An Analog Neural Network Processor and its Application to High-Speed Character Recognition

Bernhard E. Boser, Eduard Säckinger, Jane Bromley,
Yann LeCun, Richard E. Howard, and Lawrence D. Jackel

AT&T Bell Laboratories
Crawford Corner Road, Holmdel, NJ 07733

***Abstract*—A high-speed programmable neural network chip and its application to character recognition are described. A network with over 130,000 connections has been implemented on a single chip and operates at a rate of over 1000 classifications per second. The chip performs up to 2000 multiplications and additions simultaneously. Its datapath is suitable particularly for the convolutional architectures that are typical in pattern classification networks, but can also be configured for fully connected or feedback topologies. Computations are performed with 6 Bits accuracy for the weights and 3 Bits for the states. The chip uses analog processing internally for higher density and reduced power dissipation, but all input/output is digital to simplify system integration.**

Introduction

Learning from example and the ability to generalize are two features that make neural networks attractive for pattern recognition applications. However, the computational requirements, data rates, and size of neural network classifiers severely limit the throughput that can be obtained with networks implemented on serial general purpose computers. Better performance is achieved with special purpose VLSI processors that employ parallel processing to increase the processing rate.

Speed and data rates are not the only challenges faced by specialized hardware designs for neural networks. Because of the rapid progress of neural network algorithms, processors must be flexible enough to accommodate a wide variety of neural network topologies. Moreover, the size of neural networks is increasing steadily. Networks with several ten or hundred thousand connections are typical for high-accuracy pattern classifiers [1, 2], and this number is expected to grow further. To be economical, such networks must be implemented on a small number of chips. Moreover, the high-performance parallel-computing unit must be matched with an equally powerful interface to avoid bottlenecks.

In this paper, the architecture and implementation of a special purpose neural network chip that addresses these issues are described. The circuit uses analog processing internally to exploit the low resolution requirements typical of neural network, but employs an all digital external interface to simplify system integration. The practicality of the design is illustrated with results from an implementation of a neural network for handwritten optical digit recognition with over 130,000 connections. The entire network fits on a single chip and is evaluated at a rate in excess of 1000 characters per second.

0-7803-0164-1/91/0000-0415$01.00©1991 IEEE

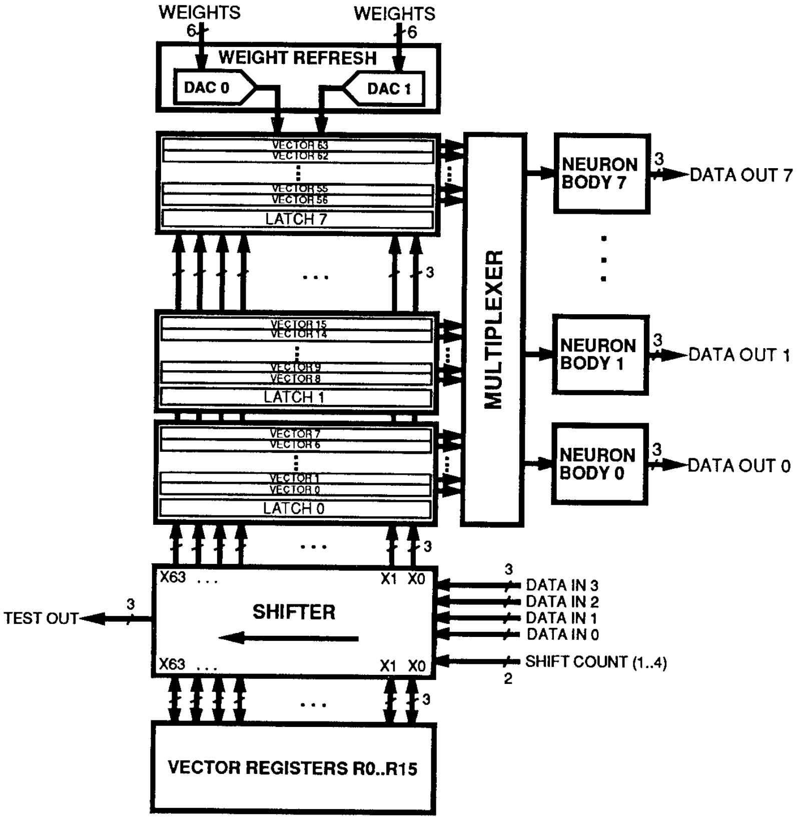

Figure 1. Block diagram of the neural network chip.

Hardware Description

Figure 1 shows the building blocks of the neural network chip. Its function is to evaluate concurrently several dot products of state and weight vectors and apply a nonlinear squashing function to the results. Data enters the chip through a 64 state (3 Bit word) deep shift register, that reads up to four values at a time. A file with 16 vector registers is used to extend the length of the input vector to more than 64 states, and as a buffer to store intermediate results when multi-layer networks are evaluated.

The actual computation is performed by eight banks of vector multipliers. Each bank consists of a latch to hold the state vector, and eight vector ALUs with 64 synapses each. The outputs from the vector multipliers are routed to the neuron bodies by a multiplexer that can be configured to combine the contributions from one to four vector multipliers. This feature, along with appropriate programming of the shifter and register file, allows the configuration of the chip to be set to extremes of 16 neurons with 256 inputs each, or 256 neurons with 16 inputs, as well as many intermediate arrangements [3]. The topology can be rearranged on a per instruction basis to permit evaluation of several layers of a network with different architectures on a single chip without performance penalty.

The neuron bodies first scale the output from the vector multipliers by a factor that can be set in the range 1/8 to 1 in eight levels to optimize the useful dynamic range of

Synapses	4096
Bias Units	256
Synapses per Neuron	16 to 256
Weight Accuracy	6 Bits
State Accuracy	3 Bits
Input Rate	120 MBits/sec
Output Rate	120 MBits/sec
On-chip Data Buffers	4.6 kBits
Computation Rate (sustained)	5 GC/sec
Refresh (all weights)	110 μs
Clock Rate	20 MHz

Table 1
System features.

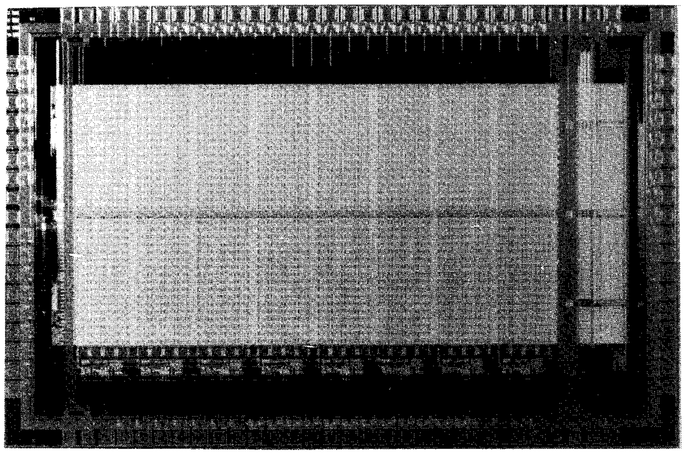

Figure 2. Die photograph. The synapse array can be seen in the center, the shifter and register file on the left, the neuron bodies at the top, and the weight refresh DACs on the right.

the circuit. Then the squashing function is evaluated and the result converted to the same 3 Bit signed magnitude representation as is used at the input of the chip.

The weights in the vector multipliers are stored as charge packets on capacitors and must be refreshed periodically. Two on-chip DACs update the values of two different synapses in each clock cycle for a refresh speed of 110 μs for the entire array.

The chip executes three instructions, CALC, SHIFT, and STORE, to perform computations, load data from an external data source, and to transfer data between the shifter, register file, and vector multiplier banks. The CALC instruction takes four cycles of 50 ns, the other two operations execute in a single clock cycle concurrent with an ongoing CALC instruction. In 200 ns the chip can, for example, load eight states and store them in a reg-

Network Topology	Average Performance
Fully Connected (single layer)	
64 inputs, 64 outputs	2.1 GC/sec
128 inputs, 32 outputs	1.2 GC/sec
32 inputs, 128 outputs	1.2 GC/sec
Local Receptive Fields	
64 × 1 receptive field, 64 features	2.3 GC/sec
16 × 16 receptive field, 16 features	4.7 GC/sec
16 × 8 receptive field, 32 features	3.6 GC/sec
Multi-Layer Network	
64 inputs, 32 hidden, 32 hidden, 32 outputs	0.8 GC/sec
Hopfield Neural Network	
64 neurons	2.1 GC/sec

Table 2
Sample network architectures and performance.

ister and two latches, and evaluate the dot product and nonlinear function of eight vectors with 256 components each. The weight refresh is performed in parallel and is transparent to the user. Table 1 summarizes the features of the chip.

The chip contains 180,000 transistors and measures $4.5 \times 7\,\text{mm}^2$ (Figure 2). It was fabricated in a single poly, double metal $0.9\,\mu$m CMOS technology with 5 V power supply. The current drawn by the chip reaches 250 mA when all weights are programmed to their maximum value, but is less than 100 mA in typical operation.

Programmability is one of the key features of the neural network chip. Table 2 lists a selection of network topologies that can be implemented and the achieved performance in each case. The chip processes networks with full or sparse connection patterns of selectable size, as well as networks with feedback at a sustained rate of over 10^9 connection updates per second.

Of particular importance for neural network pattern classifiers are neurons with local receptive fields and weight sharing, such as TDNNs [4]. In those architectures, several neurons with identical weights process different parts of the network inputs or hidden units. The neural network chip supports weight sharing in several ways: The shifter and register file enable loading of data and the computation to go on in parallel. Also, data that has been loaded on the chip once, can be buffered and reused in a later computation. Finally, rather than requiring separate hardware for all weights, neurons with identical parameters can be multiplexed.

High-Speed Character Recognition Application

An optical character recognition neural network has been selected to test and demonstrate the flexibility and power of the neural network chip [1]. The OCR identifies handwritten digits from a 20 by 20 pixel input image and employs neurons with local receptive fields as well as a fully connected layer. Overall, the network contains more than 136,000

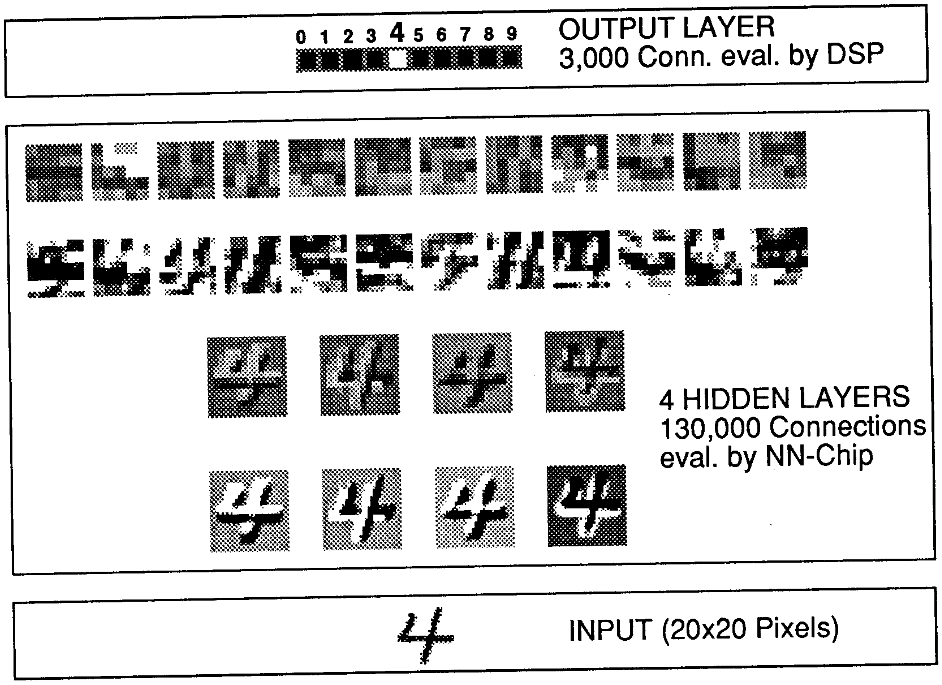

Figure 3. Example chip output for optical character recognition. The gray levels encode the neuron state.

connections organized in five layers.

Special steps are necessary to adapt the network to the low resolution of the chip. The network is trained on a workstation with floating point arithmetic using the backpropagation algorithm. Simple quantization of all weight values results in an unacceptable loss of accuracy. However, experiments reveal that the computational accuracy provided by the chip is adequate for all but the 3000 weights in the last layer of the network. Higher accuracy is needed in that layer to permit selective rejection of ambiguous or otherwise unclassifiable patterns.

The first four layers of the network with 97 % of the connections fit on a single neural network chip. The remaining 3000 connections of the last layer are evaluated on a DSP32C digital signal processor. The throughput of the chip is more than 1000 characters per second or 130 MC/sec. This figure is considerably lower than the peak performance of the chip (5 GC/sec.), a consequence of the small number of synapses of most neurons in the network for which the chip cannot fully exploit its parallelism. Nevertheless, the chip's performance compares favorably to the 20 characters per second that are achieved when the entire network is evaluated on the DSP32C. The recognition rate of the chip is far higher than the throughput of the preprocessor, which relies on conventional hardware.

The network was tested on 2000 handwritten digits provided by the U.S. postal service. A sample input, and the states of the outputs and hidden layers are shown in Figure 3. The classification error rate is 5.0 % errors when network weights and states are not quantized. After quantization, the error rate increases to 5.8 %. On the chip, a performance of 7.0 %

has been measured.

These results can be improved in several ways. For example, the effects of quantization can be reduced by retraining the last layer of the quantized network. However, the true benefits of the parallel hardware will be realized only when the topology of the network is adapted to explicitly take advantage of the chip's throughput. While the size of the current network has been constrained by the speed of conventional hardware, such issues loose importance when powerful special purpose processors are used.

Conclusions

Neural networks are attractive for pattern classification applications but suffer in practice from the limited speed that can be achieved with implementations based on classical processors. This problem can be overcome with highly parallel special purpose VLSI circuits. While a fully parallel implementation of sufficiently large networks is currently not feasible, adequately high performance can be achieved with an architecture that exploits the limited connectivity and weight sharing that are typical for pattern classifiers. This has been demonstrated with a neural network classifier with over 130,000 connections that has been implemented on a single neural network chip performing over 1000 classifications per second. This result eliminates throughput from the constraints faced by network designers. The availability of fast special purpose hardware for large applications sets the conditions to explore new neural network algorithms, and problems of a scale that would not be feasible with conventional processors.

Acknowledgement

The authors express their thanks to the U. S. Army SCD for their partial support for this project.

References

[1] Yann Le Cun, Bernhard Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne Hubbard, and Larry D. Jackel. Handwritten digit recognition with a back-propagation network. In David S. Touretzky, editor, *Neural Information Processing Systems*, volume 2, pages 396–404. Morgan Kaufmann Publishers, San Mateo, CA, 1990.

[2] Isabelle Guyon, P. Albrecht, Yann Le Cun, John S. Denker, and Wayne Hubbard. A time delay neural network character recognizer for a touch terminal. Technical report, 1990.

[3] H. P. Graf and D. Henderson. A reconfigurable CMOS neural network. In *ISSCC Dig. Tech. Papers*, pages 144–145. IEEE Int. Solid-State Circuits Conference, 1990.

[4] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang. Phoneme recognition using time-delay neural networks. *IEEE Trans. Acoust., Speech, Signal Processing*, pages 328–339, March 1989.