Intelligent Senses in Robot Based on Neural Networks

Vimal Ganotra, Rashmi Sharma, Priti Dhingra, Sourabh Sharma, and Navin Kumar

***Abstract*—This Paper describes a model and an implementation of spiking neurons for embedded microcontrollers with few bytes of memory and very low power consumption. The proposed model consists of an elementary neuron network that used Hebbian Learning to train a robot to respond to the environment implementing Artificial Intelligence (AI) in robot. The model is implemented using ATMEGA8 Microcontroller based on AVR RISC Architecture and tested with an ability to move forward and Backward according to intensity of light without human intervention and external computers.**

***Index Terms*—AI, ATMEGA8 microcontroller, hebbian learning, robots.**

I. INTRODUCTION

Now days, automation is applied in every industry. The human labor is being replaced by robots, ex. CNC machines and PLC. The time is demanding more advancement in every sector of life. AI based on speech and image processing has been developed at University of Emirates, UAE. The interest of man is increasing in AI and it has got no limits, weather it is replicating all the features of a human being or making a super human.

In the recent past, with the improvement of the technologies associated with computing and robots, there has been a broad based attempt to build embodied intelligences. But the peculiar nature of this field has resulted in the many attempts being almost entirely unconnected. Because of the difficulty and lack of success in building physical robots, there has been a tendency towards computer simulation, termed "Artificial General Intelligence" where virtual agents in a virtual reality world attempt to achieve intelligent behavior. By the 1980's AI researchers were beginning to understand that creating artificial intelligence was a lot more complicated than first thought. Given this, Brooks came to believe that the way forward in consciousness was for researchers to focus on creating individual modules based on different aspects of the human brain, such as a planning module, a memorymodule etc., which could later be combined together to create intelligence[1]. Some devoted Particular attention to the Question of what kind of basic element, possibly

Corresponding to a model of single neuron function, should be assumed in neural Spatio neural network paradigm Network research. It is to this research on alternative basic elements for neural network modeling that we turn our attention first, because a principle is emerging from this research that may have important implications for our understanding of natural and perhaps artificial intelligence[2] Fig 1. shows the spatio-temporal neural network paradigm, a neural network interacts with its environment in a real-time, closed loop fashion that is of fundamental importance, with respect to the learning mechanisms that are employed, and perhaps with regard to the neural network's function, in general.

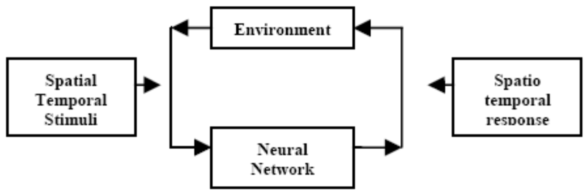

Fig. 1. Spatio temporal neural network paradigm.

The paper is organized into seven sections. Section 2 discusses the term Hebbian learning. In section 3, Background of this work, while section 4 presents the methodology and the proposed architecture. Section 5 demonstrates the Evolution model and implementation details .Section 6 presents the results. Finally, the paper is concluded in section 7.

II. HEBBIAN LEARNING

Our proposed model consisted of an elementary eight neuron network that used Hebbian Learning to train a robot to respond intelligently to input light stimuli. First, we decided upon a task that would accurately denote Hebbian learning. One of the most common examples of conditional learning such as Hebbian learning is seen in Pavlov's experiment with his dog. In this experiment, when food was offered to the dog, it caused the dog to salivate. At first the sound of a doorbell elicited no such response. However, Pavlov decided to sound the bell when he offered food to the dog. After a few repetitions of this experiment, the dog began to salivate at the sound of the bell even when no food was present. Here food was the unconditioned stimulus, and the doorbell was the conditioned stimulus. Similarly, in our work we first show that shining light in front of and or behind the robot elicits no response but pressing the push button causes the robot to move forward or backward. We

Manuscript received May 16, 2010; revised June 12, 2012.

V. Ganotra is with Department of Electronics in Videocon, Karnal, Haryana, India (e-mail: vimalganotra007@gmail.com)

R. Sharma is student in Department of Electronics and Communication Engineering, Maharshi Dayanand University, India (e-mail:dexter2011@live.in)

P. Dingra is with Research and Development team in Wipro Technologies, India (e-mail:priti.dhingra4@gmail.com)

S. Sharma is with Robotics Department in Digital Surveillance Inc, Noida, India (email:saurabhfbd@gmail.com)

N. Kumar is student in Department of Electronics and Communication Engineering, Mahamaya University, Uttar Pradesh, India (e-mail: kumarnavin.del@gmail.com)

then press the button while shining the light on the robot and the neural network programmed into the robot causes it to associate the light input with the push button input. Soon the robot moves forward or backward depending on whether the light is shined behind or in front of it in the absence of push button input. There are other neurons in this network that play an inhibitory role and prevent the robot from going too close to the light. They too display learning. Initially, the robot goes very close to a light source before it decides to move in the opposite direction. As time passes by the robot gets more responsive to the light and does not get too close to either light source.

Fig 2. Shows Experiences change the way one perceives, performs, thinks and plan. They do so physically by changing the structure of the nervous system, alternating neural circuits that participate in perceiving, performing, thinking and planning. A very simplified view of learning would state that learning modulates (changes) the input-output, or stimulus action relationship of an organism. Certainly our environment influences how we react to it, and our reactions influence our environment.

In order to reach our end goal, we first programmed a four neuron neural network in ATMEGA8 Micro Pushbuttons, with thorough testing at each level of complexity. Following this, we added the hardware interface. This involved integration of stepper motor control code into our neural network such that the stepper motors would step when the 'motor' neurons fired. We then built the chassis and added all the motors, LDR's, pushbuttons and Microcontroller board to the design.

At the neural level, many different types of changes can be imagined. For example, recordings from individual neurons in the hippocampus show that these neurons change their "place field" (i.e. responses to location in space) as the animal investigates the experiment. This change could be due to changes in the way visual stimuli affect these neurons (synaptic), or in the way the neurons respond to the same inputs (intrinsic).

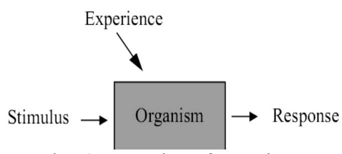

Fig. 2. Learning of organism.

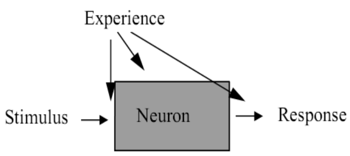

Fig. 3. Learning of neuron.

A very simple example of learning at the organism level which has been worked out a neural level (Fig 3.) is that of sensitization of the gill withdrawal reflex in Alyssa. In the sea mollusk aplysia, a light touch to the animal's siphon results in gill withdrawal.

Hebbian learning rule: The formulation of associative learning that has gathered the most attention for those studying the brain was due to Donald Hebb (see quote above). This proposition has led to a number of mathematical rules, the simplest of which is: $\Delta wij = \mu\ xi\ xj$ where Δw is the change in the synaptic weight connecting neuron j to neuron i and xi and xj are the activities (firing rates, action potentials) of neurons i and j, and μ is a scaling parameter often called *learning rate* as shown in Fig 4.

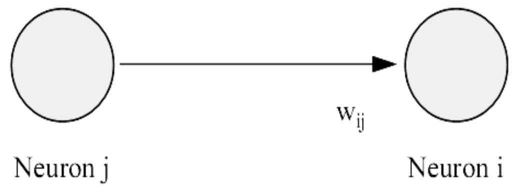

Fig. 4. Learning rate

Hebb's model, while it is a real-time learning mechanism model, utilizes time in a way that may not be valid. Hebb assumed that neurons correlate approximately simultaneous signals. Both the assumption of simultaneity and the assumption that signals are correlated by single neurons can be called into question in that it has now been demonstrated that much stronger contact with the experimental results of classical conditioning studies is obtained if one adopts two alternative assumptions: first, that neurons are correlating sequential rather than simultaneous events as a basis for learning and second, that neurons are correlating changes in signal levels (first derivatives with respect to time) rather than the signal levels, themselves**.** These Alternative assumptions yield a real-time learning.

III. Background

In previous work[3], the evolvability of spiking circuit architecture for vision based navigation of a mobile robot was investigated. The approach[4] described a large set of intelligent devices with embedded microcontrollers(PIC) with many different memory for that reduced energy consumption (<1µA) between sensory updates instead of continuously updating neural Network. This paper[5] Introduced Kaburobo's rule, performance and implemented stock Robot such as machine learning in the artificial intelligence field.

IV. Methodology

In this work, the design and Implementation of Intelligent Robot based on Neural Network using ATMEGA8 Microcontroller and a high performance approach is presented that has not been used in this manner before for such design

A. Applied Methodology

The applied methodology is based on the divide and conquers approach. Each block in the architecture was designed and tested separately and later those blocks were assembled and extra modules were added to compose the complete system. The design targeted the mandatory blocks and features and the optional coding blocks were not included.

B. Proposed Work

Fig 5. Shows the basic block diagram of proposed Intelligent Robot based on Neural Network. The input is in the forms of Light and push buttons, and controlling of movement of robot using programming in ATmega8 microcontroller. When we first shine light on the robot from different directions it has no effect on the robot. We then press the button while shining the light from a particular direction on the robot and the neural network programmed into the robot causes it to associate the light input with the push button input according to how it was trained initially. Say, every time we give light from forward direction, we press left button. Soon the robot learns to move in the forward direction when light is shined from the forward direction.

It is decided to model this type of system where there would be multiple layers of Hebbian learning.

- Environment input in form of light flashed from a torch from any of 2 directions(forward, backward)
- Output was in form of motion in 2 directions.

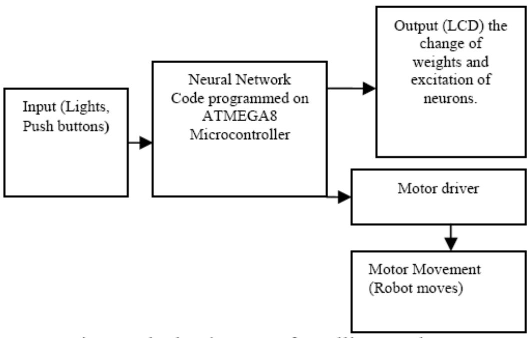

Fig. 5 . Block Diagram of Intelligent robot

- Inhibitory action was hard coded (to keep from strong light)
- A remote had basic controls to move robot while training.
- Training: The input is provided in form of light and simultaneously it is moved in a particular direction via remote to train the robot.
- Once trained, the robot can recognize light from different directions and move according to how it was trained during training period apart from inhibiting strong light.

Fig. 6 is showing the circuit diagram of intelligent robot implemented on ATMEGA8 microcontroller.

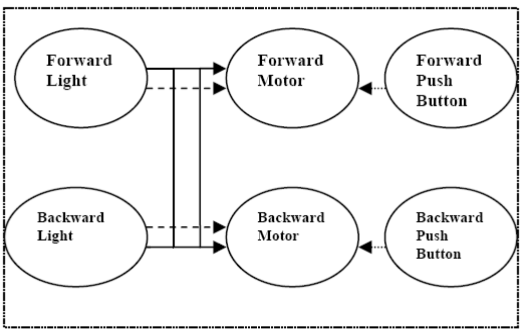

Fig. 6. Circuit diagram of intelligent robot in learning mode.

It was decided to the design process in many logical stages. The first stage: to get the hardware working. Soldering the PCBs and planned the I/O pins and other things. The simple programmers for remote control are designed, LDR usage, timers, LCD etc. After getting the motor control working, integrate this with the neural network programme to make sure that the network was working fine and showing the learning characteristics. Finally, time decaying unlearning, inhibition features and a special function to show the arbitrariness of the simplified output function and how powerful this learning technique is added. Then, shine light from any of 4 directions and show that the robot moves and there exists both excitatory and inhibitory effects. There will also be real-time learning and unlearning as the robot is moving through its light course.

Forward Pushbutton: The input to this neuron is the push button press. It has an excitatory connection to the forward motor neuron with a very large weight since it represents the 'unconditioned stimulus' to the network.

Forward Light: This takes light as input the voltage from the front light sensor. This has an excitatory connection to the forward motor neuron and an inhibitory connection to the backward motor neuron. This is because we decided our neurons can't subtract input directly, so instead we coded it so that the front sensor will excite the front motor neuron and inhibit the back motor neuron.

The backward pushbutton neuron, backward light neuron, and the backward shock light neurons all behave very similarly to the descriptions above but in the opposite directions. Finally, the forward motor output and the backward motor output neurons basically fire depending on which input neurons are firing and the strengths of those connections. After designing the neural network, we had to design the robot and motor controls. The robot is controlled by two gear motors, and they step backwards or forwards depending on which neuron fires. We also decided to build a stand-alone MEGA8 prototype unit which would just be on the robot. Furthermore, a very important part of the project was to demonstrate the learning process of the robot through the neural network.

V. Evolution Model and Implementation

Fig 7. Shows the proposed model of Intelligent Robot. The implementation used here, designed to maximize exploration while preserving the best solution obtained so far.

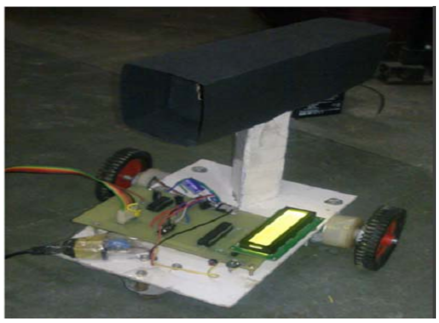

Fig. 7. Model of intelligent robot hardware design consist of

- LCD
- LDR
- Remote
- Motor Driver
- Power Sources

DC +12 V Adapters used as power sources and 7805 was used to power the mcu. To drive the motors, L293 is used since the motors cannot be run directly from microcontroller because it operates on 12 V and also mcu cannot handle so much current. This device, provides enough current and power to control the motor.L293 to control both the motors. The output from ATMEGA goes into the input of the L293. The outputs of the L293 are wired directly into the motors. The following is a schematic of how to use the L293 to drive the motors.

VI. Results

TABLE I shows the results of this proposed approach with the results of Intelligent Robot with learning mode.

If Forward button is off, Reverse button is off for the same sensor values, there is no robot movement (denoted in the table as XX) in any direction in the learning mode, when the learn button is off. If Forward button is on, Reverse button is off for the same sensor values, the robot moves in forward direction in the learning mode, when the learn button is off. If Forward button is off, Reverse button is on for the same sensor values, the robot moves in Backward direction in the learning mode, when the learn button is off.

TABLE I: Results of Intelligent Robot With Learning Mode

Learning mode						
SL.NO	Learn button	Forward Button	Reverse Button	sensor 1	sensor 2	Motion
1	OFF	OFF	OFF	986	988	XX
2	OFF	ON	OFF	986	988	Forward
3	OFF	OFF	ON	986	988	Backward
4	OFF	OFF	OFF	56	988	XX
5	OFF	OFF	OFF	986	64	XX
6	OFF	ON	OFF	58	988	Forward
7	OFF	OFF	ON	60	988	Backward
8	OFF	ON	OFF	986	62	Forward
9	OFF	OFF	ON	61	988	Backward
10	OFF	OFF	OFF	55	988	XX

Last row of table II is the learned mode, both the forward and reverse buttons are off, only learn button is on, robot moves in forward direction the sensor values are 90 and 980. Table III shows the Node potentials in Learn and Learning and Learned modes. When the robot is in the learn mode the potential values at the four nodes is zero. When the robot is in the Learning mode the node values are 65, 73, 80, and 75. When the Robot is in the learned mode the node potential value for node 1 is 155 and other nodes are having zero potential, this indicates the Robot is learned in the forward direction. Fig 8. shows the node potentials for learning mode for Forward and Backward directions. Fig 9. Shows the node potentials for learned mode for Forward direction.

TABLE II: Results of Intelligent Robot with Learned Mode

Learned Mode						
S.NO	**Learn button**	**Forward Button**	**Reverse Button**	**Sensor 1**	**Sensor 2**	**Motion**
1	**ON**	**ON**	**OFF**	**986**	**988**	**Forward**
2	**ON**	**ON**	**OFF**	**65**	**988**	**Forward**
3	**ON**	**ON**	**OFF**	**986**	**73**	**Forward**
4	**ON**	**OFF**	**ON**	**80**	**988**	**Backward**
5	**ON**	**OFF**	**ON**	**986**	**75**	**Backward**
6	**ON**	**OFF**	**ON**	**986**	**988**	**Forward**
7	**ON**	**OFF**	**ON**	**986**	**988**	**Forward**
8	**ON**	**OFF**	**ON**	**986**	**988**	**Backward**
9	**ON**	**ON**	**OFF**	**90**	**980**	**Forward**
10	**ON**	**OFF**	**OFF**	**980**	**980**	**XX**
11	**ON**	**OFF**	**OFF**	**90**	**980**	**Forward**

TABLE III: Results of Intelligent Robot with Node Potentials

		Node Potentials				
S. NO	**Mode**	**Node1**	**Node2**	**Node3**	**Node4**	**Motion**
1	**Learning**	**0**	**0**	**0**	**0**	**Forward**
2	**Learning**	**65**	**0**	**0**	**0**	**Forward**
3	**Learning**	**65**	**73**	**0**	**0**	**Forward**
4	**Learning**	**65**	**73**	**80**	**0**	**Backward**
5	**Learning**	**65**	**73**	**80**	**75**	**Backward**
6	**Learning**	**110**	**73**	**80**	**75**	**Forward**
7	**Learning**	**110**	**115**	**80**	**75**	**Forward**
8	**Learning**	**160**	**115**	**80**	**75**	**Backward**
9	**Learning**	**160**	**0**	**0**	**0**	**Forward**
10	**Learning**	**160**	**0**	**0**	**0**	**XX**
11	**Learned**	**160**	**0**	**0**	**0**	**Forward**

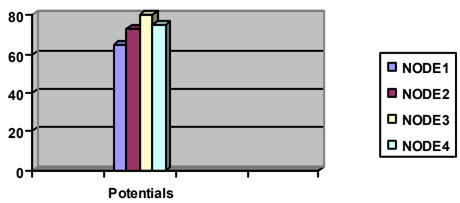

Fig. 8. Node Potentials for Learning Mode

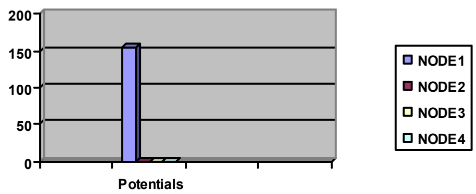

Fig. 9. Node potentials for learned mode

VII. Conclusion

The capability of designing and implementing an Intelligent Robot based on Neural Networks is proposed in this work. Based on above discussion it can be concluded that an Intelligent Robot has been implemented and trained to transport things from one place to another. Goal is to design a neural network that would model some real world phenomena using microcontroller and to learn the usage of microcontrollers and some real word electronics. It is decided to use light and model a moth's behavior via using a robot whose movement was controlled by the outputs of a neural network. Eventually the implementation of artificial intelligence in robot is studied with the help of neural networks which can makes robots to be artificially intelligent. The design implemented was targeted to be mapped on ATMEGA8 Microcontroller. Thus we have shown that the proposed design of an Intelligent Robot was successfully implemented based on Neural Networks. In future work, the design can be modified and enhanced further by adding two more LDR's and Push Buttons and Robot can be made to move left and right direction. Many extra features to the robot such as avoiding an obstacle carrying something. The robot would seem to learn to avoid an obstacle and try imitating the sense organ of human beings and also try to implement various other learning abilities.

References

[1] E. S. Brunette, R. C. Flemmer, and C. L. Flemmer, "A Review of Artificial Intelligence," in *Proc of the 4th International Conference on Autonomous Robots and Agents*, Wellington, New Zealand, pp. 10-12, 2009.

[2] A. H. Klopf and J. **S.** Morgan, "The Role of Time in Natural Intelligence: Implications for Neural Network and Artificial Intelligence Research," *Air Force Wright Aeronautical Laboratories Wright-Patterson Air Force Base*, Ohio 45433-6523.

[3] D. Floreano and C. Mattiussi, "Evolution of Spiking Neural Controllers for Autonomous Vision-Based Robots," in Gomi, T, ed, Evolutionary Robotics. From Intelligent Robotics to Arti cial Life. Tokyo: Springer Verlag.

[4] D. Floreano, N. Schoeni, G. Caprari, and J. Blynel, "Evolutionary Bits'n'Spikes," in *Articial Life VIII*, Standish, Abbass, Bedau (eds) (MIT Press) pp. 335-34, 2002.

[5] J. Wang and F. L. Zhuang, "Study and application of Stock Robot kaburobo based on Artificial Intelligence," presented at *International Joint Conference* on *Artificial Intelligence*, 2009.

Vimal Ganotra born in Karnal, Haryana, India on 25th July1989. He has obtained his Degree in Electronics in 2011 from Kurukshetra University, India. Having interest in System Architectures, he is currently working with Videocon in India

Rashmi Sharma born in New Delhi, India on 3rd December 1990. She has obtained her Degree in Electronics & Communication in 2012 from Maharshi Dayanand University, India. Having interest in wireless communication and neural networks, she is currently working under XBee based project.

Priti Dhingra born in New Delhi, India on 4th December 1988. She has obtained her Degree in Electronics & Communication in 2011 from Kurukshetra University, India. Having interest in machine learning and biped locomotion, she is currently working with Wipro Technologies, India.

Sourabh Sharma born in New Delhi, India on 25th July 1989. He has obtained his Degree in Electronics & Communication in 2011 from Kurukshetra University, India. Having interest in Artificial Intelligence and machine learning. He has many national level awards in robotics. Currently working on government projects based on underwater robotics.

Navin Kumar born in New Delhi, India on 29th October 1991.He is an undergraduate student of Mahamaya University, India. Having interest in robotics, he is currently working under GPS navigation system project.