Impact of Internet on Business Activities in Serbia and Worldwide

Uticaj Interneta na poslovanje u Srbiji i svetu

DOI: 10.15308/SINTEZA-2014-931-936

IMPLEMENTING DECISION-MAKING METHODS BASED ON MULTIPLE NEURAL NETWORKS

Duško Radaković[1], Dragan Cvetković[2]
[1]College of Professional Studies – Belgrade Polytechnic, Serbia
[2]Singidunum University, Serbia

Abstract:
This paper presents an approach to implementing a learning and decision-making method for mobile robots in manufacturing environments based on multiple neural networks. Usually, these networks are used for strongly separated architectures, but in our research, we used them to differentiate similarities in the environment, caused by low-resolution sonars, error scanning or any other noise that could omit a qualitative reading. These networks work independently in their own domain, while a decision network brings the result. We have tested different configurations of neural networks, but best results were obtained with multiple neural networks. As an experiment to support our research, we have considered a technology environment that could be improved by use of mobile robots. Given the high costs of building a real full-scale mobile robot, we have decided to downsize the problem and evaluate this possibility with LEGO Mindstorms NXT and neural networks using Matlab. The motion of the robot was determined by the choice the robot had to make. It was modeled and implemented by a software solution, set with an accuracy that enables mapping, obstacle recognition and avoidance. A higher quality evaluation would involve hardware consideration (quality of drive motors and actuators of the mobile robot, accuracy of signal processing from the ultrasonic sensor, motion tolerance of the sensor head, etc.).

Key words:
mobile robot,
artificial intelligence,
artificial neural networks,
inner transport,
decision-making methods.

INTRODUCTION

The purpose of autonomous mobile robots is to perform tasks for a wide variety of users. In order to perform tasks correctly and efficiently, the mobile robot must have the ability to interact with the environment. Navigating a mobile robot in a known environment requires almost no interaction. However, the environment usually changes in time; it is dynamic and unpredictable. For a robot to operate effectively it is crucial to acquire and handle information regarding existence and location of objects and areas along a pathway and in the haptic field of the robot. This involves extracting information from the real world while taking into account noise and inaccuracy of sensors. To develop navigation systems for autonomous mobile robots with learning capabilities requires sophisticated algorithms [1, 3, 4].

This paper presents a possible solution to the problem of enabling a robot movement in a known and partially unknown environment by implementing learning and decision-making methods with multiple neural networks. This environment learning procedure easily integrates with most general navigation learning algorithms and different types of range determining sonars.

The experiment was carried out with *LEGO Mindstorms NXT* and programming in *MathWorks Matlab*. Due to excessive costs of building a full-scale model, the production environment was downscaled to demonstrate the implementation of these methods. The example environment was a production facility in Montprojekt Co., Serbia. The goal of this research was to prove motion and recognition accuracy by use of multiple neural networks with low cost equipment. In addition, engineering students gained better insight and understanding of the process of implementing decision-making methods and found an acceptable solution.

PROBLEM SETUP

As an example environment, the layout of Monprojekt Co., Serbia, was used to for demonstration and use of an autonomous mobile robot as an alternative to traditional transport means in a production facility. The point was to show how this robot could aid workplace transportation requirements for raw material, finished parts, etc. Due to financial costs, the layout was downscaled and a mobile robot made from *LEGO Mindstorms NXT* was used to complete the research. The robot had to solve two major issues:

- recognize and classify the environment, and
- recognize and avoid obstacles on its pathway.

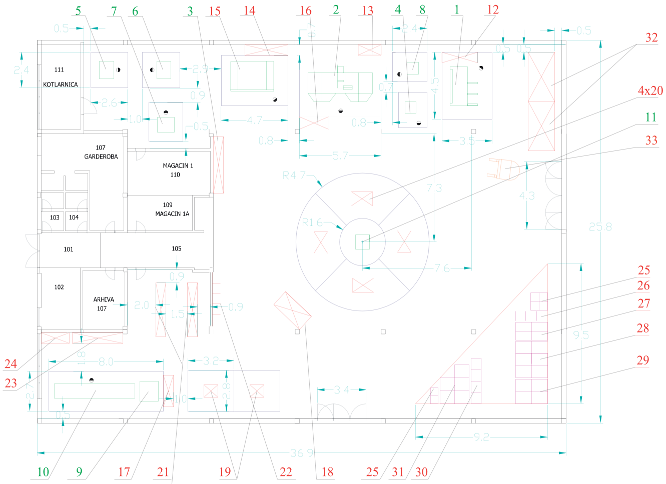

Fig. 1. *Layout of the example production facility*

The layout of the example environment of the production facility is shown in Fig. 1. Accordingly, the main machines for sheet metal processing are in linear arrangement, (machine tools: 1, 2, 3, and 4), and the allocated machines 5, 6 and 7, which are less used are displaced due to other technological processes. In addition, the machine center for copper processing (9) is fully displaced, because of the technological process of preparing isolated parts of copper in relation to other technological procedures. The idea is to replace the fork-lifter (33) by an autonomous mobile robot, serving all workplaces and storages. In order to present a solution, a downscaled model of the facility was built and a *LEGO Mindstorms NXT* robot was used as a substitute for the real full-scale robot.

Possible positions in the environment are simplified and presented in Fig. 2.

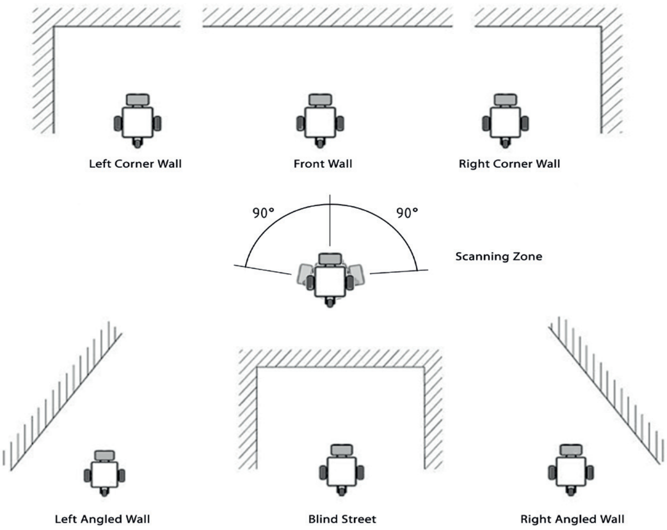

Fig. 2. Simplified environment situations.

On the other hand, obstacle recognition and avoidance was set in a similar manner. Here, an important issue was the sonar. Basically, the process by which the sonar determines the distance is very simple:

$$d = \frac{v \cdot t}{2} \tag{1}$$

where, v – speed of sound through air, t – time of flight of ultrasound and echo [7].

The profile of the amplitude of the sonar shows the strength of the sonar signal as a function of orientation with respect to the central wave. Most of the power is limited to the unit lattice (axis that best sound signal) near the central wave. Emitted waves of the ultrasonic sensor are determined by (2), and are shown in Fig. 3.

$$D(\vartheta) = 2 \cdot \frac{J_1(k\eta \sin\varphi)}{k\eta \sin\varphi} \tag{2}$$

Where, J_1 – first order Bessel's function,

$k = 2p/\lambda$, λ - wavelength,

η - ear radius,

φ - azimuth angle measured relative to center axis of broadcast.

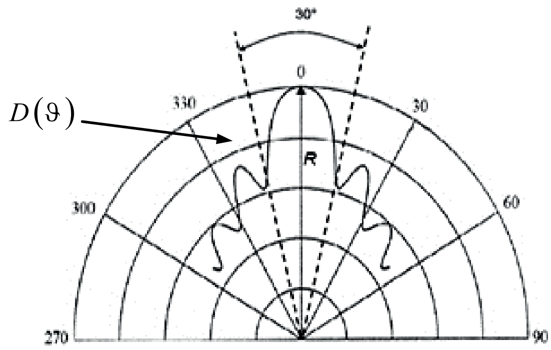

Fig. 3. Wave emission from the ultrasound sensor

Great effort has been made to use the sonar's *time-of-flight* (TOF), but results did not achieve a greater significance. This is a good reason to use sophisticated sonars with integrated frequency modulation and analysis based on the sound wave's amplitude, or to use optical distance measuring which is more accurate. When moving around the environment, the walls and obstacles are not always at a right angle. Unfortunately, the material of the object or the wall can be of low reflectance, and the signal might never return. Fig. 4 illustrates these situations.

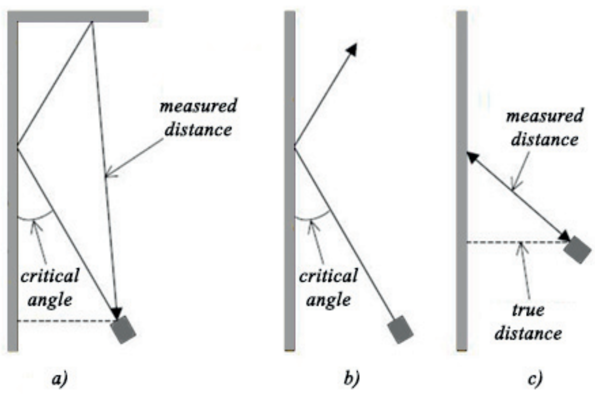

Fig. 4. Cause of errors: a) corner error; b) reflection error; c) triangular error

Two sonars were mounted on the mobile robot for the second task. Defined are only key situations and used for training the artificial neural network (ANN), instead of determining a large number of motor control parameters. All of these situations present a unique obstacle position in the work environment. Obstacles that can disrupt the work path of the mobile robot are classified into eight cases:

1. Left sonar detects an obstacle at a greater distance than the right sonar;
2. Right sonar detects an obstacle at a greater distance than the left sonar;
3. Left sonar does not detect any obstacle, while the right sonar detects an obstacle;
4. Right sonar does not detect any obstacle, while the left sonar detects an obstacle;
5. Both sonars detect an obstacle at equal distance;
6. None of the sonars detect an obstacle;
7. Both sonars detect an obstacle at angles of -30°, -45°, and -60°;
8. Both sonars detect an obstacle at angles of +30°, +45°, and +60°.

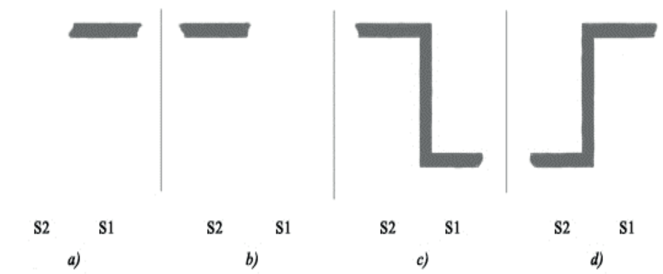

Fig. 5. Cases of detection: a) left sensor detects no obstacle in front; b) right sensor detects no obstacle in front; c) the right sensor detects a closer obstacle; d) left sensor detects a closer obstacle

The first four cases are shown in Fig. 5, while the rest are shown in Fig. 6.

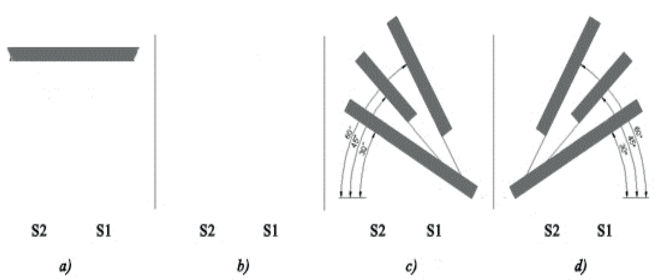

Fig. 6. Cases of detection: a) front wall; b) empty space; c) wall at angles of -30°, -45°, and -60°; d) wall at angles of +30°, +45°, and +60°

In detecting an obstacle on the pathway, the basic idea is to use readings from both sonars (S1, S2) as input for training the ANN. Based on the output from the ANN, conclusion is made in order to evaluate successful obstacle recognition. Afterward, another two neural networks were trained in order to determine parameters *AngleLimit* and *TurnRatio* depending on the output from the first ANN.

METHODOLOGY

Decision theory applies the fundamental idea that the agent (designer, robot, software, etc.) is intelligent and rational, if and only if it selects actions that provide the most expected appropriateness – usefulness. There are plenty of decision-making methods that can be used to find a solution to defined problems. Just to mention few, there are traditional expert systems, induction based rules, techniques of soft-computing like fuzzy systems and genetic algorithms [8, 9]. Another possible solution could be by applying trigonometric functions instead of ANN. This can be applied only for cases of obstacle recognition, because both sonars detect distances (in general, these distances differ) and these distances can be used to compute the angle of the obstacle relative to the sonars.

This paper presents a solution obtained by using *LEGO® Mindstorms® NXT* and MathWorks Matlab®, through a toolbox developed to control robots via a wireless Bluetooth conection [11].

In this research, multiple neural networks determine the solution for both tasks: environment classification and obstacle recognition and avoidance. As stated before, multiple networks give better accuracy in results, especially when networks come into disagreement [10]. In Fig. 7 the diagram presents the information flow and the decision tree of the proposed system of multiple neural networks.

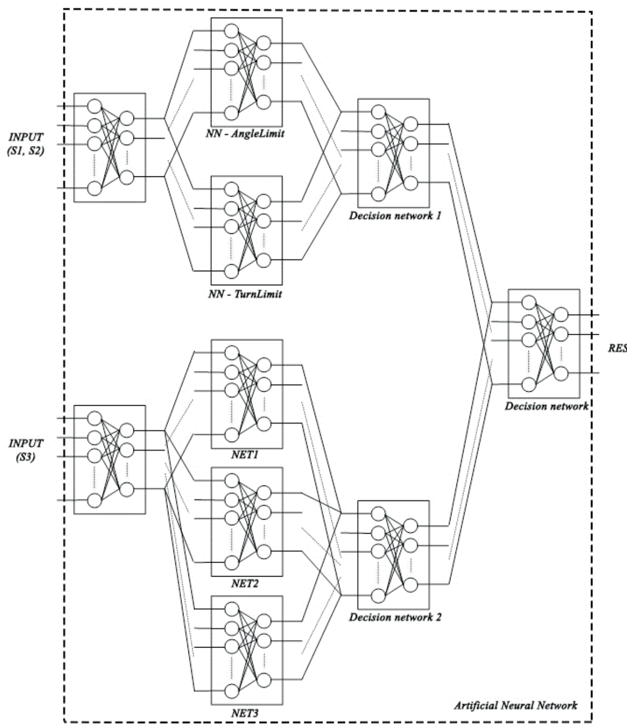

Fig. 7. Information flow through proposed system of multiple networks

Topology of the networks is shown on Fig. 8.

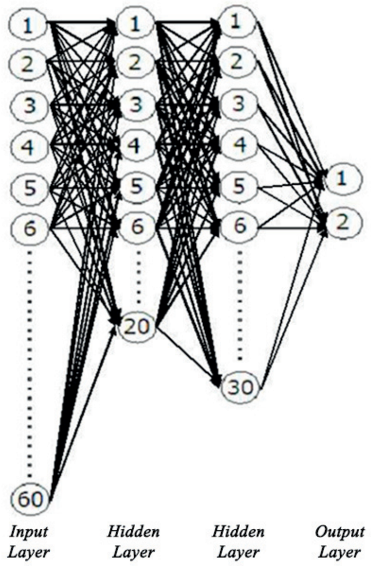

Fig. 8. Network Topology

All networks were created as feed forward backpropagation networks, trained with the Levenberg-Marquardt algorithm. Each of the 10 measurements included 60 measurement points from a scan angle of 180 degrees. First seven measurements were taken at a distance of 10 mm and three measurements at a distance of 20 mm from the wall/obstacle. The maximum fail parameter is set to 20 iterations in the first hidden layer, during which the gradient tends to decrease and when the decreasing stops, the training has ended.

EXPERIMENTAL RESULTS

Through experiments, the authors conclude that the concept solution provided satisfactory results. During the first task, the aim was to get the robot to recognize the angle of approach and successfully bypass the obstacle. The error generated during training was below 10°. All diagrams in the following figures present number of measurements on the horizontal axis, and measured distance on the vertical axis.

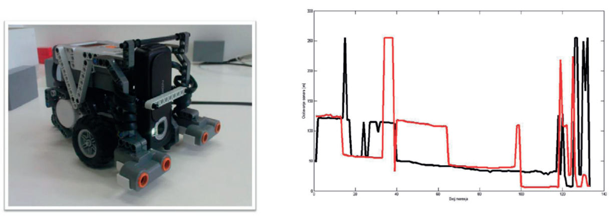

Fig. 9. Robot with parallel sonars and a reading of an obstacle

Measurements with values above 255 cm indicate that the sonar did not receive the echo (Fig. 4b illustrates this case). Such readings cannot train the ANN to give desired output values. Among other things, the problem in this diagram occurs around 120-th measurement. In fact, at that point the robot collided with a wall, and the readings show values between 20 cm and 40 cm (case is illustrated in Fig. 4a). Only when the robot has straighten according to the wall, the sonars read the real value and the robot stopped.

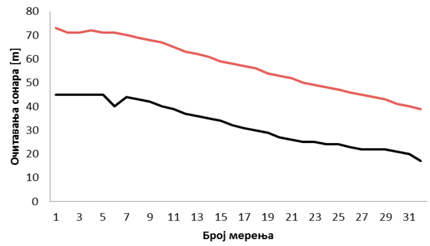

Fig. 10. Simultaneous measurements from both sonars

The curves in Fig. 10 ideally should be parallel. In practice, this is not possible, so this graph is accepted as satisfactory.

In finding a solution to this problem, different approaches were made but not one gave good results. One idea was to simulate the angle of 60° using trigonometry and then expose the ANN to learning. Unfortunately, the results were still poor, because even after the ANN passed teaching, when on-line, the readings of the sonars were so poor that the ANN could not recognize any of the situations. In addition, there was an attempt to position the robot at an unfavorable angle and move it toward the obstacle. When near to the obstacle the robot was manually stopped, and those readings were taken as input for learning. This, also, did not bring successful results.

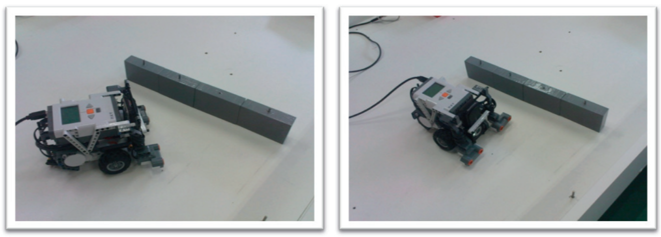

Fig. 11. Successful obstacle recognition and avoidance

In the second task, environment mapping and classification, the robot is positioned in a situation (e.g., in front of a wall) and reading is taken by turning the sonar from left to right by 180° (from -90° to +90°). The downscaled environment is an area 150 cm by 100 cm, with a diagonal of 180 cm. For every situation, measurements were made by positioning the robot 20 cm to 30 cm from the wall (obstacle). Since the whole setup was in range of the sonar, other obstacles and walls came into account. This influenced proper recognition and classification, especially at acute angles when sonar readings were false due to improper reflection (case illustrated in Fig. 4a). Besides the sonar, the motor (on which the sonar was mounted) had instable and inaccurate rotation. Even, by setting the rotation at smaller steps, in some occasions the motor would continue to rotate until collision with the wheel.

All these hardware problems had an impact on the ANN during training and later, while moving in the environment. When training a single ANN, with more than two input vectors, certain confusion omitted proper recognition and classification. This was solved by introducing multiple NNs. The ANN was set up with five additional NNs, of which two were trained for the parameters AngleLimit and TurnRatio (for detecting and avoid obstacle collision), while the other three were trained for a pair of distinct situations (NET1, NET2, NET3, i.e. for environment mapping and classification) (illustrated in Fig. 7). The Decision network 1 corrects attitude and movement, while Decision network 2 outputs a classification of the environment in the current situation. Both outputs from these decision networks are used into a final decision network with results defining location and/or bypassing obstacles.

The aim of the research was fulfilled. The robot can recognize an obstacle and bypass it accordingly. The percentage of successfulness is directly related to the accuracy of input values. Another goal of this research was to determine the degree of successful implementation of ANNs into intelligent mobile robots. A question arises: "Why ANNs?" The answer to this is that unexpected events can happen in complex environments, and the robot has to sequentially solve problems and learn at the same time.

Artificial neural networks have their limitations that are reflected demanding hardware when it comes to problems that are more complex. When training in off-line mode, this is not a big problem. Training on-line demands an immediate system response, so hardware requirements grow significantly.

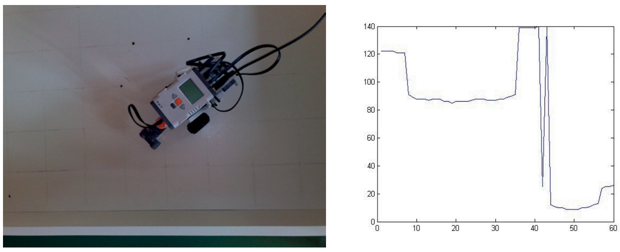

Fig. 12. Left Angled Wall

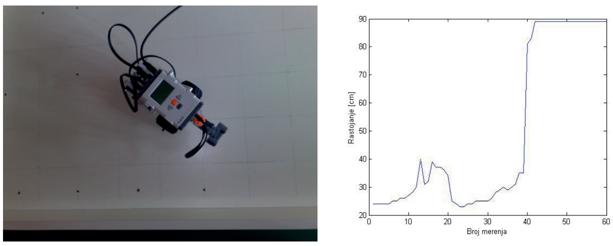

Fig. 13. Right Angled Wall

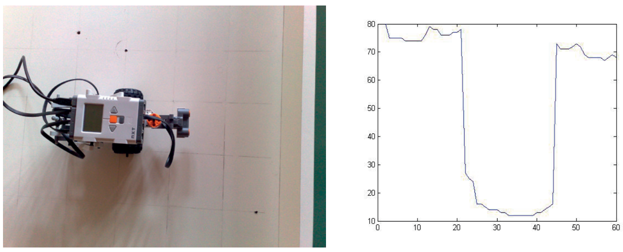

Fig. 14. Front Wall

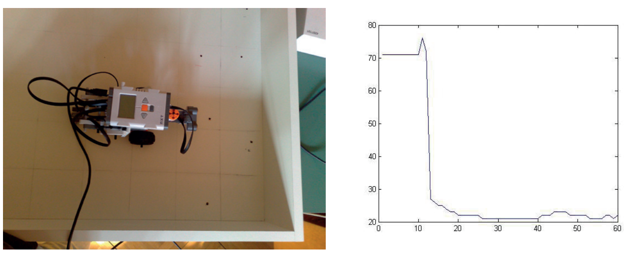

Fig. 15. Right Corner Wall

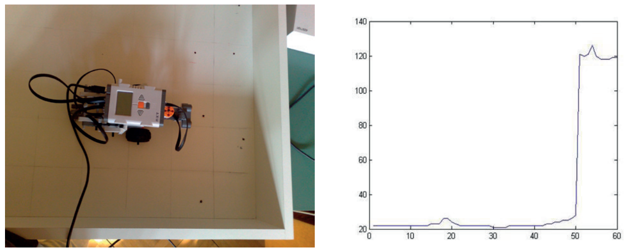

Fig. 16. Left Corner Wall

Despite these problems, measurements were successfully carried out. Ten measurements were collected for possible situations. An attempt was made to create a neural network in which the input was 60x6 matrix where each column represents the input vector and is formed based on the reading of every situation. The output would be the square matrix 6x6, in which each column represents one situation in the environment. Testing found that such a network is not good even when varying number of hidden layers and number of neurons in the hidden layer. Finally, the original ANN was modified to include multiple networks, as opposed to a modular network with no correlation between networks within it. The final ANN, during testing with two hidden layers with twenty and thirty neurons in layer, obtained acceptable results.

Acceptable results were obtained from greater distances, as well. For network training, only best samples were used, as opposed to other measurements that had some deviation. With a wider range of measurements, tolerance was introduced to the mobile robot for differentiating poor measurements having overlapping characteristics.

CONCLUSIONS

Problems and disadvantages of sonars are known in many researches and studies dealing with sonars. One of the major reasons for using sonars is their low price. Usually, they prove their use when greater accuracy is not an issue and when there are no measurements under acute angles. For this research, better results would have been obtained by infrared sensors or lasers, which are much more expensive but by far more accurate.

A solution without ANN in which the robot serving the workplaces would follow predefined markings in the production environment could replace the fork-lifter, as well. This would enable the robot to move from one workplace to another, but the problem arises when an object gets in the pathway. In this case, the robot can neither recognize nor avoid the obstacle. It does not have a mapping of the environment so it's not able to bypass unexpected objects. This is where ANNs come forth. Sensor readings will provide data to ANN, which will then know how to avoid collision and know the robot's position in the environment and where it is headed. Complex systems are very difficult to describe analytically and find interrelation within. That is the main reason to use ANNs to establish a relation between input data and desired results.

The objective was that the intelligent mobile robot, which has a pre-defined path and order of serving machines, must be able to avoid any obstacles in the environment and continue to perform its task, always knowing where it is, where it's going and where it came from. The purpose of the neural network was to identify disturbances that occurred in the system and determine the way to overcome them. Application of ANN has broad representation in intelligent mobile robots, mainly due to their ability to find a correlation between input and desired output data. The used method is based on: machine learning, application of artificial neural networks, because of their ability to adapt to changes in the system caused by disorder factors, and on the ability to learn from these situations and, hence, minimize the error after each subsequent iteration. The result is a robot that can recognize whether there is an obstacle in front of it, if it is at an angle, and how to avoid it.

An intelligent, autonomous mobile robot can replace a man-driven fork-lifter in a technological environment. Machine learning, as an application in solving this problem, proved to be a good solution. However, for this solution to give expected results it is necessary to use high quality sensors and actuators. The application of such a system would facilitate the transport in the industrial plant, and hence the productivity.

REFERENCES

[1] H.P. Moravec, "Obstacle Avoidance and Navigation in the Real World by a Seeing Rover", Phd dissertation, Stanford University, September 1980.

[2] R. A. Brooks, "Aspects of Mobile Robot Visual Map Making", Proceedings of 2nd International Symposium of Robotics Research, pp 287–293, August 1984.

[3] A. Zelinsky, "Robot Navigation with Learning", Australian Computer Journal, Vol. 20 No. 2, May 1988.

[4] A. Zelinsky, "Environment Mapping with a Mobile Robot using Sonar", 2nd Australian Joint Artificial Intelligence Conference Adelaide, Australia, November 15-18, pp. 363-378, 1988 Proceedings

[5] Chang-Hyun Kim, Joon-Hong Seok, Ju-Jang Lee, Sugisaka, M. "Adaptive robot control based on multiple incremental fuzzy neural networks", IEEE International Symposium on Industrial Electronics, Seoul, July 5-8, 2009, pp. 466-471

[6] R. Chatila, "Path Planning and Environment Learning", European Conference on Artificial Intelligence, pp. 211–215, July 1982.

[7] L. Svilainis, V. Dumbrava, "The time-of-flight estimation accuracy versus digitization parameters", ULTRAGARSAS Journal, Vol. 63, No. 1, 2008

[8] A. Elfes, "Sonar-Based Real World Mapping and Navigation", IEEE Journal of Robotics and Automation, pp. 249–265, June 1987.

[9] Chin-Teng Lin, C.S. George Lee, "Neural-Network-Based Fuzzy Logic Control and Decision Sytem", IEEE Transactions on Computers, Vol. 40, No. 12, pp 1320-1336, December 1991

[10] Sung-Bae Cho, Jin H. Kim, "Combining Multiple Neural Networks by Fuzzy Integral for Robust Classification", IEEE Transactions on Systems, Man, and Cybernetics, Vol. 25, No. 2, pp 380-384, February 1995

[11] Michael P. Perrone, Leon N. Cooper, "When Networks Disagree: Ensemble Methods for Hybrid Neural Networks", Technical Report AD-A260045, Brown University, Rhode Island, December 1992

[12] RWTH - Mindstorms NXT Toolbox, http://www.mindstorms.rwth-aachen.de/trac/wiki/Download4.07 (visited April, 2012)