A micro-sensor based on insect vision

A. Yakovleff*, A. Moini, A. Bouzerdoum, X.T. Nguyen, R.E. Bogner, K. Eshraghian, D. Abbott

Electrical & Electronic Engineering Department
and Centre for Gallium Arsenide VLSI Technology
University of Adelaide
Adelaide, SA 5005, Australia

Abstract - **A major problem faced by artificial vision systems is the computational bottleneck. The problem can be addressed by incorporating preliminary parallel signal processing in the sensor itself, thereby alleviating subsequent processing requirements.**

This paper describes a vision system implemented in analog and digital VLSI incorporating sensors and a processing unit. It extracts information such as range, depth and motion of objects in the visual environment. Advantages are its small size and its wide dynamic range which make it very suitable for real-life applications, particularly in robotics.

Key words - **Vision processing, insect vision, VLSI, sensing, micro-sensor, template.**

I. Introduction

Significant progress has been made in the theory of early visual processing. As a result, a large number of algorithms have been proposed that locate and extract edges, compute disparities, estimate motion flow fields, and find discontinuities in depth, motion, colour and intensity. However, the application of these algorithms to real-world vision problems has yielded limited success only.

The main problem is that the computational cost associated with these early vision algorithms prevents real-time machine vision implementations on anything but large-scale expensive digital computers. Most of the proposed mechanisms are formulated as relaxation algorithms which need to be iterated many hundreds if not thousands of times before a solution is reached.

Thus, one way for technology to bypass the computational bottleneck is to construct compact analog special-purpose vision hardware, whereby the complexity of vision is neutralized by the use of many processing circuits in parallel. This so-called "smart sensor paradigm", in which as much as possible of the signal processing is incorporated into the sensor and its associated circuitry in order to reduce transmission bandwidth and subsequent stages of computation, is starting to emerge as a possible competitor to more general purpose digital machines.

The work of John Tanner and Carver Mead at Caltech on the *correlating optical motion detector* in 1984 [1], was inspirational and has led to several VLSI implementations of other motion detection algorithms. Unfortunately, most implementations were based on the spatio-temporal gradient models of local motion computation [2][3], which are computationally intensive and inherently noise sensitive; the spatio-temporal gradient models are concerned with an accurate estimation of the velocity, which in turn requires an accurate estimate of the spatial and temporal derivatives of the intensity function. The result is that all the implementations were sensitive to light intensity; as the light intensity dropped, the estimate of the velocity deteriorated quickly.

Another problem hampering the success of artificial vision systems is that most of these systems, in particular computer vision systems, have been concerned with the construction of a symbolic internal model for the structure of the external world. Thus, a lot of effort is spent searching for the transformation which maps the visual stimulus into the internal world model. However, this approach faces a number of difficulties, among them are the lack of precise understanding of the transformation of the visual stimulus into the internal world model, of what constitutes an adequate primitive internal description of the stimulus, of how these descriptions are combined, of

(*) André Yakovleff is also with the Defence Science and Technology Organisation (DSTO), Information Technology Division, 171 Laboratories Area, P.O. Box 1500, Salisbury, SA 5108, Australia.

This work is supported by the Australian Research Council.

the structure of processes creating the transformation, and of how the ambiguities inherent in the sensory data may be resolved.

One solution is to go back to natural vision and look at the examples that have evolved to the appropriate level of complexity that we actually need. Insects, for example, have developed a task oriented visual system. They offer a level of visual processing which is intermediate between simply detecting the direction of a moving contrasting edge and universal discrimination of different natural objects.

This paper describes a sensing system inspired by early visual processing of insects. The design is based on the *template model*, originally proposed by Horridge and co-workers at Australian National University [4][5]. This model is based on the use of self-motion to infer the relative position of objects in the visual environment. The chip serves as a real-time visual micro-sensor; it accepts a time-varying optical image and provides outputs indicating range, bearing, and speed of objects in the visual environment. The chip carries out principally three tasks: detection, interpretation and analysis. Each of these corresponds to a separate architectural component. The first stage comprises analog circuitry and the others are digital. Whenever possible, the design is reconfigurable so as to provide a wider range of application as well as minimise errors.

The overall aim is to design a system capable of producing pre-processed sensory information directly usable by some form of robotic control system such as that for the Autonomous Land Vehicle by Payton [6] or the subsumption architecture developed by Brooks [7].

These systems derive their behaviour from a hierarchy of control layers, from the reactive collision avoidance, for instance, to the symbolic path planning. The type of sensory information required is in effect partitioned, in the sense that there is no need to interpret a global picture of the environment. At the lowest level, the inputs can be as primitive as an indication of presence at a certain bearing (proximity sensing), which is used for collision avoidance, while at the highest level it might be the image of a targeted object (image processing) necessary to perform landmark-based navigation. The basic premise of this type of system, however, is that the sensory information be adequate and relevant, and, at least at the lower level which is highly reactive, be readily interpretable.

As is shown in the next section, the template model is particularly suited to this type of application, as only straightforward processing is required. In effect, it provides a preliminary "information filter" in the sense that only the information pertaining to the task at hand is used.

We first describe the insect motion detection system and then present the template model to show how it is particularly well suited to being implemented in VLSI. Next is a high-level description of the chip's architecture and the reasons behind certain choices, followed by a more detailed analysis of its main components. Lastly, we describe one possible physical configuration.

II. The Insect Motion Detection System

The primary visual system of insects has a highly organised structure, dominated by a retinotopic organisation. It comprises three visual ganglia, namely the *lamina*, the *medulla*, and the *lobula* or *lobula complex.* Each neuropil is strictly organised into columns and strata. Both the lamina and medulla are composed of structurally identical parallel synaptic compartments, or columns, which exactly match in number the ommatidia in the retina. Each column in the lamina, known as an *optic cartridge*, receives inputs from a group of eight photoreceptors (R1 - R8) that share the same visual axis as the overlaying ommatidium, and projects outputs to the medulla column lying directly beneath it. For a detailed description of the lamina synaptic pathways, see, e.g., [8][9][10]. The lamina contains no motion-detection neurons; its major output cells are concerned with coding of contrast to help the visual system cope better with large variations in background intensities. The response of these neurons, which are thought to provide inputs to the motion-detecting neurons of the second neuropil, have the characteristics of a high pass filter [11].

The medulla is characterised by an extensive network of lateral connections [12][9]. It is the most peripheral structure in which movement detection takes place. However, the motion computation centre of insects appears to be the third visual ganglion. It is characterised by large field *directionally-selective movement-detecting* (DSMD) neurons, which seem to share a common network of presynaptic elements derived from the medulla. This group of DSMD neurons comprises several classes of tangential cells, which respond to whole-field horizontal or vertical motion [13][9].

The initial stages of movement detection in the fly appear to be based on sequence discrimination by elementary movement detectors (EMD's). Both behavioural and electrophysiological experiments indicate that movement detection takes place between neighbouring points of the sampling lattice of the eye [14][15]. Furthermore, overwhelming evidence has been accumulated showing that the DSMD neurons of the lobula plate receive both excitatory and inhibitory inputs from large retinotopic arrays of small-field EMD's which possess opposite preferred directions [Hausen].

Figure 1. illustrates the basic functional structure of a DSMD neuron. The basic information flow from a lamina cartridge (box L) to the DSMD neuron is represented by a puppet. The legs of the puppet transmit signals to its knees (M units) where they interact with the delayed signals (box τ), carried laterally by the puppet arms of the neighbouring columns. The DSMD neuron receives excitatory and inhibitory signals from an array of functionally identical EMD's, which differ only with respect to the orientation of their sampling bases. The EMD is the minimum prerequisite for directionally selective detection of motion in the visual field. Here the EMD is implemented as a spatiotemporal template which is described in the next section.

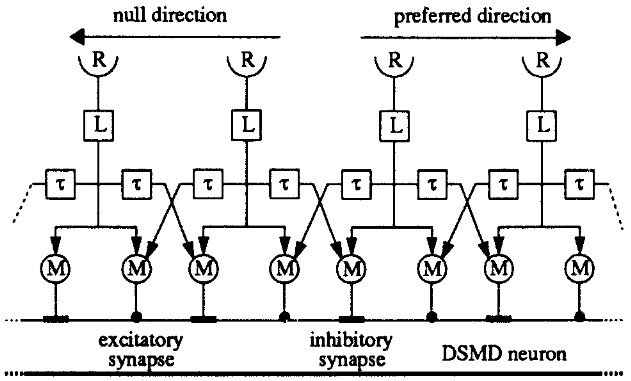

Figure 1. Schematic representation of the insect motion detection system. The R units represent the photoreceptor array, the L units are lamina cells, and the M units represent the medulla elementary movement detectors (EMD's). Adapted from [16].

III. The Template Model

Models based on insect vision differ from those based on human vision such as [17] which tend to be computationally intensive. Indeed, as will be shown, preliminary interpretation of visual information is inherent in the implementation of the detection stage.

The visual field is sampled spatially. Each sampling channel detects changes in light intensity at consecutive sampling times and will therefore show either an increase (↑), a decrease (↓), or no change (–). Hence for a pair of adjacent sampling channels there are 9 possible combinations: (– –), (↑ –), (↑ ↑), (↑ ↓), (↓ –), (↓ ↓), (– ↓), (↓ ↑), and (– ↑). These combinations are formed in the spatial domain. The temporal domain is included by associating the combinations obtained at consecutive sampling times, t_0 and t_1, yielding 81 possible combinations, or spatio-temporal "templates". Those that possess a diagonal symmetry, for instance, indicate directional motion, while others, which are symmetrical about a vertical or horizontal axis, indicate temporal change but no motion:

t_0	↓ –	↓ ↓	↓ ↑	↑ ↑	Motion to the right
t_1	↓ ↓	↑ ↓	↓ ↓	– ↑	

t_0	– –	↓ ↓	↓ ↑	– ↑	Temporal change
t_1	↓ ↓	↑ ↑	↓ ↑	– ↑	

Templates containing information on the direction of motion are shown in Figure 2. [5]. The overall picture is obtained by processing the responses in parallel. A more detailed presentation of the model, which also shows its correspondence to the biological model, can be found in [4] and [5].

← : movement to the left → : movement to the right

t_0 \ t_1	– –	↓ –	↓ ↓	↓ ↑	– ↑	↑ ↑	↑ –	↑ ↓	– ↓
– –									
↓ –			→	←	←				
↓ ↓		←		←				→	→
↓ ↑		→	→		←	←			
– ↑		→		→					
↑ ↑				→			←	←	
↑ –						→		←	←
↑ ↓			←			→	→		←
– ↓			←				→	→	

Figure 2. Template table

This approach is not dissimilar to [18], where the outputs of a bi-dimensional photoreceptor array are temporally differentiated. However a video scanner is then used to produce an image which requires further interpretation.

IV. System Description

There are essentially three tasks to be performed: detection, interpretation, and analysis (or processing). Each task is translated into an architectural component as depicted in Figure 3.

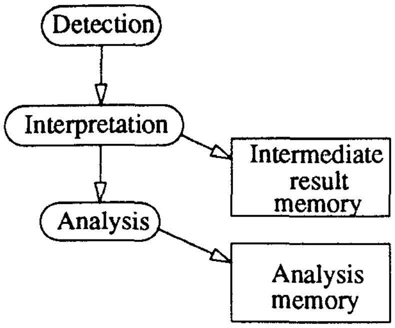

Figure 3. Functional architecture

The design needs to be adaptable to different circumstances as well as to be robust. As an experimental prototype, where adequate testing is essential, separate memories accessible from the outside are used to store the results of the interpretation and the analysis stages.

The sampling time of the detection stage is 10 milliseconds, which compares with the fastest time constant of the insect motion detection neurons. However, this parameter is not fixed and can be adjusted to suit different purposes. For this first version, we have chosen to implement the detection stage in a single row of 60 photo-detectors, each corresponding to one degree of visual angle. A cylindrical lens, which focuses the light on its surface, is placed directly on top of the chip, hence 360 degree vision is obtained by placing six chips hexagonally, as is shown in the physical implementation.

As current CMOS technology can operate at much higher frequencies than the sampling rate, processing time is not really an issue. This fact alleviates the need for interpreting the templates in parallel and allows us to multiplex the detection stage outputs into a single interpretation unit which can then be implemented in a RAM. The multiplexing rate is 1MHz. The obvious advantage of this scheme is flexibility which is important in that experimentation will be needed to show which template interpretation is best suited for different light conditions.

The analysis stage operates at 4MHz, which allows some processing to be accomplished each time a template has been interpreted. In this first version, the analysis consists in tracking the movement of certain templates from one sampling time to another within a sub-area of the 60 degree visual field.

The reasons for partitioning the visual field are mainly practical. The interest is in tracking edges of objects in movement relative to the visual system. As there might be several moving objects detected simultaneously, tracking implies correlating current template positions with the previous. Hence the problem arises of the number of edges we would have the capacity to track. Furthermore, if full capacity were to be reached before the entire visual field had been scanned, one side of it would have been favoured over the other. By partitioning the field, we assume that in normal circumstances there would be at most one moving edge in each direction within a limited area, or visual angle. Therefore there are two processing elements comprising a matching comparator and subtractor per area. Neighbouring areas can overlap so as to reduce border effects. Both the width and the amount of overlap are system parameters which can be adjusted.

We have deliberately avoided implementing a more complex processor as some of its basic parameters can only be made clear through experimentation. For instance, suppose we derive the speed of an object (or edge) relative to the sensor by integrating the movement of a given template over a certain period of time. If d is the distance travelled by the sensor in one sampling time step, s the relative speed of an edge, in terms of the displacement of a template per time step, and ϕ the angle subtended by one photodetector, then the object's range r will be:

$$r = \frac{d}{2 \times s \times \sin\frac{\phi}{2}}$$

The question is, however, how long does the object have to be tracked for the measured speed to be accurate enough for our purpose. Hence, instead of making assumptions with the danger of being too inaccurate if the integration time is too short, or of slowing down the system by choosing a long integration time, we have used the approach of optimising the processing through experimentation.

The general timing is shown in Figure 3. The sampling clock is used for the change detectors, and is in the 100Hz range, while the template and processing clocks are 1MHz and 4MHz respectively.

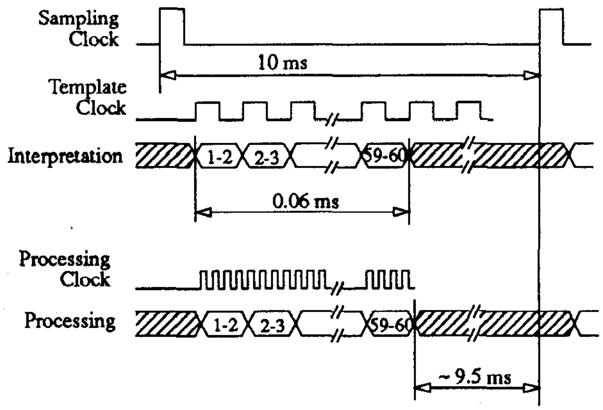

Figure 4. Timing (the numbers 1 to 60 indicate the sampling channels being interpreted and processed)

It is clear that there is a lot of time available for the results to be accessed from the outside and downloaded, for instance, onto a computer for further analysis, as the internal processing is done concurrently with the interpretation and ends shortly after the last template has been stored.

V. Architecture

Figure 5. depicts the hardware organisation of the chip. The address and data buses are connected to I/O pins. The interpretation is done by simply reading a memory, using the template as the address. The template RAM output is then written into the intermediate result RAM while being concurrently processed in the analysis stage. The latter partitions the global visual angle into smaller areas which can overlap and detects pre-defined templates within each area (see further), and writes its results into another RAM.

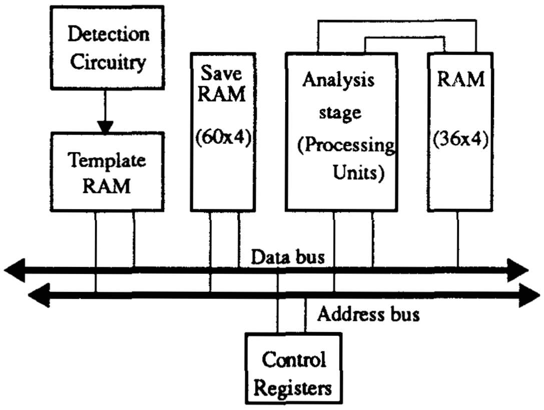

Figure 5. Hardware organisation

The control registers are loaded from the outside and specify the current operating mode, such as normal operation, result download, initial template RAM loading and various testing modes. In effect, all RAMs can be accessed from the outside for reading and writing. The target templates for the analysis stage and the width and overlap parameters are also held in the control registers.

A. Detection Stage

The chip comprises 60 photo-detectors each followed by a contrast change detector. The light is focused onto the chip surface by a gradient index lens. The photodetectors are spaced apart in such a way that each corresponds to a visual angle of one degree. The detection circuitry itself is analog, but the contrast change detector outputs are digital. Two bits are used to represent the following states: no change, increasing and decreasing light intensity.

Of the several kinds of photodetectors available in a standard CMOS process, we have chosen to use the well-substrate parasitic diode as a photodetector. The current generated by this type of device, while being in the pico/nano-ampere region, varies within several orders of magnitude and is converted into a voltage using the circuit shown in Figure 6. [1], which has the advantage of acting as a wide range automatic gain control (AGC). Due to the very low photocurrents, the MOS transistors operate in the subthreshold region, resulting in a logarithmic function. The output voltage of the circuit is proportional to the number of these diodes, which is thus chosen according to the required sensitivity, taking into account the fact that the voltage drop across the photodetector element should be kept within a reasonable range to have sufficient reverse bias. In this design there are three diodes, as we also need to compensate for the voltage drop of the next stage which is a source follower buffer.

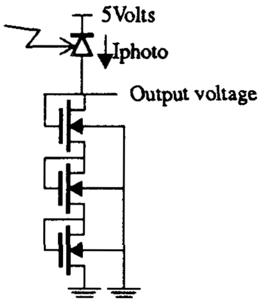

Figure 6. Logarithmic current to voltage converter

It should be noted, for future work, that a Gallium Arsenide (GaAs) implementation is likely to provide better results than the current silicon implementation, as the responsivity of the GaAs element would be in the order of 25% higher, based on previous work such as [19]. A GaAs implementation would also be advantageous for cases where demands are made on increasing the processing speed, whilst maintaining low power for mobile applications.

B. Differentiation Stage

Time differentiation of a signal can be approximated using either feedback or feedforward methods. Because of the low frequency nature of intensity changes and the high compression of the current in the AGC, the sensitivity of the circuit should be as high as possible. Feedback methods offer the advantage of decreasing the necessity for precise matching between the elements. We have chosen continuous differentiation using an active resistor and a limiter.

The highest capacitance available in VLSI is in the pico-Farad range, therefore the resistor will determine the lowest differentiation frequency, and would have to be in the giga-Ohm range which is not feasible for a passive element in VLSI, hence the use of an active resistor which has a large dynamic range [20].

The transfer function of the feedback network, as shown in Figure 7., comprising *C*, the active resistor *R* and the feedback amplifier of gain *-k* is:

$$H(s) = \frac{-kRCs}{1+k+RCs}$$

which can be approximated to:

$$H(s) \approx -RCs$$

The time constant $RC/(1 + k)$ is in the order of microseconds, hence our 10 millisecond sampling time is adequate. Other configurations, such as those using a differential amplifier, do not result with *k* in the denominator and are therefore more sensitive to high frequencies.

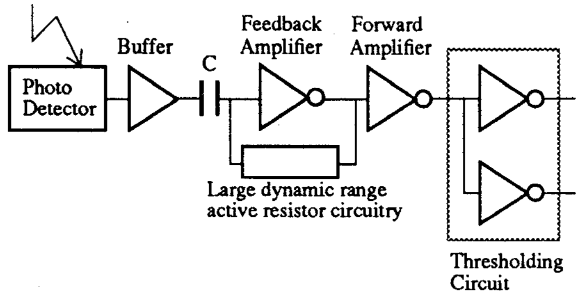

Figure 7. Contrast change detector

C. *Interpretation Stage*

The contrast change detector outputs are multiplexed pairwise and, together with the previous sampling time states of the same pair, form the template described previously. The template is used as an address to a RAM containing its interpretation (movement/no movement, direction, speed, light-to-dark/dark-to-light).

Physical layout limitations have the change detector outputs on either side of the row of photo-detectors. Hence one output from each side will form the template, therefore two 30-to-1 multiplexers can be used. Their select signals are the outputs of 5-bit counters which are incremented alternatively so as to select the first two detectors, then the second and third, then the third and fourth and so on.

Details of the template interpretation stage are given in Figure 8. The template RAM can only be written into during initialisation. At that time, the RAM address is provided from outside through the address bus and the multiplexer is switched accordingly by a control register bit.

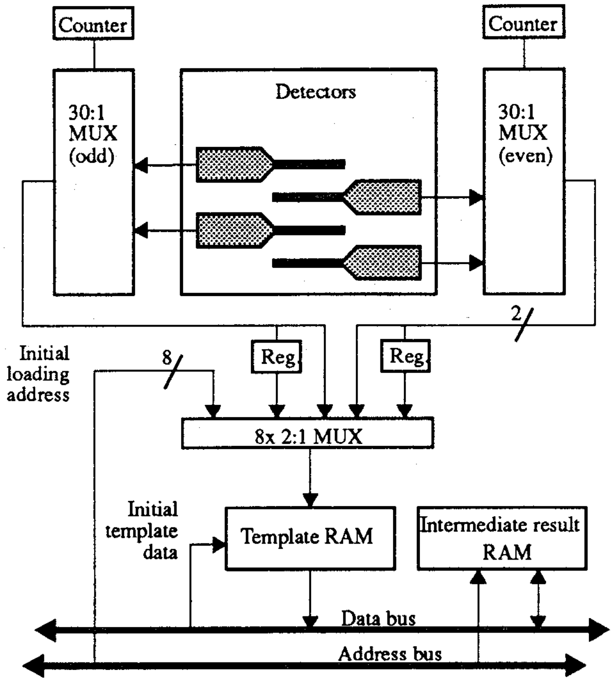

Figure 8. Template interpretation

D. *Analysis Stage*

The analysis stage detects the occurrence of pre-determined templates and calculates their relative displacement within pre-set areas of the visual field, which can be partitioned into up to 6 areas that may overlap. Hence, while the template RAM output is stored in an intermediate result memory at an address corresponding to its position in the visual field, the analysis stage compares the current template with a desired value. If a match is found, the current position is stored and subtracted from the previously stored value, if any, thus yielding the displacement. The latter is in turn stored along with the current position into another result memory at an address corresponding this time to the area.

The analysis stage, shown in Figure 9., comprises a controller and 12 processing units as we have chosen to track two templates in each partition of the visual field. The internal data bus is 12 bits wide, as a processing unit output comprises the current address, or position within the 60 degree visual angle (6 bits), and the displacement (5 bits + sign bit). The RAM is organised accordingly, which means that during processing it operates as a 12x12 memory (two 12-bit locations per area) while results are read on the 4-bit external data bus.

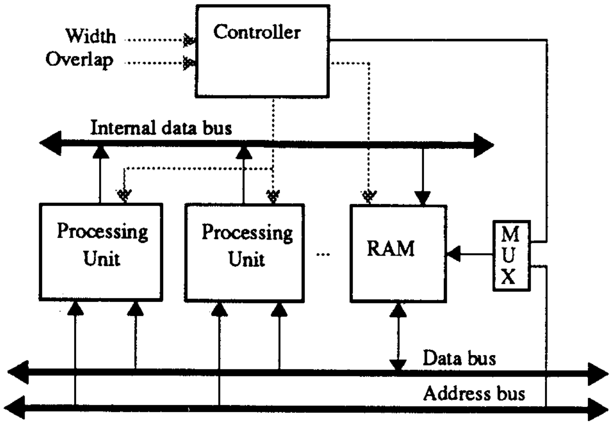

Figure 9. Analysis stage architecture

The controller only requires a signal from the interpretation unit that indicates the presence of valid data and address (that signal is essentially the "write enable" for the intermediate result memory). It produces the control signals to the appropriate processing units and generates the RAM address and write enable at each area boundary. Hence processing is finished a few cycles after the last template has been analysed. There are 4 system parameters: width, overlap and two target template values. Hence areas are identical, and two templates are tracked independently by each area.

The controller comprises two 5-bit counters, Count1 and Count2, which are multiplexed into two comparators, one each for width and overlap as shown in Figure 10.

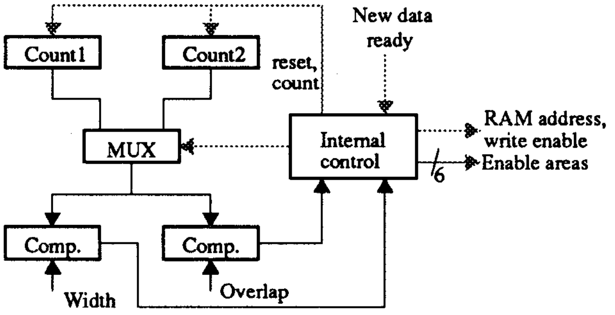

Figure 10. Controller

Initially, only Count1 and the first area are enabled. When the overlap value is reached, Count2 and the second area are enabled (The overlap value is therefore the position at which the next area begins, rather than the amount of overlap; its value must be more than half the width to ensure correct operation). When Count1 reaches the width value, the multiplexer switches to Count2, Count1 is reset and the first area is disabled. Later on, when Count2 reaches the overlap value, Count1 and the third area are enabled. The second area is disabled when Count2 reaches the width value and the multiplexer switches back to Count1. Operations proceed in this manner until either the sixth area is the last enabled, or the address has reached 60. The processor output values are stored in the result RAM each time the width value is reached, which corresponds to the switching of the multiplexer. Figure 11. depicts the activation of consecutive areas.

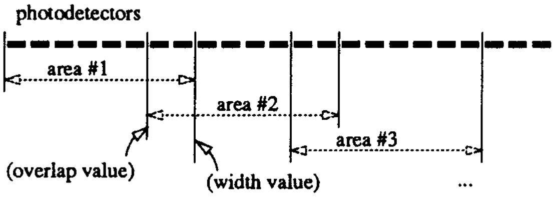

Figure 11. Example area activation

A processor comprises three registers, a comparator and a subtractor (see Figure 12.). Within the enabled area(s) it proceeds as follows: Each time a new template is on the data bus, its value is compared with a target value. If there is a match, the current address is stored in Reg1, subtracted from Reg2, and the result is stored in Reg3. Reg1 is then copied into Reg2, which will therefore hold the previous position of the template the next time that particular area is scanned.

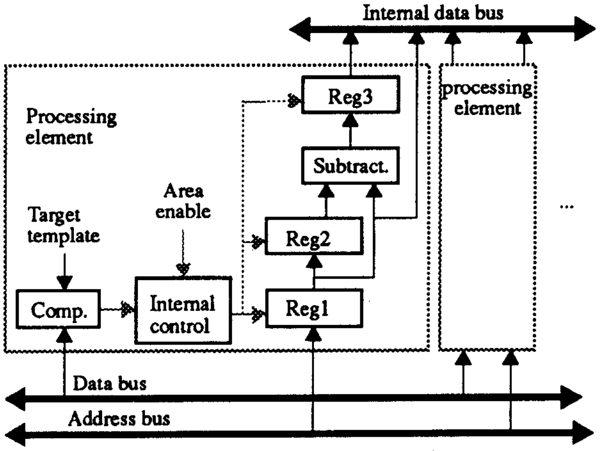

Figure 12. Processing unit

VI. Implementation

The gradient index lens mentioned earlier has a diameter of 1.8 mm and an aperture of 72 degrees, hence the photodetectors are 25 μm apart. The lens is positioned directly above the chip, as shown in Figure 13. The chip was fabricated in a 2μm p-well, double metal, double poly CMOS process. The die size is 4.5 by 4.6mm. Figure 14. shows the placement of the chip's main components.

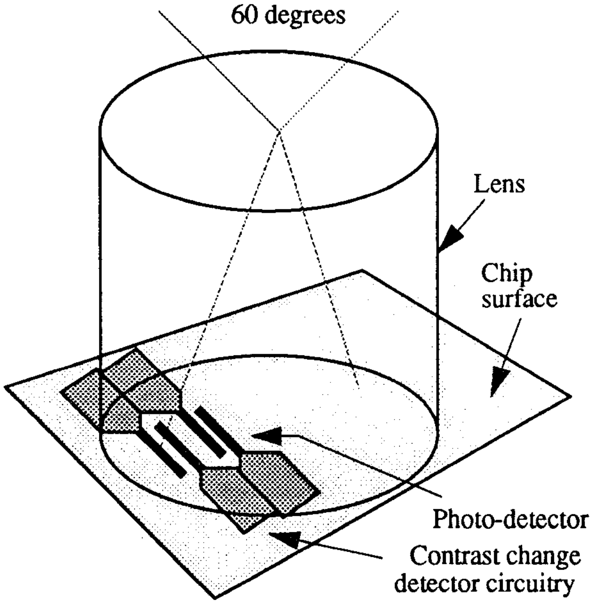

Figure 13. Lens/chip configuration

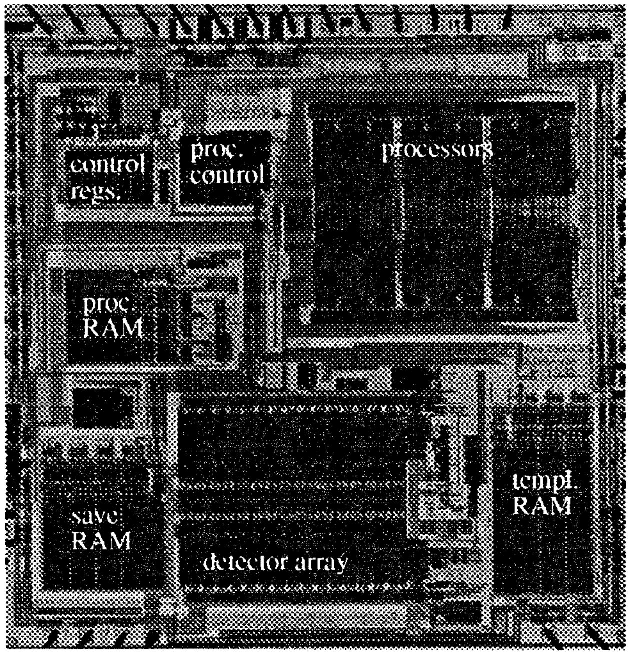

Figure 14. Chip photograph/floorplan

360 degree vision is accomplished by placing a chip with its lens on each surface of a hexagonal rod (Figure 15.) which is then held vertically. By extension, it can be associated with a horizontal combination so that both vertical and horizontal edges can be detected.

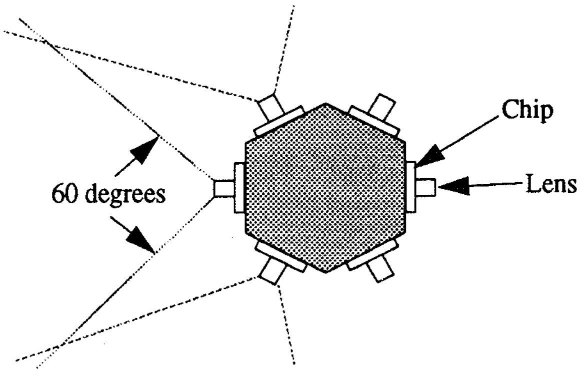

Figure 15. Hexagonal configuration cross-section

VII. Testing and Results

A test rig comprising adjustable bias voltage generators for the chip, a clock generator and a small amount of control has been designed to interface with a PC. The software developed permits access to the chip's internal memories and control registers, as well as control of the photodetector sampling clock.

The chip has been designed in such a way that most of the digital logic can be accessed from outside. For instance, the processor can be tested by providing data and addresses externally rather than internally, as would normally be the case. No errors have been found in either memories or processing elements.

The simplest test involving the photodetectors consists in a loop where at each iteration a sampling pulse is applied and the save RAM is read (see Figure 5.). The results can be shown on the PC in real time.

The chip was first tested in ambient light without the lens by moving a laser beam across the photodetector array so as to produce changes in light intensity under easily controllable conditions. We have found that approximately 15 templates occur, some of which being directionally sensitive, which corresponds both to our expectations and to previous simulations carried out while the chip was being fabricated.

We have begun further testing, this time with the lens, and preliminary results are encouraging. For instance, motion is accurately detected when a sheet of paper with strips of sufficient contrast is moved in front of the lens at a distance of up to one meter approximately. In general, the higher the contrast between the strips and the background, the further away from the lens can they be detected. However, a performance increase is expected with the lens correctly centred and as close as possible to the chip's surface.

VIII. Conclusions

The implementation of a vision processing system based on insect vision has been described. Its main features can be summarised as follows: The design combines analog and digital circuitry on a VLSI chip; the Automatic Gain Control built into the detection stage provides a large dynamic range; the system is small and easily expandable to perform 360 degree vision.

The system is meant as a development tool as well as a concept demonstrator. It is expected that real-time experimentation, with the system connected to a host, will enable us to produce a more autonomous implementation incorporating advanced processing.

It would also be desirable to incorporate a larger visual angle on a single chip, which infers that the optical engineering aspect needs to be re-addressed. The ideal solution would be to re-direct light from 360 degrees onto a flat two-dimensional array.

Acknowledgement

The authors wish to thank Professor Adrian Horridge, of Australian National University, for his original work which led to this implementation.

References

[1] Tanner J. & Mead C., "A Correlating Optical Motion Detector", in *Proceedings MIT Advanced Research in VLSI*, 1984.

[2] Tanner J. & Mead C., "An Integrated Analog Optical Motion Sensor", *VLSI Signal Processing II*, S-Y. Kung, R.E. Owen and J.G. Nash eds, IEEE Press, NY, pp 59-76, 1986.

[3] Moore A. & Koch K., "A Multiplication Based Analog Motion Detector", *SPIE Vol. 1473, Visual Information Processing: From Neurons to Chips*, pp 66-75, 1991.

[4] Horridge G.A., "The compromise between seeing spatial layout and making visual discriminations", *Current Science*, Vol. 60, No. 12, pp 686-693, 25 June 1991.

[5] Sobey P., "Determining range information from self motion - the template model", *SPIE Vol. 1382, Intelligent Robots and Computer Vision IX: Neural, Biological, and 3-D Methods*, pp 123-131, 1990.

[6] Payton D., "An Architecture for Reflexive Autonomous Vehicle Control", in *Proceedings IEEE International Conference on Robotics and Automation*, San Fransisco, CA, Vol. 3, April 1986.

[7] Brooks R., "A Robust Layered Control System for a Mobile Robot", *IEEE Journal of Robotics and Automation*, RA-2, No. 1, 1986.

[8] Meinertzhagen L.A. & O'Neil S.D., "Synaptic organization of columnar elements in the lamina of the wild type in *Drosophila melanogaster*", *J. Comp. Neurol.*, Vol. 305, pp 232-236, 1991.

[9] Strausfeld, N.J., "Beneath the compound eye: neuroanatomical analysis and physiological correlates in the study of insect vision", *Facets of Vision*, D.G. Stavenga & R.C. Hardie eds., Springer-Verlag, Berlin, pp 317-359, 1989.

[10] Shaw, S.R., "The retina-lamina pathway in insects, particularly Diptera, viewed from an evolutionary perspective", *Facets of Vision*, D.G. Stavenga & R.C. Hardie eds., Springer-Verlag, Berlin, pp 186-212, 1989.

[11] Laughlin S., "Coding efficiency and design in visual processing", *Facets of Vision*, D.G. Stavenga & R.C. Hardie eds., Springer-Verlag, Berlin, pp 213-234, 1989.

[12] Baussenwein B., Dittrich A.P.M. & Fischbach K.-F., "The optic lobe of *Drosophila melanogaster*: II. Sorting of retinotopic pathways in the medulla", *Cell T. Res.*, Vol. 267, pp 17-28, 1992.

[13] Hausen K., "The Lobula-Complex of the Fly: Structure, Function and Significance in Visual Behaviour", in *Photoreception and Vision in Invertebrates*, M.A. Ali ed, London New York: Plenum Press, pp 523-559, 1984.

[14] Kirschfeld K., "The Visual System of Musca: Studies on Optics, Structure and Fusion", in *Information Processing in the Visual Systems of Arthropods*, R. Wehner ed, Berlin New York: Springer-Verlag, pp 61-74, 1972.

[15] Buchner E., "Elementary Movement Detectors in an Insect Visual System", *Biol. Cybern.*, Vol. 24, pp 85-101, 1976.

[16] Bouzerdoum A., "The elementary movement detection mechanism in insect vision", *Phil. Trans. R. S. Lond*, Vol. 339, pp 375-384, 1993.

[17] Wyatt Jr. J.L., Stanley D.L. & Yang W., "The MIT Vision Chip Project: Analog VLSI Systems for Fast Image Acquisition and Early Vision Processing", in *Proceedings IEEE International Conference on Robotics and Automation*, Sacramento, CA, pp 1330-1335, April 1991.

[18] Delbrück T., Mead C., "Time-derivative adaptive silicon photoreceptor array", *SPIE Vol. 1541, Infrared Sensors, Detectors, Electronics, and Signal Processing*, pp 92-99, 1991.

[19] Abbott D., Cui S., Eshraghian K., McCabe E., "Photovoltaic gate biasing edge effect in GaAs MESFETs", *IEE Electronics Letters*, Vol. 27, No. 21, pp 1900-1902, 1991.

[20] Moini A., Bouzerdoum A., Yakovleff A., Abbott D., Kim O., Eshraghian K., Bogner R.E., "An Analog Implementation of Early Visual Processing in Insects", in *Proceedings International Symposium on VLSI Technology, Systems, and Applications*, Taipei, Taiwan, pp 283-287, May 1993.