A Road-Following Computational-Sensor Prototype

Andrew Gruss

Computer Science Department
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, Pennsylvania 15213-3890

Phone: (412) 268-5622, *FAX:* (412) 268-5571
EMail: <gruss@cs.cmu.edu>

Abstract *— This paper describes the ongoing development of an intelligent VLSI imaging sensor for road following. This sensor generates heading information required to steer a robotic vehicle by "watching" the road. On-chip processing is based on ALVINN (Autonomous Land Vehicle In a Neural Network), a neural network trained to drive without human intervention on public highways. Circuitry for neural computations is integrated with a photosensor array using VLSI in order to directly sense and process road-image information.*

A prototype chip has been fabricated. It consists of a 5 × 5 array of sense-and-process cells which perform the early neural processing needed by the ALVINN algorithm. In this paper, I discuss the design and implementation of this computational sensor and present experimental results obtained from the prototype.

1 Introduction

VLSI technology provides the opportunity to integrate raw sensing with processing. The result is a new robotic sensor methodology — *computational sensing*. A computational sensor intelligently extracts information from transduced data at the point of sensing. The bottleneck between transducer and computer, present in traditional sensing system implementations, is eliminated.

In this research, an intelligent imaging sensor for road following is being developed. The chip integrates photosensing with ALVINN, a neural-network architecture which assists an autonomous vehicle to drive over single and multi-lane roads[7]. Image information, acquired using a video camera, is processed by the neural network to generate the vehicle heading required to stay on the road (Fig. 1 & Fig. 2).

An implementation of ALVINN running in software on SUN4 workstations has successfully driven the CMU NAVLAB I and NAVLAB II HMMWV (High-Mobility Multi-purpose Wheeled Vehicle) at 55 miles/hour speeds for distances exceeding 20 miles. ALVINN demonstrates

This research is supported by the National Science Foundation, under grant MIP-9305494.

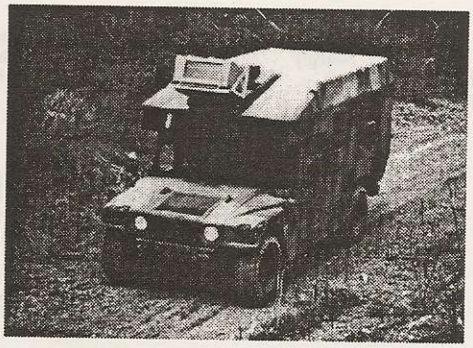

Fig. 1: The HMMWV testbed.

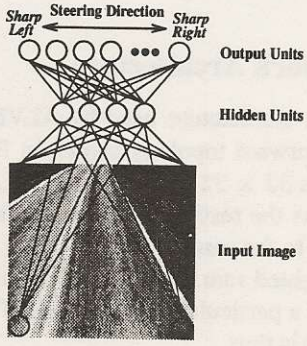

Fig. 2: The ALVINN network.

that neural networks are suitable for performing the image processing required by complex, real-time robotic tasks.

In this research, the functionality and learning ability of the ALVINN neural network is mapped onto VLSI and combined with a means to sense light. Neural processing is done directly on output from photoreceptors onto which a scene is imaged. Research issues include:

- how to map the neural-net architecture into silicon,
- how to inject intensity image data directly into parallel neural-network processing circuitry, and
- how to train the system.

The integration of transducing and computation is not a new idea. Perhaps the best known work is that done by Carver Mead and his group at Caltech[5, 8], though many other examples can be found[1, 3, 9, 10]. A good example of a practical task-oriented sensor is the light-stripe range sensor[4].

2 The ALVINN Network

ALVINN is a connectionist system for processing image data. Patterns in input images generate an output response that has been learned through training. For an autonomous

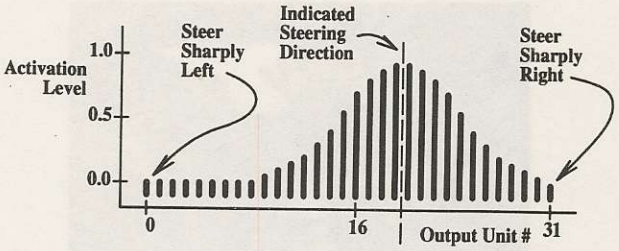

Fig. 3: Representation of generated steering direction.

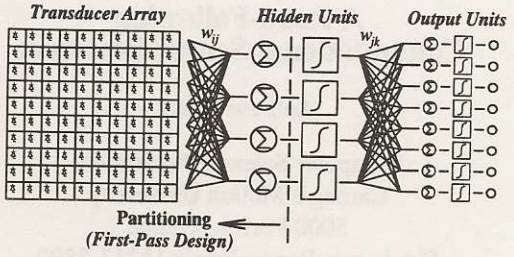

Fig. 4: Partitioning the ALVINN task for VLSI.

vehicle, the output from the system is interpreted as a steering direction which keeps the vehicle on the road.

2.1 Network Architecture

The network architecture used by ALVINN is the two-layer, feed-forward topology shown in Fig. 2. The first layer weights 32 × 32 inputs I_i ($i = 0..1023$) with values w_{ij}, sums the results, and then applies an *activation* function $\alpha()$ to the result. The activation function thresholds the weighted sum in order to determine the effect on the output of a particular network path. The output H_j of a hidden unit is thus

$$H_j = \alpha\left(\sum_{\text{image}} w_{ij} I_i\right) \qquad (1)$$

ALVINN uses $j = 1..4$ hidden units in the intermediate layer. Hidden unit outputs are processed by the second layer in a similar manner to generate outputs O_k.

In both layers, each connection has an associated weight or *connection strength* w_{ij}. Connection strengths are inhibitory (negative) as well as excitory (positive). The capability to provide one constant (thresholding) input is also required.

2.2 Output Representation

The continuous outputs O_k ($k = 0..31$) encode the decision on vehicle heading made by the neural network. ALVINN is trained to produce a Gaussian representation of output heading, as illustrated in Fig. 3. In operation, the output activation levels O_k are converted into a steering direction by fitting a Gaussian of the width specified during training. The peak position of the best-fit Gaussian along the output vector is the vehicle heading inferred by the network.

This multiple-output representation of steering direction allows the network to indicate competing responses. The system in which the network is embedded can judge the best course of action in the case of a conflict. For example, consider an input image of a fork in the road. The network can respond to the input with an output distribution which indicates two peaks, one for taking the left fork and the other for taking the right fork. Intelligence external to the network can be invoked at this point to resolve the conflict, perhaps by consulting a map. If a single continuous output had been used, one value would have been produced, possibly the average of the two in this example. Therefore, no indication of the conflict could be given.

3 An ALVINN-On-A-Chip Design

The integration of an entire ALVINN network on a single chip is this project's ultimate goal. Currently, ALVINN neural processing is performed on digitized video images by an attached workstation. This implementation is bulky and expensive. The use of standard video cameras and frame buffers introduces latency that limits performance. A VLSI implementation permits compact, low-cost ALVINN driving systems to be deployed. Performance is enhanced through the elimination of system latency.

3.1 Partitioning the Computation

The computational and bandwidth requirements of an imaging connectionist architecture like ALVINN are highest in the input layer — between photosites and hidden units. A natural point at which to partition the computation is just inside the hidden units as illustrated in Fig. 4. There a straightforward linear computation, between transducers and hidden units, distills information from many photodetectors.

The bulk of the processing done by a neural network is matrix multiplication. Each hidden unit sums its weighted inputs $w_{ij} I_i$ and then applies the activation function $\alpha()$ to the result (Eqn. 1). The summing operation reduces data from many input connections to a small number of values — one for each hidden unit. In the ALVINN architecture, a 32 × 32 image (1024 inputs) is represented as four hidden unit values. On a VLSI chip, the weighted-sum computation can be done directly on photocurrent inputs.

152

3.2 Intensity Readout

The ALVINN learning algorithm must access the raw input information in order to converge on a set of weights. Typically, the learning system has direct access to the camera images processed by the network. However, a VLSI computational sensor processes sensor input locally. Raw sensor data need not be available at the chip interface. To facilitate training, each cell includes additional circuitry that permits intensity information to be read directly.

3.3 Weight Representation and Storage

An analog representation of data values is used to reduce die area required for computations. Storage elements for analog values are compact. In addition, output values can be transmitted via a single output pin. Compact storage is especially important because each connection in the neural network requires a distinct weighting value. To implement the fully-connected, four hidden-unit design, eight weights (four positive, four negative) must be provided at each photosite.

Capacitors provide a means for compact storage of trainable weight values. The 2 μm n-well BiCMOS fabrication process available to us provides a second poly layer out of which very high quality capacitors are made. With precise control of geometry, tight capacitance tolerances can be specified and maintained. In addition, weights held on capacitors are readily changed as a part of chip training procedure.

However, values held as charge on capacitors are not permanent. Stored charge leaks away over time and must be refreshed periodically. Therefore, input image information must be processed in discrete time, to allow for periodic refreshing of weights. While undesirable, the need to refresh these values is not a severe limitation. ALVINN, as currently implemented in software, operates in discrete time. Successful vehicle control has been demonstrated at 10 frames/second input frame rates. Weights can be easily refreshed, between each frame if necessary, at this processing rate.

Nonvolatile memory (NVSD) technology is a mechanism for analog storage which does not require refreshing. However, such devices also have drawbacks. It is difficult to control the amount of charge injected and thus the exact value held. Complex control circuitry is required. In addition, the number of write cycles is limited. Despite its drawbacks, nonvolatile weight storage is important for a practical implementation. NVSD elements have been successfully implemented in fabrication technologies similar to that available to us[2]. The use of such structures will be investigated in future prototypes.

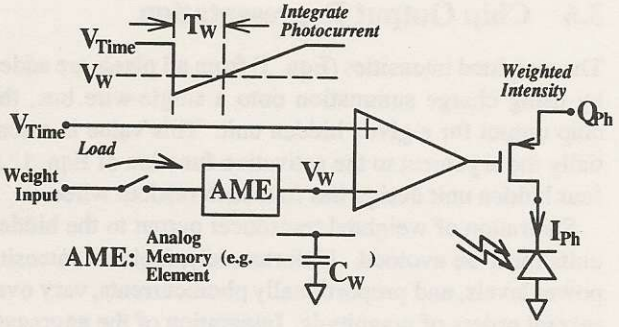

Fig. 5: Photocurrent weighting.

3.4 Area-Efficient Weighting

The primary computation performed within each cell is multiplication of incident intensity by stored weights. The area of the circuitry required to perform the multiplication must be minimized in order to keep cell area small.

At a given pixel, weighting voltage $\mathbf{V_W}$ (analogous to w_{ij} in Eqn. 1, where index i is a pixel photosite) is loaded from off chip and held constant during a frame. $\mathbf{V_{Time}}$ is a linearly-increasing ramp function whose slope is determined by the desired frame rate. This is adjusted according to the overall scene brightness. The integration time $\mathbf{T_W}$ is determined by $\mathbf{V_{Time} = V_W}$.

The product of the weight and input signal at a single cell is equal to photocurrent integrated over $\mathbf{T_W}$

$$\mathbf{Q_{Ph}} = \int_{\mathbf{T_W}} \mathbf{I_{Ph}}\, \mathrm{dt}. \qquad (2)$$

As a result, multiplication is performed through integration of photocurrent over a period of time proportional to the programmed weight.

3.5 Excitory and Inhibitory Weights

The circuit of Fig. 5 is excitory only, but both excitory and inhibitory weights must be supported. Excitory and inhibitory weights are provided using a "push-pull" scheme (shown in the cell schematic of Fig. 6). Current mirrors generate equal source and sink currents proportional to the sensed intensity. Two values are required — one which indicates the strength of the excitory stimulus and the other for the strength of stimulus inhibition.

The weighted intensity output, in term of the total output charge $\mathbf{Q_{Ph}}$ as a function of the positive weighting time $\mathbf{T_{Wp}}$ and the negative weighting time $\mathbf{T_{Wn}}$ for this circuit is

$$\mathbf{Q_{Ph}} = \int_{\mathbf{T_{Wp}}} \mathbf{I_{Ph}}\, \mathrm{dt} - \int_{\mathbf{T_{Wn}}} \mathbf{I_{Ph}}\, \mathrm{dt}. \qquad (3)$$

Its value may be positive, negative, or zero depending on the relative strengths of excited and inhibited connections.

3.6 Chip Output Representation

The weighted intensities (Eqn. 3) from all pixels are added by using charge summation onto a single-wire bus, the chip output for a given hidden unit. This value is essentially the argument to the activation function of Eqn. 1. A four hidden unit design has four such readout wires.

Saturation of weighted transducer output to the hidden units must be avoided. Unfortunately, incident intensity power levels, and proportionally photocurrents, vary over several orders of magnitude. Integration of the aggregate weighted photocurrents is subject to saturation. The mechanism used in the prototype to avoid saturation is to output the aggregate weighted photocurrent directly from the chip. One such output for each supported hidden unit is required. The resulting current profile can be directly digitized and then integrated numerically in the host computer.

3.7 Operation

The sensor processes intensity information a frame at a time. While $\mathbf{V_{Op}}$ is asserted and $\mathbf{V_{Time}}$ is active (Fig. 6), the chip outputs, for each hidden unit, the instantaneous sum of weighted photocurrents from each cell. Simultaneously, individual cells measure and record the incident light they observe. If an intensity image is desired, it is read from the chip in a separate operation before the next frame is begun.

At some point, cell weights must be loaded into the chip. If NVSD analog memory elements are used, this need be done only once when the chip is trained. Capacitor-based storage, however, requires that weights be loaded frequently, perhaps at the beginning of each frame. Loading of values onto capacitors can be done quickly, however, and this should not lengthen the time between frames excessively for arrays of 1024 cells.

4 Prototype Chip

The prototype chip consists of a 5×5 array of photosensitive cells which perform the neural computation. Support for one hidden unit is provided. Multiple hidden units will be supported in future versions through replication of circuitry within each cell (as shown in Fig. 6). Raw intensity image readout is also provided to facilitate training. Once the prototype sensor is functional, a complete first-stage ALVINN architecture (32×32 pixels each supporting 4 hidden units) will be built by replicating the number of cells and the hidden units within each cell.

4.1 Cell Implementation

Fig. 6 shows the circuitry implemented in the first-pass sensing cell. Important features of the cell include

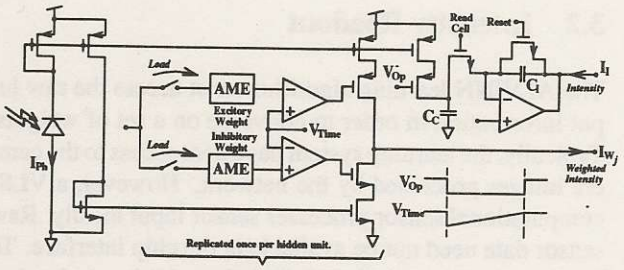

Fig. 6: An ALVINN-on-a-chip cell implementation.

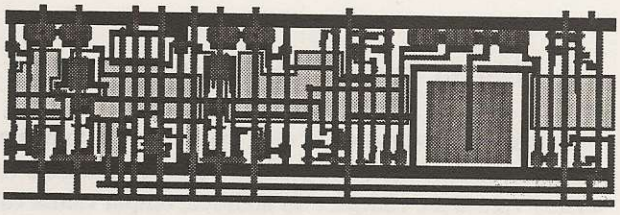

Fig. 7: Prototype ALVINN pixel.

- an integrated photodetector,
- photocurrent buffering, replication and negation,
- per-hidden-unit weighted intensity output, and
- unweighted cell intensity output.

Circuitry in the upper-right corner of Fig. 6 measures and records intensity information seen by the cell during each frame. Readout of this information from the chip is done through a time-multiplexed bus. The "**Read Intensity**" signal provides the means for gating the charge held in the intensity circuit onto the readout bus.

4.2 Cell Layout

Fig. 7 is a plot of the prototype cell currently under test. The cell measures 120 μm $\times$ 374 μm and has been built using a 2 μm CMOS double-metal, double-poly process provided by MOSIS. A well-substrate diode is used as the photoreceptor. It is approximately 3200 μm^2 in area and can be seen on the right side of the cell.

4.3 Pixel Response

The relative response of a prototype cell was determined using a set of neutral-density filters. The result of one experiment is plotted in Fig. 8. In this experiment, the sensor was placed approximately one meter from a white, diffuse surface which reflected back typical room ambient lighting. A 25mm, $f1.4$, 1"–format lens focused light onto the chip. For each transmittance value, as determined by the interposed neutral-density filter, photocurrent was integrated on a 0.68 pF capacitor within the cell. The time required to change the capacitor voltage by 3 volts was

154

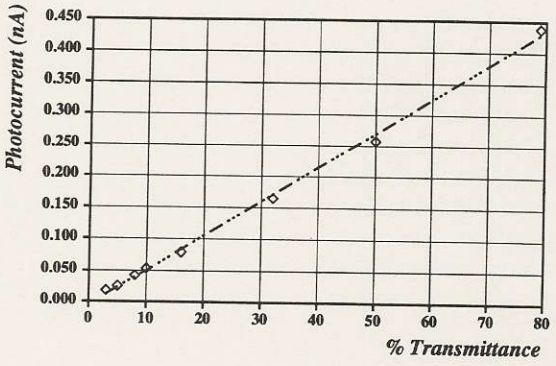

Fig. 8: Cell intensity readout relative response.

recorded. Fig. 8 shows the resulting (and expected) linear relationship between light intensity reaching the sensor and photocurrent.

5 Conclusion

This paper has described a prototype sensor which integrates sensing and processing required by the ALVINN road-following neural network. Using VLSI, the computation required by the early stages of the neural algorithm is done on directly-sensed image information. Latency is reduced and frame processing rates are increased.

A prototype ALVINN chip has been fabricated and is being tested. Early results which show the sensitivity of this chip to light were presented.

ALVINN-on-a-chip will continue to evolve. In the near term, the design will be scaled up to support first-level processing for the full 1024 pixel, four hidden unit ALVINN architecture. Next, activation function and second layer processing through to the output layer will be added. In the final phase, the design will incorporate the learning algorithm for training connection weights.

References

[1] Philippe Aubert, Henri J. Oguey, and Raymond Vuilleumier. Monolithic optical position encoder with on-chip photodiodes. *IEEE Journal of Solid-State Circuits*, 23(2):465–473, April 1988.

[2] David A. Durfee and F. S. Shoucair. Comparison of floating gate neural network memory cells in standard VLSI CMOS technology. *IEEE Transactions on Neural Networks*, 3(3):347–353, May 1992.

[3] Ralph R. Etienne-Cummings, Sanath A. Fernando, and Jan Van der Spiegel. Real-time 2–D analog motion detector VLSI circuit. In *Proceedings of the International Joint Conference on Neural Networks*, Baltimore, MD, 7 June 1992. IEEE.

[4] Andrew Gruss. *A VLSI Smart Sensor for Fast Range Imaging*. PhD thesis, Carnegie Mellon University, 13 November 1991.

[5] Carver A. Mead. Adaptive retina. In Carver A. Mead and Mohammed Ismail, editors, *Analog VLSI Implementation of Neural Systems*, chapter 10, pages 239–246. Kluwer Academic Publishers, Boston, MA, 1989.

[6] Hans Melchior, Mahlon B. Fisher, and Frank R. Arams. Photodetectors for optical communication systems. *Proceedings of the IEEE*, 58:1466–1486, October 1970.

[7] Dean A. Pomerleau. *Neural Network Perception for Mobile Robot Guidance*. PhD thesis, Carnegie Mellon University, 16 February 1992.

[8] Massimo A. Sivilotti, Michelle A. Mahowald, and Carver A. Mead. Real-time visual computations using analog CMOS processing arrays. In Paul Losleben, editor, *Advanced Research in VLSI — Proceedings of the 1987 Stanford Conference*, pages 295–312. The MIT Press, 1987.

[9] D. L. Standley and B. Horn. An object position and orientation IC with embedded imager. In *Proceedings of the IEEE International Solid State Circuits Conference*, pages 38–39, San Francisco, CA, 13–15 February 1991.

[10] J. L. Wyatt, Jr., D. L. Standley, and W. Yang. The MIT Vision Chip Project: Analog VLSI systems for fast image acquisition and early vision processing. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pages 1330–1335, Sacramento, CA, April 1991.