http://doi.org/10.24867/JPE-2019-02-025

Journal of Production Engineering
Vol.22 No.2

JPE (2019) Vol.22 (2) Original Scientific Paper

Crnokić, B., Grubišić, M.

FUSION OF INFRARED SENSORS AND CAMERA FOR MOBILE ROBOT NAVIGATION SYSTEM - SIMULATION SCENARIO

Received: 15 December 2018 / Accepted: 26 March 2019

Abstract: *This paper presents a simulation scenario of a mobile robot navigation system. Navigation system is based on data collecting from three infrared sensors and camera, on which basis the fusion process of detection and avoidance of obstacles is realized. Information from camera was used to detect edges of the obstacles in the environment, while infrared sensors were used to measure the distance from the obstacles. Multilayer perceptron network, trained with backpropagation algorithm, was used for classification of detected obstacles. Experiment was realized through simulation in the simulation environment "Robotino SIM". The control algorithm was implemented in MATLAB. Sensor fusion has proven to be a much better solution than using infrared sensors or cameras separately. This experiment showed that the developed algorithm gives very good results (accuracy: 89.61%), and the navigation system itself performs required tasks of detecting and avoiding most of the obstacles on which it was tested.*

Key words: *Mobile robot navigation, Sensor fusion, Infrared sensors, Camera, Artificial neural networks.*

Fuzija infracrvenih sensora i kamera za mobilni robot navigacijskog sistema - simulacioni scenario. *Ovaj rad predstavlja scenario simulacije navigacijskog sistema za mobilne robote. Navigacioni sistem zasnovan je na prikupljanju podataka sa tri infracrvena senzora i kamere na osnovu kojih je ostvaren proces fuzije detekcije i izbegavanja prepreka. Informacije sa kamere korišćene su za otkrivanje ivica prepreka u okolini, dok su infracrveni senzori korišćeni za merenje udaljenosti od prepreka. Za klasifikaciju otkrivenih prepreka korišćena je višeslojna perceptrona mreža, obučena algoritmom "backpropagation". Eksperiment je realizovan simulacijom u simulacionom okruženju „Robotino SIM". Algoritam upravljanja implementiran je u MATLABu. Pokazalo se da je fuzija senzora mnogo bolje rešenje od zasebne upotrebe infracrvenih senzora ili kamera. Ovaj eksperiment je pokazao da razvijeni algoritam daje veoma dobre rezultate (tačnost: 89.61%), a sam navigacioni sistem obavlja tražene zadatke otkrivanja i izbegavanja većine prepreka na kojima je testiran.*

Ključne reči: *Navigacija mobilnih robota, fuzija senzora, infracrveni senzori, kamera, veštačke neuronske mreže.*

1. INTRODUCTION

Navigation is one of the main problems in the design and development of intelligent mobile robots. Different navigation systems, based on one or more types of sensors, have been developed. Common problems of all these systems are robot self-location, route planning and environmental map construction. To solve this problems, accurate and reliable information are needed to determine the status of the environment in which mobile robot is located. [1] Navigation system of a mobile robot which is based on environmental data obtained from only one type of sensor is a major problem due to the disadvantages of each type of sensor individually. Solution of this problem is in combining multiple different types of sensors, or fusion of data (measurements) from multiple sensor.

There are many methods for fusion of data obtained from sensors used in a mobile robot, such as: Bayes' theory, Dempster-Shafer's (DS) evidence theory, weight factor model method, Kalman filter method, fuzzy fusion, artificial neural networks fusion method, etc.[2] Conventional numerical methods require quite a lot of computing time and large memory capacity, which creates difficulties and limitations in time-limited applications and functions. Artificial neural networks, including Support vector machines (SVM), Principal component analysis (PCA), etc., and their applications, in recent decades have an important place in many fields such as information processing and control systems.[3] In the paper [4] a multilayer model of neural network was introduced for avoidance of obstacles through increased learning. The MONODA system was implemented on the mobile robot NOMAD, where fusion of infrared and ultrasonic sensors was used to detect and avoid obstacles.[5] In paper [6] it is shown that the mobile robot NOMAD can categorize (note and classify) information from the environment obtained from different sensors (camera, infrared sensors, microphone, contact sensors, ...), and link those categories to the values of the loaded signal patterns, and then take the appropriate action. Fusion of infrared sensors and sonar, based on preconfigured neural networks, is presented in the paper [7]. The use of computer vision, image processing and distance measurement using infrared sensors was used in [8] for detecting different shapes and colours. In the mobile navigation system NEURO-NAV feedforward neural network was also used to determine the approximate angular values between the robot motion direction and the corridor orientation to keep the robot on the path in the middle of the corridor. [9], [10] In the paper [11],

Note: The manuscript was presented at the 13. International Scientific Conference MMA held in Novi Sad, Serbia, 28.-29. September, 2018.

neural units with higher-order synaptic operations have been used for image processing applications, i.e. for edge detection and for processing of edge detection data with Hough's transformation. Kohonen type of neural networks have been used to identify and provide coordinates of landmarks with use of laser sensing measurements. [12]

This paper presents a simulation scenario of a mobile robot navigation system based on artificial neural networks.

2. HARDWARE AND SOFTWARE USED

Hardware used in this paper includes mobile robot Robotino 2 (Figure 1.a)) and mobile PC (laptop), while the navigation system algorithm is implemented in programming environment MATLAB/Simulink and through software package RobotinoView2. Robotino SIM simulating environment was used for the simulation of navigation system based on artificial neural networks, implemented on Robotino in the environment cluttered with obstacles.

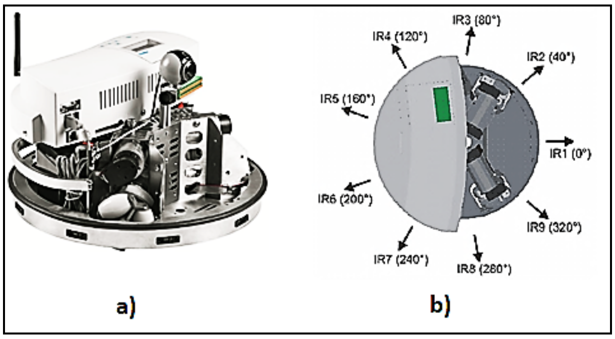


Fig. 1. a) Robotino®2, b) Arrangement of 9 IR sensors on Robotino [14]

Mobile mechatronic system Robotino 2 is manufactured by the German company "Festo Didactic". Robotino possesses various types of sensors, actuators and software interfaces which are at the highest level in the field of mobile robotics [13].

Robotino moves with three independent, omnidirectional drive units, mounted at an angle of 120° relative to another drive. This drive system enables movement in all directions: forward, backward, sideways and rotation. The system can work independently as well as linked to the external computer via Wi-Fi connection. Robotino has various types of sensors. Some of these sensors come as standard equipment, such as: 9 infrared sensors (IR sensors - Figure 1.b)), VGA camera, incremental encoder and anti-collision sensor. Infrared sensors have the ability to detect objects at a distance from 3 to 40 cm, and 9 infrared sensors (IR1 - IR9) on robot are arranged on a chassis at an angle of 40º. Robotino is equipped with a camera Logitech C250 whose height and inclination can be regulated. Resolution can be set in Robotino View by selecting Camera (Block Camera) in the Program dialog box.

MATLAB (MATrix LABoratory) is a multifunctional numerical computational environment that allows manipulation of matrices, drawing of the functions, creating user interfaces and interfaces of programs written in other programming languages. [15] RobotinoView2 is an intuitive graphical programming environment designed specifically for the creation and implementation of control algorithms for mobile robot Robotino [16]. Robotino SIM is a Windows program for 3D simulation of Robotino in predefined virtual experimental environment. This program allows control of Robotino with the use of RobotinoView2 or MATLAB programming interface [16].

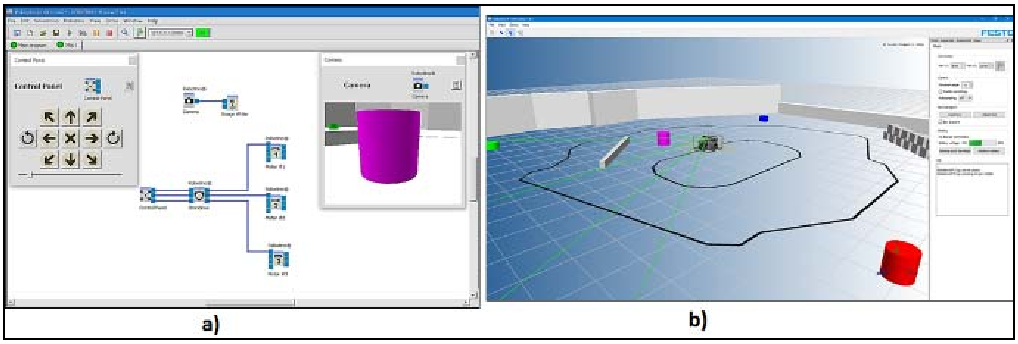


Fig. 2. a) RobotinoView2 programming interface, b) Robotino SIM simulation environment

3. STRUCUTRE OF THE PROPOSED NAVIGATION SYSTEM

Infrared sensors (IR1 - IR9) and cameras installed on Robotino are used to provide information about robot's environment.

In order to simplify the system and to reduce the computational complexity of the control algorithm, in the experiment only three front IR sensors are used, labelled as IR1, IR2 and IR9 in Figure 1.b). Using only these three sensor proved to be sufficient to detect obstacles in front Robotino, or left and right of the Robotino.

Obstacles detection is separated in two parts, for readings from the cameras and from IR sensors. The camera serves to detect landmarks and for localization, i.e. to detect edges, overhangs, and free space for unobstructed locomotion of the robot. IR sensors are used to measure distance from obstacles and to detect additional obstacles in free space in real time. For image pre-processing, i.e. for detection of edges and extraction of features Canny method is used combined with LPQ descriptor. Artificial neural network performs the task of learning the system to detect obstacles in the environment using the information obtained from the camera. After training the network, the system is able to decide whether the robot should continue to move forward, if no trace of obstacles is detected, or to avoid an obstacle if any of them is detected. Additional information on detected obstacles are coming from 3 IR sensors. These data sets merge with the previous information gathered from the camera to give the final confirmation to the robot which direction of motion should be selected. Computer (laptop) is used to process all information (sensor information, pre-image processing, and all processes in neural networks), which is connected over Wi-Fi network with control unit of Robotino.

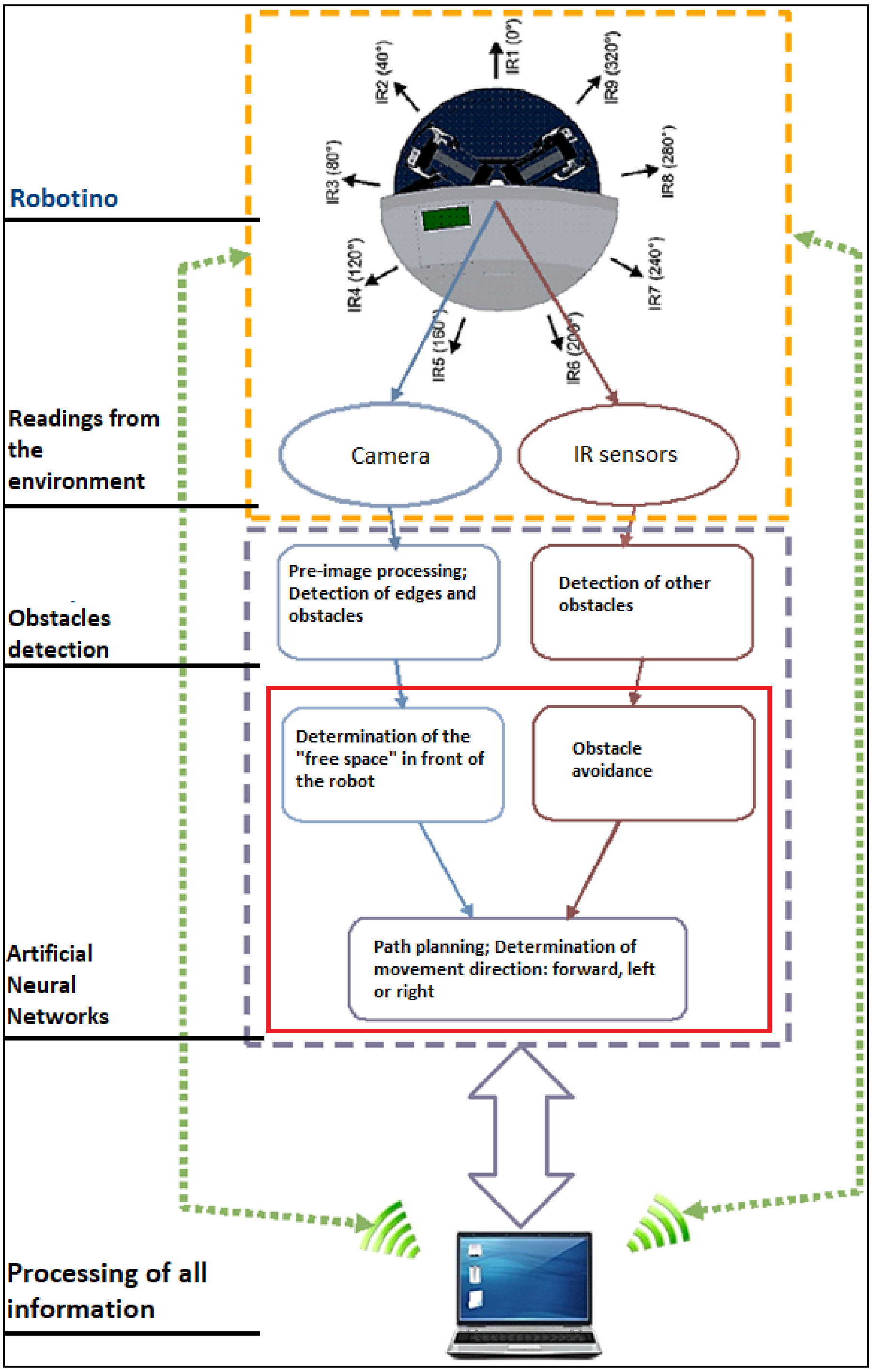


Fig. 3. Structure of the proposed mobile robot navigation system [16]

4. PROPOSED APPROACH AND RESEARCH METHODS

4.1. Collecting of information from the sensors

Robotino View program was used to collect images for training of the neural network and also to set image resolution. It is possible to lead Robotino to any position in the environment using the block "Control Panel". This block is connected to the omni-directional module "Omnidrive" which controls the three motors (Motor #1, Motor #2, Motor #3), and thus allows movement of the robot in any direction. The camera module is activated via the "Camera" block and the image is saved using the "Image Writer" block. In this module it is possible to select two image resolution, 320x240 and 640x480, and the resolution of the collected images for training of the neural network in the displayed navigation system is 320x240. Obtaining information from three infrared sensors is done in MATLAB. Where the three infrared sensors are marked in sequence as DistanceSensor0, DistanceSensor1 and DistanceSensor8. A voltage of 0.7 V or a distance of about 17 cm is taken as the reference distance at which the obstacle will be detected.

4.2. Avoidance of obstacles detected by IR sensors

Depending on whether one of the three IR sensors has detected the obstacle, an obstacle avoidance vector is defined, i.e., the velocities v_x and v_y are set in the direction of robot movement along *x axis* or *y axis*. The velocity values are expressed in *mm/s*. Depending on the direction of the obstacle avoidance, the speed values are 0, 100 or -100 mm/s. By combining different speed values, and considering the position of the obstacle, the robot can move in the following directions:

- forward, for $v_x = 100$, $v_y = 0$ (if there is no detected obstacle in front of the robot),
- right, for $v_x = 0$, $v_y = 100$ (if obstacle is detected on the left side of the robot),
- left, for $v_x = 0$, $v_y = -100$ (if obstacle is detected on the right side of the robot).

4.3. Artificial neural network

After data collection, two steps must be taken before using the training data:

- perform image pre-processing (Canny + LPQ),
- the data needs to be divided into different sub-groups.

To train the neural network in this experiment, 36 images were taken, and 9 more images of the same obstacles were taken for testing. Obstacles are divided into three image classes. The first class consists of cylinders (red, blue, yellow and pink). These obstacles should simulate obstacles of a similar shape in a real environment, such as furniture parts or a particular bulkhead. The second class of obstacles is formed by the images of obstacles that represent all types of walls.

This class should simulate obstacles in a real environment such as walls, corridors, or furniture parts of a similar shape, such as cabinets. The third class are lines on the floor. This class should simulate negative obstacles, such as stairs edges.

The artificial neural network used in this experiment is a multilayer perceptron feedforward network for pattern detection with one input, one output and one hidden layer:

- Input layer consists of 256 input units (nodes) representing the image obtained after pre-processing. Since 36 pictures were taken for training, the network input is a 256x36 matrix.
- Hidden layer has 10 nodes.
- Output layer has 3 nodes. Nodes represent outputs that determine one of the three input image classes, and finally determine the direction of movement of the robot (forward, right or left).

5. ANALYSIS AND IMPLEMENTATION OF THE RESULTS

Table 1. shows the classification of obstacle detection by the type of obstacles, by the type of sensor that has detected an obstacle and by accuracy of detecting a certain obstacle.

This section shows simulation results of control algorithm tested in the simulation environment Robotino SIM. Since this is a demo version of this program, it is not possible to make an arbitrary environment. However, although the simulation environment is limited, the results have shown that this simulation environment is very suitable to simulate proposed control algorithm. The table was formed in a way that 20 attempts were made to detect each of the obstacles. Robotino approached obstacles from different sides, at different angles of approach and from different distances. Successful detection of a certain obstacle is marked with "YES", while unsuccessful obstacle detection is marked with "NO". Successful detection is further classified according to the type of sensor (camera - "Cam", infrared sensors - "IR") that has detected an obstacle.

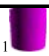
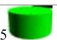
Type of obstacle	Detection			Sensor		Sensor (%)		Accuracy (%)
	YES	NO	Σ	Cam	IR	Cam	IR	
Ob 1 	19	1	20	5	14	26,32	73,68	95,00
Ob 2 	19	1	20	6	13	31,58	68,42	95,00
Ob 3 	19	1	20	5	14	26,32	73,68	95,00
Ob 4 	18	2	20	7	11	38,89	61,11	90,00
Ob 5 	20	0	20	4	16	20,00	80,00	100,00
Ob 6 	20	0	20	10	10	50,00	50,00	100,00
Ob 7 	19	1	20	11	8	57,89	42,11	95,00
Ob 8 	20	0	20	8	12	40,00	60,00	100,00
Ob 9 	0	20	20	0	0	0	0	0
Σ	**154**	**26**	**180**	**56**	**98**	**36,36**	**63,64**	**85,56**

Table 1. Classification of obstacles detection accuracy

6. CONCLUSIONS AND RECOMMENDATIONS

The total number of attempts to detect all obstacles was 180. In 154 attempts obstacles were detected successfully, while 26 attempts were unsuccessfully (obstacles were not detected). Such a number of detected and undetected obstacles resulted in 85.56% accuracy of obstacle detection. Since the system is based on camera and infrared sensor fusion, it is also interesting to see the ratio of detected obstacles for each sensor. Three infrared sensors detected 98 times some of the obstacles (63.64%) while the camera detected 56 times some of the obstacles (36.36%). IC sensors had better efficiency when detecting the cylinders and the camera had better efficiency when detecting the walls. In the Robotino SIM environment it is not possible to manipulate obstacles by using certain commands, however, with the robot movements some of the obstacles could be "knocked down". These "knocked down" obstacles have become a good test for the proposed system since the neural network is not trained for such a "new" obstacles. In the case of detecting these and similar obstacles, the system proved to be the same accurate as in case of detecting obstacles that were represented to the neural network through the learning process.The sensor fusion has proven to be a much better solution than using only infrared sensors or cameras separately. The advantage of the IC sensor is expressed through a faster response, i.e. through a higher detection rate.

Disadvantages of the proposed algorithm are first of all the ability to move the robot only left, right and forward, without rotation, backward motion and combined motions. The most important upgrade to improve the system is the use of a greater number of IC sensors and omnidirectional camera that will have the ability to rotate and thus to cover a larger area around the robot. Future work can be focused on the application of other methods and sensors, which can be incorporated into the existing algorithm. Also, the use of other edge detection methods should be considered in the future work.

7. REFERENCES

[1] B. Crnokić and S. Rezić, "Robots as an Important Factor in Development of Science and New Technologies," in *1th Internacionalna konferencija „NOVE TEHNOLOGIJE" razvoj i primjena „NT-2014"*, 2014, vol. 1, no. 1, pp. 123–133.

[2] Y. Qingmei and S. Jianmin, "A data fusion method applied in an autonomous robot," *2008 27th Chinese Control Conf.*, pp. 361–364, Jul. 2008.

[3] Z.-G. H. Z.-G. Hou, M. T. M. Tan, M. M. Gupta, P. N. Nikiforuk, and N. Homma, "Neural network methods for the localization and navigation of mobile robots," *Can. Conf. Electr. Comput. Eng. 2005.*, no. May, pp. 1057–1060, 2005.

[4] I. E. T. Fujii, Y. Arai, H. Asama, "Multilayered reinforcement learning for complicated collision avoidance problems," *Robot. Autom. 1998. Proceedings. 1998 IEEE Int. Conf.*, pp. 2186– 2191, 1998.

[5] C. Silva and M. Cris, "MONODA : A Neural Modular Architecture for Obstacle Avoidance Without Knowledge of the Environment," *Proc. IEEE-INNS-ENNS Int. Jt. Conf. Neural Networks. IJCNN 2000. Neural Comput. New Challenges Perspect. New Millenn.*, vol. 6, pp. 334–339, 2000.

[6] J. L. Krichmar, J. A. Snook, J. Jay, and H. Drive, "A Neural Approach to Adaptive Behavior and Multi-Sensor Action Selection in a Mobile Device," *Robot. Autom. 2002. Proceedings. ICRA '02. IEEE Int. Conf.*, vol. 4, no. May, pp. 3864–3869, 2002.

[7] H. M. Barberá, A. G. Skarmeta, and M. Z. Izquierdo, "Neural Networks for Sonar and Infrared Sensors Fusion," *Proc. Third Int. Conf. Inf. Fusion*, vol. 2, p. 18.-25., 2000.

[8] J. Eduardo and C. Ortiz, "Visual Servoing for an Omnidirectional Mobile Robot Using the Neural network - Multilayer Perceptron," *Eng. Appl. (WEA), 2012 Work.*, pp. 1–6, 2012.

[9] A. C. Meng, M., Kak, "Fast Vision-Guided Mobile Robot Navigation Using Neural Networks," *Proc. IEEE Int. Conf. Syst. Man Cybern.*, vol. 1, pp. 111–116, 1992.

[10] A. C. Meng, M., Kak, "Mobile robot navigation using neural networks and nonmetrical environment models," *Control Syst. IEEE*, vol. 13, no. 5, pp. 30–39, 1993.

[11] Z.-G. Hou, K.-Y. Song, M. M. Gupta, and M. Tan, "Neural Units with Higher-Order Synaptic Operations for Robotic Image Processing Applications," *Soft Comput.*, vol. 11, no. 3, pp. 221–228, Mar. 2006.

[12] H. Hu and D. Gu, "Landmark-based navigation of industrial mobile robots," *Ind. Robot An Int. J.*, vol. 27, no. 6, pp. 458–467, 2000.

[13] B. Crnokic, M. Grubisic, and T. Volaric, "Different Applications of Mobile Robots in Education," *Int. J. Integr. Technol. Educ.* , vol. 6, no. 3, pp. 15–28, Sep. 2017.

[14] Festo Didactic, "Robotino-Workbook With CD-ROM Festo Didactic 544307 en," Denkendorf, 2012.

[15] P.H. Pawar R.P. Patil, "Image Edge Detection Techniques using MATLAB Simulink," *Int. J. Eng. Res. Technol.*, vol. 3, no. 6, pp. 2149–2153, 2014.

[16] B. Crnokić, "Development of the algorithm for control of mobile robot navigation with application of artificial neural networks," University of Mostar, 2016.

Authors: **Assist. Prof. Boris Crnokić, Assist. Prof. Miroslav Grubišić**, Matice hrvatske bb, 88000, Mostar Bosnia and Herzegovina, Phone.: +387 36 337-001, Fax: +387 36 337-012.

E-mail: boris.crnokic@fsre.sum.ba,
miroslav.grubisic@fsre.sum.ba