Pattern Memory and Analysis in Bat-Inspired Echolocation Systems

MERIT FAIR - BIEN 2010
Patrick B. Ellis
Advisors: Tarek M. Massoud, Prof. Timothy Horiuchi

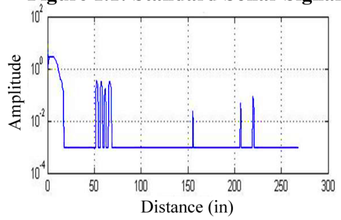
The Objective

To build a system that uses a neural network to locate, learn, and identify different objects in an environment through use of echolocation. This system will be written in MATLAB® then implemented onto an FPGA. It will eventually be used as a calibration signal for a larger system.

The Inspiration

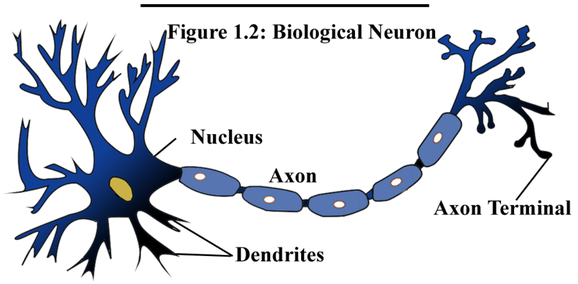
One of the goals is to recognize how many entities are being viewed. Figure 1.1 depicts a typical sonar reading. Initially it seems there are 8 entitities, but in fact there are 4.

Figure 1.1: Standard Sonar Signal



This tricky circumstance is caused by sound waves reflecting off of the 4 entities. Bat-inspired techniques of deciphering sonar can help understand this and other problems.

Figure 1.2: Biological Neuron



~100 billion neurons are thought to exist in the human brain - each receiving, processing, and sending short voltage pulses. The dendritic tree receives and integrates the charge given from many other neurons. If the integrated charge exceeds a threshold, the neuron "fires" and sends a voltage pulse to other neurons. This is what the neural network in our system seeks to replicate.

The Neural Network

- A single layer Neural Network
- 60 Neurons in total
- 15 Neurons per object, each watching a range of 3.7 inches

What are Objects?

An "Object" is a set of entities (like number of poles) that the Neural Network looks to identify.

Figure 2.1: Object Chart

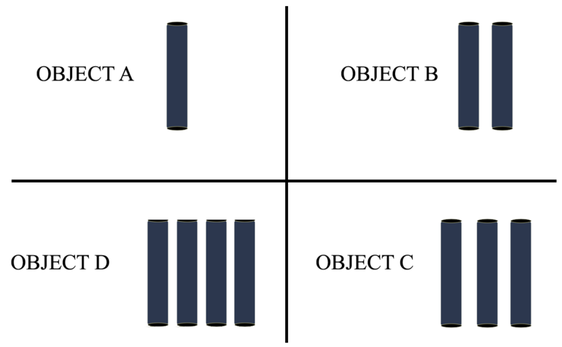
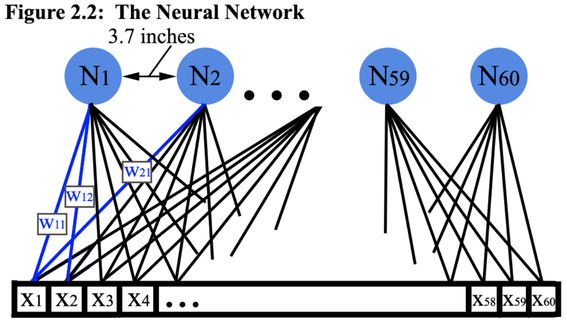


Figure 2.2: The Neural Network



$$N_{output} = \sum_{i=1}^{60} W_i X_i$$

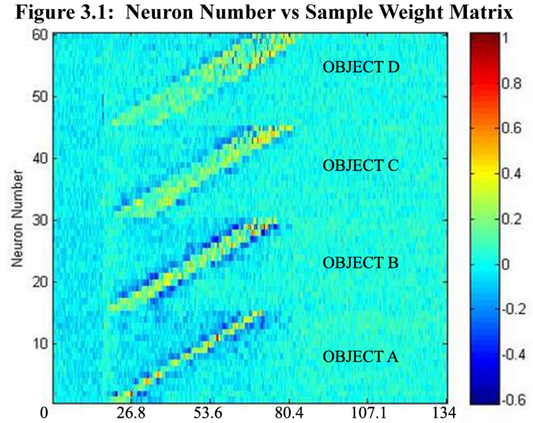
The 60 Neurons process the input amplitude values (X's) through a weight matrix (set of W's) producing an output from each neuron.

The numeric order of the neuron with the largest number will tell which of the 4 objects is being "seen."

Implementation

First Step: Entire System on MATLAB®

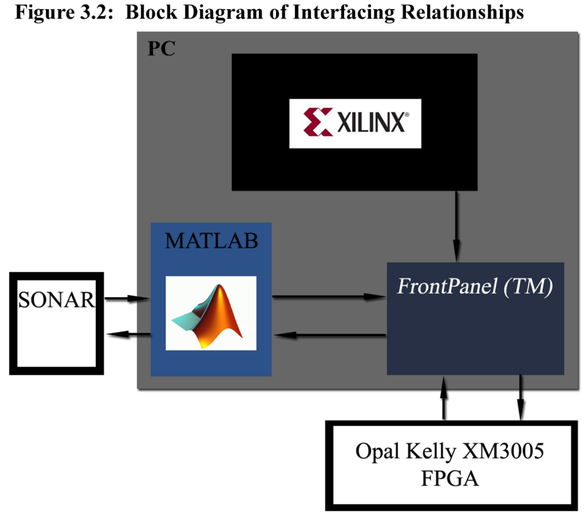
Figure 3.1: Neuron Number vs Sample Weight Matrix



- Demonstrates 15 neurons for the 4 objects at different distances.
- Represents successful catagorization 80% of the time with only 1 million iterations.

Second Step: Implementing onto the FPGA

Figure 3.2: Block Diagram of Interfacing Relationships



The XEM3005 comes with software called FrontPanel(TM) which has libraries that allow an API such as MATLAB® powerful control over experimentation.

Conclusion & Future Work

Completed:

- The system was completed on MATLAB®
- Complete interfacing between MATLAB,® Xilinx, and FrontPanel.(TM)
- The schematic shown in Figure 3.2 was accomplished with full functionality of neural network with small matrix sizes but not the dimensions sought.
- Three separate general approaches of storage were used:
  1. 1-Dimensional Arrays
  2. 2-Dimensional Arrays
  3. Case Statements with 1D arrays

Future Work:

- Work on minimized storage of values.
- Optimization and Refining Needed:
  - Less Neurons - less data to store
  - Less Samples - less data to store
  - More Iterations - adds higher percentage of correct readings of objects.
- Full Neural Functionality on FPGA

Acknowledgments

I would like to thank all the assistance and guidance I have received from my mentors Professor Horiuchi and Tarek Massoud, the MERIT program, the Computational Sensormotor Systems Lab, and National Science Foundatation.

References

1. J. Hertz, A. Krogh, R. Palmer, *Introduction to the Theory of Neural Com putation*, Redwood City, CA: Addison-Wesley Publishing Company 1991.
2. K. Mehrotra, C. Mohan, S. Ranka, *Artificial Neural Networks*, 2nd Edition, London, England: Bradford Book 2000.

Bat Picture (upper left): Acquired from nanowrimo.org.
Neuron Picture (bottom left): Acquired from commons.wikimedia.org.