A New Classification Technique in Mobile Robot Navigation

Siti Nurmaini, Bambang Tutuko
Computer Engineering Department, Faculty of Computer Science, Sriwijaya University
Jl. Raya Palembang-Prabumulim Km. 32 Indralaya, Ogan Ilir 30662, Palembang, Indonesia
e-mail: siti_nurmaini@unsri.ac.id, bambang_tutuko@unsri.ac.id

Abstrak

Makalah ini menyajikan algoritma pengenalan pola yang baru menggunakan teknik weightless neural network (WNNs).Teknik ini berperan sebagai classifier untuk menilai situasi di sekitar lingkungan mobile robot dan membuat keputusan pengendalian dalam navigasi mobile robot. Teknik WNNs dipilih karena memiliki keuntungan yang signifikan dibandingkan neural network konvensional, antara lain dapat dengan mudah diimplementasikan menggunakan perangkat keras RAM standar, lebih cepat dalam tahap pelatihan dan menghasilkan jumlah komputasi yang kecil. Menggunakan algoritma klasifikasi sederhana, data yang sama akan dikelompokkan dalam satu kelas tertentu dan juga untuk menyatakan kelas-kelas yang sama untuk suatu lokasi tertentu dalam lingkungan mobile robot. Strategi ini diimplementasikan pada mobile robot sederhana menggunakan mikrokontroler yang memiliki 512 bytes RAM dan sensor yang murah. Hasil eksperiment menunjukkan ketika jumlah neuron bertambah pengenalan pola lingkungan rata-rata meningkat dari 87.6% menjadi 98.5%. Teknik WNNs memungkinkan mobile robot untuk mengenali setiap pola lingkungan dan menghindari rintangan di lingkungan secara real time. Selain itu dengan menggunakan teknik WNNs yang diusulkan, mobile robot telah berhasil mencapai target pada lingkungan yang dinamis dibandingkan dengan teknik logika fuzzy dan fungsi logika, mampu menangani ketidakpastian dalam pembacaan sensor, mencapai kinerja yang baik dalam melakukan kegiatan pengendalian dengan sumber komputasi yang minimal dengan rata-rata error kecepatan 0.56%

Kata kunci: kasifikasi lingkungan, mobile robot, weightless neural network

Abstract

This paper presents a novel pattern recognition algorithm that use weightless neural network (WNNs) technique.This technique plays a role of situation classifier to judge the situation around the mobile robot environment and makes control decision in mobile robot navigation. The WNNs technique is choosen due to significant advantages over conventional neural network, such as they can be easily implemented in hardware using standard RAM, faster in training phase and work with small resources. Using a simple classification algorithm, the similar data will be grouped with each other and it will be possible to attach similar data classes to specific local areas in the mobile robot environment. This strategy is demonstrated in simple mobile robot powered by low cost microcontrollers with 512 bytes of RAM and low cost sensors. Experimental result shows, when number of neuron increases the average environmental recognition ratehas risen from 87.6% to 98.5%.The WNNs technique allows the mobile robot to recognize many and different environmental patterns and avoid obstacles in real time. Moreover, by using proposed WNNstechnique mobile robot has successfully reached the goal in dynamic environment compare to fuzzy logic technique and logic function, capable of dealing with uncertainty in sensor reading, achieving good performance in performing control actions with 0.56% error rate in mobile robot speed.

Keywords: environmental classification, mobile robot, weightless neural network

1. Introduction

In autonomous navigation, mobile robot is capable of sensing environments, interpreting the information to obtain the knowledge of its position, and planning real-time routes from initial position to targets with obstacle avoidance, and target reaching [1]-[4]. For a mobile robot to navigate in a completely unknown environment, it has to perceive its environment locally or globally, reason about its perceptions, and act accordingly. The ultimate goal of an autonomous mobile robot is to navigate safely to its target [5]. Many methods have been introduced in the field of mobile robot navigation such as model-based, behavior-based, and

hybridapproaches [6]-[8]. However, these approaches lack of adaptation and perception capabilities.In many cases, autonomous mobile robot has been required to navigate in more complex domains, where the environment is uncertain and dynamic.Autonomous navigation in these environments demands adaptation and perception capabilities. Other approach that has received attention by several researchers is the map-based navigation [2]. In this approach, the mobile robot relies on a map that models the environment. When such map is not given to the mobile robot a priori, it must be built by the mobile robot itself using a sensory system such as ultrasonic and infra-red sensors [2]. However, this approach suffer from the fact thatthey must be fine enough to capture the details of the environmentandthe position of the mobile robot must be precisely determined which is a difficult task in the presence of sensory uncertainty and imprecision [9].

To overcome the limitation, some researchers utilize fuzzy logic (FL) and conventional neural networks (CNNs) techniquein mobile robot navigation [5], [10], [11]. These techniques are emerging approaches which parallel the remarkable ability of the human mind to reason and learn in an environment of uncertainty and imprecision. They bring effectivetechnique and improve the intelligence in mobile robot navigation [10]. However, these approaches suffer from the following drawbacks; (i) deriving and fine-tuning the entire rule set and membership functions (MFs) are often difficult [12], (ii) the performance of NN depends on connecting synaptic weights where the best selection of these could be a tedious job [13], (ii) NN technique often needs a long time-consuming training process, due to real-time floating point calculations involved [14], (iv) particularly in the case of real time mobile robot application, which lead to low speed processing [15].

Hence as a solution, the neural network model called the weightlessneural network (WNNs) is proposed. WNNs have single layer architecture. These neuron computing models are based on binary inputs and outputs, and 'no weight' between nodes, which can be implemented using commercially available random access memories (RAM). In WNNs, training is thus a single epoch process with one training step for each input-output association preserving their high speed [16]-[19]. This also makes associative memories computationally simple and transparent. In contrast, in most conventional neural networks (CNNs) used for classification such as multi-layer perceptron (MLP) or radial basis function (RBF), training takes time and the resultant network is effectively a "black-box" [19]. Storage is efficient in WNNs due to their simple matrix structure. New input patterns do not require additional memory allocation as they are overlaid with existing trained patterns [20].Unlike the CNNs such as MLP or RBF, WNNs are able to partially match records during retrieval, this partial matching is a central concept for this strategy [21], [22]. Thus WNNs can rapidly match records that are close to the input but do not match exactly.The WNNs-based RAM methodology has introduced a look-up table for thresholding value that can retrieve the highest$n$matches unlike the other RAMs [21]. This strategy producefast training, computational simplicity, network transparency and partial match capability make this approach perfect to produce better performance in mobile robot navigation application.

This paper presents a novel pattern recognition algorithm that use WNNs technique to control mobile robot navigation in real-time. The WNNs technique plays a role of situation classifier to judge the situation around the mobile robot environment and makes control decision in mobile robot navigation. Using a simple classification algorithm, the similar data will be grouped with each other and it will be possible to attach similar data classes to specific local areas in the mobile robot environment. In the eksperiment with several environmental patterns, the proposed technique is compare to fuzzy logic dan logic fungction to perform navigation tasks.

2. Proposed Method
2.1.Environmental Uncertainty
Every autonomous robot needs some sensing devices to get first a perception of its environment and then to be able to move in this environment. Generally, high quality and high precision sensors raise the costs of a robot significantly, while the advantage of low precision sensors, such as, for example, infrared or ultrasonic sensors, is their cost. That is why reliable navigation based on low cost sensors would open the possibility of wider application of autonomous, mobile robots. However, low cost sensors provide an imprecise perception of the

environment, making navigation using theses sensors a non-trivial task, especially if the environment is unstructured, unknown to the robot, and contains an unknown number of obstacles of arbitrary shape [2] ,[9].

The imprecise environmental perception of ultrasonic sensors is a result of the fact that these sensors provide a relatively accurate measurement of the distance to an object, but poor information about its exact location due to the angular resolution [1]. The same perception can be obtained from an object placed at different locations. Another source of uncertainty is a consequence of specular reflection and well-known problems such as cross talking and noise. At the same time, the problem of blind sectors between two ultrasonic sensors and the large blind areas of a mobile robot with few sensors may not be neglected [9]. Several procedures have been developed to overcome the disadvantages of low-cost sensors to map first the mobile robot's environment and then to generate the appropriate control commands [9]. However, it should be noted that these representations of the environment still present some inherent imprecisions and uncertainty. Moreover, for some on-board sensorial layout, the computational cost of building up any of such representations could be very high or they could even be quite difficult to obtain when considering a mobile robot with few sensors.It shows the convenience of consideringthe constraints on the mobile robot perception when computing the control action.

2.2. Weightless Neural Network

WNNs are artificial neural networks with random access memories (RAM) devices, which is composed of sets of inputs and outputs [12], [18]. Depending on the sets of information contained on inputs, different memory locations in the RAM are accessed. The WNNs structure has two-layer for pattern-classification network [19] as shown in Figure 1. The first layer is formed with K nodes each of which is a random-access memory (RAM). The second layer consists of a summation device that adds up the output values of all K RAM nodes. As there are K RAM nodes, and each RAM has N input variables (storing 2^N memory locations), the network receives at the input KN one-bit variables. Each RAM samples a different portion of the input field that has KN bit-positions.
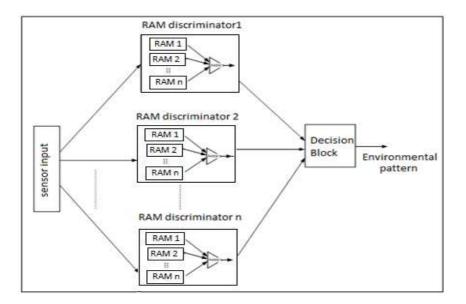

Figure 1. WNNs Structure

Neuron structure is used to classify the environmental pattern of mobile robot as shown in Figure 2.The WNNs inputs$S =(S_1,…, S_n)$ are connected through the RAM address line in groups of discriminator that correspond to one of the possible neuron classes m. The neuron classifiesaccording to environmental pattern, 1 byte data for each sensor is utilized to produces

n groups of neuron. For each class of input patterns one discriminator is needed, which is trained solely on the input data of its own class. The groups are connected to an output adder *O* =($o_{1,1}$, $o_{1,2}$, …,$o_{m,n}$) that counts the number of active neurons in the group,

WNNs have two operational modes: the learning mode, in which the neuron assigns the value on the teach terminal to the addressed content and the recalling mode in which the neuron puts the addressed content on the output terminal [17].In learning mode a test pattern x_p is presented to the discriminator, each discriminator fires accordingly. The discriminator with the highest output sum, therefore, should represent the class of the presented pattern. For j^{th} discriminator the output (z_j) are given by [8]:

$$z_j = \sum_{k=1}^{M} o_j^k \qquad (1)$$

where, o_j^k is the output of the kth RAM of the jth discriminator.
The normalized discriminators output vector for an input pattern is given by:

$$x = [x_1, x_2, ..., x_n], \ 0 \le x_j \le 1 \qquad (2)$$

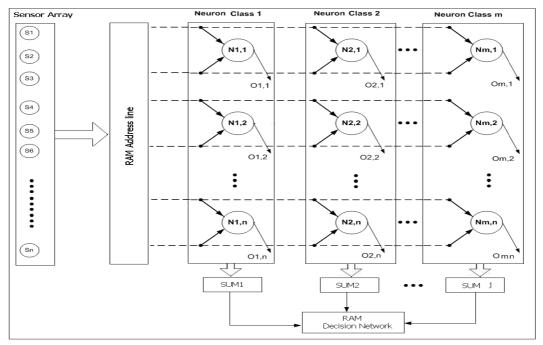For the jth element of the response vector x that belong to class *j* is equal to unity.

Figure 2.Neurons Structure

In the recalling mode each discriminator responds with the number of matching sub-patterns (*sum 1, …, sum j*). The pattern is classified according to the discriminator with the highest response and the network could set a threshold in look up table (LUT). The recognition is accomplished through the sum of all neurons output in RAM decision network, which is associated the corresponding pattern to identify the current environment and choose the winner class. The pattern is accepted and recognized if the output of a discriminator is higher than threshold as shown in Figure 2.

WNNs operation must able not only of recalling the output values stored for the training patterns but also of outputting values for non-trained patterns (unknown patterns) as well or generalization ability.   Generalization is a very important characteristic and relates to the WNNs ability to respond to a novel input (which can be a noisy version of a learnt pattern) according to what was previously learnt [23]. In this work, discriminator response is considered as a measure to the degree of similarity of the applied input pattern to its class. The non-accessed memory locations are set up in terms of how similar they are to memory locations which are set up during training, similarity measure to be used which considers is the Hamming distance [24].

The conventional Hamming distance (CHD) that is the number of bits in which two binary vectors are different. It can be used as the measure of distance between the addresses. However, the drawback of the CHD is that it measure absolute differences in bit without indicating the similarity measure [24], [25]. It does not recognize the notion of approximate similarity, because it measures only the exact extent to which the corresponding bits in two bit vectors agree [24]. For overcome the limitation, this study uses, Extended Hamming distance (EHD). The distance value gives interval for being close. The HD value D_{ij} measure between input bit pattern x_i and test bit pattern x_p, then output vector $O=[o_1, ..., o_D]$ be a bit vector of dimension D and $N(O)$ be the number of 1 bit in O. To compute the similarity measure a cost is associated relative to the distance value $D_s(x)$. To indicate the nearest neighbor distances between x_i and x_p equation (3) is used [26],

The similarity distance $D_s(x) = D_{ij}$ - min distance/max distance (reference)   (3)

Each discriminator output z_j is from being the 'winner' given input vector x_i, a similarity distance degree $D_s(x)$ can be specified, yielding the degree to which x satisfies the corresponding pattern.

3. Research Methods
3.1 Sensors Direction
In order to acquire the information about the environment around the mobile robot, 8 ultrasonic sensors mounted on the front of mobile robot are used to recognize the environment and to avoid obstacle. Ultrasonic sensor has been used extensively in mobile robot for obstacle detection, ranging, and mapping [23], [27]. The WNNs, whose inputs are preprocessed by ultrasonic sensors readings, informs the mobile robot of the environmental situation. The operating frequency of the sensor is 40 kHz. The maximum effective examination distances under short distance pattern and long distance pattern are 10 cm and 3 m respectively. Sensor measurement of the short distance pattern is used. In order to reduce the computing data then 8 sensors are divided into 4 groups to arrange at the four corners and four sides of mobile robot as shown in Figure 3.
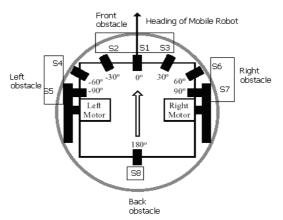

Figure 3. Arrangement of ultrasonic sensors installed in mobile robot

The groups are left obstacle, front obstacle, right obstacle and back obstacle respectively. The sensor ultrasonic positions in all arrays in the mobile robot are fixed; one on back side, and seven facing outwards at 30 degree intervals starting from sensor 1 up to sensor 7. The placement of the sensors permits the detection of an obstacle in different positions, as shown in Figure 3. Sensor 8 is used if the mobile robot is in emergency condition to detect obstacle at the back of the mobile robot for backward movement. Sensor module is configured using 8 microcontrollers PIC16F84 where it controls the eight ultrasonic and one infrared range finder pulse as shown in Figure 3. It calculates the sensor readings, and selects which sensors are active according to their position in the robot periphery.However, typical applications do not exploit the full range of information carried by ultrasonic echoes, focusing only on the time of arrival of the first detected echo [28].

3.2 Environmental Classification

The pattern of sensor data base representing a variety of possible situations of the environment should be built beforehand to classify the whole environment into one of the prototype pattern.Then for every prototype environment is trained by the appropriate module contained in the proposed RAM network architecture. The pattern of environments formed by sensor measurements gives the information with reference to the environment situation. Variety of environment situation may arise in front of mobile robot when mobile robot goes through the environment. In this work, 7 ultrasonic sensor values are input to the WNNs to generate one of the classified environments. The result is used to identify different patterns of obstacle feature include cylinder, plane, and corridor feature, also environments feature with different angle such as, concave, convex and others. Training consist of placing each object at various distances from the ultrasonic beam, collecting the echoes for each object at each position and train the WNNs for each object at each position.
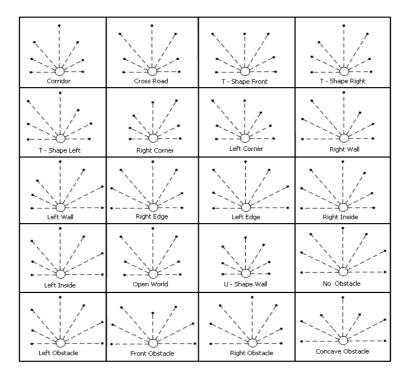

Figure 4. Sensory pattern for 20 feature of environment

To navigate efficiently the WNNs is trained with 50 patterns representing typical scenarios, some of which are depicted in Figure 4. The example of some sensory patterns registered for 20 environmental entire prototypes.Ultrasonic sensors distance is set in range 10 cm to 300 cm appropriate to data bases of sensors measure. Although a set of corridor patterns has been

used to train WNNs, but the training process must be generalized into all kinds of environments, including scattered obstacle in the environment that may be encountered by a mobile robot during navigational tasks. Hence, the output of the environmental classification network can be used to build a topology map to determine the mobile robot high-level behavior.

3.3 Simple WNNs Algorithm

WNNs must be trained to recognize the correct patterns from incorrect ones and it works by generating all the possible input combination. This allows the network to generalize previously unseen cases, as well as make correct approximation. Training is a single-pass process in which each discriminator is trained individually on a set of patterns that belongs to it. In the training process all the sites of discriminator that are addressed by this pattern are set to 1. This process is repeated until all pattern of set are trained and when the process is complete, the WNNs has encodes the information in its memory register about the training subset.

During learning, the input vector, which is an n-bit binary number, acts like an address generator, where n defines the length of the input vector, and $0 < n < \infty$. Every location addressed is a signed register, and all of the registers in the network locations are initially set to zero interpreted as an undefined value. When all training pattern are presented to the WNNs, the data stored to all the addressed register of RAM which is in write mode and the networks is counting the frequency of a bit pattern sequence encountered in the binary matrix.

```
Start;
    1.  Set the contents of all address line of memory location to initial state
    2.  Present an input pattern to the input terminals together with the desired
        response to the output terminals.
    3.  The contents of the memory locations are propagated forward through the
        layers of the network until they reach the output node.
    4.  The response of the network is compared with the desired output
        a.  if they are similar, then the contents of the addressed memories will
            assume their current values (i.e. the addressed memories will be
            rewarded), or
        b.  if they are different, then the network is allowed to run again until:
            - The output matches the desired response, then neuron active.
            - The output mismatches the desired response β times (β is a parameter
              which should be set before training). In this case, the addressed
              memories are punished by reverting their contents back to the initial
              state.
    5.  Repeat by going back to Step 2 for all the N training patterns.
    6.  The algorithm halts when consistent success (the correct output produced N
        times consistently) indicates that all patterns have been learned.
End;
```

Figure 5.Learning algorithm of WNNs

Shown in Figure 5 is the learning algorithm, initially all RAM register stores 0. The training set consists of an equal number of patterns from each class. Each neuron has n address lines of memory and 2^n single bit storage locations. The WNNs will generate a feature vector *(x)* when an unknown pattern is detected to the networks that describes the class of the input vector. The outputs of all discriminator are summed together to give its response. The data is stored in the discriminator after all training patterns are presented to the network and give the generalization ability. Generalization is one important characteristic and in WNNs and

enables it to respond to a current input which can be a noisy version of a learnt pattern according to what was previously learnt. This allows the network not only to perform content addressability but also to perform default assignment [22]. This means that it is possible to extract the general features of a given concept based on the input information.

In this work, generalization is introduced by considering the hamming distance from training patterns. The Hamming distance between two binary patternsis the number of bit positions in which they differ. It is simply defined as the number of bit that is different between two bit vectors and this gives a rough idea of how similar thetwo patterns are [29].

4. Results and Discussion

In order to verify the feasibility of the technique presented in this paper, a navigational experiment of the mobile robot is done.The proposed WNNs contains several microcontrollers to avoid require large memory that lead to the robust implementation. It could also improve the system flexibility by using multi-processor. In performing navigational tasks, all the data detection from ultrasonic sensor are input to WNNs, which tell to WNNs the obstacle-scenarios in the environment. Thus WNNs outputs are the environmental and the obstacle pattern.Depending on the pattern, a decision is made by the WNNs sent to the actuator to produce suitable commands for mobile robot action.

As WNNs is memory-based it requires a number of RAM and ROM inside a microcontroller. In this work, the neurons are realized in RAM. The size of RAM used is 8 x 128 byte for sensor array, 2 x 512 byte for navigation module with size of ROM is 20 K x 8 bit for processing the controller. This size provides plenty of storage to investigate any changes in the structure, size and number of neurons.To identify how the neurons are organized in RAM, Table 1, shows WNNs with 7 classes and 8 neurons belong to each class is considered, thus the number of neurons are 56. Each neuron (n) represents specific pattern in the environment.

Table 1.A typical WNNs with eight address lines

Address	class 1	class 2	class 3	class 4	class 5	class 6	class 7
Line 1	n11	n21	n31	n41	n51	n61	n71
Line 2	n12	n22	n32	n42	n52	n62	n72
Line 3	n13	n23	n33	n43	n53	n63	n73
Line 4	n14	n24	n34	n44	n54	n64	n74
Line 5	n15	n25	n35	n45	n55	n65	n75
Line 6	n16	n26	n36	n46	n56	n66	n76
Line 7	n17	n27	n37	n47	n57	n67	n77
Line 8	n18	n28	n38	n48	n58	n68	n78

All neurons that are connected with different address line, which connects each pattern to the address input of the neurons.In this work, all neurons are placed in 8 RAMs with a different memory location.1 neuron is placed in 1 block of memory address, 1 block comprises of 8 memory locations with 1 memory location is equivalent to 8 bit, thus 1 block has 64 bit as shown in Figure 6.
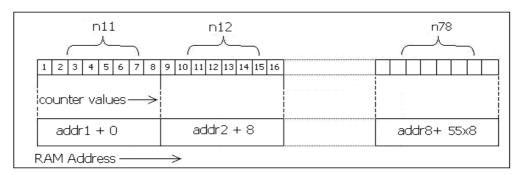

Figure 6.Each neuron has eight counting states

Hence, look-up tables are applied to implement the navigation control circuit. The network uses 8 binary inputs from sensor arrays are represented by a black box structure. The neurons connection is specified. It can generate many binary vectors as environmental patterns and all patterns data stored in look-up table. This strategy is used as the navigational control of the mobile robot.

This research considers WNNs technique in mobile robot navigation to be applied to complex real environment. The controller receives the inputs connected to the address lines of RAM device. Implementation of WNNstechnique in this work is limited by the amount of available memory since the neurons are assembled as binary vectors and stored all input pattern into 11 commercial RAM as a look-up table in pyramidal shape. The result of the proposed solution is shown in Table 2, indicates the amount of memory required by adopted WNNs.

Implementation of the proposed design strategy is to find WNNs structures including neuron numbers and input numbers that show the best results for the environmental recognition problem. WNNs topologies are analyzed in order to specify which neuron architecture presents the best recognition level. It is noticed that the WNNs can saturate if the number of presented patterns is high, instead of improving its recognition rate [29], [30]. This research investigates the error rate of recognition produced by the approach after training process. Table 3 shows summary of percentage of WNNs recognition level, the table shows that the performance test using maximum 9 input neurons produce rejection and recognition level which gives pattern similarity of the WNNs. The average recognition rate based on that technique has risen from 87.6% to 98.5%.

Table 2.Memory used in proposed architecture

Input number	Neuron number	Pattern number	Memory (bit)	Memory (byte)
3	60	30	12,288	1,280
4	60	30	12,208	1,536
5	40	30	16,384	2,048
6	40	30	16,384	2,048
7	40	50	22,528	2,816
8	32	50	24,576	3,072
9	24	50	28,672	3,584

Table 3.WNNs recognition level using proposed architecture

Input	Neuron	Pattern	Recognition (%)	Error (%)	Rejection (%)
3	60	30	87.6	1.1	11.3
4	60	30	90.0	1.2	8.8
5	40	30	94.0	0.8	5.2
6	40	30	95.3	0.8	3.9
7	40	50	95.5	1.1	3.4
8	32	50	96.7	1.0	2.3
9	24	50	98.5	0.9	1.3

To validate the performance of mobile robot to perform navigation tasks, the experimental is conducted to see changes in motor speed in dealing with changing environmental patterns. Previously mentioned, that there are 50 patterns of the environment has been entered on the look-up table in WNNs. When the mobile robot moves, the environment changes dynamically and it must be solved by a mobile robot. From graph in Figure 7 shows that the speed of mobile robot movement is relatively stable although the environmental changes. Error rate of the motor speed when the mobile robot movement is only 0.56 % as shown in Figure 8, this occurred because of the presence of cross talk in ultrasonic sensor and produce noise in the speed data when it detects every patterns of the environment.

The experimental results shown by the proposed technique has yielded promising results that indicatethe mobile robot is able to recognize the environmental pattern and achieve robust performance with less computational cost. Furthermore, it has successfully performed in several mobile robotenvironment outperform other controllers including logic function and fuzzy logic as shown in Figure 9. From all experimental results, this technique could improve the

flexibility, since it allows fast processing, decreases computational cost and improves performance in mobile robot navigation.
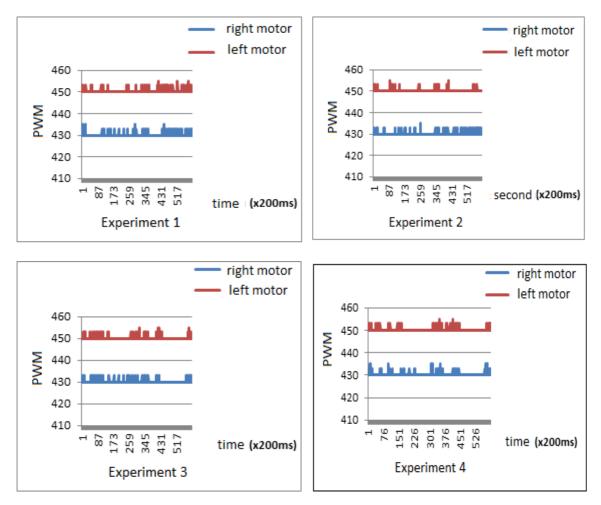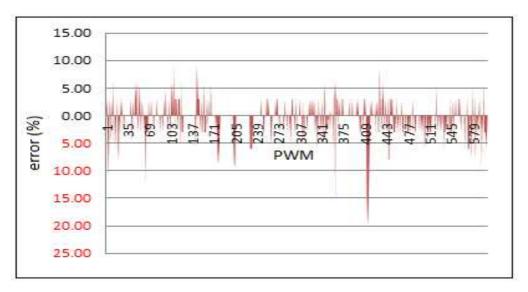

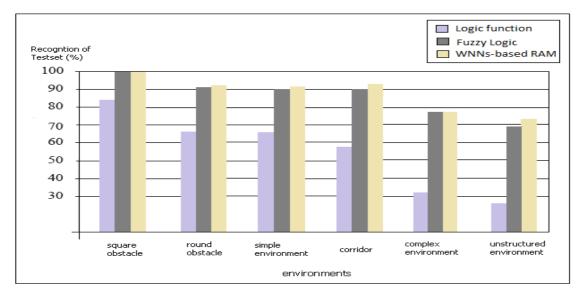Figure 7. Response of motor speed when environmental changes

Figure 8. Error rate for motor speed in mobile robot movement

Figure 9.Comparison of recognition rate for several environments

5. Conclusion

WNNs help to overcome the sensor noise that exist in a real time environment, generating a good reactive system appropriate for complex environment, with static as well as dynamic obstacles. It is apparent the ability of WNNs controller in avoiding obstacles, the experiment shown that WNNs-based RAM has successfully generated 0.56% error rate of motor speed when mobile robot moves, only 3.854 byte memory resources and high recognition rate compare to logic function and fuzzy logic controller. This indicates the proposed algorithm has been verified via real experiments and being compared to logic function and fuzzy logic controllers. The mobile robot has successfully reached the goal in dynamic environment, capable of dealing with uncertainty in sensor reading, achieving good performance in performing control actions with minimal computational resources.

References

[1] Cuesta F, Ollero A. Intelligent Mobile Robot Navigation. *Springer-Verlag*. 2005; STAR 16:79–122.
[2] Maaref H, Barret C. Sensor-based navigation of a mobile robot in an indoor environment. *Robotics and Autonomous System*. 2002; 38:1-18.
[3] Magnenat S, Retornaz P, Bonani M, Longchamp V, Mondada F. *ASEBA: A modular architecture for event-based control of complex robots*. Proceedings of International conference on Fun and Games.Springer. 2008; 1-9.
[4] Skrzypczyk K. Time Optimal Target Following by a Mobile Vehicle. *Springer-Verlag*. 2009; 671–678.
[5] Hui BN, Pratihar KD. A comparative study on some navigation schemes of a real robot tackling moving obstacles. *Robotics and Computer Integrated Manufacturing*. 2009; 25:810-828.
[6] Carlson J, Murphy RR, Nelson A. *Follow-up analysis of mobile robot failures*. Proceedings of International Conference on Robotics & Automation, New Orleans, LA, 2004; 4987-4994.
[7] Stoytchev A, Arkin R. Combining deliberation, reactivity, and motivation in the context of a behavior-based robot architecture. *Journal of Advanced Computational Intelligence and Intelligent Informatics*. 2004; 8(3):100-105.
[8] Gregor K, Igor S. Tracking-error model-based on predictive control for mobile robots in real time. *Robotics and Autonomous Systems*. 2007; 55: 460–469.
[9] Hagras, H. *Developing a type-2 FLC through embedded type-1 FLCs*. IEEE World Congress on Computational Intelligent. Hong Kong. 2008: 148-155.
[10] Hoffmann, F. Soft computing techniques for the design of mobile robot behaviors. *Information Science*. 2000; 122: 241-258.
[11] Widodo, S,N. Penerapan multi-mikrokontroller pada model robot mobil berbasis logika fuzzy. *TELKOMNIKA Indonesian Journal of Electrical Engineerin*g. 2009; 7(3): 213-218.
[12] Simoes, E.D.V. *An embedded evolutionary controller to navigate a population of autonomous robots*. Frontiers in Evolutionary Robotics. 2008; 439-464.

[13] Arbib AM, Fellous JM. Emotions: from brain to robot. *Trends in Cognitive Science*. 2004; 8(12): 554-561.

[14] Ayrulu B, Barshan B. Neural networks for improved target differentiation and localization with sonar. *Neural Networks*. 2001; 355-373.

[15] Pratihar DK. Algorithmic and soft computing approaches to robot motion planning. *Machine Intelligence and Robotic Control*. 2003; 5(1): 1–16.

[16] Corradini M, Ippoliti G, Longhi S. Neural networks based control of mobile robots: development and experimental validation. *Journal of Robotic Systems*. 2003; 20(10):587-600.

[17] Austin J. RAM based neural networks, a short history. *Neural Processing*. 1998; 3-17.

[18] Ludemir TB, De Souto MCP, De Oliveira WR. On a hybrid weightless neural system. *International Journal Bio Inspired Computation*. 2009; 1(2): 93-104.

[19] Al-Alawi R. Performance evaluation of fuzzy single layer weightless neural network. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*. 2007; 15(3): 381–393.

[20] Aleksander I, De Gregorio M, França FMG, Lima PMV, Morton H. *A brief introduction to weightless neural systems*. Proc. on Artificial Neural Networks - Advances in Computational Intelligence and Learning. 2009; 299-305.

[21] Arguelles AJ, Leon DE, Yanez JL, Camacho C. *Pattern recognition and classification using weightless neural networks and Steinbuch Lernmatrix*. SPIE Optics and Photonics. 2005; 591-600.

[22] Howells G, Fairhurst MC, Rahman F. An exploration of a new paradigm for weightless RAM-based neural networks. *Connection Science*. 2000; 12(1):65–90.

[23] Corragio P, De Gregorio M, Forastiere M. Robot navigation based on neurosymbolic reasoning over landmark. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*. 2008; 22(5): 1001-1014.

[24] Cha HS, Tappert C, Yoon S. Enhancing binary feature vector similarity measures. *Journal of Pattern Recognition Research*. 2006; 1(1): 63-77.

[25] Canuto AMDP. Combining neural networks and fuzzy logic for application in character recognition. PhD Thesis. University of Kent; 2001.

[26] Howells WGJ, Kola S, Statheros T, McDonald-Maier K. *An intelligent fast-learning multi-classifier system based on weightless neural architectures*. Proceeding of Recent Advances in Soft Computing. 2006: 72–77.

[27] Coraggio P, De Gregorio M. WiSARD and NSP for robot global localization. *In: nature inspired problem-solving methods in knowledge engineering*. Part II, J. Mira & J.R. Alvarez eds. LNCS. Springer. 2007; 4528: 449-458.

[28] Aguirre E, Gonzalez A. A Fuzzy perceptual model for ultrasound sensors applied to intelligent navigation of mobile robots. *Applied Intelligence*. 2003; 19: 171–187.

[29] Hannan MA, Dimond KR. *Design of an FPGA based adaptive neural controller for intelligent robot navigation*. Proceedings Euro-micro Symposium on Digital System design. (2002); 283-290.

[30] Howells WGJ, Kola S, Statheros T, McDonald-Maier K. *An intelligent fast-learning multi-classifier system based on weightless neural architectures*. Proceeding of Recent Advances in Soft Computing. 2006; 72–77.