|| Volume 7 || Issue 9|| November 2023 || ISSN (Online) 2456-0774
INTERNATIONAL JOURNAL OF ADVANCE SCIENTIFIC RESEARCH
AND ENGINEERING TRENDS

Machine Learning-Based Movement Scheduling and Management for Autonomous Mobile Robot Navigation

Sumoli Lahu Vaje, Prof. Anita Mahajan, Dr. Pankaj Agarkar,

PG Student: Department of Computer Engineering, Ajeenkya DY Patil School of Engineering Pune

Prof.: Department of Computer Engineering, Ajeenkya DY Patil School of Engineering Pune

HOD: Department of Computer Engineering, Ajeenkya DY Patil School of Engineering Pune

Abstract: Autonomous mobile robot navigation refers to the ability of robots to navigate and move in their environment without human intervention. It is of great significance in various industries as it enables robots to perform tasks efficiently and safely, leading to increased productivity and cost-effectiveness. Industries such as manufacturing, logistics, healthcare, and agriculture all benefit from the implementation of autonomous mobile robot navigation. An analysis of the challenges involved in achieving efficient and flexible movement planning and control for autonomous robots is necessary. Highlight the benefits of employing machine learning methodologies to address these issues. This work proposes the utilization of an Augmented Gradient Support Vector Machine (AG-SVM) to facilitate movement scheduling and management in the context of autonomous mobile robot navigation. Create a thorough dataset containing historical data on the locomotion of mobile autonomous robots in various scenarios. Collect data regarding the robots' positions and velocities, the surrounding environment, the sequence of jobs, and any relevant sensor information. To ensure data cleanliness and preprocessing, it is necessary to eliminate outliers, handle missing values, and normalize the acquired dataset. If needed, conduct feature engineering to extract relevant characteristics for the task of movement scheduling and management. The most advantageous elements of the dataset that aid in movement planning and management are extracted using the Histogram of Oriented Gradients (HOG). This method aids in decreasing dimensionality and enhancing the efficacy of learning algorithms. AG-SVM is utilized for the management and coordination of movements. In order to enhance the implementation of self-governing robots across different sectors, it is crucial to underscore the significance of adaptable and efficient movement scheduling and administration.

Keywords: autonomous mobile robot navigation, Augmented Gradient Support Vector Machine (AG-SVM), movement scheduling, management, Histogram of Oriented Gradients (HOG)

1. Introduction

Machine learning-based movement refers to the utilization of machine learning algorithms and techniques to enable independent or partially independent movement in various systems, such as robots, autonomous vehicles, drones, and virtual characters. It involves instructing these systems to progressively enhance their mobility capabilities over time without explicit programming. In the field of machine learning-driven mobility, the system frequently utilizes sensor inputs, such as camera images, lidar data, or other environmental information, to identify and understand its surroundings [1]. After the data is processed and evaluated, machine learning algorithms are used to detect and extract relevant patterns and traits. The system has the ability to generate appropriate movement commands by analyzing acquired patterns and making informed decisions. During the training phase, a substantial assortment of samples, which may be labeled or unlabeled, is often collected to represent different movement scenarios. The goal is to enable the system to extrapolate from the training data and adapt its movement strategies for unfamiliar, unexpected situations. The term "machine learning-based movement" encompasses a range of tasks including as navigation, route planning, obstacle avoidance, object tracking, gesture recognition, and locomotion control [2]. Machine learning can enhance the effectiveness, adaptability, and responsiveness of these systems, hence increasing their reactivity to changing surroundings. Autonomous mobile robot navigation is the term used to describe a robot's ability to independently and without human intervention move about its environment. The robot utilizes a diverse range of algorithms, sensors, and control systems to facilitate its ability to perceive its environment, plan a trajectory, and execute the necessary actions to reach its destination. The robot perceives its environment through the utilization of sensors such as cameras, lidar, sonar, or infrared sensors. These sensors [3] give obstacles, landmarks, and other crucial elements. Utilizing the sensor data, the robot generates a map or model of its immediate environment. To understand the arrangement, positioning of barriers, and other navigational cues, make use of this map. The robot autonomously determines its position within the mapped environment. Simultaneous localization and mapping (SLAM) techniques, which combine sensor data and movement information to accurately anticipate the robot's position, could be beneficial in addressing this issue. Once the robot obtains a map of the surrounding region and determines its own location, it calculates a route that avoids all potential collisions to reach the desired destination. In order to establish an optimal or ideal path, considerations such as robot dynamics and other limitations must be taken into account [4]. A robot consistently identifies and circumvents impediments in its trajectory throughout its locomotion. In order to safely maneuver around obstacles, the system modifies its trajectory, speed, or direction by utilizing real-time sensor data. The control system translates the desired course into actual robot movements. In order to ensure the accurate implementation of the intended path, this includes motor control, motion planning, and feedback systems. Autonomous mobile robots can utilize machine learning techniques to improve their ability to navigate [5]. They can adapt their behavior in accordance with fluctuating environmental conditions, enhance their route planning algorithms, or acquire knowledge from past encounters. A self-governing mobile robot has the ability to independently navigate various environments, including restricted spaces, uneven surfaces, and complex industrial situations, by integrating these components. The goal is to enable the robot to navigate swiftly and safely to its intended location, while also avoiding potential dangers and making astute decisions based on its perception of the environment [6]. The process of using machine learning techniques to improve the efficiency of scheduling and controlling the movements of robots in a changing environment is referred to as machine learning-based movement scheduling and management for autonomous mobile robot navigation. Machine learning can be employed to devise the movements of multiple robots operating inside a shared environment. Machine learning models have the ability to determine the optimal method for allocating movement tasks, considering factors such as robot capabilities, energy usage, workload distribution, and job prioritization [7]. Machine learning approaches can enhance the trajectory planning process for autonomous mobile robots. These algorithms acquire information from historical or simulated data to predict optimal paths and generate seamless trajectories that conserve energy, mitigate the risk of collisions, and account for robot dynamics and environmental constraints. Traffic management Machine learning techniques can be used to aid in the management of traffic and the prevention of collisions in environments where multiple autonomous mobility robots operate simultaneously. Robots have the ability to make intelligent decisions in order to avoid traffic, navigate lanes, and synchronize their motions to optimize overall efficiency. They achieve this by analyzing sensor data and incorporating knowledge gained from previous experiences [8]. Machine learning technologies enable autonomous mobile robots to make adaptive judgments during navigation. Robots have the ability to modify their movement techniques, select alternative paths, and adjust their actions in order to enhance efficiency and adapt to evolving circumstances. They achieve this by continuously acquiring knowledge from sensor inputs, environmental alterations, and human preferences. Machine learning algorithms can be employed to detect anomalies in robot movements or system malfunctions. Machine learning models have the ability to identify abnormal behavior, trigger alerts, and initiate corrective actions to ensure secure and reliable navigation by constantly monitoring sensor data, control signals, and robot responses [9]. Mobile autonomous robots can acquire information through user interactions and feedback. Machine learning models can adapt their movement strategies to align with user expectations by considering user preferences, feedback, or demonstrations. This enhances customer satisfaction and the quality of the navigation experience. Machine learning-based movement scheduling and management algorithms enable autonomous mobile robots to traverse efficiently, adjust to dynamic circumstances, and optimize their motions to enhance performance. These techniques utilize historical and present data, along with human interactions, to detect patterns, predict results, and enhance the navigational capabilities of autonomous robots [10].

2. Related Works

This paper presents a standardized framework for integrating task scheduling and routing control on a shop floor operated by mobile autonomous robots, which is an increasingly popular industrial manufacturing pattern. We propose a multi-agent architecture that explicitly incorporates human beings, machines, and mobile robots. The effectiveness of IM systems is influenced by the architecture of the core software platform and the choice of the underlying algorithm, as is the case with any other cyber-physical system [11]. The article proposes a reinforcement learning approach where an agent creates paths on a pre-determined layout and receives rewards depending on several criteria that represent the desired characteristics of the system. The results indicate that when there is a high number of AMRs operating in the system, the proposed method performs better than the traditional shortest-path-based approach in terms of both throughput and reliability. Although the area in which the robots work may be relatively small, the need for high throughput requires a significant number of AMRs to be in operation. Therefore, it is recommended to implement the suggested technique [12]. The objective of the project was to determine the potential of geographic data mining, simulation-based digital twins, and real-time monitoring technologies in improving the capabilities of remote sensing robots. The flow diagram of evidence-based data was generated using the Shiny software, following the Preferred Reporting Items for Systematic Reviews and Meta-analysis (PRISMA) standards. The initial bibliometric mapping employed Dimensions for data visualization, and VOSviewer for the layout algorithms [13]. The intricacy of robotic tasks and environments tends to escalate in tandem with the intricacy of control interfaces. Conventional input techniques such as touch, voice, and gesture may not be suitable for every user. Individuals with restricted mobility may be unable to operate such devices, despite being the ones who are most in need of robotic assistance. While certain users may exert effort to familiarize themselves with a robotic system [14]. This study aims to develop a unique architecture that enables customers to communicate with a robotic service assistant just through their thoughts in a closed-loop environment. The system includes a brain-computer interface (BCI) as one of its interconnected components. The BCI utilizes non-invasive methods to record neural signals and employs co-adaptive deep learning techniques. Additional elements encompass advanced task planning that relies on reference expressions, planning for navigation and manipulation, and environmental perception [15]. This study addresses the challenge of autonomously mapping unfamiliar small celestial objects during near encounters. This study proposes a Deep Reinforcement Learning (DRL)-based forecast strategy to enhance the effectiveness of surface mapping. This is achieved through the intelligent autonomous selection of the image capture epochs. The comparison between learned policies and standard policies is conducted in a range of possible circumstances, and the methodologies of Neural Fitted Q (NFQ) and Deep Q Network (DQN) are analyzed [16].

3. Methodology

Machine learning-based movement scheduling and management is a crucial element of autonomous mobile robot navigation, as it allows for efficient and intelligent navigation in many circumstances. Figure 1 illustrates the progression of this study.

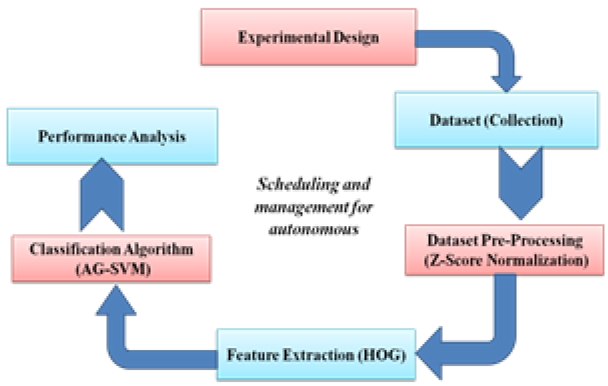

Fig.1. Flow of this study

3.1 Data set

The inputs for the proposed adaptive method are the distances between the left and right wheels, as well as their estimated probabilities obtained from the AMR at each time. The output velocity in this context is denoted as Vout, and it represents the disparity between the computed probabilities of velocities for both wheels. The computed value is compared to the desired velocity or VD. The discrepancy between VD and VOUT is utilized to compute the error signal, E(x). The block coordinate descent method utilizes the signal to iteratively update the weights W1, W2, and W3. These weights are represented by a combined weight vector, denoted as Pk = [D P(VRW) P(VLW)]. The dataset consists of a total of 2920 data points, with each of the six sets containing 17520 data points in total (2920 x 6). Validation is performed using the remaining 5256 data points, while 12264 data points are allocated for training.

3.2 Preprocessing using Z-Score Normalization

Z-score normalization, sometimes referred to as zero-mean normalization, is the procedure of standardizing output descriptors by computing the normalized mean and standard deviation for each parameter over many test datasets. The mean and standard deviation are assigned to each attribute. The equation 1 provides a general description of the replacement process:

$$v' = \frac{v - \mu A}{\sigma A} \qquad (1)$$

The term "A differences in values" refers to the standard deviation of the attribute, denoted as σ_(A). Consequently, every characteristic in the dataset exhibits no variation and no importance. Prior to creating a trainee collection and commencing the training procedure, each training sample in the data set undergoes the Z-Score normalization process. The average, variance, and statistical significance of a training data collection must be computed, recorded, and employed as weights in the final system design. The preprocessing phase is an integral part of the neural network architecture. Due to the neural network being trained on a dissimilar dataset, its outputs may exhibit substantial variations compared to the normalized data. Statistical normalization reduces the volume of data, hence mitigating the impact of data anomalies.

3.3 Feature extraction using Histogram of Oriented Gradients (HOD)

The Histogram of Oriented Gradients (HOG) is a popular technique in computer vision for extracting features. It is commonly employed for tasks such as object detection and identification. The algorithm captures the gradient information present in an image and converts it into a feature vector. The HOG method is frequently employed to assess the quality of a certain distribution of local gradients. When employed for target identification, it yields exceptional outcomes. The HOG feature can be efficiently utilized to represent the distribution of local gradients.

Determine the gradient's magnitude and orientation:

Using these formulas, we can calculate the A and B orientation intensity gradients:

$Hw = (w + 1, z) - (w - 1, z)$ (2)

$hz = (w, z + 1) - (w, z - 1)$ (3)

where Hw and hz are the horizontal and vertical gradients, respectively; the gradient amplitude m(x, z) indicates the variance in the size of the grey level.

Equations (4) and (5) may also be used to compute gradient amplitude and direction:

$$\mathrm{n}(\mathrm{w}, \mathrm{z}) = \sqrt{H_w^2 + H_w^2} \qquad (4)$$

$$\theta(\mathrm{w}, \mathrm{z}) = \frac{\arctan(H_2\,)}{H_w} \qquad (5)$$

3.3.1 The Process of a Cell Histogram

Histograms are constructed by counting the frequency at which each orientation of a gradient occurs in each of the bins assigned to that orientation. The orientation of each gradient is utilized to establish the specific bin to which it is allocated. The histogram depicts the distribution of internal gradient orientations of cells, which in turn shows regional edge directions. Normalization is an accessible option to enhance the histogram's tolerance towards slight variations in brightness and contrast. In L1 normalization, each bin value is divided by the total bin value, while in L2 normalization, each bin value is divided by the Euclidean norm of the histogram vector. The HOG approach captures the local edge information in an image by constructing histograms of cells that are dependent on gradient orientations.

These histograms are utilized as features to provide a more detailed description of the picture in tasks such as item identification and recognition.

3.3.2 Block specification

A feature vector is formed by combining the normalized histograms of each block in the image. The feature vector not only contains information about local gradients in individual cells but also include spatial correlations, which are quantified using block normalization. Normalizing cell contributions within blocks is an essential step in HOG for addressing variations in light and contrast. Applying the HOG feature descriptor to a cluster of cells within a block and subsequently normalizing it enhances its ability to withstand changes in illumination and contrast.

3.3.3 The block's gradient is normalized

The HOG technique is frequently employed for picture feature extraction. By dividing an image into small cells, calculating histograms of gradient orientations within each cell, and concatenating the resulting feature vectors, it is possible to capture detailed local gradient information. The density of each histogram is determined.

$$U^* = \sqrt{u||U||L + 1.1s} \qquad (6)$$

Density can be defined as the proportion of all occurrences or values that are contained within a specific interval. Density measures the degree to which gradients are focused within histograms.

3.4 Augmented Gradient- Support Vector Machine (AG-SVM)

AG-SVM is a supervised learning algorithm that utilizes a predetermined function to forecast the label of an output by considering the input values. Minimizing the errors of the sample points and reducing structural hazards can enhance the model's ability to generalize. Let's suppose that there are l data points and n indices in the datasets that require categorization.

The AG-SVM algorithm, a widely recognized supervised machine learning technique, is employed for both classification and regression tasks. It has the ability to handle complex data and operate in feature spaces with a large number of dimensions. The main goal of AG-SVM is to identify a hyperplane that effectively divides data points into their respective classes while minimizing the distance between the hyperplane and the closest data points. The objective of AG-SVM is to optimize the margin by maximizing its size and minimizing the classification error. To commence, gather a substantial amount of training data that has been previously annotated, ensuring that each data point is already assigned to its respective category. AG-SVM is a binary classifier, hence it is necessary to divide the data into two distinct groups. Complex problems including diverse categories often necessitate the utilization of various approaches for resolution. Refine the data to extract pertinent attributes for the categorization process. While AG-SVM excels with numerical characteristics, it is still feasible to encode categorical features. To ensure that no single feature dominates the learning process, it is advisable to normalize or standardize the values of the features. Min-max scaling and z-score normalization are two frequently used methods of scaling. The AG-SVM model is trained during the SVM Model Training step by providing it with the labeled data. The ideal hyperplane is the one that maximizes the distance between the two sets of data. By employing a mathematical optimization technique, AG-SVM is capable of solving this problem. AG-SVM utilizes kernel functions to transfer data into a higher dimensional space, enabling linear separation. The linear kernel, polynomial kernel, Gaussian (RBF) kernel, and sigmoid kernel are examples of common kernel functions. The choice of kernel is contingent upon the intricacy of the problem and the nature of the data being processed. The resolution of a quadratic programming issue is necessary in order to determine the ideal hyperplane by AG-SVM optimization. The optimization procedure aims to maximize profit by minimizing a cost function that penalizes misclassified data points. In order to identify the support vectors, the Lagrange multipliers are calculated using the data points that lie on the boundary or in close proximity to it. One way to achieve this is by use techniques such as grid search or randomized search. The AG-SVM algorithm has been applied in various domains, such as text categorization, visual perception, biology, and even the financial industry. They are extensively utilized due to their strong theoretical basis, capacity to handle diverse data distributions, and resilience against overfitting.

4. Results and Discussion

This section provides a comprehensive analysis of the results obtained from the suggested methodology, AG-SVM, in comparison to the existing methods employed in this research, namely Conventional Neural Networks (CNN), Deep Neural Networks (DNN), and long short-term memory (LSTM). This research utilizes criteria like as accuracy, precision, recall, and f1-score to analyze the efficacy of the suggested strategy. TP represents the number of true positive cases, TN represents the number of true negative cases, FP represents the number of false positive cases, and FN represents the number of false negative cases.

Table.1. Numerical outcomes of proposed and existing methods

Methods	Accuracy %	Precision %	F1- score %	Recall %
CNN [17]	75	70	80	73
DNN [18]	78	77	70	87
LSTM[19]	83	85	83	89
AG-SVM [Proposed]	96	89	92	95

A. Accuracy

Inadequate accuracy is the reason of the discrepancy between the result and the true value. The proportion of observed results indicates the general equilibrium of the data. Accuracy is evaluated through the utilization of a mathematical equation.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

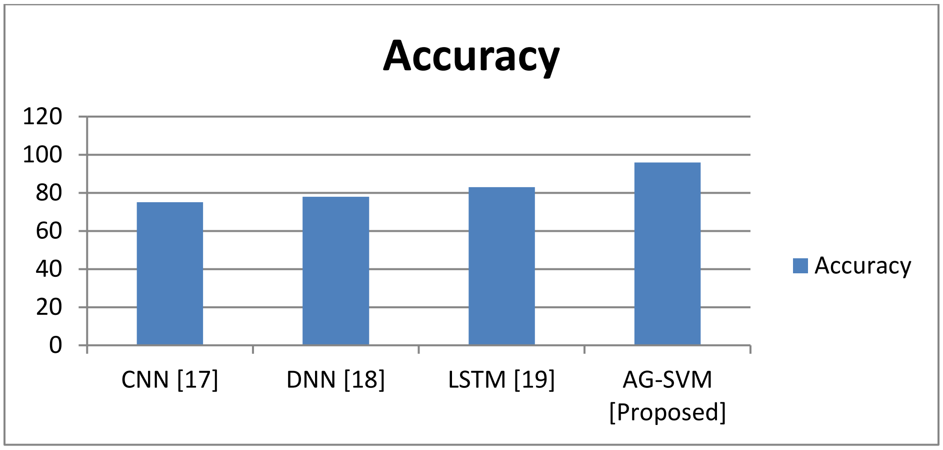

Fig.2. Comparison of Accuracy

Figure 2 displays the corresponding values for the accuracy metrics. When compared to established methods such as CNN, which achieves an accuracy rate of 75%, DNN, which achieves an accuracy rate of 78%, and LSTM, which achieves an accuracy rate of 86.64%, the suggested method's AG-SVM value is 96%. The AG-SVM suggested exhibits superior accuracy compared to current methodologies and effectively categorizes autonomous mobile robot navigation.

B. **Precision**

Precision is the key criterion for accuracy, and it is precisely defined as the proportion of correctly classified cases to all occurrences of positively predicted data. The equation is utilized to calculate the precision.

$$Precision = \frac{TP}{TP + FP}$$

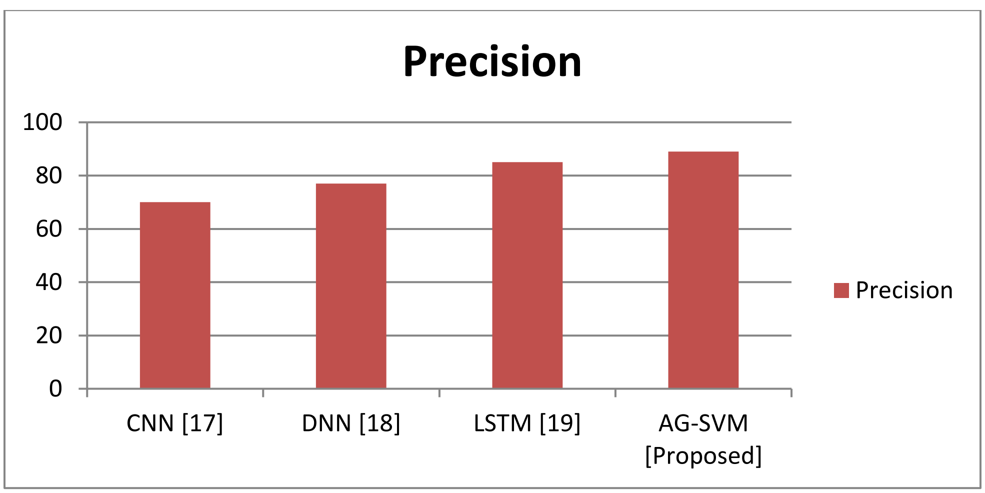

Fig.3. Comparison of Precision

The accuracy measurements' equivalent values are displayed in Figure 3. This demonstrates that the proposed strategy has the potential to yield performance outcomes that surpass those achieved with the present study methods. The suggested approach AG-SVM achieves a precision of 89%, surpassing the performance of existing methods. The precision rates for DNN, CNN, and LSTM are 77%, 70%, and 85% respectively.

C. Recall

Recall refers to the capacity of a model to accurately recognize and classify significant samples within a dataset. The recall is determined by utilizing a mathematical calculation.

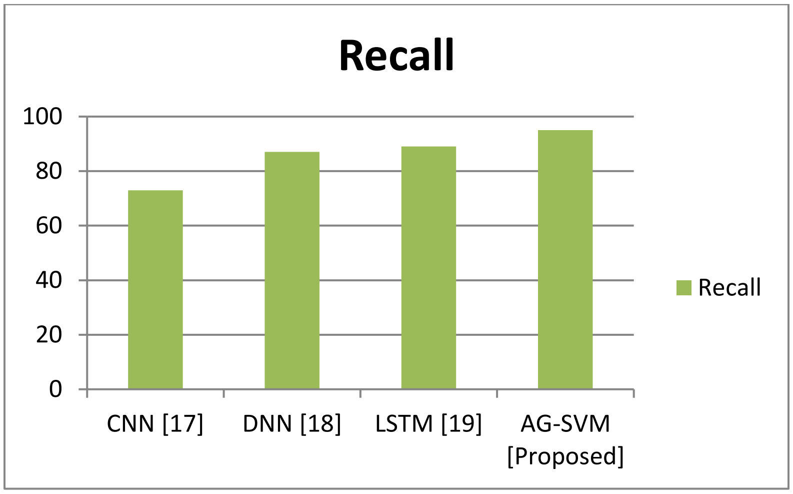

Fig.4.Comparison of Recall

Figure 4 displays the comparative data for the recall measures. The recall rates for CNN, DNN, LSTM, and AG-SVM were 73%, 87%, 89%, and 95% respectively. The proposed strategy outperformed the present results, with a recall rate of 95%.

D. F1-score

The f1-score is calculated as the harmonic mean of the recall and precision measures in the proposed model. Equation (18) is utilized for calculating the f1-score.

$$\mathrm{F1-score} = \frac{2 \times \mathrm{precision} \times \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}$$

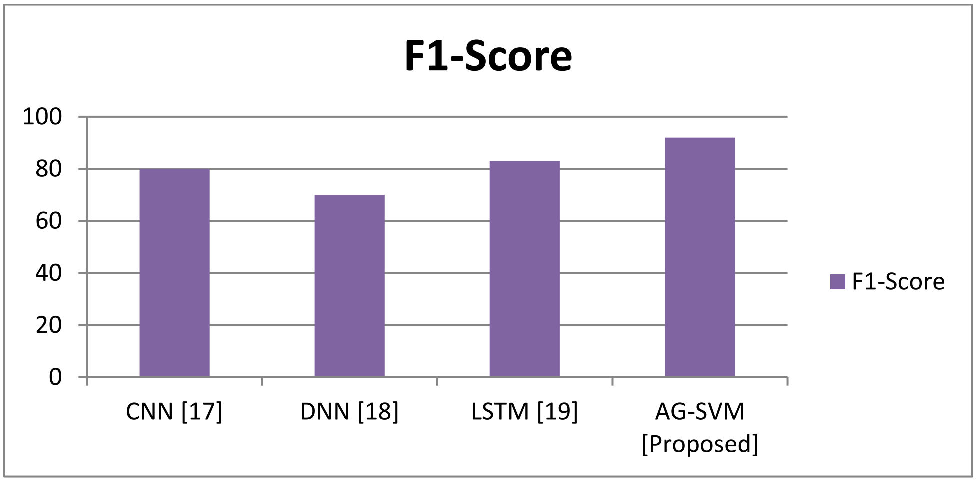

Fig.5. Comparison of F1-score

Figure 5 displays the comparative data for the recall measures. The recall rates for CNN, DNN, LSTM, and AG-SVM are 80%, 70%, 83%, and 92% respectively. The proposed approach surpasses the existing outcomes, achieving an F1-score of 95%.

5. Conclusion

The AG-SVM algorithm, specifically developed for this study, utilizes augmented gradient optimization techniques to improve decision-making abilities, leading to more accurate and efficient movement scheduling and control. Overall, the AG-SVM algorithm and HOG feature extraction are suggested for their significant contribution to the system's exceptional performance. Specifically, the system achieves an accuracy of 96%, precision of 89%, recall of 95%, and F1-score of 92% in the context of movement scheduling and management for autonomous mobile robot navigation [20]. These measurements are related to the system's capacity to navigate independently within its surroundings. The machine learning-based movement scheduling and management system, which incorporates HOG feature extraction and the recommended AG-SVM method, greatly enhances the navigation of autonomous mobile robots. The impressive performance indicators demonstrate its potential in various domains, including manufacturing, supply chain management, and security. Through additional study, it may be feasible to create intuitive and seamless interfaces that allow individuals to provide advanced commands or preferences to the robot. This will enable seamless collaboration between people and robots in shared workspaces or collaborative projects.

References

[1] Jokić A, Petrović M, Kulesza Z, Miljković Z. Visual Deep Learning-Based Mobile Robot Control: A Novel Weighted Fitness Function-Based Image Registration Model. InNew Technologies, Development and Application IV 2021 May 12 (pp. 744-752). Cham: Springer International Publishing.

[2] Hu H, Jia X, He Q, Fu S, Liu K. Deep reinforcement learning based AGVs real-time scheduling with the mixed rule for the flexible shop floor in industry 4.0. Computers & Industrial Engineering. 2020 Nov 1;149:106749.

[3] Zeng J, Ju R, Qin L, Hu Y, Yin Q, Hu C. Navigation in unknown dynamic environments based on deep reinforcement learning. Sensors. 2019 Sep 5;19(18):3837.

[4] Eskandari M, Savkin AV. Deep Reinforcement Learning Based Joint 3D Navigation and Phase Shift Control for Mobile Internet of Vehicles Assisted by RIS-equipped UAVs. IEEE Internet of Things Journal. 2023 May 18.

[5] Fan T, Long P, Liu W, Pan J. Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios. The International Journal of Robotics Research. 2020 Jun;39(7):856-92.

[6] Wesselhöft M, Hinckeldeyn J, Kreutzfeldt J. Controlling fleets of autonomous mobile robots with reinforcement learning: a brief survey. Robotics. 2022 Aug 30;11(5):85.

[7] Le AV, Kyaw PT, Veerajagadheswar P, Muthugala MV, Elara MR, Kumar M, Nhan NH. Reinforcement learning-based optimal complete water-blasting for autonomous ship hull corrosion cleaning system. Ocean Engineering. 2021 Jan 15;220:108477.

[8] Pei M, An H, Liu B, Wang C. An improved dyna-q algorithm for mobile robot path planning in an unknown dynamic environment. IEEE Transactions on Systems, Man, and Cybernetics: Systems. 2021 Jul 26;52(7):4415-25.

[9] Ho TM, Nguyen KK, Cheriet M. Federated deep reinforcement learning for task scheduling in a heterogeneous autonomous robotic system. IEEE Transactions on Automation Science and Engineering. 2022 Nov 14.

[10] Chehelgami S, Ashtari E, Basiri MA, Masouleh MT, Kalhor A. Safe deep learning-based global path planning using a fast collision-free path generator. Robotics and Autonomous Systems. 2023 May 1;163:104384.

[11] Agrawal A, Won SJ, Sharma T, Deshpande M, McComb C. A multi-agent reinforcement learning framework for intelligent manufacturing with autonomous mobile robots. Proceedings of the Design Society. 2021 Aug;1:161-70.

[12] Kozjek D, Malus A, Vrabič R. Reinforcement- learning-based route generation for heavy-traffic autonomous mobile robot systems. Sensors. 2021 Jul 14;21(14):4809.

[13] Andronie M, Lăzăroiu G, Iatagan M, Hurloiu I, Ştefănescu R, Dijmărescu A, Dijmărescu I. Big Data Management Algorithms, Deep Learning-Based Object Detection Technologies, and Geospatial Simulation and Sensor Fusion Tools in the Internet of Robotic Things. ISPRS International Journal of Geo- Information. 2023 Jan 21;12(2):35.

[14] Thakur A, Das S, Kumari R, Mishra SK. Machine Learning based Intelligent Model for Path Planning Obstacle Avoidance in Dense Environments for Autonomous Mobile Robot.

[15] Piccinin M, Lunghi P, Lavagna M. Deep Reinforcement Learning-based policy for autonomous imaging planning of small celestial bodies mapping. Aerospace Science and Technology. 2022 Jan 1;120:107224.

[16] Ho DK, Ben Chehida K, Miramond B, Auguin M. Learning-Based Adaptive Management of QoS and Energy for Mobile Robotic Missions. International Journal of Semantic Computing. 2019 Dec;13(04):513-39.

[17] Jokić A, Petrović M, Kulesza Z, Miljković Z. Visual Deep Learning-Based Mobile Robot Control: A Novel Weighted Fitness Function-Based Image Registration Model. InNew Technologies, Development and Application IV 2021 May 12 (pp. 744-752). Cham: Springer International Publishing.

[18] Lee S, Kim Y, Kahng H, Lee SK, Chung S, Cheong T, Shin K, Park J, Kim SB. Intelligent traffic control for autonomous vehicle systems based on machine learning. Expert Systems with Applications. 2020 Apr 15;144:113074.

[19] Niranjan DR, VinayKarthik BC. Deep learning-based object detection model for autonomous driving research using Carla simulator. In2021 2nd international conference on smart electronics and communication (ICOSEC) 2021 Oct 7 (pp. 1251- 1258). IEEE.

[20] Sundar Ganesh, C. S. & Daisy Mae, R. B. (2022). A Study on the Usage of Robot Navigation to Determine Robot's Own Position in its Frame of Reference and then to Plan a Path towards Some Goal Location. Technoarete Transactions on Industrial Robotics and Automation Systems (TTIRAS). 2(4), 1–6.

[21] Kapoor, E. ., Kumar, A. ., & Singh , D. . (2023). Energy-Efficient Flexible Flow Shop Scheduling With Due Date and Total Flow Time. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 259–267. https://doi.org/10.17762/ijritcc.v11i2s.6145

[22] Paul Garcia, Ian Martin, Laura López, Sigurðsson Ólafur, Matti Virtanen. Enhancing Student Engagement through Machine Learning: A Review. Kuwait Journal of Machine Learning, 2(1). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/163

[23] Kumar, S.A.S., Naveen, R., Dhabliya, D., Shankar, B. M., Rajesh, B. N. Electronic currency note sterilizer machine (2020) Materials Today: Proceedings, 37 (Part 2), pp. 1442-1444.