Vision-Based Mobile Robot Controllers: A Scientific Review

Adnan Mohsin Abdulazeez, Fayez Saeed Faizi

a Presidency of Duhok Polytechnic University, Duhok, Kurdistan Region, IRAQ
b Energy Department, Technical College of Engineering, Duhok Polytechnic University, Duhok, Kurdistan Region, IRAQ

Corresponding author: fayez.faizi@dpu.edu.krd

Article History: Received: 10 November 2020; Revised 12 January 2021; Accepted: 27 January 2021; Published online: 5 April 2021

Abstract: Today, there are different types of self-controlled robots. Some of them had critical effects on our lives like industrial and medical robots. Others are for military usages such as drones and the pets robots just for entertainment. The crucial differences between this kind of robot and the controlled ones are their ability to move on their own and make decisions based on their observations of the world around them. Mobile robots must have a data source used as an input dataset and processed to change their behavior; for instance, moving, stopping, rotating, or doing any required action based on the information gathered from the surrounding environment. Different types of sensors were used to feed robots controllers with data. Such data source could be ultrasonic sensor, laser sensor, torque sensor, or vision sensor. Robots integrated with cameras were becoming an essential field of study. They recently attracted significant attention from researchers it has been commonly used in many sectors of healthcare, manufacturing, and many other services. The robot needs a controller with a powerful mechanism of realization to deal with such incoming data. The world of mobile robot controllers is discussed in this paper, and the latest trends were reviewed. This review aims to provide a general understanding of robot controllers and navigation methods developed over the last few years.

Key words: Mobile Robots, Robots Control, Navigation Systems, Computer Vision, and Machine Learning.

1. Introduction:

Currently, mobile robotics is one of the fastest fields growing in the scientific research topic. Thanks to their skills, robots replace humans in many sectors. Including different applications, especially in extreme environments such as petrochemical applications, planetary exploration, mines detection, and many medical implementations [1]. Autonomous robots can move, determine an action, and do a task without any intervention from a human. Instead, it depends on its build-in controller, or we can call it its brain [2], [3]. It’s worth mentioning that the mobile robot consists of several portions having different technologies. These sections allow the robot to perform the required task. The main sub-systems are sensors, motion system, navigation, and localization system. These sub-systems need a control system or cognition unit that will enlighten the robot’s life. Besides, coordinate other components operation to accomplish the required mission coherently [3]. The general process in the mobile robot is explained in (Figure 1).

The robot control unit is considered the most critical concept among all other sub-systems; nevertheless, the navigation task is the main impact of mobile robots [4], [5]. At most, there are two fundamental types of navigation local and global. Local navigation deals with surrounding space with a short distance range to implement the desired task, including collision prevention, obstacle avoidance, or path tracking. Global navigation points to the robot’s
locomotion in a broad range environment by employing a pre-specified map, or a whole location map to help it move in that area [5].

Here the local navigation type will be considered; these kinds of mobile robots are linked with sensors that give information about the extrinsic environment, which assist the automaton in creating a map of that location and localizing itself. Prevalent mobile robots employ different sensors, including (laser, ultrasonic, or infrared sensors). To discover the surrounding and collect data [2], [6]. The controller via some algorithms will do the required computational process for the incoming data which takes a form of a reflected light beam or reflected sound signal initially transmitted from a robot’s sensor. The signal will feed the data in an imperceptible form, and the robot will sense or understand its environment. Such sensors may receive inaccurate data reverberate from the objects in some situations due to obstruction of the incoming signal [7].

Consequently, a camera (or vision sensor) is a better substitution for the sensors mentioned above in a mobile automaton. The incoming data is a visualized information as an image format, which will be processed and analyzed by controller algorithms to convert it into useful data used in performing requested tasks [8]. A vision-sensing-based movable robot is typical for an indoor environment. Robots with attached cameras can perform their jobs more accurately than robots with other sensor-based; whoever, they can be found outdoor as well [9], [10]. (Figure 2) represent a basic vision system for a mobile robot.
The rest of this paper will be as follows: in section 2, the challenges related to mobile robot design will be explained, while theoretical bases are presented in section three. A literature review about the mobile robot controller’s methods is in the follower section. Eventually, a brief conclusion is given in section 5.

2. Mobile Robot Challenges

Several challenges arise while dealing with mobile robots. These challenges include kinematic, sensing, controlling, and navigating issues. Covering a wide range of engineering fields includes mechanic, electric, and computer engineering [11].

Generally, challenges can be divided into four classes as the issues related to:

2.1 Robot Locomotion

The first challenge related to the robotic field is the robot’s movement. Usually, the camera-based robot moves in known and controlled areas like stores or factories. Nevertheless, in some cases, it needs to move in extreme places with inhospitable environments. The robot’s motion system is an essential portion of designing a robot [11], [12]. It depends mainly on the area where the robot will operate; if it works in the air (drone), on the ground surface, or underwater [13]? Other critical technical and mechanical issues are related to the robot’s motion. For instance: stability, efficiency, controllability, maneuverability, and smooth turning. These troubles are mostly related to mechanical engineering, which can be solved by understanding the kinetic theories’ and dynamic mechanisms [14].

2.2 Robot Perception

Self-controlled mobile robots must collect information about its operating area. This knowledge can be formed from the sensors’ data. The sensors’ presence helps the robots carry out some tasks, including localization, mapping, and object recognition [15]. Sensors are classified broadly according to two main features. The first classification is done according to Proprioceptive and Exteroceptive characteristics. Other variety determined according to Passive and Active features [16].

Proprioceptive sensors measure some parameters inside the robot such as battery energy level, steering angle, and load on the motor, and its speed. In comparison, Exteroceptive sensors refer to the type that collects data about the surrounding area like sound level, distance to an obstacle, or light intensity [13], [14].

On the other hand, Passive sensors represent the ones that measure the energy entering it from its environments; Such as light-sensing Complementary Oxide Semiconductor (CMOS) and Charge Coupled Device (CCD), sound sensing (microphone), and temperature measuring (thermometer). On the other hand, the Active sensor will transmit a signal into its environment and calculate the reflected echo. The last type may be affected by noise or interference; however, it outperforms other types in working areas [17].

During the robot design stage, it’s critical to use the appropriate sensor that suits robot tasks and the operating environment; because the measurements collected by those sensors will be presented to the robot controller, which will determine the next step movement [12].

2.3 Robot Control System

It is the brain of the robot and its planner to the tasks that the robot should accomplish. The cognition system coordinates between other mechanical and electrical parts. Also, controls how it interacts with its environment. The incoming data from input sensors is fed to the cognition system, organizing, analyzing, and processing this information [17], [18]. The controller mainly contains models and algorithms that decide how the robot will interact to its
environment like selecting the proper path or track objects. Also, it may include other algorithms that build a map for the surrounding area. Lastly, artificial intelligence or motion planning methods were used to specify how the robot will develop its perspective and make the required decision [19], [20]. After deciding what to do, the controller will send a command to the actuators to move the mechanical parts to accomplish a specific job. It is critical to implement the correct control system, which contains the necessary algorithms and models to perform the required tasks [18].

2.4 Robot Navigation

A vital challenge in building a mobile robot is the ability to navigate. Navigation refers to the mobile robot’s capability to move safely from the initial point to the destination point without colliding with obstacles; also, without considering if the environment is known (training place) or not (test place) [21]. Generally, obstacle avoidance algorithms and motion planning models are required since the robot will not move in a straight line, while there are obstacles from the original point to the destination point [3], [22].

There are three main types of navigation forms. These categories depend on how the robot will calculate the path to the endpoint. The classes are:
- Creating a map of the whole environment, including the available trajectories.
- Determining a complete obstacles-free path.
- The robot moves over the track without colliding with any obstacles.

With knowing this, to build a navigation skill in a robot, it’s essential to feed the robot with sufficient data about its location to navigate [21]. That is to say, that the cornerstone in navigation procedure is the Localization process. Before the robot starts to navigate, it should calculate its location in the test environment; robot positioning or localization refers to its place in the workplace and its location according to the destination [23], [24]. Sensitive cooperation among locomotion, sensing, and localization all under the cognition system’s control should be performed to create a suitable robot navigation system [25]. Ultimately, to overcome the navigation challenge, a good knowledge of artificial intelligence, information theories, and path planning algorithms are required.

3. Theory of Robot Controlling

This paper will cover the ideas of controlling a robot with a vision sensor attached. Three related concepts should be defined to understand this kind of robot. These concepts are: visual processing, required feature extraction, and controlling via artificial intelligence. Due to the interference, or the noise that may affect other types of sensors, vision sensor (camera) rises as an excellent replacement to such sensors [17]. Whoever sensed signal will be visual-based as an image that needs to be analyzed then processed to produce the desired information [20], [22]. In the following sub-sections, these points will be covered. (Figure 3) show the main steps that vision-based robots follow.

---

**Figure 3 General processes of vision-based robot systems**
3.1 Computer Vision (CV)

We can define computer vision in two ways. It can be defined as a science field that works on extracting information from a digital image [26]. It can also be defined as the process of getting an image and building an algorithm that tries to interpret its contents and deploy it in any other applications [8], [27]. Computer vision sector is not a new field it was considered a sub-discipline of Artificial Intelligence (AI) topics since 1970. Its task was a simple recognition by specifying some objects. This sector developed over the years, and it is considered an essential subject in scientific and industrial fields. However, it still has notable limitations despite its growth over near to 50 years [2].

Nowadays, video clips and photos are omnipresent, taking a big part of our interest every day. Far from personal usage, cameras can be used in critical cases related to medical, scientific, and military issues that are difficult to or may not be resolved without computer vision [28]. That is to say, the Computer Vision (CV) term goes beyond just taking (importing) images and capturing videos, but it also covers the idea of understanding what this image is. Computer Vision covers a wide range field for the topics related to machine vision, path tracing, and image processing. Precisely, objects detection, classification, recognition, and features extraction [26], [29].

3.2 Visual Based Feature Extraction

Feature extraction is a crucial phase in vision-based mobile robot control [30]. As humans, we can explain the meaning of a picture, depending on what we see and understand via recognizing a particular part of that photo [26]. Is it possible for a computer algorithm or program to identify semantic features from a picture? Due to the development in this field, the answer is yes. However, extracting features that reflect an image’s primary content is still challenging in the image processing field [30], [31]. The main features that the human eye and computer vision can recognize are colors, shapes, and spatial characteristics. Most of the recognition and detection feature systems were build based on these three aspects [32]. Nonetheless, other methods were proposed that used a combination of these concepts as a hybrid system, or segmenting the image and use the dominant colors in each segment to detect features [31], [33].

3.3 Visual-Based Motion Controller

The motion controller is a critical mechanism for leading robots in an environment with movable obstacles. The main aim of robots’ controller is determining a path for the robot, which will be used to travel from the initial point to the finishing point successfully; meanwhile, avoiding colliding with any obstacle [34], [35]. If the environment contains dynamic obstacles such as humans or other robots, the robot must predict their trajectory to avoid them [19], [36]. Based on the viewing range and mapping size, path planning will be classified into local planning and global planning. The first term means that the robot is only aware of the obstacle situation around it, while the other term refers to the knowledge about general test area [34], [36], [37]. The planning procedure is illustrated in (Figure 4). Several controlling methods are available that will be discussed extensively in the next section (section 4).
4. Related Work for Controlling Methods

After walking through the essential concepts of mobile robots, explaining the challenges and theories behind designing vision-based mobile robots; now, it is the time to present the most common and efficient controlling methods. Here, several recent kinds of research dealing with the mobile robot controlling system will be given and discussed. After that, each model, algorithm’s power, and weak points will be summarized in a table (1).

Harandi et al. [38] proposed a method called Transition Certainty based Feature Selection (TCFS) a feature selection method based on state transition probability to control a mobile wheeled robot. The proposed model is originally a part of Supervised Deep Learning (SDL) method. As the input sensor is a Kinect camera, the incoming data in a depth image form with high dimensions; the proposed model tries to extract the required features via deep learning to reduce the input data dimensions. The model will employ clustering procedure with a genetic algorithm. As it is a certainty based model, TCFS will maximize the motion certainty from the present state to the next state. The experimental results show that the TCFS model overcomes the standard SDL method regarding some selected tasks.

Aparanji et al. [39] utilized a multi-layers Auto Resonance Network (ARN) to build a new network structure to control a robot’s movement. The configuration of this network was unlike the traditional Convolutional Neural Networks (CNN) and other architectures deployed in Deep Learning techniques. The presented network joints characteristics from Self Organizing Maps and ARN to improve the performance. The nodes in lower layers will try to map the incoming data to the output via ARN network architectures. On the other hand, the upper layers will resolve the locomotion issue by distinguishing, then optimizing the usable trajectories. This structure will allow the proposed network to scan the environment in order to determine several routes around obstacles, including the dynamic ones. After simulating the presented system in R simulation, the results demonstrate that the complexity of kinematic expressions can be entirely avoided and the overall robot’s performance was improved.

Al-Jarrah et al. [40] combine the fuzzy image processing and Genetic Algorithm (GA) for building a new model to control a mobile robot; their algorithm consists of two stages. In the first stage, the captured image was equalized to get more benefits from its details. After that, the system works on edge detection via a fuzzy system, that had been improved by the bacterial
algorithm for the goal of computational time reduction. Each pixel in the image will be categorized as edge or not. The output of stage one will be utilized to build a two-dimensional map for the test environment. The second phase is responsible for calculating the robot’s best path to move from the starting point to the end; this is done by passing the constructed map to GA and A* search algorithms to cooperate in achieving this task. Additionally, the proposed model presents a time-based path, which means that the robot can predict the velocity depending on the selected route. The introduced model has experimented with a real navigating robot, and the testing results show increasing in edge detection efficiency while reducing the time required for computations.

Jafar et al. [41] introduced a new model to control a vision-based robot. They exploited the idea of visual feedback to determine localization and navigation of the robot. The robot could specify its location by utilizing environment characteristics, where the features will be extracted from the captured image and then presented to Neural Network (NN). The implemented path planning algorithm allows the robot to determine its location and orientation using one camera, which will reduce the cost of designing such robots. For controlling and computation purposes, four layers of NN were implemented to perform these tasks. The input layer number stands for the numbers of the shapes and colors features extracted from the image. Finally, NN’s backpropagation rules were applied to modify the network’s biases and weights to minimize the squared mean error. The robot will move one step at a time, and it will take one image at each point to determine its position and orientation toward the destination. That is an advantage of this approach, where the robot doesn’t have to know the whole trajectory; instead, it will move from one node to another until reaching the destination.

Mnih et al. [42] presented a new model to improve the NN-based controller via utilizing asynchronous gradient descent for deep reinforcement learning. The proposed framework uses four reinforcement algorithms that work asynchronously to train the NN controller in different domains. The four algorithms were, one-step Q-learning, one-step Sarsa, n-step Q-learning, and advantage actor-critic. These algorithms work in parallel to train and update the NN that shared to all algorithms. The presented framework was applied to four different experiments, and the results of all tests indicate the stability effect of the framework. The four algorithms cooperate in training the NN controller. The system was stable in any situation; nevertheless, the findings show that the training process was faster.

Imen et al. [43] build a two-stages controller for the track-control task in a mobile robot. The initial controller is a fuzzy logic controller, and it takes four inputs:

• Vc: the current velocity.
• C: path curvature.
• dR: the distance from the current location to the destination location.
• d: the difference between the previous heading angle and the robots’ current orientation.

These data will be processed in the first controller to output one variable representing the trajectory curvature. This variable will be presented to the second controller, an Adaptive Neuro-Fuzzy Interface System (ANFIS) to resolve the trajectory tracking issue. The proposed system utilized the gradient descent algorithm to modify the parameters. Testing the presented (ANFIS) based system shows an improvement in tracking job, high precision, and better noise resistance than the fuzzy-only system.

Fathinezhad et al. [44] provided a new strategy to merge reinforcement learning and supervised learning. The proposed model named Supervised Fuzzy Sarsa Learning (SFSL) aims to exploit the power points of reinforcement learning and supervised learning. The zero-order Takagi-Sugeno fuzzy was applied as the central controller, which was utilized as obstacle avoidance. In the first step, the robot was trained by a human to collect training data from the training place. In the next step, each candidate’s value was initialized via training data. Lastly, the SFSL model was used to perform final fine-tuning toward the destination. Results indicate
that the computation complexity and cost were reduced. Also, an improvement in analyzing time was noticed.

Liu et al. [45] used a Convolutional Neural Network (CNN) to build an end-to-end paradigm as an obstacle avoidance controller in a mobile robot. The presented model contains 5 CNN layers followed by three fully connected layers. The single-camera captured images, and then features were extracted via deploying deep learning. The signal flows through the CNN and reaches the fully connected layer, that adjusted to three nodes representing steering control commands: turn right, turn left, and go straight. Authors claimed that their model has high accuracy in a testing environment.

Bakken et al. [46] worked on almost the same idea as [45] in building a model, but they design their robot to works in the agriculture section (crop row-following). In the test results, they also referred to the accuracy of the presented model.

Gaya et al. [47] investigated Deep Learning (DL) to build an obstacle avoidance model to control Autonomous Underwater Vehicles (AUVs). The AUV captured images using a single monocular camera and utilized a deep neural network to build a transmission map. The transmission map can specify the Region of Interest (RoI) for the taken video frames to determine the next state direction, leading to avoiding obstacles. The results depicted that the approach can efficiently determine the RoI and direct the robot to escape through free areas and avoid obstacles.

Li et al. [48] merge both Primal-Dual Neural Network (PDNN) and Model Predictive Control (MPC) techniques to present a new steering model that works on dynamic and kinematics field. The proposed paradigm’s focus was the optimization, where it iteratively calculated as a quadratic programming (QP) then it was resolve via PDNN. The developed scheme firstly controls the robot’s velocity as a part of the kinematic part. After that, in the dynamic aspect, the torques were changed to handle the steering task. Their test results indicate that the presented model was better in steering control compared to CNN only.

Sharma et al. [49] proposed DyHS algorithm, a hybrid scheme that combines the Lyapunov theory and Harmony Search (HS) to build a fuzzy tracking system to control mobile robot navigation. The controller consists of two sections, one for X-axis and the other for Y-axis direction motion. DyHS exploit the stability of Lyapunov theory and control ability in HS to achieve the required automation system. The presented model was tested in real-life and simulation experiments as well, and the result demonstrates that DyHS shows better performance than particle swarm optimization and genetic algorithm.

Harandi et al. [50] worked on combining three algorithms, Reinforcement learning (RL), Supervised Learning (SL), and state-representation learning to produce a new paradigm; this model extracts features more efficiently and control a mobile robot. The proposed model was based on a weighted sum of the extracted characteristics. The controller has two levels in calculating the weights in NN, where SL was used for hard-tuning while RL was utilized for fine-tuning. The experimental outcomes show that the model was effective and powerful in a path tracking task.

Franco et al. [51] present a new trajectory tracking scheme that builds on two mechanisms. The first technique uses the Extended Kalman Filter (EKF) algorithm to train a discrete-time Recurrent High-Order Neural Network (RHONN). The second one uses the inverse optimal model to prevent solving the Hamilton Jacobi Bellman (HJB) equation. These two techniques were cooperated to determine the best path and use it. After testing the controller, the high efficiency of the tracking task was evident.

Tai et al. [52] merge Convolutional Neural Network (CNN) and fully connected layers as a decision making in a complex form to perform steering control for a mobile indoor robot. The system accepts a raw image as input then decide the orientation according to that. The captured depth image will be presented to CNN for feature extraction and selecting the effective
ones; this information will be passed to the fully connected network that utilizes a regression method to determine the results. Steering command output from regression process will take five values each defines a specific direction control: ‘0’ for ‘turn to the full right’, ‘1’ for ‘turning half-right’, ‘2’ for ‘move directly’, ‘3’ for ‘turn half-left’, and ‘4’ for ‘turn to the full left’. The results indicated high obstacle avoidance performance, and the authors claimed that the proposed model is similar to that human make decisions.

Giusti et al. [53] used a Deep Neural Network (DNN) as a supervised classifier to create a mobile robot model to recognize and follow forest trails. The network firstly was trained with (17,119) frames to adopt network structure and help it in the classification task. The system getting input data from one camera, the incoming image was resized to 101x101 pixels; as an RGB format, the image will have a dimension of (3x101x101) and will be passed to the input layer of the network. The input image will finally be classified to one of the three available classes: turning left, go straight, and turn right. The training phase’s advantage makes the proposed scheme’s output layer put each image into one of the classes based on the probability. According to the selected category, the robot moves to that direction. Testing results show that this system over-perform other models.

Lei and Ming [54] introduced a new paradigm for mobile robot controlling based on Deep Q-Network (DQN). The proposed model utilizes a supervised approach for the feature extraction and reinforcement method to process and predict the output. The convolutional neural network architecture was formalized in the Q-value prediction of Q-network model. The robot will navigate in a corridor by taking RGB-D images and passes it to the CNN for feature extraction. The data go to the Q-learning network to determine the output (as a reinforcement process) and the next movement to avoid obstacles. Findings of testing the robot in a different corridor (testing areas) show the robustness of the proposed scheme and its efficiency.

English et al. [55] provide a new scheme to control an autonomous agricultural vehicle that detects crops rows in a field. The vision-based robot captures images and utilizes the 3D-structure, texture, and colors parameters to do the guiding task. The input information was processed via the Support Vector Machine (SVM) algorithm to perform a regression in calculating the output. The proposed model used SVM with Radial Basis Function (RBF) kernels, $\gamma=0.5$, $\nu=0.1$, and $c=12.5$ to perform an efficient regression process. The proposed system learns online and utilizes the gained knowledge to recognize the offset space between crops rows. The results demonstrate that the robot can apply to a wide range of fields and do online steering efficiently.

Jia et al. [56] utilized both Convolutional Neural Networks (CNN) and Deep Belief Network (DBN) to create a Deep Neural Network (DNN) model for the prepuce of obstacle detection and avoidance. While CNN is used to generalize some blocks’ local information (candidate ones), the DBN will generalize the complete image’s global data. However, the selected blocks’ position was determined. Merging the available information from blocks location, local, and global information. The model will recognize the segments with obstacles; nevertheless, the proposed model also calculates the obstacles’ depth. The model was trained with a large dataset to classify and identify obstacles from other blocks. The results indicate the ability of the scheme to detect obstacles and infer its depths.

Salavati and Mohammadi [57] propose almost the same model as [56]. The difference is that they used the unsupervised model (UnspVGG16) to extract the global features. At the same time, GoogleNet was utilized as a CNN supervised model to extract the local features. The other difference that they utilized the neighbouring blocks as well in the classification task. Their results show an improvement in accuracy compared to other models.

Zhu et al. [58] presented two models based on reinforcement learning. Besides, tried to solve the lack of generalization capability and multi-training issues related to that learning method. The two collaborated to give best results to perform visual-based navigation. To
diagnosed the first problem, the authors introduced an actor-critic scheme, that provides better generalization for the features. The second issue was addressed by proposing AI2-THOR framework, which offers high-quality 3-Dimensional scenes and efficiently provides many training data. Experiment outcomes indicate that the proposed models converge faster than regular reinforcement learning model. Furthermore, it gives a better generalization, and it can be applied to continuous and discrete domains.

Telles et al. [59] worked on building a navigation controller for an autonomous underwater robot. Combining the linear iterative clustering algorithm with the nearest neighbor classification model. The proposed model will capture an image and define the Region of Interest (RoI), then try to divide it into super-pixels. The model will then classify the super-pixels and check if they represent water or an obstacle object in the water. It is done according to position, shape, texture, and colors characteristics. The super-pixel will consider as an obstacle when an irregularity appears compared to the neighbors ones. The controller will determine the new direction toward the obstacle-free path and escape to it. The proposed model was tested in simulation and real-life robot, and both results show the effectiveness of the model.

Kaufmann Et al. [60] introduced a new scheme to control an autonomous drone for obstacles avoidance and trail tracking tasks. The proposed model merges path planning algorithms and CNN. The network will get the captured images and maps it in the shape of a waypoint to determine the next direction and the current speed. That is done via the planner algorithm, which instructs the corresponding motor to respond. Then the robot will reach the desired destination through the planned trajectory. The proposed model was tested in real life and simulation as well. The results demonstrate the efficiency of the scheme compared to the professional human pilot and state-of-the-art navigation models.

Sales et al. [61] combined Artificial Neural Networks (ANN) and Finite State Machines (FSM) to build an approach for mobile robot control. The robot takes images and feeds it to the ANN that segment it, analyze it, and classify the region in the image and consider the RoI to move toward it. Then, the ANN’s output will be passed to the FSM to determine the robot’s current state, and calculate the appropriate behavior that the robot should do based on the information from the previous stage (ANN stage). The results indicate convenient results and show that the proposed algorithm is a promising method used in self-driving cars.

Ronecke and Zhu et al. [62] present a new paradigm to efficiently navigate a self-driving vehicle in the road without colliding with other obstacles. The proposed model was based on reinforcement learning in collaboration between a deep Q-Network learning and the control theory. Images were captured, and the Q-Network was trained to make an action to avoid obstacles and plan the path. The proposed model was tested on two different roads, and the results show that the model can be used to drive a car efficiently and safely.

Manderson et al. [63] proposed a model to control an underwater vehicle based on Convolutional Neural Network (CNN). Consisting of five layers that finally determine the yaw and pitch angles. The captured image processed by the controller as a classification task to detect obstacles and avoid them.

Shkurti et al. [64] introduced a scheme near the one proposed in [63]. However, it can be deployed to serval robots to collaborate to perform the navigation task. And it works on long-distance obstacle avoidance, not a short distance.

Chuixin and Hanxiang et al. [65] build an Automatic Guided Vehicle (AGV) with a vision-based machine learning controller. The proposed model utilizes deep learning in Convolutional Neural Network (CNN) form. The network consists of 11 layers, seven of them were convolution layers, while the remaining four were fully connected layers. The captured image was resized to \((129*225)\) before entering the network; after that, it will feed to the CNN and go through the first five layers with a \(5*5\) core size. Here the system will rescale and extract
features. Features number will be 24,36,48,64, and 64 respectively in each layer. The two remaining convolution layers with 3*3 core size will extract features without resizing. The signal then goes through the four fully connected layers with 1146, 100, and 50 neurons in the first three layers; then the last one represents the output steering control direction. Test results indicated the proposed system’s effectiveness and how it can be deployed in many industrial fields.

<table>
<thead>
<tr>
<th>Ref.</th>
<th>Robot Type</th>
<th>Technique used</th>
<th>Contribution</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>Wheeled Mobile Robot (WMR)</td>
<td>Combined Deep Learning with Genetic Algorithm</td>
<td>Proposed (TCFS) a feature selection based on state transition probability to control a mobile wheeled robot.</td>
<td>Reduce input dimensional while saving the performance</td>
<td>Depend on state probability</td>
</tr>
<tr>
<td>39</td>
<td>Simulated Robot</td>
<td>Joints characteristics from Self Organizing Maps and ARN</td>
<td>Lower layers nodes will map the incoming data to the output via ARN, and the upper layers will resolve the locomotion issue by then optimizing the usable trajectories.</td>
<td>Avoid non-linear inverse expression when controlling joint angles and torque</td>
<td>Difficult to apply in real-world robot</td>
</tr>
<tr>
<td>40</td>
<td>Wheeled Mobile Robot (WMR)</td>
<td>Merged fuzzy processing and Genetic Algorithm (GA)</td>
<td>Image processed and edges were detected via fuzzy system improved by bacterial algorithm, then the path calculated via GA and A*.</td>
<td>Reduce computation time</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>Zen360 Wheeled Robot</td>
<td>Four layers of backpropagation Neural Network (NN)</td>
<td>The presented framework uses four reinforcement algorithms that work asynchronously to train the NN controller in different domains.</td>
<td>Simple calculations</td>
<td>Not sensitive to fast change</td>
</tr>
<tr>
<td>42</td>
<td>Simulated via Atari domain</td>
<td>NN and utilizing asynchronous gradient descent for deep reinforcement learning</td>
<td>The controller has two-stage, first one to determine path curvature, and the other one to track the calculated path, with a gradient descent algorithm.</td>
<td>Multi-algorithms provide more system stability</td>
<td>Extensive computations</td>
</tr>
<tr>
<td>43</td>
<td>Wheeled Mobile Robot (WMR)</td>
<td>Adaptive Neuro-Fuzzy Interface System (ANFIS)</td>
<td>The zero-order Takagi-Sugeno fuzzy was applied as the central controller, which was utilized as obstacle avoidance.</td>
<td>Robust control method</td>
<td>Time delay due to two cascaded controllers</td>
</tr>
<tr>
<td>44</td>
<td>E-puck mobile robot</td>
<td>Combined Reinforcement learning, Supervised Learning, and Fuzzy</td>
<td>The proposed model contains 5 CNN layers followed by three fully connected layers</td>
<td>Relatively high performance.</td>
<td>Difficulty deal with dynamic obstacles</td>
</tr>
<tr>
<td>45</td>
<td>iRobot Roomba</td>
<td>Convolutional Neural Network (CNN)</td>
<td></td>
<td></td>
<td>Need more marks on the</td>
</tr>
<tr>
<td>Page</td>
<td>Field</td>
<td>Model/Technique</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-------------------------------</td>
<td>------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>Agriculture mobile robot</td>
<td>Convolutional Neural Network (CNN)</td>
<td>that directly gives three outputs (orders) to turn right, turn left, and go straight</td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>Autonomous Underwater Vehicles (AUV)</td>
<td>Deep Neural Network (DNN)</td>
<td>Almost the same to [45] but adapted to follow crop rows.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>Wheeled Mobile Robot (WMR)</td>
<td>Merged primal-dual neural network (PDNN) and Model predictive control (N MPC) techniques</td>
<td>Utilized a deep neural network to build a transmission map from a captured image. The map can specify the Region of Interest (RoI) for the taken video frame to determine the next state direction. The proposed paradigm focuses on optimization, where iteratively calculated as a quadratic programming (QP), it was resolved via PDNN. The developed scheme controls the robot's velocity and direction.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>Wheeled Mobile Robot (WMR)</td>
<td>Joints Lyapunov theory and harmony search (HS)</td>
<td>Introduced (DyHS) controller consists of two sections, one for X-axis and the other for Y-axis direction motion. It exploits the stability of Lyapunov theory and control ability in HS.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>Simulation via WEBOTS and MATLAB</td>
<td>Combines Reinforcement learning, Supervised Learning, and state-representation learning</td>
<td>The presented model based on a weighted sum of the extracted features. The controller has two levels in calculating the weights in NN, where SL was used for hard-tuning while RL was utilized for fine-tuning.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>Simulated Robot</td>
<td>Cooperation between extended Kalman filter (EKF) and discrete-time recurrent high-order neural network (RHONN)</td>
<td>The scheme builds on two mechanisms; the first technique uses (EKF) algorithm to train an (RHONN). Secondly, to prevent solving the Hamilton Jacobi Bellman (HJB) equation, the inverse optimal model was used.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>TurtleBot</td>
<td>Convolutional Neural Network (CNN)</td>
<td>The image is presented to CNN for feature extraction; then it will be passed to the fully connected network that used a regression method to control decision. It uses ‘Discrete classification’ which is not enough to be environment to work well.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Not smooth in turning and changing directions.

Build a transmission map as a relative depth map.

Need clear images that not always available.

It controls velocity and orientation to avoid obstacles.

Determines both local and global search to present a high level of stable automation.

Time delay in decision making.

Stable against uncertainty and scalability for broad range areas.

Relatively high No. of failures during a training phase.
**Research Article**

<table>
<thead>
<tr>
<th>Page</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>Quadrotor drone</td>
<td>Deep Neural Network (DNN) determine steering commands. Theses commands take five values; each defines a specific direction. DNN as s supervised classifier was used to process the input image and be classified into one of the three classes representing the controlling director: Turning Left, Go Straight and Turn Right.</td>
</tr>
<tr>
<td>54</td>
<td>Simulation via Gazebo</td>
<td>Deep Q-Network (DQN) Work on whole input image at once Lose some frames when processing High stability Need extensive training</td>
</tr>
<tr>
<td>55</td>
<td>Wheeled Mobile Vehicle</td>
<td>Merge Support Vector Machine (SVM) algorithm and Radial Basis Function (RBF) The provided controller utilizes the 3D-structure, texture, and colors parameters, and it used SVM with Radial Basis Function (RBF) kernels to perform the guiding task.</td>
</tr>
<tr>
<td>56</td>
<td>Wheeled Mobile Vehicle</td>
<td>Use both Convolutional Neural Networks (CNN) and Deep Belief Network (CBN)</td>
</tr>
<tr>
<td>57</td>
<td>Simulated Robot</td>
<td>Joint Unsupervised model (UnspVGG16) with CNN Not only recognize obstacles but determine their depth as well Short (point to point) trajectory tracking</td>
</tr>
<tr>
<td>58</td>
<td>SCITOS mobile robot</td>
<td>Reinforcement learning</td>
</tr>
<tr>
<td>59</td>
<td>Autonomous Underwater Vehicles (AUV)</td>
<td>Combining linear iterative clustering algorithm with the 'nearest neighbor' classification model</td>
</tr>
<tr>
<td>60</td>
<td>Autonomous drone</td>
<td>Merge path planning algorithms and Convolutional neural network (CNN)</td>
</tr>
<tr>
<td>61</td>
<td>Surveyor SRV-1Q mobile robot</td>
<td>Used both artificial neural networks (ANN) and finite state machines (FSM)</td>
</tr>
<tr>
<td>62</td>
<td>Simulated robot</td>
<td>Reinforcement learning</td>
</tr>
<tr>
<td>63</td>
<td>Autonomous Underwater Vehicles (AUV)</td>
<td>Convolutional Neural Network (CNN)</td>
</tr>
<tr>
<td>64</td>
<td>Autonomous Underwater Vehicles (AUV)</td>
<td>Deep Learning</td>
</tr>
<tr>
<td>65</td>
<td>Wheeled Mobile Robot (WMR)</td>
<td>Convolutional Neural Network (CNN)</td>
</tr>
</tbody>
</table>

Finally, it is worth mentioning that in this paper, we concentrated on the control techniques of Machine Learning (ML). However, there are other methods to build robot controllers. For instance, using Hamming distance and intersection over union using the Robot Operating System (ROS) [66]. Or depend on some image processing filter like Kernelized Correlation.
Filter (KCF) tracker [67]. Or attaching an FPGA control system to the robot to perform the image processing and guiding tasks [68]. While [69] utilized Python to build a model using the ‘Raspberry Pi 3 model b+’; or even some techniques related to stability like gain scheduling technique [70]. It’s worth to mention that this technique (Machine Learning technique) can be used for different applications that may be merged with some robots to perform important tasks [71-75].

5. Conclusion

All things considered, autonomous mobile robots get significant attention from academics and industrial sectors in the last decades. They can be found in medical, scientific, and manufacturing fields. Subsequently, finding a robust control system is essential to such robots to prevent damages. This paper presents a scientific and global overview of vision-based robot controller techniques. The focus here was on machine learning, including NN and fuzzy algorithms. The article introduces the controller from different points of view, including path planning, navigation, trajectory tracking, and obstacle avoidance. We strongly believe that designing a vision-based robot utilizes a 3D camera and artificial intelligent NN control system to overcome the most confused and failure in autonomous mobile robots.

References:


