Realization and Control of a Mobile Robot based on a Dynamic Model with the Implementation of Artificial Intelligence

ABDELGHAFOUR SLIMANE TICH TICH
Mechanical Engineering Department LGMM Laboratory
University of Skikda
ALGERIA

FOUED INEL
Mechanical Engineering Department, Automatic Laboratory,
ALGERIA

MOHAMMED KHADEM
Mechanical Engineering Department LGMM Laboratory
ALGERIA

Abstract: - In this paper, a mobile robot (tricycle robot) and its kinematic and dynamic model are presented. Moreover, its simulation and model control are derived. For a mobile robot to be autonomous, it must perform command tasks and perceive the environment. In this context, navigation plays an important role in the interaction of the robot with its environment. It consists in the determination of possible trajectories by the robot to follow a predefined trajectory. To accomplish this task, our approach relies on the dynamic motion of the robot to generate admissible trajectories. The reference trajectory is first constructed based on a predefined trajectory. To eliminate the navigation problem, an optimization problem with constraints, it is necessary to reduce the difference between the predicted trajectory of the robot and the desired trajectory. Moreover, it is possible to control the behaviour of the robot by using a trajectory parameterized with the dynamic model and its control. Finally, the display of experimental results up to the implementation of the object detection..

Key-Words: - Non-holonomic, mobile robot, sliding mode, fuzzy control, recognition of objects.

Received: April 12, 2022. Revised: November 22, 2022. Accepted: December 17, 2022. Published: December 31, 2022.

1 Introduction

A mobile robot can be considered synonymous with a vehicle, as it consists of the physical and mechanical structure that allows the robot to move, [1]. Other disciplines, such as automation engineering, are involved and may shed a different light on the matter, but do not change the problems to be solved, [2].

The interest in the determination of mobile robots is to perform certain tasks for humans, in different environments, with boundaries and uneven terrain, even in places inaccessible to humans [3], [4].

Important research work has been devoted to motion planning. The purpose of which is to transition a system from a certain initial state to a certain final state when the motion command is to solve basic navigation problems, [5]; follow a reference path, stability to a desired position, [6]. It is not easy to take control of a non-holonomic system from a start to a finish configuration. This problem has attracted the attention of robotics automation engineers, hence the possibility of applying robust control [7], [8]. The aim of the project was to develop a non-holonomic mobile robot. The presentation of the obtained simulation results and the experimental part show the realization of the prototype with the application of object recognition.

2 Non-holonomic Mobile Robot Model

A characteristic illustration of a non-holonomic mobile robot can be seen in Fig.1.

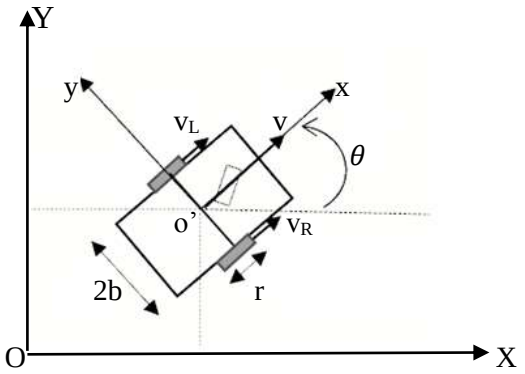

Fig. 1: Representation of the mobile robot

The wheeled mobile robot has two rear wheel drives and a self-supporting passive support wheel. To enable translational and rotational motion, the rear wheels are controlled independently. The resulting linear velocity at the point midway along the axis between the two rear wheels is given by: $v = \frac{V_L+V_R}{2}$ and the angular velocity of the moving robot is given by: $w = \frac{V_L-V_R}{2b}$. Two wheels have the same radius as at and two driving wheels are separated by 2b, the ground center of the mobile robot is at point o', [9].
The mobile robot is parameterized by the vector.

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (1)$$

The location of the Cartesian coordinates of the points (x, y) and θ the orientation of the robot. The linear velocity of the point o' and the angular velocity of the robot respectively represent the reference quantities v and ω, [10].

2.1 Kinematics
By deriving equation (1), we obtain the following transformation which describes the kinematic model of the robot, [11], [12].

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} cos\theta & 0 \\ sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (2)$$

With: $$J(\theta) = \begin{bmatrix} cos\theta & 0 \\ sin\theta & 0 \\ 0 & 1 \end{bmatrix} \quad (3)$$

2.2 Dynamics
The dynamic model of a mobile robot with electric rotor inertia and torque-based steering can be described as follows [11], [12]:
The inertia matrix of the system is given by:

$$M = \begin{bmatrix} m + \frac{2b^2 I_k}{r^2} + \frac{2b^2 I_m}{r^2 n_g^2} & 0 \\ 0 & I_c + \frac{2I_k}{r^2} + \frac{2I_m}{r^2 n_g^2} \end{bmatrix} \quad (4)$$

With:
m: the total mass of the vehicle.
I_c: moment of inertia of the vehicle.
I_m: moment of inertia of the wheel.

$$H = \begin{bmatrix} \frac{2}{r^2}(\boldsymbol{\xi} + \frac{\boldsymbol{\xi_m}}{n_g^2}) & 0 \\ 0 & \frac{2b^2}{r^2}(\boldsymbol{\xi} + \frac{\boldsymbol{\xi_m}}{n_g^2}) \end{bmatrix} \quad (5)$$

$$B = \begin{bmatrix} \frac{1}{r.n_g} & \frac{1}{r.n_g} \\ \frac{b}{r.n_g} & -\frac{b}{r.n_g} \end{bmatrix} \quad (6)$$

The forces that produce the torque τ_1 and τ_2 are:

$$F_a = \begin{bmatrix} D_{av}.|v|.v \\ D_{\omega}.|\omega|.\omega \end{bmatrix} \quad (7)$$

With:
τ: the external disturbances
D_{ω}: The coefficient of the wheel viscous friction.

$$F_r = \begin{bmatrix} \frac{1}{2}.m.g.C_r.\tanh(v_f.r.\omega_p) + \tanh(v_f.r.\omega_L) \\ \frac{b}{2}.m.g.C_r(\tanh(v_f.r.\omega_p) - \tanh(v_f.r.\omega_L)) \end{bmatrix} \quad (8)$$

The system matrix is:

$$E = M^{-1}[B.T_m - H.u(t) - F_a - F_r] \quad (9)$$

With the transformation matrix:

$$J = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{2b} & -\frac{r}{2b} \end{bmatrix} \quad (10)$$

2.3 Control System Architecture
The vector below gives the tracking error.

$$q_e = e(t) = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} x_d \\ y_d \\ \omega_d \end{bmatrix} - \begin{bmatrix} x_q \\ y_q \\ \omega \end{bmatrix} \quad (11)$$

A PID sliding surface is chosen to improve the performance and flexibility of the robot, [13].

$$\dot{\sigma}(t) = k_p e(t) + k_i \int e(t)dt + k_d \dot{e}(t) \quad (12)$$

To avoid $\sigma(t)$ tending to zero, we should contribute the damping of $\sigma(t)$ that results:

$$\dot{\sigma}(t) = k_p e(t) + k_i \int e(t)dt + k_d \dot{e}(t) \quad (12)$$

Table.1. Rule base interface for relationship between error and derror

		Error				
Derror		NB	NS	ZE	PS	PB
	NB	NB	NB	NS	NS	ZE
	NS	NB	NS	ZE	ZE	PS
	ZE	NS	ZE	ZE	ZE	PS
	PS	NS	ZE	ZE	PS	PB
	PB	ZE	PS	PS	PB	PB

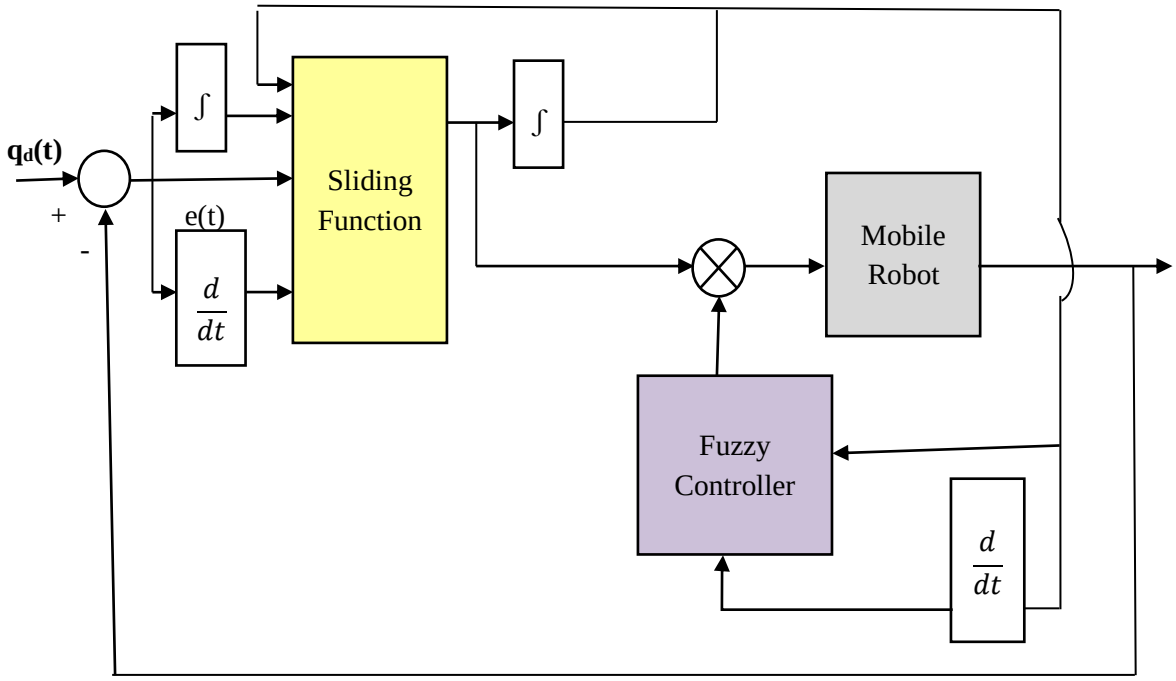

Fig. 2: Architecture of the control system

$$\dot{\sigma}(t) = k_p e(t) + k_i \int e(t)dt + k_d \dot{e}(t) - k\sigma(t) \quad (13)$$

Fuzzification is the process of converting net values into degrees of agreement with the linguistic terms of fuzzy sets [14].

In order to match the direction of the robot, the delta error and the error values are used as input parameters.

These rules specify the relationship between the error sensor and the delta error sensor in terms of linguistic values. In the present work, a Mamdanian participation method is used. After the inputs are fuzzified, FLC determines the degree to which the prior rule is satisfied. The equation of the Mamdanian method to check the fuzzy implication, [15].

$$\mu_k = \max[\min\{\mu_k, \mu(error'i), derror(j)\}] \quad (14)$$

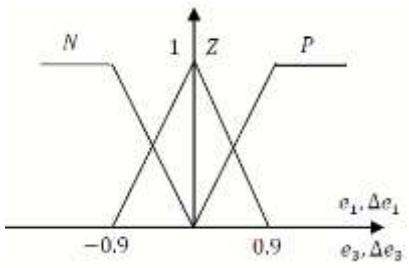

Fig. 3: Membership fuzzification

3 Results and Experiments

The goal of the control system is to track the exact references of the trajectories desired by the robot by applying the resolved equations of the kinematic and dynamic models in combination with the fuzzy sliding control. The simulations show the performance of the robot in tracking a predefined

trajectory. The characteristics of the robots are listed in Table 2.

Table.2. Parametric values

Parameter Symbol	Unit	Value
Mass of mobile robot	kg	4
Moment of inertia of mobile robot	kg m^2	0.0945
Radius of wheel	m	0.05
Length of platform	m	0.25
Breadth of platform	m	0.38
Motor coefficient	N/V	0.087
Motor coefficient	kg/s	11.4

Simulation results illustrating tracking performance are shown in the figures below.

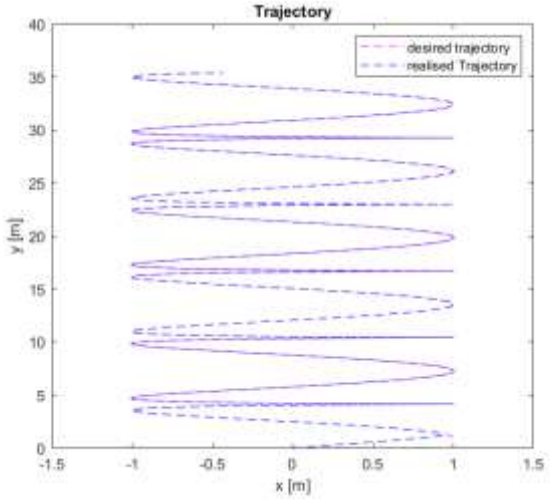

Fig. 4: Trajectory followed by the robot

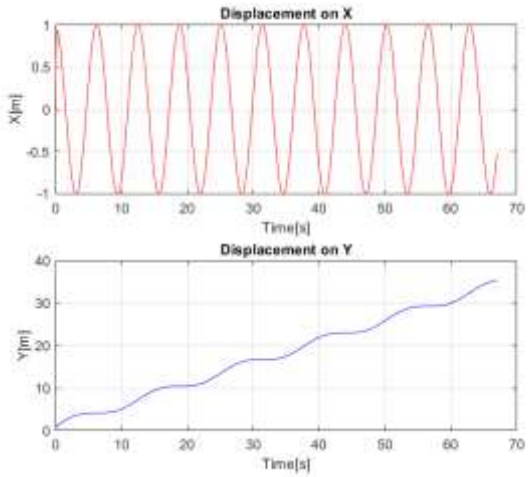

Fig. 5: Displacement in X and Y

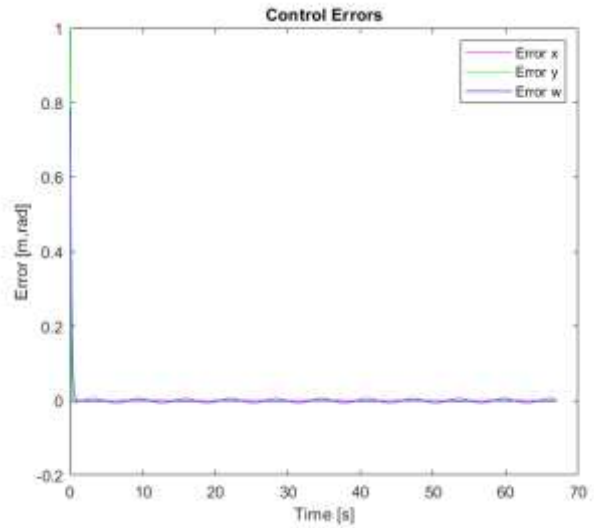

Fig. 6: Following errors

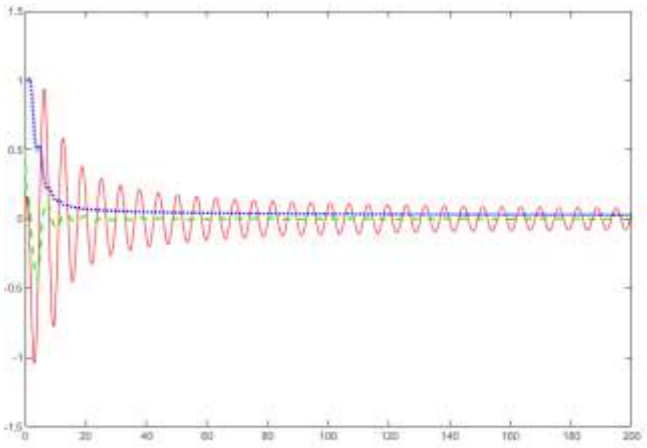

Fig. 7: control errors and derrors

The difference between the control effort without disturbance and the control effort with disturbance that results from solving a forward dynamic problem is shown graphically in Figures 8 and 9.

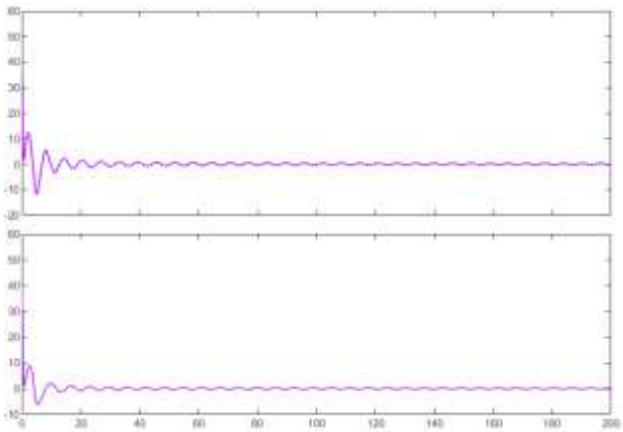

Fig. 8: Comparison between control efforts without interference

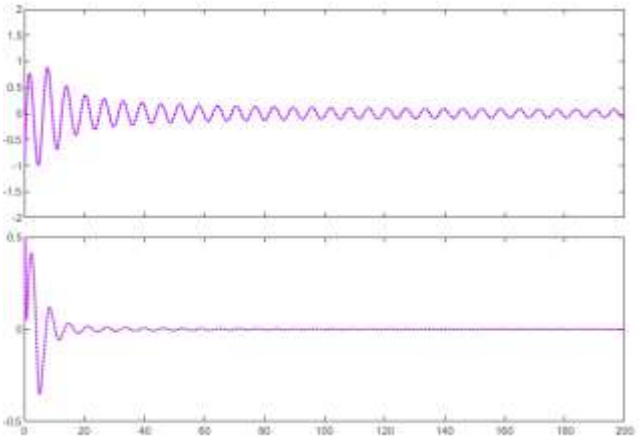

Fig. 9: Comparison between control efforts with disturbance

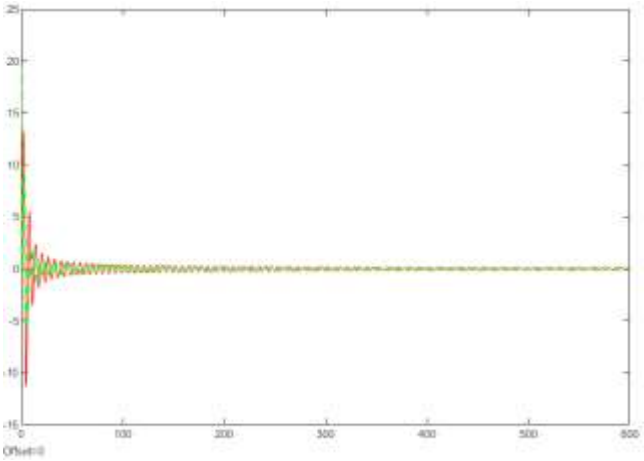

Fig. 10: Second derivative of the sliding function

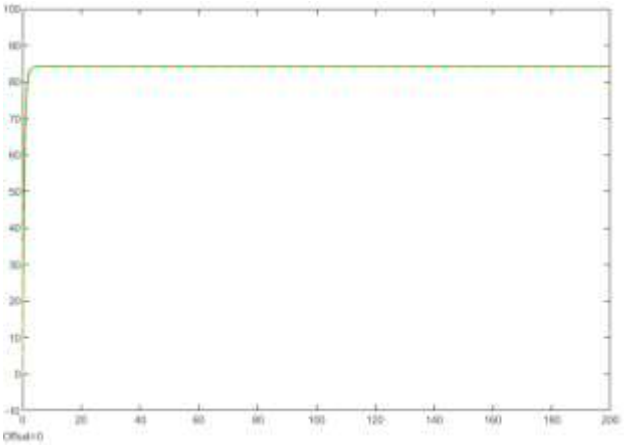

Fig. 11: Desired inputs and control inputs

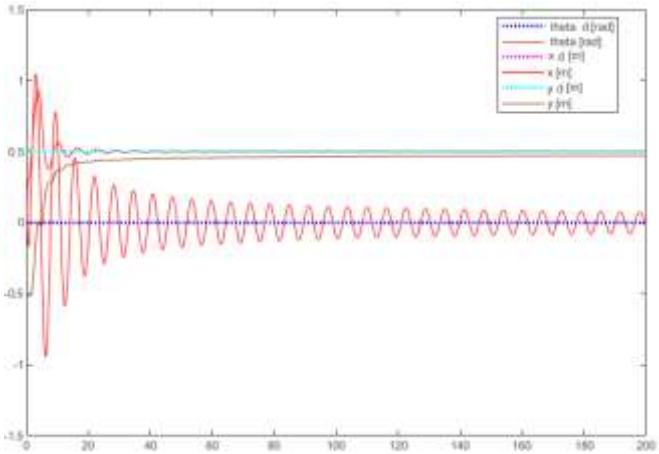

Fig. 12: Comparison between displacement, desired orientation and actual position

The results obtained show that the tracking is almost perfect, although certain factors were not taken into account in the calculation of the command.
The posture error of the robot converges to almost zero, and the actual robot velocities converge with the selected reference velocities. According to the simulation results, the dynamics of the system requires special attention, with the relative velocity errors requiring minimal movement.

3.2 Implementation of Object Detection

The goal of this study is to distinguish three robots (humanoid robot, drone, wheeled robot) by first using a small database (60 images per class).
Then, finally, we will study the performance of our model by varying the size of our database without changing the parameters, and thus be able to infer an average size.

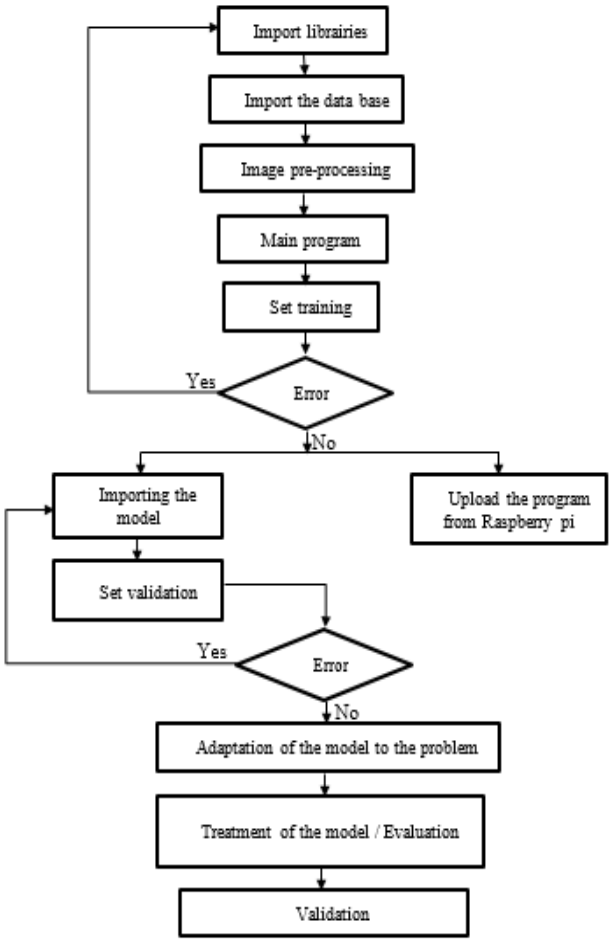

Fig. 13: Organizational chart

If we keep the same parameters and vary the size of the database, we can see that the number of correctly classified images increases when the database is larger. We find that the classification rate is highest for a database of 700 images/class.

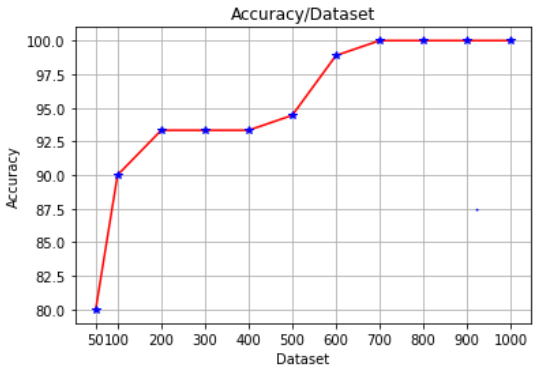

Fig. 14: Performance of the model based on the dataset.

In this test, however, we have fixed the values of the hyperparameters, which play an extremely important role in the performance of the model; we have correctly adjusted the hyperparameters for each model. It is possible that we have a powerful model with a database much less than 700 images/class.

The following figures show the results of classification between wheeled robots, humanoid robots, and drones without increasing the amount of data.

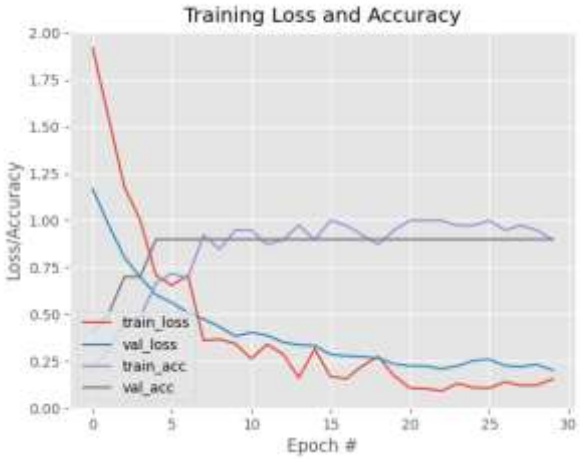

Fig. 15: Training loss and accuracy

And the table represents the model parameters that result after running our algorithm.

Table.3.Model parameters

Parameters	precision	recall	F1-score	support
Drone	0.75	1.00	0.86	18
Humanoid robot	1.00	0.50	0.67	19
Mobile robot	0.75	1.00	0.86	20
Accuracy			0.88	57
Macro avg	0.83	0.83	0.79	57
Weighted avg	0.85	0.80	0.78	57

Figure 14 shows the evolution of our metrics as a function of the evolution of the epochs; we are interested in the accuracy of validation, which is an indicator of the progress of our model. Indeed, the accuracy indicates the percentage of correct predictions.

The following figures show the results of the classification between wheeled robots, humanoid robots, and drones as the amount of data increases.

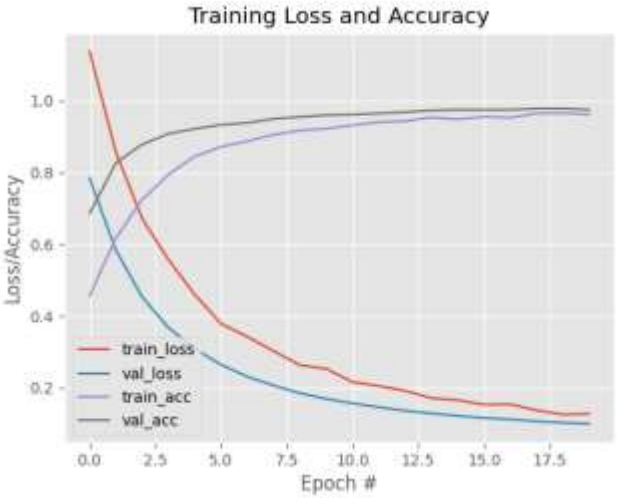

Fig. 16: training loss and accuracy

And the table presents the model parameters that result after running our algorithm.

Table 4. Model parameters

Parameters	precision	Recall	F1-score	support
Drone	1.00	1.00	1.00	18
Humanoid robot	1.00	1.00	1.00	19
Mobile robot	1.00	1.00	1.00	20
Accuracy			1.00	57
Macro avg	1.00	0.92	1.00	57
Weighted avg	1.00	0.90	1.00	57

Figures 17, 18 and 19 show the image processing for our three classes in real time.

Fig. 17: Image classification of a mobile robot

Fig. 18: Image classification of the humanoid robot Microsofot

Fig. 19: Image classification of a drone

The prototype mobile robot used in this part is the same as in Figure 20.

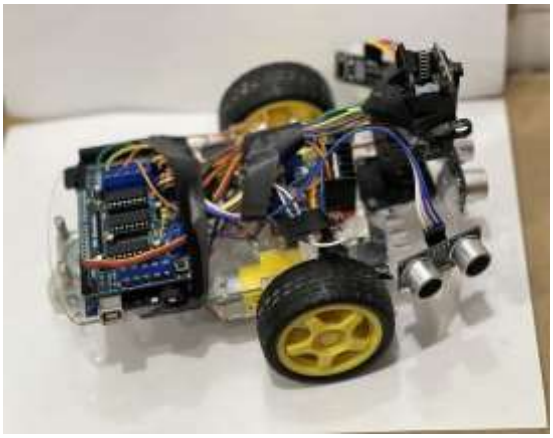

Fig. 20: Image of the physical prototype of a mobile robot.

4 Conclusion

In this paper, path tracking by a combination of two controllers for non-holonomic mobile robots is presented. The proposed controller is tested for different trajectories. Simulation and experimental

results show that it is very capable of following any trajectory. In addition, a new implementation of a computer vision solution using deep learning algorithms to detect and identify different objects on the omnidirectional robot has been developed. We adapted a variant of the MobileNetv2 neural network architecture to quickly recognise and classify images of objects. Our visual system provides a reliable solution for recognising images even when the object is partially occluded. We expect to use the data computed by this system to integrate advanced tracking into the robot in the future.

References:

[1] D. Filliat, "Robotique mobile", *Fondation UNIT et GDR robotique*, 2012.

[2] J. P. Laumond, "La robotique mobile," *Edition Hermes Science Publicatio*n, Paris, 2001.

[3] A.Slimane Tich Tich, F.Innel, G.Carbone, "Realization and Control of a Mecanum Wheeled Robot Based on a Kinematic Model". *Advances in Italian Mechanical Science*, vol 122(1), 2022, pp. 663-676.

[4] G. Tont, L. Vldreanu, M. S. Munteanu, D. G.Tont, Markov approach of adaptive task assignment for robotic system in non-stationary environments, *WSEAS Transactions on Systems*, 9(3), 273-282, (2010).

[5] M. Papoutsidakis, D. Piromalis, F. Neri, M. Camilleri, Intelligent Algorithms Based on Data Processing for Modular Robotic Vehicles Control, *WSEAS Transactions on Systems, 13, WSEAS Press (Athens, Greece),* 242-251, (2014).

[6] J. Zhang and H. Huang, "Steering angle prediction for autonomous cars based on deep neural network method*," *in 2020 Australian and New Zealand Control Conference (ANZCC*), 2020, pp. 205–208.

[7] E. Guechi, "Suivi de trajectoires d'un robot mobile non holonome : approche par modèle flou de Takagi-Sugeno et prise en compte des retards," *Phd Thesis, Annaba University*, Algeria, 2010.

[8] ZIDANI Ghania, Commande Robuste d'un Robot Mobile à Roues, *Phd Thesis, Batna University,* Algeria, 2017.

[9] R. Siegwart and I. R. Nourbakhsh, "Introduction to autonomous mobile robots," *Bradford Book, A MIT Press Cambridge,* Massachusetts, London, England, 2004.

[10] E. Hashemia, M. G. Jadidi and N. G. Jadidi, "Model based PI fuzzy control of four wheeled omni-directional mobile robots,'' *Robotics and Autonomous Systems* 59, 2011, pp 930–942.

[11] Au F. Pourboghrat and M. P. Karlsson, "Adaptive control of dynamic mobile robots with nonholonomic constraints," *Computers and Electrical Engineering*, 2002, pp. 241–253.

[12] Y. T. Wang, Y. C. Chen and M. C. Lin, "Dynamic object tracking control for a non holonomic wheeled autonomous robot," *Tamkang Journal of Science and Engineering*, vol.12,2009, pp. 339-350.

[13] Wang, H., Liu, L., He, P., Yu,M., Do,M.T., Kong, H.,Man, Z.:Robust adaptive position control of automotive electronic throttle valve using PID-type sliding mode technique. *Nonlinear Dyn*. 85(2),2016, pp 1331–1344.

[14] Radwan A.G., Noaman, I., "Generating Fuzzy Sets and Fuzzy Relations Based on Information", *WSEAS TRANSACTIONS on MATHEMATICS*, 2021, pp 178-179.

[15] Fahmizal, Chung-Hsien Kuo, Trajectory and Heading Tracking of a Mecanum Wheeled Robot Using Fuzzy Logic Control, *International Conference on Instrumentation, Control and Automation (ICA)*, 2016, pp. 57-58.