IOP Conference Series: Materials Science and Engineering

PAPER • OPEN ACCESS

# Collision avoidance using neural networks

To cite this article: Shilpa Sugathan *et al* 2017 *IOP Conf. Ser.: Mater. Sci. Eng.* **263** 052041

View the article online for updates and enhancements.

Related content

- Obstacle Avoidance for Quadcopter using Ultrasonic Sensor
Muhammad Fazlur Rahman and Rianto Adhy Sasongko

- Smart Collision Avoidance and Hazard Routing Mechanism for Intelligent Transport Network
Gurpreet Singh, Pooja Gupta and Mohd Helmy Abd Wahab

- Real-time Collision Avoidance and Path Optimizer for Semi-autonomous UAVs.
A.F Hawary and N.A Razak

IOP ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

This content was downloaded from IP address 79.110.19.189 on 17/01/2019 at 05:00

# Collision avoidance using neural networks

**Shilpa Sugathan, Sowmya Shree B V, Mithila R Warrier and Vidhyapathi C M**
School of Electronics Engineering, VIT University, Vellore 632014, Tamil Nadu, India

E-mail: vidhyapathi.cm@vit.ac.in

**Abstract.** Now a days, accidents on roads are caused due to the negligence of drivers and pedestrians or due to unexpected obstacles that come into the vehicle's path. In this paper, a model (robot) is developed to assist drivers for a smooth travel without accidents. It reacts to the real time obstacles on the four critical sides of the vehicle and takes necessary action. The sensor used for detecting the obstacle was an IR proximity sensor. A single layer perceptron neural network is used to train and test all possible combinations of sensors result by using Matlab (offline). A microcontroller (ARM Cortex-M3 LPC1768) is used to control the vehicle through the output data which is received from Matlab via serial communication. Hence, the vehicle becomes capable of reacting to any combination of real time obstacles.

## 1. Introduction

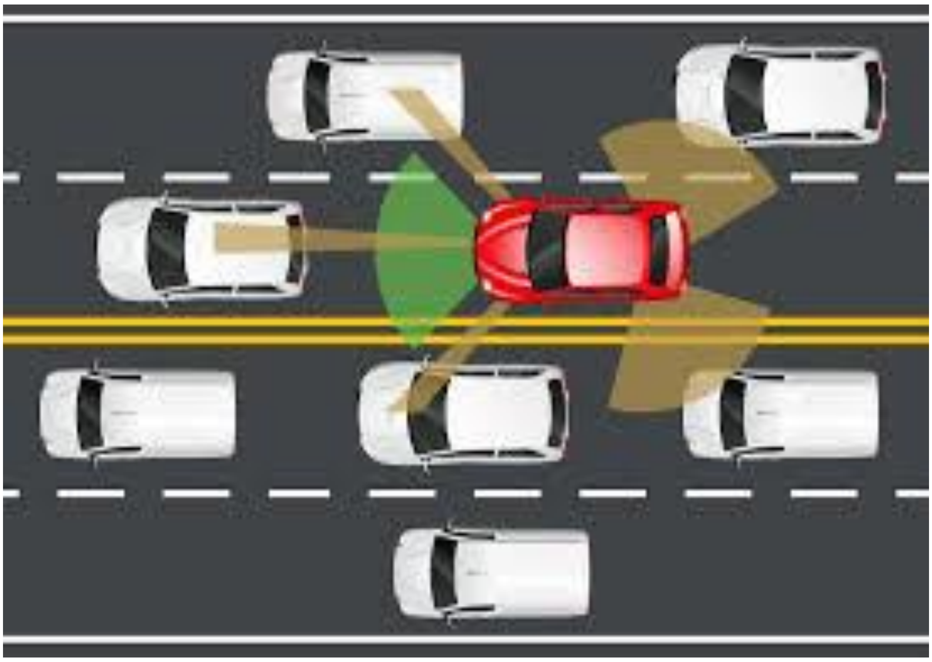

**Figure 1.** Sketch of the proposed system on road

Collision avoidance system plays an important role in reducing number of accidents and saving human lives. [12] Robots and autonomous systems are becoming more popular day by day. One of the best method for successful operation of autonomous vehicles and robots are neural network. But

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

obots have some difficulties in real time such as real time obstacles. [11] To avoid this sensors are used. Sensors like IR proximity sensor are used to detect obstacles and sensor is also low cost. Neural Network is an important information processing model that is inspired by the way biological nervous systems, such as the process information and brain. [9] A perceptron neural network is used. It is an algorithm of supervised learning. It takes both binary and bipolar input and output. [6] Supervised learning, training and testing is done offline using Matlab. Matlab is a tool which is used for numerical computations and manipulations, plotting graphs, etc. This is developed by Math Works.[10] This can be used as a user interface and interfacing with programming languages written in like C, C++, java, Fortran, etc. LPC1768 is an ARM Cortex-M3 microcontroller. It is a 32-bit RISC ARM processor. Mbed online compiler is used for compiling purpose for LPC1768. [7] Our project mainly deals with collision avoidance in real time world. In this collision avoidance system, fourIR Proximity sensors are used to detect the real time obstacles. The four sensors are placed on four critical sides of the vehicle (robot) left, right, front and rear. Sensor gives a digital output such as '0' and '1'.[3] If obstacle is detected then sensor output will be '0' otherwise '1'. Based on the sensor output the vehicle will react to real time obstacles. For all different possible combinations of sensor results a perceptron neural network is trained and tested in Matlab. [2]A microcontroller LPC1768 is used to control the vehicle from the data received via serial communication from Matlab. Hence, this model reacts well to any combinations of real time obstacles. [5] This system helps and also assists the drivers for a smooth travel without any accidents.

## 2. Related work

As per the part of the study related to collision avoidance using neural network, it is found that although mobile robots is an most upcoming field in robotic field.[12] But this also has some difficulties such as real time obstacles. To avoid this sensors are used. If blind spots are detected then it leads to accidents. Solution is to use more number of sensors.[8] Eight sensors are used in Low cost Mobile Robot using Neural Networks in Obstacle Detection, this has 256 input patterns, and all these are trained and tested using neural network and interfaced with LabVIEW.[5] In this paper, training and testing is difficult for all 256 combinations and also interfacing with labview also has some difficulties.[6] So, we purposed a simple and cost effective collision avoidance system with four IR proximity sensors.[6] First, sensors used should of good range in detecting obstacles so that there are no blind spots. Blind spots are main reason for accidents. [1] Based on the different possible combinations of the sensor results the neural network will be trained and tested in Matlab and interfaced with LPC1768 to control vehicle via serial communication using Matlab.[4] Thus, our system has a capability to react fast to real time obstacles and our main aim is to avoid accidents.

## 3. Methodology

### *3.1 Hardware setup*

The hardware setup comprises of 4 proximity IR sensors. These sensors are placed at the front, left, right and rear sides of the vehicle, which are common blind spots for the driver. The block diagram is shown in figure 2.

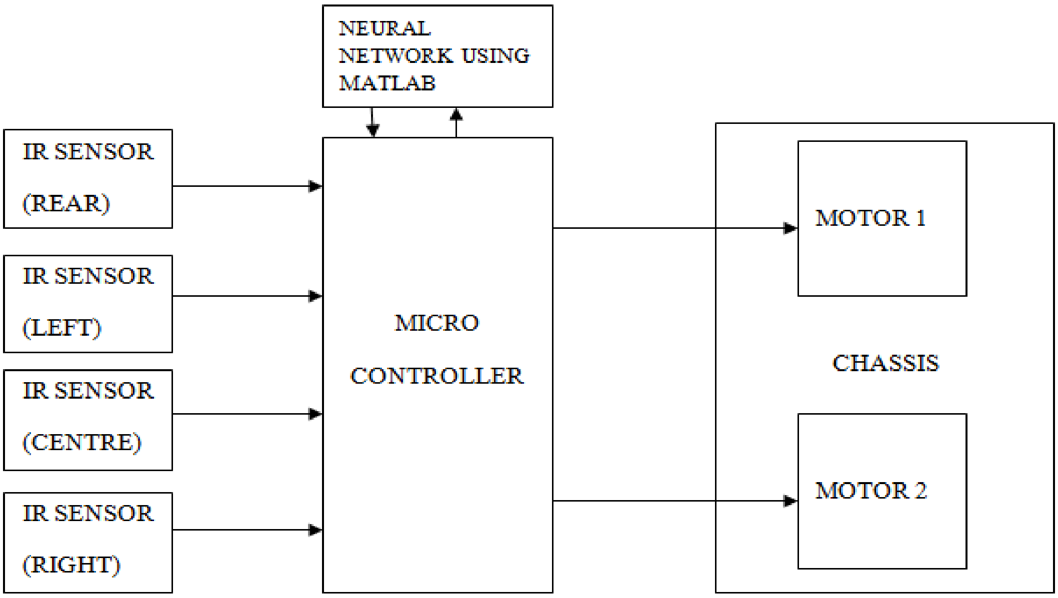


**Figure 2.** Block diagram of the proposed system

The sensors interfaced are active low sensors i.e. they will output '0' when obstacle is sensed and '1' when no obstacle is present. With the 4 sensors $2^4$ combinations of input will be processed and accordingly the vehicle will be told to move forward, reverse, left, right or stop. The ARM Cortex M3 board, LPC178 is used to acquire the sensor patterns.[3] These patterns are transferred to pc serially. The neural network is already trained offline in MATLAB, as per the input sensor pattern the network will produce a result which is transferred to the microcontroller via serial link. [3]The controller then controls the movement of the chassis interfaced to it. This is shown in real time. The hardware setup is shown in figure 8.

*3.2 Hardware description*

*3.2.1. Proximity sensor.* The IR sensors used in this paper are IR Obstacle Proximity. They are active low sensors, giving an output of '0' when obstacle is detected. The figure shows a picture of the sensor

**Figure 3.** IR proximity sensor

*3.2.2. ARM cortex m3 board, lpc 1768.* The controller used is LPC1768. It has a flash memory up to 512 KB and SRAM up to 64 KB. The serial peripherals are a 12 Mbps USB 2.0 device, OTG controller with on chip PHY, four UARTS' with fractional baud rate generation. Its analog peripherals comprises of 12-bit ADC with eight channels and 10-bit DAC [4]. The other peripherals are up to 70 GPIO, motor control PWM, Internal RC Oscillator of 4MHz.

IOP Conf. Series: Materials Science and Engineering **263** (2017) 052041 doi:10.1088/1757-899X/263/5/052041

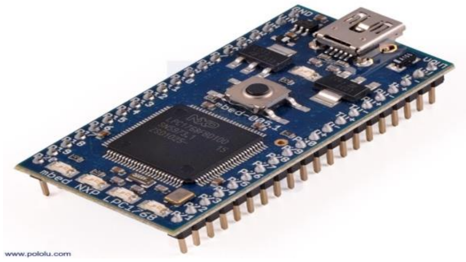

**Figure 4.** ARM LPC1768

*3.2.3. Chassis and dc motors.* A two wheel chassis is used. The motors used to drive the wheels are two 60 rpm dc motors. They run on a voltage of 5V. An L293D driver circuit is used to control the direction of the motor.

**Figure 5.** Chassis set

*3.3 Software description*

*3.3.1. ARM mbed online compiler.* The compiler used for interfacing the sensors and chasses is the mbed online compiler. ARM provides an IDE for easy and quick deployment of ideas on hardware platforms.

*3.3.2. MATLAB.* MATLAB is used to train the neural network offline. MATLAB is the most efficient and productive software environment for researchers, engineers, students alike. The matrix-based MATLAB language helps express computational mathematics in a simple format.

## 4. Neural network controller

*4.1 Perceptron algorithm*

The neural network adopted for training is the Perceptron Network. It is a supervised learning algorithm with re-enforcement where after each input-output pattern example is presented, we know if the target is achieved. Therefore only the input vector can be used for weight correction. Here the input and target vectors are initially fed to the system. Training the network is done by updating the weights after each epoch depending on if the target has been achieved or not. If the target is achieved, the corresponding weights are recorded. Figure shows basic model of Perceptron network

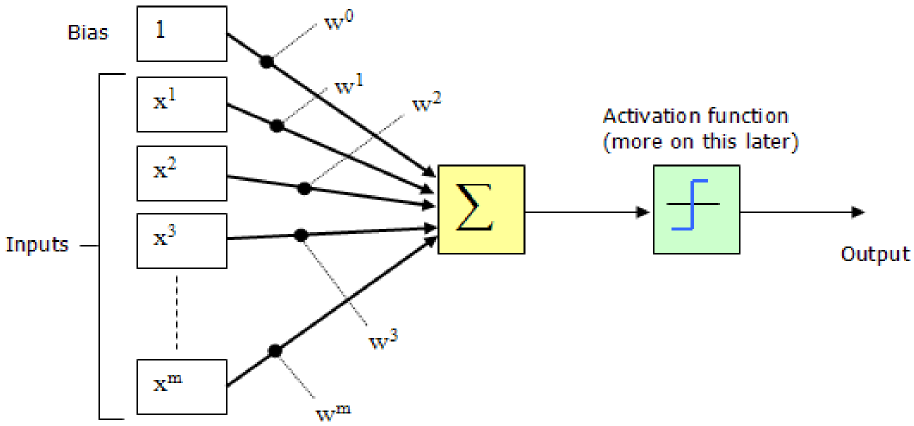


**Figure 6.** Model of perceptron network

It consists of one input layer, one output layer and no hidden layer. As the targets are linearly separable there is no need of a hidden layer. The inputs are multiplied with their corresponding weights and then summed with a bias to obtain an intermediate output.This intermediate output is then passed through an activation function giving us the desired output. The activation function used in this paper is a binary step function given by equation 1

$$f(x) = \begin{cases} 1 & \text{if } w \cdot x + b > 0 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

The weight updating function is given by equation 2, where η is the learning rate. For Perceptron networks it is usually assumed to be 1.

$$w^{new} = w^{old} + \Delta w \qquad \Delta w = \eta(d - o) \tag{2}$$

*4.2 Neural network design and implementation*

Neural network structure is implemented using MATLAB environment. Input layer consists of 4 neurons and output layer consists of five neurons. The four input neurons are '0' and '1' from four sensors according to detection of obstacle. Output represents the action to be taken by the vehicle. Table 1 shows the truth table of the network. In this case the outputs are forward, reverse, left turn, right turn and stop.

| LEFT | RIGHT | CENTRE | REAR | L.TURN | R.TURN | FWD | REV | STOP |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |

IOP Conf. Series: Materials Science and Engineering **263** (2017) 052041 doi:10.1088/1757-899X/263/5/052041

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |

**Table 1.** Truth table of the proposed system

## 5. Results

The offline training and testing of the neural network is done offline using MATLAB. The training, performance of the net and hardware setup with sensors and model is shown in below figures.

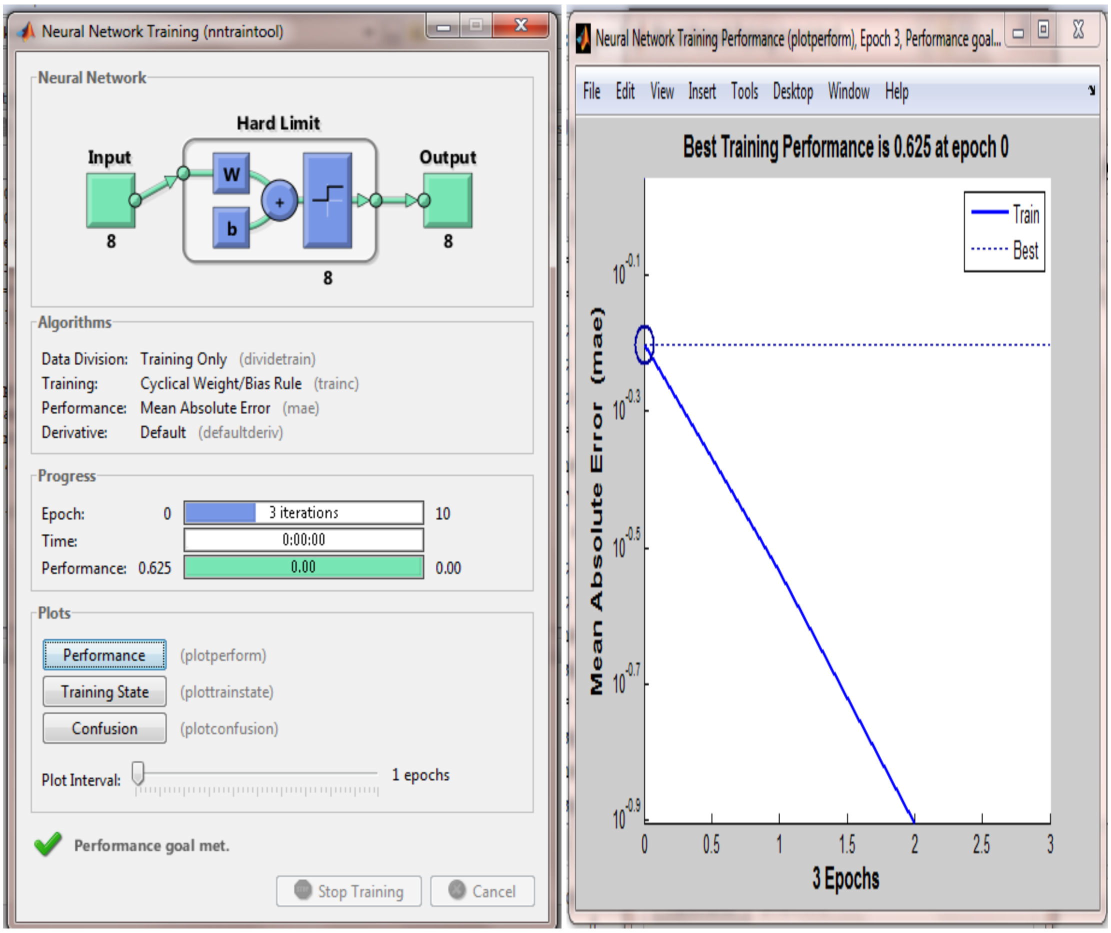


**Figure 7.** Training performance of the perceptron network on MATLAB

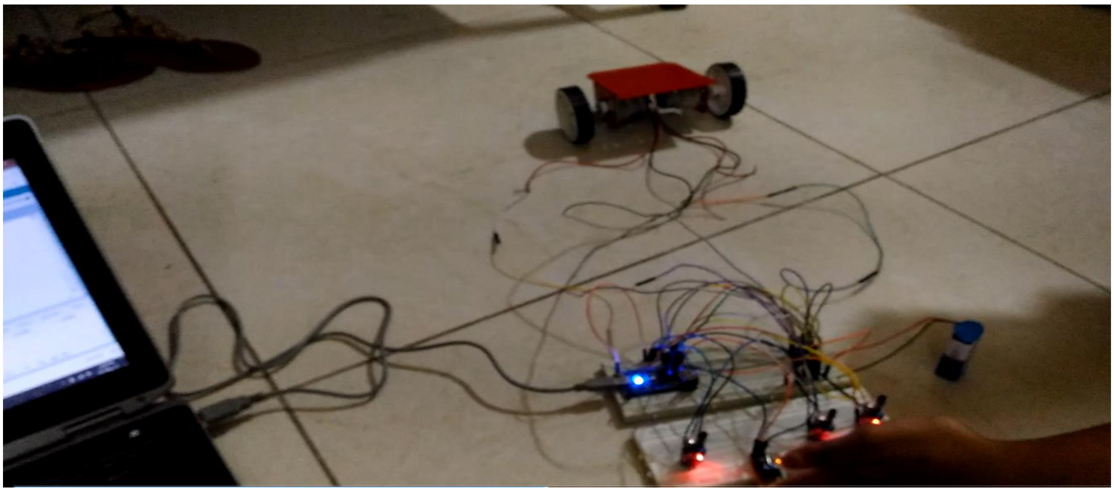

**Figure 8.** Hardware setup of the implemented system

## 6. Conclusion

In this project, collision which occurs due to unexpected real time obstacles is avoided using IR proximity sensors with the help of a single layer perceptron neural network, a microcontroller

LPC1768 and Matlab. FourIR Proximity sensors are used to detect the obstacles. The four sensors are placed on fourcritical sides of the vehicle left, right, rear and front. Based on the different possible combinations of the sensor results the neural network will be trained and tested offline in Matlab and interfaced with LPC1768 to control vehicle via serial communication from Matlab. The vehicle hence becomes capable of reacting to any combination of real time obstacles. Main aim is to avoid accidents; this can be achieved through this.

**References**

[1] Durand N, Alliot JM, Noailles J. 1999 Collision avoidance using neural networks learned by genetic algorithms *Proc. of 9th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert systems*.

[2] Thorsson JL, Steinert O. 2013 *Neural Networks for Collision Avoidance.*

[3] Zimmerman TA. 2014 Neural Network Based Obstacle Avoidance Using Simulated Sensor Data.

[4] Park KS, Choi HS. 2006 Neural Network Based Path Planning Plan Design of Autonomous Mobile Robot *Proc. of IEEE Internatinal Conference on InSICE-ICASE* 3757-3761.

[5] Yang X, Meng M. 1999 A neural network approach to real-time motion planning and control of robot manipulators. *Proc. Of IEEE International conference on InSystems, Man, and Cybernetics* **4** 674-679.

[6] Meng M, Yang X. 1998 A neural network approach to real-time trajectory generation [mobile robots]. InRobotics and Automation *Proc. Of IEEE International Conference on Robotics and Automation* **2**1 725-1730.

[7] Nagarani R, Nithyavathy N, Parameshwaran R. 2013 Low cost Mobile Robot using Neural Networks in Obstacle Detection.

[8] Pawan Deshpande 2014 Road Safety and Accident Prevention in India A review *Int J Adv. Engg Tech* **V** II 64-68.

[9] Mark Campbell, Magnus Egerstedt, Jonathan P. How, Richard M. Murray 2010 Autonomous driving in urban environments: approaches lessons and challenges *Philosophical Transactions of the Royal Society A: Mathematical Physical and Engineering Sciences* **368** 1928 4649-4672.

[10] Marc Cranswick, Pontiac Firebird 2013 The Auto-Biography, Veloce Publishing Ltd.

[11] Todd Litman 2015 Autonomous Vehicle Implementation Predictions Implications for Transport planning Victoria Transport policy Institute.

[12] James M. Anderson, Nidhi Kalra 2014 Autonomous Vehicle Technology: A Guide for Policymakers RAND Corporation.