Neural Network Controller Design for a Mobile Robot Navigation; a Case Study

Tresna Dewi, Pola Risma, Yurni Oktarina, and M. Taufik Roseno
Department of Electrical Engineering
Politeknik Negeri Sriwijaya
Palembang, Indonesia, 30139
Email: tresna_dewi@polsri.ac.id, polarisma@polsri.ac.id, yurni_oktarina@polsri.ac.id, and m_taufik_roseno@polsri.ac.id

***Abstract*—Mobile robot are widely applied in various aspect of human life. The main issue of this type of robot is how to navigate safely to reach the goal or finish the assigned task when applied autonomously in dynamic and uncertain environment. The application of artificial intelligence, namely neural network, can provide a "brain" for the robot to navigate safely in completing the assigned task. By applying neural network, the complexity of mobile robot control can be reduced by choosing the right model of the system, either from mathematical modeling or directly taken from the input of sensory data information. In this study, we compare the presented methods of previous researches that applies neural network to mobile robot navigation. The comparison is started by considering the right mathematical model for the robot, getting the Jacobian matrix for online training, and giving the achieved input model to the designed neural network layers in order to get the estimated position of the robot. From this literature study, it is concluded that the consideration of both kinematics and dynamics modeling of the robot will result in better performance since the exact parameters of the system are known.**

***Index Terms*—Dynamics modeling; kinematics modeling; mobile robot navigation; neural network controller.**

I. Introduction

Nowadays, mobile robot are widely applied in various aspect of human life, such as a service robot, military robot, transport robot, underwater robot, and so on. The main issue of this robot is how to navigate safely to reach the goal or finish the assigned task when applied autonomously in dynamic and uncertain environment [1]- [9]. The most applied type of mobile robot is car-like robot, since this type is the easiest one to build but not necessarily the easiest one to control. Car-like mobile robot suffers from the non-holonomic constraint that requires a solution of a complex mathematical modeling, and many researches have been dedicated to address this problem.

Some studies have been done on developing trajectory tracking generation [7] [10] [11], path generator [12], pre-mapping algorithm [13] [14], and SLAM [15] are among the strategies designed to achieve a safe and reliable navigation for the robot. All of the previous mentioned strategies are meant to equip the robot with intelligent control, therefore if only the robot has its own "brain" than it can navigate itself autonomously in a known or unknown environment. "Brain" installed in robotics technology is called artificial intelligence, and the most used one is neural network. By applying neural network, the complexity of mobile robot control can be reduced by choosing the right model of the system [1]- [28].

In order to function properly according to its assigned task, a robot needs sensing devices [18] [19]. These sensing devices or sensors have to be modeled with a mathematical solution, however with the application of neural network controller, this complexity can be avoided. The large data and information from the installed sensor will be utilized by the training of neurons or synaptic weight. This learning process will be the feedback to the robot controller system [20]- [22]. Letting this controller to work alone or combine with conventional PID controller will be the choice of the researchers [23] [24].

A neural network application works best for a mobile robot applied with many types of installed sensors on it. The learning process in neural network will help in organizing the data, predicting the output and minimizing the errors and in the end creating the brain for the robot to decide the best path for its navigation. A simple application of neural network is a wall follower robot, this application can be easily extended into a "smart" wall follower wheel chair that is very useful in hospital application.

However, in spite the fact that neural network application makes the robot work effectively, in some cases, the only application of neural network was not enough. Researchers have developed strategies to combine it with other type of artificial intelligence algorithm, such as fuzzy logic [25] [26].

This paper discusses the application of neural network in mobile robot navigation by comparing the methods presented by previous researchers. This comparison is started by examining the kinematics [13] [14] [23] [24] [27] [28] and dynamics [11] [13] [17] [20]- [24] of mobile robot by considering its non-holonomic constraint and ignoring the possibility of slipping and skidding [28]. This paper is a review article in finding the simplest and yet effective neural network controller design for a car-like mobile robot. This study is the preliminary research, and the result will be a reference to design a suitable controller for the simulation and experiment of a real robot.

II. Mobile Robot Modeling

The first step of designing the best controller for mobile robot shown in Fig. 1 is by modeling the system. Modeling means putting the robot into an ideal version of the real system. There are two types of modeling, kinematics [2] [13] [14] [23] [24] [27] [28] and dynamics modeling [11] [13]

978-1-5386-0549-3/17/$31.00 ©2017 IEEE

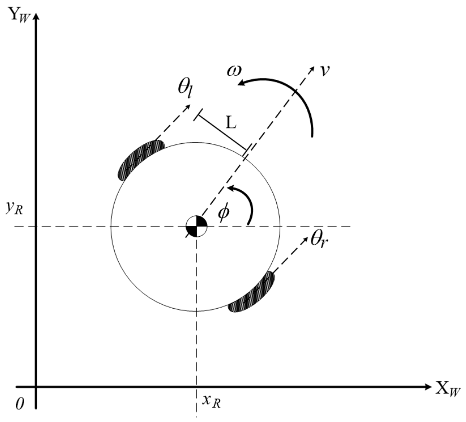

Fig. 1. Two-driven wheels and its coordinate frame

[17] [20]- [24], and in some studies, researchers also included modeling of robot actuators [23] to get the exact torque needed by the robot to follow the designed tracking trajectory.

A. Kinematics modeling

Kinematics modeling means learning the way robot moves in robot coordinate frame (x_R, y_R) relative to world coordinate frame (x_W, y_W). This position/motion presentation is not considering any forces applied to the system. Kinematics modeling establishes the equations of robot velocity $\dot{q} = \begin{bmatrix}\dot{x} & \dot{y} & \dot{\phi}\end{bmatrix}^T$ as the function of tires velocities, θ_L and θ_R.

The forward kinematics of two wheels differential driven mobile robot shown in Fig. 1 is

$$\dot{q} = \begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\phi}\end{bmatrix} = f\left(\dot{\theta_R}, \dot{\theta_L}, L, r, \phi\right), \tag{1}$$

where v and ω are the translational and rotational velocity of the robot, ϕ is the orientation of the robot, L is the distance from driving wheels to the robot axis in y-direction, θ_L and θ_R are left and right tires' angles, $\dot{x}$, and $\dot{y}$ are the translational velocities in x and y axis respectively, $\dot{\theta_L}$ and $\dot{\theta_R}$ are the velocities of the left and right tires, r is the tire radius, and $\omega = \dot{\phi}$

The inverse kinematics from eq. 1 is given by

$$\begin{bmatrix}\dot{\theta_R}\\ \dot{\theta_L}\end{bmatrix} = f\left(\dot{x}, \dot{y}, \dot{\phi}\right). \tag{2}$$

The relation between robot's translational (v) and rotational (ω) velocities and both tires velocities are calculated as follow

$$v = r\frac{\dot{\theta_R} - \dot{\theta_L}}{2}, \quad \text{and} \quad \omega = \frac{r}{2L}\left(\dot{\theta_R} - \dot{\theta_L}\right). \tag{3}$$

As it can be seen from Fig. 1 that $\dot{q_w} = R(\phi)\,\dot{q_R}$, therefore we have

$$\begin{aligned}\dot{q_w} &= \begin{bmatrix}\dot{x}\\ \dot{y}\\ \omega\end{bmatrix} = \begin{bmatrix}\cos\phi & -\sin\phi & 0\\ \sin\phi & \cos\phi & 0\\ 0 & 0 & 1\end{bmatrix}\frac{r}{2}\begin{bmatrix}\dot{\theta_R} + \dot{\theta_L}\\ 0\\ \frac{\dot{\theta_R} - \dot{\theta_L}}{L}\end{bmatrix}\\ \dot{q_w} &= \begin{bmatrix}r\frac{\dot{\theta_R}+\dot{\theta_L}}{2}\cos\phi\\ r\frac{\dot{\theta_R}+\dot{\theta_L}}{2}\sin\phi\\ \frac{r}{2L}\left(\dot{\theta_R} - \dot{\theta_L}\right)\end{bmatrix}.\end{aligned} \tag{4}$$

The main complexity of car-like mobile robot is the non-holonomic constraint [29] that can not be solved with simple integral solution. The non-holonomic constraint is represented by

$$\dot{x}_c\cos\phi + \dot{y}_c\sin\phi - a\dot{\phi} = 0, \tag{5}$$

where a is the distance from the center of gravity to the driving wheels axis in x-direction. However in this study, the driving wheel is located in the center of gravity as shown in Fig. 1, therefore $a = 0$.

The pure rolling constraints of the robot [29] are given by

$$\begin{aligned}\dot{x}_c\cos\phi + \dot{y}_c\sin\phi + L\dot{\phi} &= r\dot{\theta_R},\\ \dot{x}_c\cos\phi + \dot{y}_c\sin\phi - L\dot{\phi} &= r\dot{\theta_L}.\end{aligned} \tag{6}$$

The constraints state in eq. 5 and 6, can be rewritten as

$$A(q)\,\dot{q} = 0, \tag{7}$$

where

$$A(q)\begin{bmatrix}-\sin\phi & \cos\phi & 0 & 0 & 0\\ \cos\phi & \sin\phi & L & -r & 0\\ \cos\phi & \sin\phi & -L & 0 & -r\end{bmatrix}, \tag{8}$$

and

$$\dot{q} = \begin{bmatrix}\dot{x}_c & \dot{y}_c & \dot{\phi} & \dot{\theta}_R & \dot{\theta}_L\end{bmatrix}. \tag{9}$$

B. Dynamics modeling

Dynamic modeling is the study of the robot system by modeling the applied forces, and all the speeds related to the motion of the robot.

The dynamics modeling of mobile robot given in Fig. 1 is

$$M(q)\,\ddot{q} + V(q,\dot{q}) + F(\dot{q}) + G(q) + \tau_d = B(q)\,\tau - A^T(q)\,\lambda \tag{10}$$

where $M(q)$ is the symmetric positive definite inertia matrix, $V(q,\dot{q})$ is the centripetal and coriolis matrix, $F(\dot{q})$ is the surface friction matrix, $G(q)$ is the gravitational vector, τ_d is unknown disturbance, $B(q)$ is the input transformation matrix, τ is the input vector, $A^T(q)$ is the constraints matrix, and λ is the vector of forces for constraints.

By calculating and substituting parameters achieved from kinematics and dynamics mathematical modeling, Jacobian matrix is derived as $\frac{\partial q}{\partial v_c}$, or it can be expanded into

$$J = \frac{\partial q}{\partial v_c} = \frac{\partial q}{\partial v}\frac{\partial v}{\partial \tau}\frac{\partial \tau}{\partial e_c}\frac{\partial e_c}{\partial v_c}, \tag{11}$$

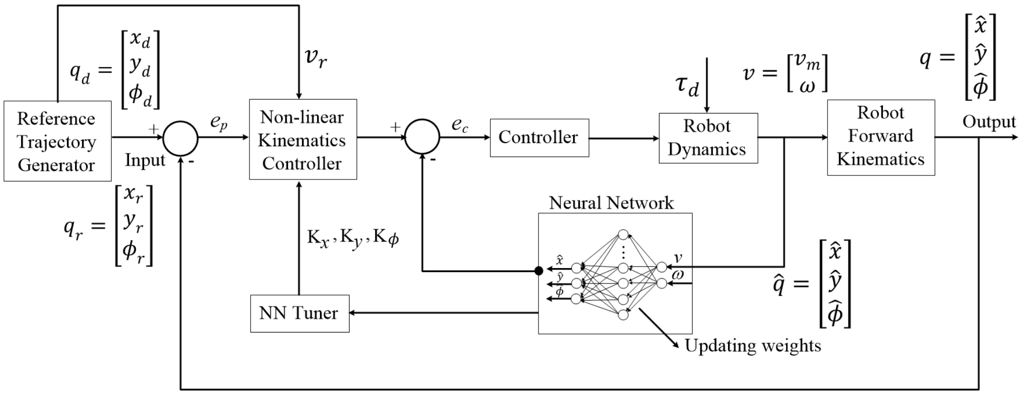

Fig. 2. The applied neural network for robot in Fig. 1

where v_c is the control law, calculated as

$$v_c = \begin{bmatrix} v_r \cos(e_\phi) + K_x e_x \\ \omega_r + K_y v_r e_y + K_\phi v_r \sin(e_\phi) \end{bmatrix}. \quad (12)$$

Control law v_c is calculated from the error dynamics (e_c) presented by the backstepping controller in [20], given as follow

$$e_c = \begin{bmatrix} e_x \\ e_y \\ e_\phi \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_r - x_d \\ y_r - y_d \\ \phi_r - \phi_d \end{bmatrix}, \quad (13)$$

where K_x, K_y, and K_ϕ are the arbitrary constants, e_x, e_y, and e_ϕ are error in x, y and ϕ, $q_r = \begin{bmatrix} x_r & y_r & \phi_r \end{bmatrix}^T$ is the reference position of the robot, and $q_d = \begin{bmatrix} x_d & y_d & \phi_d \end{bmatrix}^T$ is the desired position of the robot.

The resulted Jacobian matrix is needed to know the exact of all system parameters. Online calculation of the Jacobian matrix will be done by neural network that continuously train robot's models.

In this study, the actuator modeling is neglected, the torques parameters are achieved by dynamics equation in eq. 10 and motors' frictions are considered very small and negligible.

III. Neural Network Controller Design

Neural networks are a system that involves a numbers of a special type processors named neurons. The neurons imitate the work of human brain by training the internal parameter called weight(s). The weights are adjusted according to the assigned task. Neurons are represented by a state, from the list of inputs, and govern the dynamic operation of the system. The three main ideas of neural network are utilizing sensors information, collecting processing capability, and learning and adapting. The learning and control are done simultaneously through the neurocontrollers and the processes are conducted continuously as the feedback to the system. In this study, we choose the application the multilayer perceptron (MLP) as the most suitable type. The block diagram of the proposed neural network controller is shown in Fig. 2

The applied topology of neural networks in this study is shown in Fig. 3. It has a feedforward structure, one input layers, one hidden layers with N neurons, and one output layers. The output of the neural network can be written in

$$y_i = \sum_{j=1}^{l} \left[v_{ij} \sigma \left(\sum_{k=0}^{n} w_{jk} x_k \right) \right], \quad i = 1, 2, ..., m \quad (14)$$

where $x_k \, (k = 0, 1, 2, ..., n)$ is the control input (v_c, ω), l is the amount of neurons inside the hidden layers, m is the output neurons, y_i is the output $\left(\hat{q} = \begin{bmatrix} \hat{x} & \hat{y} & \hat{\phi} \end{bmatrix}^T\right)$, w_{jk} are weight functioning as the input to hidden layers, v_{ij} are the weight of hidden layers output to interconnection weights, and σ is the nonlinear hyperbolic tangent activation function, given by

$$y = \sigma(z) = \frac{1 - \exp^{-z}}{1 + \exp^{-z}} \quad (15)$$

For the simplification, eq. 14 can be written in compact form,

$$y = V^T \sigma \left(W^T x \right). \quad (16)$$

as shown in Fig. 3, where $N_1, N_2, N_3, ..., N_n$ is the vector function; related to neurons, and W and V are weight of neurons.

The parameters in eq. 16 is derived by using Jacobian in eq. 11, and the resulted output will be trained and updated online as the neural network is learning the robot's navigation or as the feedback shown in Fig. 2. The combination of derived Jacobian from neural networks neurons performs more precise approximation of the system parameters.

IV. Result and Discussion

The best modeling and controller design using kinematics and dynamics modeling of robot parameters and multilayer perceptron neural network for navigating the applied robot in Fig. 1 have been theoretically derived. In this section, the comparison of neural networks research for mobile robots that basically alike with robot shown in Fig. 1 is discussed, and the result presented in this section is given by [19].

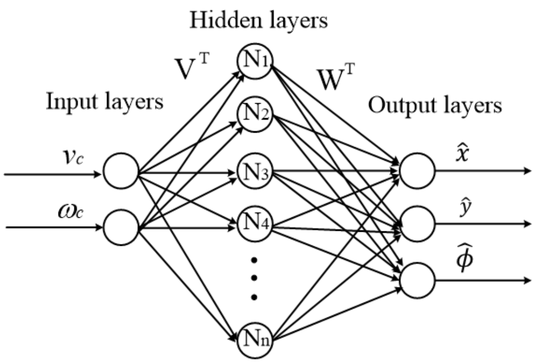

Fig. 3. Neural network layers presentation

A. *Without considering Kinematics or Dynamics of the robot*

The application of neural networks in mobile robots is possible without getting into details of modeling the kinematics and dynamics of robot system. These studies utilized the data obtained from installed sensors as the input to the neural network controllers. The weights can be trained online or offline using software and mostly using backpropagation learning and tangent sigmoid activation function. Below some works used as the references for this study.

Farooq et al. designed two neural network controller, hurdle avoidance and goal reaching to a car-like mobile navigation applied in an unknown environment such as a corridor environment. They utilized the data of the installed sensors as the input to the controller (hidden layers), the distance sensors data was used for hurdle avoidance and distances to the goal data was the input for goal reaching controller design. The weights were trained offline using MATLAB with backpropagation learning using the tangent sigmoid activation function [1].

Sabto et al. used RFID sensor as the main data input to the designed controller. The RFID tags functioning as the landmark were randomly distributed and the experimental robot were able to autonomously navigate itself in the unknown environment [4].

Karakuş et al. presented probabilistic neural network (PNN) as the learning algorithm for a wall-following robot equipped with 20 distance sensors [5].

Pessin et al. used an artificial neural network to estimate the position of robot node applied in indoor environment using a set of wireless network. ANN trained to get the exact location and to track any displacement of robot nodes. They used error measurement calculated by the distance formula $d = \sqrt{(x_2 - x_1^2) + (y_2 - y_1^2)}$, where x and y are the distances in cm [6].

Capi et al. compared the performance of three neural network architectures applied in urban dynamic environment. The sensory data were achieved from GPS and compass [8].

Folgheraiter et al. developed a motor path generator to a quadruped robot based on a chaotic neural network by utilizing the installed sensory system [12].

Janglová is among one of the pioneers in designing neural network for mobile robot navigation without considering kinematics or dynamics. Janglová proposed a path planning for the robot to move autonomously in partially unknown environment by utilizing the installed ultrasound range finder data [16].

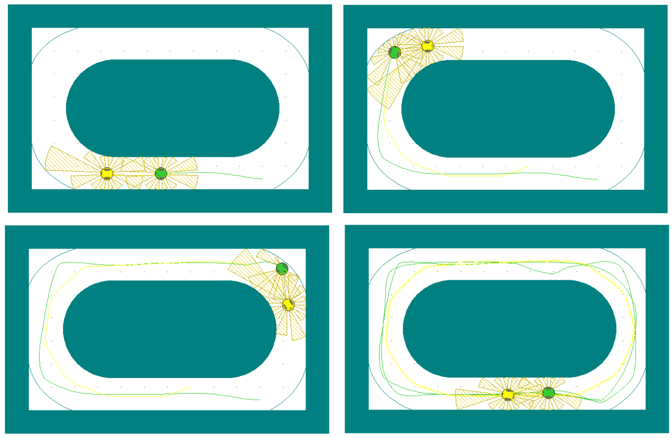

Fig. 4. Simulation result from Neta et al.'s study [19]

Neta et al. proposed a simple neural network designed by taking data from only 4 installed distance sensors to an object following mobile robot combined with a wall following algorithm. They proved the feasibility in a simulation program shown in Fig. 4 [19].

B. *Considering kinematics of the robot*

Some studies only considered kinemetics of the robot, by deriving the kinematics modeling, positions and velocities can be known. In kinematics only case, the Jacobian matrix needed for the online training is achieved by taking the derivation of velocities. For example, Velagic et al. designed a recurrent neural network with one hidden layer to train and learn the relationship between robot's linear velocities and error position. The neural network is trained online using the backpropagation algorithm with adaptive learning rate [27].

C. *Considering kinematics and dynamics of the robot*

R. Fierro and F. L. Lewis are the pioneers in developing neural network controller for mobile robot, since their publication in 1998, the works have been used as the reference for the further neural network controller design and implementation. They proposed the backstepping algorithm where the stability was proven with Lyapunov analysis [20]. The complete derivation of kinematics and dynamics modeling of the robot will give the exact parameters needed to be the input of neural network controller [11] [13] [17] [20]- [24].

D. *Other consideration*

In some studies, the actuators applied to the robot were also modeled to get the exact torques needed for the systems and how much friction resulted from actuators (motors) rotation [23].

The application of mobile robot in unknown dynamics environment might result in wheels skidding and slipping, these conditions are considered by Yoo. Yoo proposed neural network based adaptive control system using systematic and recursive design techniques used to compensate the occurrence of skidding and slipping [28].

In some cases, researchers combined neural network controller with other type of artificial intelligence, such as fuzzy logic, combining the training and learning process of neurons with rules and membership function of fuzzy logic algorithm. Rules and membership function became the input to neural network layers instead of giving numerical data obtained directly from sensors [25] [26].

V. Conclusion

This paper is a review paper that investigates the application of neural network controller for a mobile robot navigation system by comparing the methods used by previous researchers. The study was started by considering the derivation of kinematics and dynamics model of two wheels differential driven mobile robot. The derived mathematical model was combined with the generic equation of neural network output, and Jacobian matrix was achieved. The Jacobian matrix would be trained online to get the exact parameters for the system. The neural network controller was designed by taking the input from robot's mathematical model and the output would be the estimated position of the robot. In some previous studies, neural network controller designed by taking direct input from sensory information data without considering the kinematics and dynamics of the robot, however including complete model resulted in better system performance for it means considering complete design of a robot controller.

References

[1] U. Farooq, M. Amar, M. U. Asad, A. Hanif, and S. O. Saleh, "Design and Implementation of Neural Network Based Controller for Mobile Robot Navigation in Unknown Environment", *International Journal of Computer and Electrical Engineering*, vol. 6, no. 2, pp. 83-89, 2014.

[2] P. K. Mohanty and D. R. Parhi, "A New Intelligent Motion Planning for Mobile Robot Navigation using Multiple Adaptive Neuro-Fuzzy Interference System", *International Journal Applied Mathematics & Information Sciences*, vol. 8, no. 5, pp. 2527-2535, 2014.

[3] I. Engedy and G. Horváth, "Artificial Neural Network Based Mobile Robot Navigation," in *2009 IEEE International Symposium on Intelligent Signal Processing*, Budapest, 2009, pp. 241-246. doi: 10.1109/WISP.2009.5286557

[4] N. A. Sabto, and K. Al Mutib, "Autonomous Mobile Robot Localization based on RSSI Measurements using an RFID Sensor and Neural Network BPANN", *Journal of King Saud University - Computer and Information Sciences*, vol 25, pp. 137-143, 2013.

[5] M. O. Karakuş, and O. Er, "Learning of Robot Navigation Task by Probabilistic Neural Network", in *Second International Conference on Advanced Information Technologies and Applications*, 2013, 24-24.

[6] G. Pessin, F. S. Osório, J. Ueyama, D. F. Wolf, and T. Braun, "Mobile Robot Indoor Localization using Artificial Neural Network and Wireless Network", 2011.

[7] S. X. Yang, and M. Meng, "Neural Network Approaches to Dynamics Collision-Free Trajectory Generation", *IEEE Transaction on Systems, Man, and Cybernetics - Part B*, vol. 31, no. 3, pp. 302-318, 2001.

[8] G. Capi, S. Kaneko, and B. Hua, "Neural Network based Guide Robot Navigation: An Evolutionary Approach", in *2015 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS2015)*, pp. 74-79, 2015.

[9] K. H. Chi and M. F. R. Lee, "Obstacle avoidance in mobile robot using Neural Network," in *2011 International Conference on Consumer Electronics, Communications and Networks (CECNet)*, pp. 5082-5085, 2011.

[10] M. Seyr, S. Jakubek, and G. Novak, "Neural Network Predictive Trajectory Tracking of an Autonomous Two-wheeled Mobile Robot", in *16 Triennial World Congress, Prague, Elsevier IFAC Publication*, pp. 385-390, 2005.

[11] M. Boukens, and A. Boukabou, "Design of an Intelligent Optimal Neural Network-Based Tracking Controller for Non-holonomic Mobile Robot System", *Neurocomputing*, vol. 226, pp. 46-57, 2017.

[12] M. Folgheraiter, and G. C. Gini,"A Chaotic Neural Network as Motor Path Generator for Mobile Robotics". *ROBIO*, pp.64-69, 2014.

[13] D. Gu, and H. Hu, "Neural Predictive Control for Car-Like Mobile Robot", *International Journal of Robotics and Autonomous Systems*, vol 39, no. 2-3, pp. 1-15, 2002.

[14] G. Yanfeng, Z. Hua, and Y. Yanhui, "Back-Stepping and Neural Network Control of a Mobile Robot for Curved Weld Seam Tracking", *Procedia Engineering*, vol 15, pp. 38-44, 2011.

[15] C. C. Lai, and K. L. Su, "Development of an Intelligent Mobile Robot Localization System using Kinect RGB-D Mapping and Neural Network", *Computer and Electrical Engineering*, 2016. http://dx.doi.org/10.1016/j.compeleceng.2016.04.018.

[16] D. Janglová, "Neural Network in Mobile Robot Motion", *International Journal of Advanced Robotics Systems*, vol. 1, No. 1, pp. 15-22, 2004.

[17] G. Bayar, E. I. Konukseven, A. B. Koku, "Control of a Differentially Driven Mobile Robot Using Radial Basis Function Based Neural Network", *WESAS Transactions on Systems and Control*, vol. 13, pp. 1002-1013, 2008.

[18] F. Shamsfakhr, and B. S. Bigham, "A Neural Network Approach to Navigation of a Mobile Robot and Obstacle Avoidance in Dynamic and Unknown Environments", *Turkish Journal of Electrical Engineering & Computer Science*, Available online: 05.06.2016, http://journals.tubitak.gov.tr/elektrik/accepted.htm

[19] N. Larasati, T. Dewi, and Y. Oktarina, "Object Following Design for a Mobile Robot Using Neural Network", *Computer Engineering and Application Journal*, vol 6, no. 1, 2017.

[20] R. Fierro, and F. L. Lewis, "Control of a Nonholonomic Mobile Robot Using Neural Network", *IEEE Transactions on Neural Networks*, vol. 9, no. 4, pp. 589-600, 1998.

[21] T. Dierks, and S. Jagannathan, "Neural Network Control of Mobile Robot Formations Using RISE Feedback," *IEEE Transactions on Systems, Man and Cybernetics: Part B, Institute of Electrical and Electronics Engineers (IEEE)*, vol. 39, no. 2, pp. 332-347, 2009.

[22] A. Burghardt, and Z. Hendzel, "Neural Control of a Mobile Robot Amigobot", *Mechanics and Mechanical Engineering, Technical University of Lodz*, vol. 15, no. 4, pp. 21- 29, 2011.

[23] M. Mohareri, R. Dhaouadi, and A. B. Rad, "Indirect Adaptive Tracking Control of a nonholonomic Mobile Robot Via Neural Network," *Neurocomputing*, vol. 88, pp. 54-66, 2012.

[24] A. Singh, G. Bisht, and P. K. Padhy, "Neural Network based Adaptive Non-linear PID Controller for Non-Holonomic Mobile Robot", in *2013 International Conference on Control, Automation, Robotics and Embedded Systems (CARE)*,2013, pp. 1-6. doi: 10.1109/CARE.2013.6733732

[25] K. C. Ng, and M. M. Trivedi,"A Neural-Fuzzy Controller for Real-time Mobile Robot Navigation", in *Proc. SPIE 2761, Applications of Fuzzy Logic Technology III*, 12pages, 1996. doi: 10.1117/12.243253.

[26] M. Harb, R. Abielmona, and E. Petriu, "Speed Control of a Mobile Robot Using Neural Network and Fuzzy Logic", in *Proceedings of International Joint Conference on Neural Network*, pp. 1115-1121, 2009.

[27] J. Velagic, N. Osmic, and B. Lacevic, "Neural Network Controller for Mobile Robot Motion Control", *International Journal of Electrical, Computer, Energetic, Electronic and Communication* , vol. 2, no. 11, 2008.

[28] S. J. Yoo, "Adaptive Neural Tracking and Obstacle Avoidance of Uncertain Mobile Robot with Unknown Skidding and Slipping", *Information Science*, vol. 238, pp. 176-189, 2013.

[29] S. G. Tzafestas, "Introduction to Mobile Robot Control", *Elsevier*, 2014.