve path optimization, and dynamic planning rules were formulated for emergency response, and the LSTM-RNN neural network was designed and adjusted.

- Using ant colony algorithm for static planning:

Ant colony optimization (ACO) is a bio-inspired intelligent algorithm that simulates the foraging behavior of ants. In nature, after carrying food for a certain period of time, almost all ants will crawl along the shortest path from the nest to the food source. This is because ants release a type of pheromone during crawling, which evaporates over time. Subsequent ants tend to choose the path with a higher concentration of pheromone, and the shorter the path from the nest to the food source, the less the pheromone evaporates [7]. More ants will follow the path with a higher pheromone concentration, which increases the pheromone concentration further and attracts more ants to crawl, thus forming a positive feedback loop. Ultimately, most ants tend to crawl on the shortest path, which is the optimal convergence process in nature. ACO simulates this process and establishes a mechanism for releasing and evaporating pheromones. However, artificial ants are different from natural ants in that they can have memory functions and heuristic information can be introduced in the process of exploring paths. The key to implementing ACO is designing pheromone updating rules and state transition rules.

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (1)$$

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{When ant } k \text{ passes by edge } e_{ij} \\ 0 & \text{other} \end{cases} \quad (2)$$

Among them, i, j —status point number, m —the total number of ants in the ant colony, ρ —volatility coefficient, Q —constant, L_k —the price paid by ant k foraging, which is taken as the path length.

To accelerate the convergence speed, the paths taken by the top 25% ranked ants in each generation are reinforced with pheromone:

$$\tau_{ij}(t+1) = \lambda \times \tau_{ij}(t) \quad (3)$$

$$\lambda = 1 + 0.5 \times \frac{D}{L_k} \quad (4)$$

Among them, D —distance from start point to end point.

In order to make the algorithm still have a certain exploratory ability in the later stage of the search, the pheromone concentration is limited after the pheromone is updated and strengthened:

$$\tau_{ij}(t+1) = \begin{cases} \tau_{min}, & \text{if } \tau_{ij}(t+1) < \tau_{min} \\ \tau_{max}, & \text{if } \tau_{ij}(t+1) > \tau_{max} \\ \tau_{ij}(t+1), & \text{other} \end{cases} \quad (5)$$

Among them, τ_{min} , τ_{max} —self-set upper and lower limits of pheromone concentration.

The state transition rule refers to the selection rule for the next state when the ant transitions from the current state to the next. In the application of path planning, this refers to the selection of visible points at the current position, as typically there are multiple visible points at any given location, and the choice of which visible point to move to next is the most important aspect to optimize the planning process. In this paper, probability selection is used to implement probabilistic transition, and the state transition probability is designed as follows:

$$p_{ij}^k = \frac{(\tau_{ij})^\alpha (\eta_j)^\beta}{\sum_{r \in allow_i} (\tau_{ir})^\alpha (\eta_r)^\beta} \quad (6)$$

This formula describes that ant k is currently at point i , and the probability of its next move to visible point j of i is p_{ij}^k . Among them, $allow_i$ —a set of visible points for ant k that allows state transition of point i , η —heuristic function, α —the importance of pheromones, and β —importance of heuristics;

- System dynamic programming strategy design:

Global planning refers to the process of planning a safe path based on known environmental information. However, the working environment of unmanned underwater vehicles (UUVs) is usually unknown and may contain unknown obstacles, hazards such as mines, and even intelligent threats such as torpedoes and surface vessels [8]. Additionally, UUVs may need to change their mission plan due to internal reasons or urgent requirements. Therefore, we call this planning method that can respond to emergencies as dynamic planning. In practical applications, UUVs usually act as an off-board resource of the host vessel and collaborate with other devices. There are two ways to obtain unknown environmental information: communication-based and self-sensing through sensors carried by UUVs. Dynamic planning requires UUVs to respond promptly to new environments, which is essentially the same as global planning but with a faster requirement. The algorithm adopted in this article is dependent on the complexity of the environmental factors, and the more complex the environment, the longer the planning time required.

In the dynamic planning strategy designed in this paper, UUV obtained unknown environmental information through autonomous sensing. For dynamic planning of static obstacles, this paper's design method had the following three characteristics:

In each dynamic planning iteration, all known obstacles were included in the planning process rather than just focusing on information within the environmental window. Additionally, the endpoint was always used as the target point, and no temporary targets were constructed.

This paper stored all static obstacles within the sensing range of UUV in a known information linked list, and performed global planning based on the information in the list for each planning cycle. Although this approach led to a continuous increase in environmental information involved in dynamic planning, it also enhanced the “memory” function of the algorithm and prevented UUV from oscillating in traps. Due to the high efficiency of the ant colony planning algorithm proposed in this paper, it can store a large amount of “memory”. If excessive “memory” is found to cause slow response of UUV, some distant “memory” can be discarded.

Dynamic programming adopts an unknown obstacle-triggered mechanism instead of a fixed time interval triggering mechanism.

During the UUV’s navigation, if it encounters information that does not exist in the known information linked list, the information will be added to the list and trigger planning. The advantage of this approach is that it can respond in a timely manner to newly discovered unknown obstacles while avoiding triggering dynamic planning too frequently, thus reducing unnecessary computation. Even if planning is triggered multiple times when no new environmental information appears, the UUV’s path will not change significantly. The triggering mechanism is also an important step in avoiding moving obstacles.

If any part of an unknown obstacle enters the UUV’s perception range, the complete information of the obstacle will be stored in the known information linked list, giving the UUV a “foresight” capability.

Although this does not correspond to the real world, it simplifies the simulation planning process. The use of unrealistic “prophetic” assumptions to simplify models or computations is not unique to this study, but rather has been widely applied in various fields [9]. A typical example is the Naive Bayes classification method, which assumes conditional independence of the probability distribution. This is a very strong assumption, and the “naive” in its name reflects this fact. With the assumption of conditional independence, the following equation holds.

$$P(X = x|Y = c_k) = P(X^{(1)} = x^{(1)}, \dots, X^{(n)} = x^{(n)} | Y = c_k) \quad (7)$$

Among them, input space $\chi \subseteq R^n$, output space $\gamma = \{c_1, c_2, \dots, c_k\}$, input features $x \in \chi$, output features $Y \in \gamma$. The assumption of conditional independence considers each feature used for classification under specific conditions to be conditionally independent. Therefore, the final product rule symbol holds. It is evident that the features are not truly conditionally independent, but this assumption simplifies the Bayesian method and usually yields optimistic classification results.

- Design of UUV Dynamic Programming based on LSTM-RNN:

Long short-term memory (LSTM) is a highly successful method for solving the problem of long-term dependency in recurrent neural networks (RNNs), and has been widely applied in the field of natural language processing (NLP) [10]. In this study, a simulated sensor was designed based on the forward-looking sonar detection feature of UUVs, which was used to collect and generate a sample set in the system. Next, LSTM was used to address the issue of long-term dependency in RNNs, and an LSTM-RNN structure suitable for UUV dynamic programming was designed. Finally, the LSTM-RNN was trained using the sample set and its learning effect was validated.

The primary feature of LSTM networks is the replacement of the hidden layer in traditional RNN networks with a memory module, as illustrated in Figure 3. The memory module included a cell, which serves as a storage location for the current state of the hidden layer. Additionally, there were three important gate units within the memory module that control the input, output, and forgetting of the cell through multiplication units. The dashed line in the figure represents the “peephole connection,” which indicates that the gate units adjust their controls based on the current state of the Cell. Unfolding the LSTM network revealed the flow of signals between the memory module and the gate units, as depicted in Figure 4. Using this model for training, under ideal conditions, it was possible

to achieve absolute memory updates and forgetting by replacing the activation function of the gating unit.

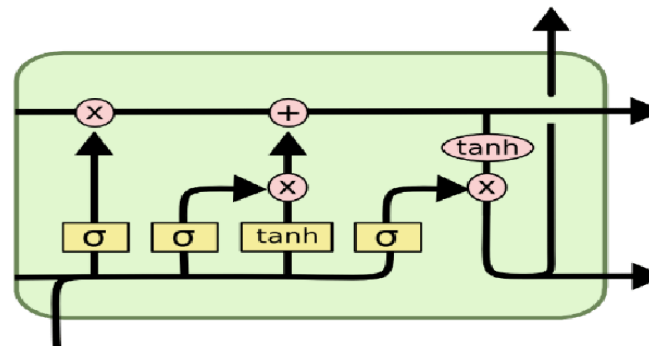


Figure 3. LSTM Memory Module.

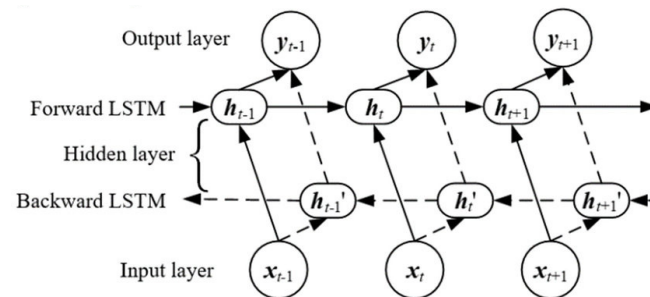


Figure 4. Expansion of LSTM network.

The design of the gate unit enabled the LSTM memory module to store and access information from a long time ago, thereby alleviating the problem of vanishing gradients. For example, as long as the input gate remains closed, the state of the cell will not be overwritten, allowing the state from a long time ago to influence the output far into the future. Figure 5 illustrates the process of gradient preservation in LSTM, where the “○” and “—” in the hidden layer represent the fully open and fully closed states of each gate unit, respectively. It can be seen that from the first input entering the memory module until the seventh step, the input gate remained closed and the forget gate was fully open, meaning that the cell retained 100% of its memory. This is shown as orange circles in Figure 5 [11–13]. Therefore, even if there are new inputs later on, the state of the cell will not be overwritten and the state of the first input can be obtained at any time by opening the output gate. At the seventh step, the input gate was fully open and the forget gate was closed, causing the cell to forget its previous memory and update its state. Figure 5 analyzes an extreme example, as the activation function of the gate unit is the sigmoid function, which ranges from 0 to 1 and rarely reaches the extremes of “0” or “1”. As a result, the state of the cell will always be slowly forgotten or updated [14–16].

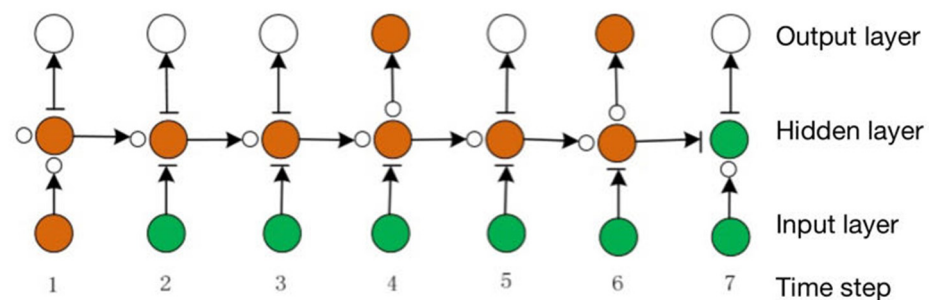


Figure 5. LSTM memory protection process.

The LSTM-RNN network structure design of UUV is to realize dynamic planning mode learning and obstacle avoidance decision-making in unknown environments. The design of the LSTM-RNN network structure was divided into the following parts:

Input Layer: The input layer receives UUV perceptual data, such as sensor readings, images, or depth images. These data are used to provide information about the environment and obstacles to support obstacle avoidance decisions.

LSTM Layer (LSTM Layer): The LSTM (long short-term memory) layer is a variant of RNN, which has the structure of a memory unit and a gating unit. The LSTM layer can process sequence data and can effectively capture and remember long-term dependencies in the sequence. In the LSTM-RNN network structure of UUV, the LSTM layer is used to learn and remember the dynamic planning mode of UUV in an unknown environment.

Hidden Layer: The hidden layer is the middle layer of the network, which is used to extract and combine features in the input data. Neurons in the hidden layer can introduce nonlinear relationships through activation functions to increase the expression ability of the network.

Output Layer: The output layer generates corresponding obstacle avoidance decisions based on the learned dynamic planning mode. The output can be one or more continuous variables or discrete categories, representing the action or path planning that the UUV should take.

Backpropagation algorithm (Backpropagation): In the process of network training, the backpropagation algorithm is used to adjust the network parameters so that the network output is as close as possible to the desired output. In this way, the network can gradually learn the optimal dynamic planning mode and obstacle avoidance strategy.

The design of the LSTM-RNN network structure needs to be adjusted and optimized according to specific problems and data characteristics. In practical applications, the performance and adaptability of the network can be improved by adjusting the number of layers of the network, the number of neurons, and the selection of activation functions. In addition, regularization techniques (such as Dropout) and optimization algorithms (such as Adam, SGD) can also be used to improve the training and generalization capabilities of the network [17,18]. The LSTM-RNN network structure of UUV can learn and remember the dynamic planning mode of UUV in an unknown environment by using the memory and sequence processing power of LSTM, thereby supporting the obstacle avoidance decision-making of UUV. The design of this network structure can be adjusted and optimized according to specific problems and needs to achieve better performance and results.

3. Results

This part mainly analyzes the model training results, and uses the teacher system and the LSTM-RNN system to conduct comparative experiments to verify the dynamic programming simulation function of the system [19].

It is worth mentioning that the optimal path obtained by the LSTM-RNN method was symmetric, that is, whether it was from the starting point to the end point or from the end point to the starting point, the results of the two experiments were the same. The same error loss and motion trajectory were not available in the teacher system. In the optimal trajectory problem, symmetry is very important. Through symmetry, we can intuitively verify whether the experimental result is the optimal result, the optimal trajectory must be symmetrical.

3.1. UUV Dynamic Planning LSTM-RNN Training

In this paper, the well-known robot simulation tool ROS (Robot Operating System) was used to generate and simulate UUV data, and related algorithm development and training were carried out.

In this study, we took the following steps: first, use the corresponding software to automatically generate sample data, and use the LSTM-RNN model to train it to meet specific requirements. Then, the software was used to randomly generate obstacles, and

then, the appropriate tools were used to calculate the optimal path. We used the teacher system and the LSTM-RNN intelligent system to automatically plan the path, and record the heading and speed of the vehicle in the process. By comparing the difference between the actual path and the planned path, we can calculate the final error. Using the obtained speed and heading angle data, we can analyze the operating state of the vehicle. These steps will help to evaluate and verify the performance and effectiveness of the LSTM-RNN model in autonomous obstacle avoidance and path planning of UUV.

After sampling, this study obtained a training set with a size of 40,000 and a test set with a size of 1000. During the training process, the average error (calculated using the loss function) was output every 500 iterations, and 50 sequences were randomly selected from the test set for testing using the current weights, and the average error was output. The convergence property of the error on the test set reflects the generalization ability of the network, with smaller test errors indicating stronger generalization ability [20–22].

Table 1 shows the convergence process of heading during 4 million training iterations, the error value is the average value. It can be seen that the average error of heading converged to about 0.19 radians, or approximately 11 degrees, on the test set. Although this may seem like a relatively large error, analysis revealed that 95% of the heading errors converged to within 2 degrees. Clearly, this was due to the cyclical nature of the heading angle, and the error evaluation simply treated it as a linear numerical value within the interval $(-\pi, \pi)$, such assessments do not carry negative consequences; in addition, the speed error in the test set converged to within 1 kn, which is an acceptable error range.

Table 1. Model test error table.

Training Times	1,000,000	2,000,000	3,000,000	4,000,000
Heading error	0.28 rad	0.24 rad	0.21 rad	0.19 rad
Speed error	1.2 kn	1.0 kn	0.85 kn	0.81 kn

3.2. Simulation Verification of System Dynamic Planning

After the network training was completed, the weight matrices were saved and the network was used to control the UUV to sail in the simulation training ground. The input sequence of the network was the latest 10 sets of continuous perceptual data detected by the analog perceptron. When the UUV was distributed, all perceptual data were initialized to a zero vector. As the UUV moved, the input sequence was constantly updated [23–25].

Figure 6 is experimental scenario 1, which shows the dynamic planning effect of LSTM-RNN and the teacher system in a static unknown environment. It can be seen that LSTM-RNN learned to avoid obstacles in static unknown environments, and the heading change was quite smooth.

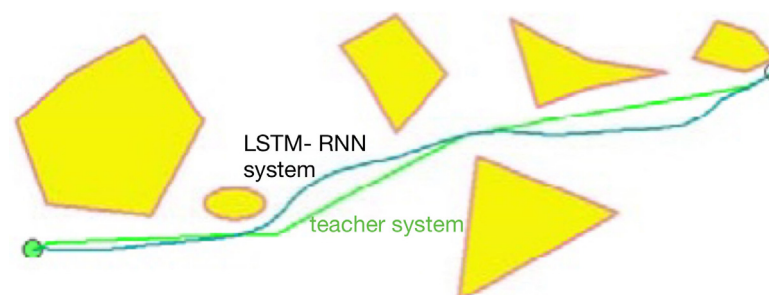


Figure 6. Experimental scenario 1: Comparison of planning in a static unknown environment.

Figure 7 is a comparison of the output values of the teacher system and LSTM-RNN in scene 1. The teacher system updated the heading by triggering the planning mechanism of unknown obstacles, so its heading changed in a stepped manner, while LSTM-RNN was in a continuous planning state, and the heading and speed were updated all the time.

The teacher system maintained uniform speed movement in scenes with static unknown obstacles, LSTM-RNN learned this characteristic, and the speed jitter was very small.

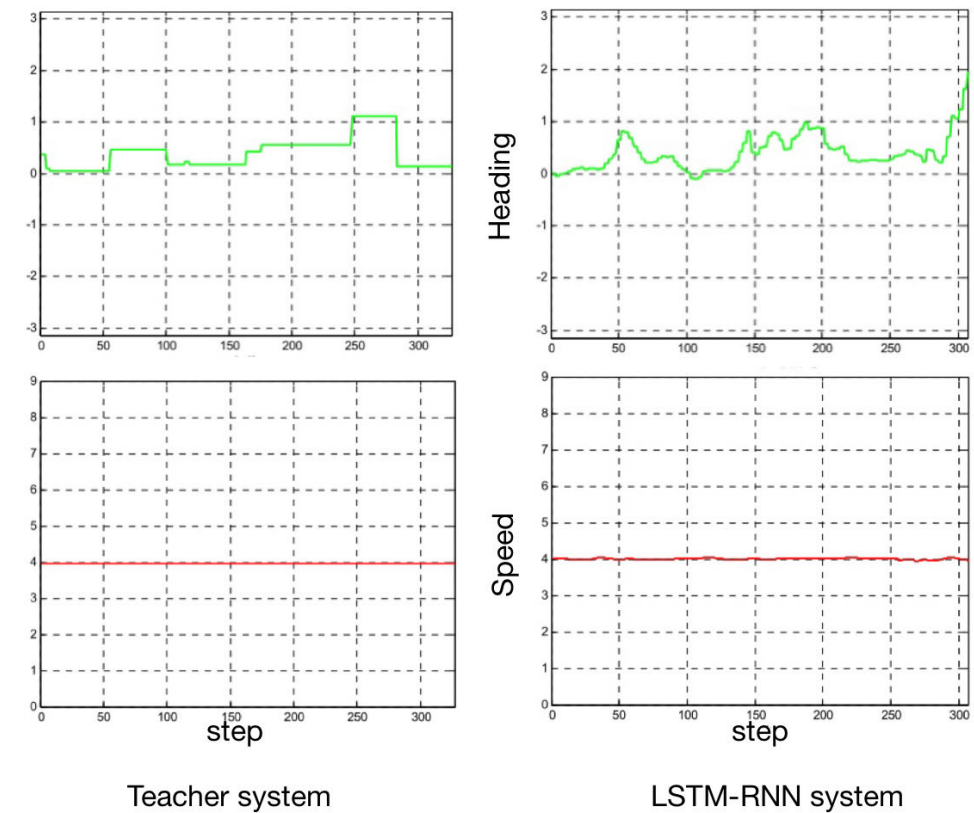


Figure 7. Experimental scenario 1: LSTM-RNN and teacher system output values.

Experimental scenario 2 shows the dynamic planning effect of LSTM-RNN and the teacher system when facing movement disorders. In order to make the effect clearer, only one movement obstacle was placed in the scene, which was in a state of uniform linear motion [26–28].

Figures 8 and 9 show the situation of UUV sailing against the movement barrier. The strategy adopted by the teacher system was to change the course and detour at the maximum speed V_{max} . After leaving the danger zone, the speed dropped to the cruise speed U . The effect of LSTM-RNN output was similar.

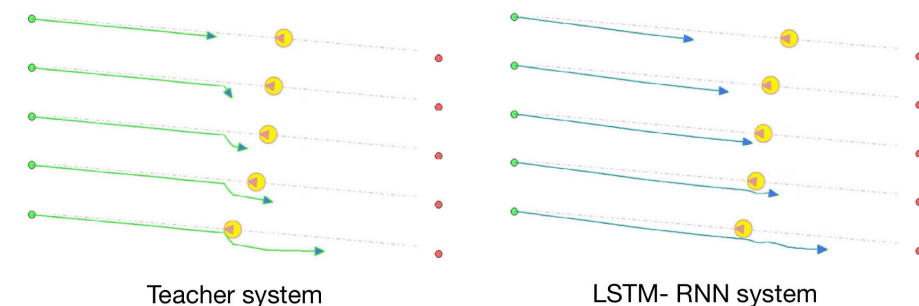


Figure 8. Experimental scenario 2: The dynamic planning effect of UUV and movement disorders when sailing against each other.

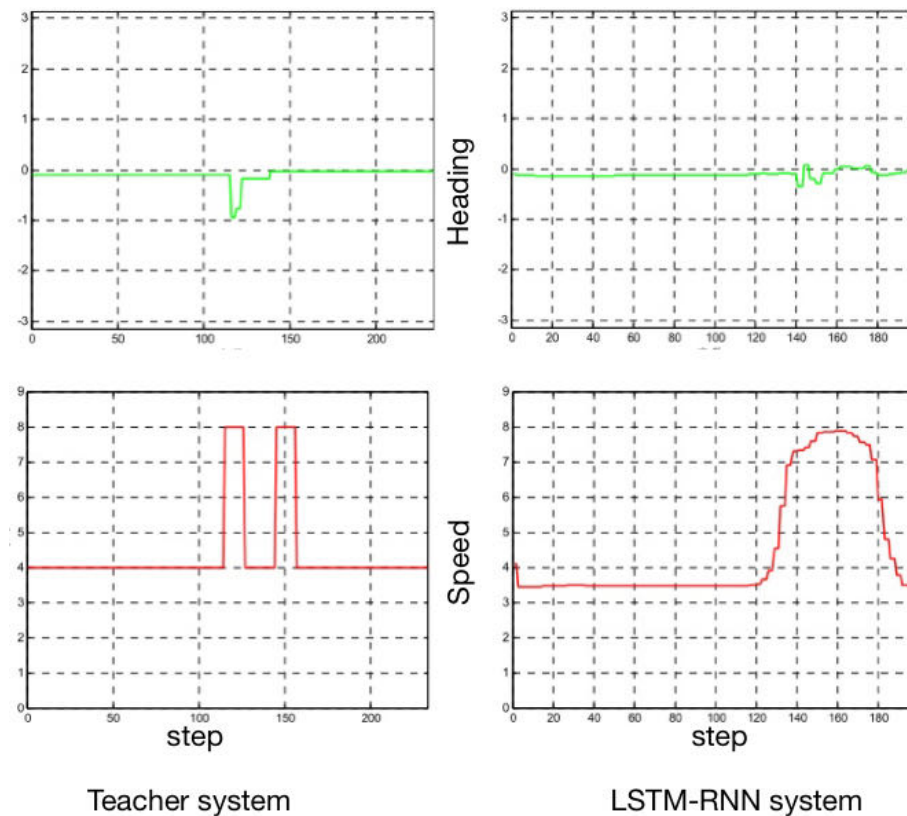


Figure 9. Experimental scenario 2: LSTM-RNN and teacher system output values.

Through the comparison in scenario 1, it can be concluded that in terms of overall performance, underwater vehicles using the LSTM-RNN model performed better than traditional methods. By learning the dynamic planning patterns in the unknown environment, the LSTM-RNN model enabled underwater vehicles to more intelligently avoid obstacles and find the optimal path. Compared with the traditional method, the LSTM-RNN model had higher autonomy and adaptability, thereby improving the overall performance and efficiency.

Through the comparison in scene 2, it can be concluded that in terms of detail processing, underwater vehicles using the LSTM-RNN model also showed better performance. The LSTM-RNN model can sequence the input data and capture the time dependence and dynamic laws in the data. This allowed underwater vehicles to better handle complex details, such as rapidly changing obstacles or emergencies. Compared with the traditional method, the LSTM-RNN model can make more accurate decisions and adjustments, and improve the accuracy and responsiveness of detail processing.

From the above two basic experimental scenarios, it can be seen that LSTM-RNN basically learned the planning method of the teacher system, and the model function was effective [29–31]. Using LSTM-RNN to realize dynamic planning in an unknown environment, it can be found that LSTM-RNN can learn and discover the optimal path in the unknown environment. By analyzing and learning a large amount of sample data, the LSTM-RNN network can capture the dynamic planning patterns in the environment and find the optimal path in the obstacle avoidance decision-making process. This means that UUV can intelligently avoid obstacles in order to achieve efficient goal achievement, and the LSTM-RNN network structure gave UUV higher autonomy.

By learning planning patterns in unknown environments, UUV can reduce its dependence on precise environmental models and make autonomous decisions. The LSTM-RNN method is a data-driven learning method that obtained the dynamic planning pattern of the environment by learning a large amount of sample data. Compared with traditional

methods that require accurate environmental models, the LSTM-RNN method learned directly from raw data and did not rely on pre-built environmental models. This makes the use of LSTM-RNN method in unknown environments more flexible and adaptable. The LSTM-RNN method can directly use sensor data as input without the need for complex sensing equipment. Sensor data can be various sensor readings of UUV, such as lidar, camera, sonar, etc. The LSTM-RNN method realized the perception and understanding of the environment by learning the environmental characteristics in the sensor data, avoiding the need to rely on cumbersome perception equipment. This enhancement of autonomy enabled UUV to better adapt to different environmental conditions and task requirements, improve its application capabilities in complex environments, and the LSTM-RNN network structure exhibited strong anti-interference performance in the face of interference and noise.

Because the network had memory units and recursive connections, it can adapt and adjust when encountering uncertainties and changes in the environment to provide more stable and reliable obstacle avoidance decisions. This allowed UUV to cope with the complex and dynamically changing marine environment, reducing the risk of misjudgment and conflict.

There were mainly the following five difficulties in applying this research to practice:

1. Data collection and labeling: Obtaining sufficient training data and labeling can be a challenge. Large-scale, high-quality data sets are essential for training LSTM-RNN models. In addition, labeling data may require professional domain knowledge and time costs.
2. Network design and tuning: Designing and optimizing the LSTM-RNN network structure for optimal performance may require a lot of experimentation and tuning. Choosing the appropriate network architecture, number of layers, number of neurons, and activation function is a complex task.
3. Computational resource and time cost: The LSTM-RNN method usually requires large computational resources and time to train the model. The complexity of the model and the size of the training data set may lead to increased demand for computing resources, and the training time may be longer.
4. Real-time requirements: In practical applications, the LSTM-RNN method may need to meet the real-time requirements, that is, to generate obstacle avoidance strategies within a limited period of time. Real-time requirements may require further optimization and acceleration of the algorithm.
5. Environmental complexity: The complexity and dynamic changes of the actual marine environment may have an impact on the performance of the LSTM-RNN method. Dealing with complex terrain, ocean currents, multiple moving obstacles, etc., may require higher model robustness and generalization capabilities.

4. Discussion and Future Work

Against the backdrop of the booming development of artificial intelligence, this study attempted to explore a deep-learning-based dynamic path planning method that is suitable for the working environment and perception characteristics of unmanned underwater vehicles (UUVs), so as to make it simpler, cheaper, and more efficient in practical applications. The main design work of this paper focused on two parts: the design of the overall system and the implementation of dynamic programming based on LSTM-RNN [32–35].

This paper applied the LSTM-RNN network to the field of dynamic programming for UUVs for the first time, while achieving certain results, there were also deficiencies and expandable areas, such as the following:

1. System improvement. The system designed in this paper performed well in terms of real-time and planning effectiveness, but the planned path was not smoothed, and when it is in a complex multi-motion obstacle environment, strategy conflicts may occur, leading to slow or oscillatory UUV movement [36–39].

2. Data cleaning. During the sampling process, there may be a small number of erroneous samples due to system defects, and the entire dataset is concatenated from sampling data in many different random environments, resulting in a large number of sequence discontinuities. Therefore, in the case of sufficient resources, it is advisable to create independent sequence-label sample pairs instead of randomly selecting sequences from the training dataset. This approach can eliminate erroneous samples and avoid the issue of sequence discontinuity.
3. Training on a larger dataset. Due to equipment limitations, the dataset used in this paper was relatively small, and a larger dataset will undoubtedly lead to better performance.
4. Using deep reinforcement learning. This approach can enable the system to automatically try and learn from errors without the need for sampling, achieving true intelligence implementation

In addition, the practical significance of the research results was reflected in the following aspects:

1. Enhanced autonomy and safety: By utilizing the LSTM-RNN network to learn dynamic planning patterns, UUVs can autonomously navigate through complex and unknown underwater environments. This capability reduces the reliance on human operators for real-time decision-making, increasing the autonomy of UUVs and improving their overall safety during underwater missions.
2. Efficient resource exploration: UUVs are widely used in marine resource exploration, and accurate path planning is crucial for maximizing the efficiency of these missions. The proposed approach enabled UUVs to dynamically plan their paths in unknown environments without the need for extensive environment modeling or cumbersome perception devices. This not only saves time and resources but also enhances the efficiency and effectiveness of resource exploration activities.
3. Applications in marine science and research: UUVs play a significant role in marine scientific research, enabling data collection and analysis in remote and inaccessible marine environments. The research findings contribute to the development of intelligent UUV systems that can adapt to unknown environments and autonomously avoid obstacles. This opens up new possibilities for conducting in-depth studies in marine biology, geology, oceanography, and other scientific fields.
4. Enhanced operational capabilities: The ability of UUVs to autonomously plan paths and avoid obstacles in challenging environments expands their operational capabilities. They can perform tasks such as underwater inspection, maintenance, and surveillance with greater efficiency and accuracy. This has practical implications for applications such as underwater infrastructure inspection, offshore oil and gas operations, and marine security.

The importance of this study lies in the application of deep learning techniques to dynamic path planning for underwater unmanned vehicles (UUVs). By integrating the LSTM-RNN network, the research enabled the UUV to autonomously navigate and avoid obstacles in complex and unknown underwater environments. This advancement improved the autonomy, safety, and efficiency of UUVs in areas such as marine scientific research, resource exploration, and marine security. The research results contribute to the development of intelligent UUV systems that can adapt to challenging underwater environments, thereby improving the effectiveness and practicality of underwater operations.

Author Contributions: Conceptualization, Y.K.; Methodology, Y.K.; Software, Y.M. and V.B.; Validation, Y.K.; Formal analysis, Y.K.; Investigation, Y.K.; Resources, Y.M.; Data curation, O.P. and Y.M.; Writing—original draft, Y.K. and V.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

MDPI	Multidisciplinary Digital Publishing Institute
UUV	Unmanned Underwater Vehicles
LSTM	Long Short-Term Memory
RNN	Recurrent Neural Network
ACO	Ant Colony Optimization
DL	Deep Learning
NN	Neural Network

References

- Wang, J.; Yang, Y.; Mao, J.; Huang, Z.; Huang, C.; Xu, W. CNN-RNN: A Unified Framework for Multi-label Image Classification. *arXiv* **2016**, arXiv:1604.04573.
- Yu, H.; Wang, J.; Huang, Z.; Yang, Y.; Xu, W. Video paragraph captioning using hierarchical recurrent neural networks. *arXiv* **2015**, arXiv:1510.07712.
- Wang, J.X.; Kurth-Nelson, Z.; Tirumala, D.; Soyer, H.; Leibo, J.Z.; Munos, R.; Blundell, C.; Kumaran, D.; Botvinick, M. Learning to reinforcement learn. *arXiv* **2016**, arXiv:1611.05763.
- Gupta, O.; Raskar, R. Distributed learning of deep neural network over multiple agents. *J. Netw. Comput. Appl.* **2018**, *116*, 1–8. [[CrossRef](#)]
- Osband, I.; Blundell, C.; Pritzel, A.; Van Roy, B. Deep Exploration via Bootstrapped DQN. *arXiv* **2016**, arXiv:1602.04621.
- Foerster, J.; Assael, I.A.; De Freitas, N.; Whiteson, S. Learning to Communicate with Deep Multi-Agent Reinforcement Learning. *arXiv* **2016**, arXiv:1605.06676.
- Dai, Z.; Li, L.; Xu, W. CFO: Conditional Focused Neural Question Answering with Large-scale Knowledge Bases. *arXiv* **2016**, arXiv:1606.01994.
- Carta, S.; Ferreira, A.; Podda, A.S.; Recupero, D.R.; Sanna, A. Multi-DQN: An ensemble of deep Q-Learning agents for stock market forecasting. *Expert Syst. Appl.* **2021**, *164*, 113820. [[CrossRef](#)]
- Li, Y.; Fan, C.; Li, Y.; Wu, Q.; Ming, Y. Improving Deep Neural Network with Multiple Parametric Exponential Linear Units. *arXiv* **2016**, arXiv:1606.00305. [[CrossRef](#)]
- Lutonin, A.; Shklyarskiy, J. Topology and control algorithms for a permanent magnet synchronous motor as a part of a vehicle with in-wheel motors. In Proceedings of the E3S Web of Conferences, Khabarovsk, Russia, 7–9 September 2021; EDP Sciences: Ulysses, France, 2021; Volume 266, p. 04001.
- Brigadnov, I.; Lutonin, A.; Bogdanova, K. Error State Extended Kalman Filter Localization for Underground Mining Environments. *Symmetry* **2023**, *15*, 344. [[CrossRef](#)]
- Yapar, C.; Levie, R.; Kutyniok, G.; Caire, G. Real-time outdoor localization using radio maps: A deep learning approach. *arXiv* **2021**, arXiv:2106.12556. [[CrossRef](#)]
- Xu, W.; Cai, Y.; He, D.; Lin, J.; Zhang, F. Fast-lio2: Fast direct lidar-inertial odometry. *IEEE Trans. Robot.* **2022**, *38*, 2053–2073. [[CrossRef](#)]
- Kakani, V.; Nguyen, V.H.; Kumar, B.P.; Kim, H.; Pasupuleti, V.R. A critical review on computer vision and artificial intelligence in food industry. *J. Agric. Food Res.* **2020**, *2*, 100033. [[CrossRef](#)]
- Sankowski, D.; Nowakowski, J. *Computer Vision in Robotics and Industrial Applications*; World Scientific: Singapore, 2014; Volume 3.
- Tian, H.; Wang, T.; Liu, Y.; Qiao, X.; Li, Y. Computer vision technology in agricultural automation—A review. *Inf. Process. Agric.* **2020**, *7*, 1–19. [[CrossRef](#)]
- Bertolini, M.; Mezzogori, D.; Neroni, M.; Zammori, F. Machine Learning for industrial applications: A comprehensive literature review. *Expert Syst. Appl.* **2021**, *175*, 114820. [[CrossRef](#)]
- Kashyap, P.; Kashyap, P. Industrial applications of machine learning. In *Machine Learning for Decision Makers: Cognitive Computing Fundamentals for Better Decision Making*; Apress: Bangalore, India, 2017; pp. 189–233.
- Abiodun, O.I.; Jantan, A.; Omolara, A.E.; Dada, K.V.; Mohamed, N.A.; Arshad, H. State-of-the-art in artificial neural network applications: A survey. *Heliyon* **2018**, *4*, e00938. [[CrossRef](#)] [[PubMed](#)]
- Zhou, H.; Zhao, H.; Zhang, Y. Nonlinear system modeling using self-organizing fuzzy neural networks for industrial applications. *Appl. Intell.* **2020**, *50*, 1657–1672. [[CrossRef](#)]
- Chen, F.; Wang, M.Y. Design optimization of soft robots: A review of the state of the art. *IEEE Robot. Autom. Mag.* **2020**, *27*, 27–43. [[CrossRef](#)]
- Takishima, Y.; Yoshida, K.; Khosla, A.; Kawakami, M.; Furukawa, H. Fully 3D-printed hydrogel actuator for jellyfish soft robots. *ECS J. Solid State Sci. Technol.* **2021**, *10*, 037002. [[CrossRef](#)]
- Runciman, M.; Darzi, A.; Mylonas, G.P. Soft robotics in minimally invasive surgery. *Soft Robot.* **2019**, *6*, 423–443. [[CrossRef](#)]
- Zakharov, L.A.; Martyushev, D.A.; Ponomareva, I.N. Predicting dynamic formation pressure using artificial intelligence methods. *J. Min. Inst.* **2022**, *253*, 23–32. [[CrossRef](#)]

25. Zhukovskiy, Y.L.; Kovalchuk, M.S.; Batueva, D.E.; Senchilo, N.D. Development of an Algorithm for Regulating the Load Schedule of Educational Institutions Based on the Forecast of Electric Consumption within the Framework of Application of the Demand Response. *Sustainability* **2021**, *13*, 13801. [[CrossRef](#)]
26. Yakovleva, T.A.; Romashev, A.O.; Mashevsky, G.N. Digital technologies for optimizing the dosing of flotation reagents during flotation of non-ferrous metal ores. *Min. Inf. Anal. Bull.* **2022**, *6*, 175–188. [[CrossRef](#)]
27. Zemenkova, M.Y.; Chizhevskaya, E.L.; Zemenkov, Y.D. Intelligent monitoring of the condition of hydrocarbon pipeline transport facilities using neural network technologies. *J. Min. Inst.* **2022**, *258*, 933–944. [[CrossRef](#)]
28. Alanazi, A.K.; Alizadeh, S.M.; Nurgalieva, K.S.; Nestic, S.; Grimaldo Guerrero, J.W.; Abo-Dief, H.M.; Eftekhari-Zadeh, E.; Nazemi, E.; Narozhnyy, I.M. Application of Neural Network and Time-Domain Feature Extraction Techniques for Determining Volumetric Percentages and the Type of Two Phase Flow Regimes Independent of Scale Layer Thickness. *Appl. Sci.* **2022**, *12*, 1336. [[CrossRef](#)]
29. Ushakov, E.; Aleksandrova, T.; Romashev, A. Neural network modeling methods in the analysis of the processing plant's indicators. In *International Scientific Conference Energy Management of Municipal Facilities and Sustainable Energy Technologies EMMFT 2019*; Springer: Berlin/Heidelberg, Germany, 2021; Volume 2, pp. 36–45.
30. Filippov, E.V.; Zaharov, L.A.; Martyushev, D.A.; Ponomareva, I.N. Reproduction of reservoir pressure by machine learning methods and study of its influence on the cracks formation process in hydraulic fracturing. *J. Min. Inst.* **2022**, *258*, 924–932. [[CrossRef](#)]
31. Islamov, S.; Grigoriev, A.; Beloglazov, I.; Savchenkov, S.; Gudmestad, O.T. Research Risk Factors in Monitoring Well Drilling—A Case Study Using Machine Learning Methods. *Symmetry* **2021**, *13*, 1293. [[CrossRef](#)]
32. Montiel, J.; Halford, M.; Mastelini, S.M.; Bolmier, G.; Sourty, R.; Vaysse, R.; Zouitine, A.; Gomes, H.M.; Read, J.; Abdessalem, T.; et al. River: Machine learning for streaming data in python. *J. Mach. Learn. Res.* **2021**, *22*, 4945–4952.
33. Vasilev, I.; Slater, D.; Spacagna, G.; Roelants, P.; Zocca, V. *Python Deep Learning: Exploring Deep Learning Techniques and Neural Network Architectures with Pytorch, Keras, and TensorFlow*; Packt Publishing Ltd.: Birmingham, UK, 2019.
34. Sun, W.; Akashi, N.; Kuniyoshi, Y.; Nakajima, K. Physics-informed recurrent neural networks for soft pneumatic actuators. *IEEE Robot. Autom. Lett.* **2022**, *7*, 6862–6869. [[CrossRef](#)]
35. Sultanbekov, R.; Beloglazov, I.; Islamov, S.; Ong, M.C. Exploring of the Incompatibility of Marine Residual Fuel: A Case Study Using Machine Learning Methods. *Energies* **2021**, *14*, 8422. [[CrossRef](#)]
36. Romashev, A.O.; Nikolaeva, N.V.; Gatiatullin, B.L. Adaptive approach formation using machine vision technology to determine the parameters of enrichment products deposition. *J. Min. Inst.* **2022**, *256*, 677–685. [[CrossRef](#)]
37. Chen, A.; Yin, R.; Cao, L.; Yuan, C.; Ding, H.; Zhang, W. Soft robotics: Definition and research issues. In Proceedings of the 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), Auckland, New Zealand, 21–23 November 2017; pp. 366–370.
38. Makhovikov, A.B.; Kryltsov, S.B.; Matrokhina, K.V.; Trofimets, V.Y. Secured communication system for a metallurgical company. *Tsvetnye Met.* **2023**, *4*, 5–13. [[CrossRef](#)]
39. Matrokhina, K.V.; Trofimets, V.Y.; Mazakov, E.B.; Makhovikov, A.B.; Khaykin, M.M. Development of methodology for scenario analysis of investment projects of enterprises of the mineral resource complex. *J. Min. Inst.* **2023**, *259*, 112–124. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.