

A Biologically Inspired Modular VLSI System for Visual Measurement of Self-Motion

Charles M. Higgins and Shaikh Arif Shams

Abstract—We introduce a biologically inspired computational architecture for small-field detection and wide-field spatial integration of visual motion based on the general organizing principles of visual motion processing common to organisms from insects to primates. This highly parallel architecture begins with two-dimensional (2-D) image transduction and signal conditioning, performs small-field motion detection with a number of parallel motion arrays, and then spatially integrates the small-field motion units to synthesize units sensitive to complex wide-field patterns of visual motion. We present a theoretical analysis demonstrating the architecture's potential in discrimination of wide-field motion patterns such as those which might be generated by self-motion. A custom VLSI hardware implementation of this architecture is also described, incorporating both analog and digital circuitry. The individual custom VLSI elements are analyzed and characterized, and system-level test results demonstrate the ability of the system to selectively respond to certain motion patterns, such as those that might be encountered in self-motion, at the exclusion of others.

Index Terms—Address event representation, analog VLSI, asynchronous digital communication, biomimetic, self-motion, spatial integration, visual motion.

I. INTRODUCTION

ONE OF THE most fundamental computations that can be performed on a visual image sequence is the detection of motion. This primitive operation is so powerful that it is nearly ubiquitous in modern biological organisms which have true visual systems [1], [2]. Comparisons of fossil eyes with modern eyes suggest that it may be possible to trace motion detection back through the evolutionary pathways as far as 400 million years [61]. However, detection of visual motion by artificial systems is quite computationally intensive, requiring (in some form or another) the comparison of image sequences at high resolution in very short amounts of time.

Biological organisms solve this computational problem with a massively parallel continuous-time approach to small-field motion detection. A large number of elementary motion detectors (EMDs) operate in parallel on small portions of the image, and the patterns of motion (or optical flow) are later integrated

spatially to detect moving objects, compute a measure of depth from motion parallax, estimate self-motion, and so forth.

This abstracted approach to visual motion processing is general enough that it applies to organisms from insects to primates. In fact, although the underlying neural hardware is certainly quite different, the leading models describing elementary motion detection in insects [3] and primates [4] have been shown to be mathematically equivalent. Similarly, a strong analogy can be made between cells in the brain of a fly which spatially integrate highly specific arrays of small-field motion detectors all over the visual field (lobula plate tangential cells [5], [6]) and cells in the brain of a monkey (in visual cortical area MST [7], [8]). A strong hypothesis for the functional significance of all these cells is that they are involved in the measurement of self-motion.

The impressive performance and efficiency of biological organisms in real-world environments motivates us to use their underlying principles to design artificial systems. However, in doing so, we must do more than blindly mimic their operation. Rather we must choose models which, like the motion processing system described above, have been widely selected by evolution and are more general than a particular organism.

In this paper, we introduce and analyze a computational architecture for small-field detection and wide-field spatial integration of visual motion which is inspired by the principles of the biological systems systems described above. A custom VLSI implementation of this architecture is also presented and characterized. We show how this compact hardware system might be used in the visual estimation of the self-motion of a moving platform, which is particularly valuable for small airborne or underwater platforms that not only have very tight power and weight constraints, but also for which ground speed is not necessarily related to speed of movement through the air or water.

II. RELATED WORK

A. Computational Architecture

Extraction of self-motion information has been a topic of research in the machine vision community for decades and has generated volumes of research [9]–[14]; see [15] for a review. While many algorithms exist for estimating heading direction and other self-motion parameters, most rely on iteration and few are suitable for efficient real-time hardware implementation.

Following the seminal work of Gibson [16], much effort in the biological community has been dedicated to understanding how organisms estimate their own motion in the world. Of particular interest have been methods of estimating the heading direction [17], closely related to the focus of expansion (FOE). Perrone

Manuscript received January 31, 2002; revised October 16, 2002. This work was supported by the National Science Foundation's Center for Low Power Electronics (CLPE) and the Office of Naval Research under Agreement N68936-00-2-0002. The associate editor coordinating the review of this paper and approving it for publication was Dr. Ralph Etienne-Cummings.

C. M. Higgins is with the Department of Electrical and Computer Engineering and Arizona Research Laboratories, Division of Neurobiology, The University of Arizona, Tucson, AZ 85721 USA (e-mail: higgins@ece.arizona.edu).

S. A. Shams was with the Department of Electrical and Computer Engineering, The University of Arizona, Tucson, AZ 85721 USA. He is now with the Intel Corporation, Chandler, AZ 85226 USA.

Digital Object Identifier 10.1109/JSEN.2002.807304

et al. [18]–[20] have developed a detailed model of responses in primate cortical area MST which is closely related to the computational architecture presented here. Many similar models in varying levels of detail have been proposed for area MST [21], [8]. Franz *et al.* [22]–[24] have shown how to compute optimal matched filters for estimation of self-motion from optical flow and compared these filters to the patterns seen in fly tangential neurons. Their optimal matched filters are completely compatible with the present system.

A very similar architecture to the one described here has been independently developed by Douglass and Strausfeld [25], [26] based on the optic lobe organization of Dipteran insects. Their computational system is very similar to the one described here, but their study concentrates on the effects that details of EMD function and innervation matrix composition have on the tuning of the output to wide-field motion patterns, whereas the present work is more concerned with the range of computations that are possible with the architecture given a fixed set of simple EMDs and innervation matrices.

B. Hardware Implementation

Because small-field visual motion processing is very well matched to continuous-time fully parallel focal plane arrays, a large number of monolithic integrated sensors of this type have been fabricated [27]–[36] based on a variety of algorithms. An extensive review of analog VLSI motion sensors can be found in [37].

Integrated hardware attempts at self-motion processing have only begun recently, with the work of Indiveri *et al.* [38]. The zero crossing in a one-dimensional (1-D) array of CMOS velocity sensors was used to detect one component of the focus of expansion. In a separate chip, the sum of a radial array of velocity sensors was used to compute the rate of flow field expansion, from which the time-to-contact can be calculated. McQuirk [39] built a CCD-based image processor which used an iterative algorithm to locate consistent stable points in the image, and thus the focus of expansion. Higgins and Koch [40] designed a monolithic chip to compute flow field singular points, the function of which is subsumed by the present system. More recently, Deutschmann and Wenisch [41] have extended Indiveri's work to two dimensions by summing rows and columns in a two-dimensional (2-D) CMOS motion sensor array and using software to detect zero crossings and find the flow field singular point.

A growing number of multichip integrated hardware systems have been designed to address vision and image processing problems, a review of which can be found in [42]. Many of these systems, including the present work, use asynchronous digital communication techniques to transmit information between chips. The origin of the asynchronous interchip communications protocol used in the present work is in the work of Mahowald [43], but the protocol has been formalized and the performance greatly improved by Boahen [44], [45], who designed and provided the circuitry used here.

In a related approach to the present work, Boahen [46] has published a multichip vision processor that computes motion by emulating a model of primate motion computation. The photosensitive sender chip has four output channels per pixel,

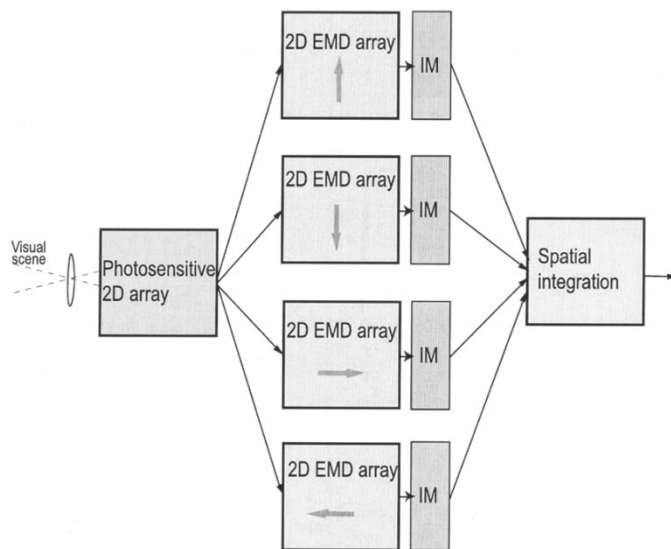


Fig. 1. High-level vision system architecture. The image to be analyzed is focused onto a photosensitive front end, which sends changing contrast information to four parallel arrays of EMDs. The output of these motion detectors is multiplied by innervation matrices (IMs) before being spatially integrated to form the output of the system.

modeling on and off, sustained and transient responses to light stimulation. By using a serial processor to combine the outputs of channels from neighboring pixels in a receiver chip, motion-sensitive outputs were synthesized. This system does not address wide-field spatial integration of visual motion, but rather synthesis of small-field motion-sensitive units with a clever algorithm.

Another closely related project is the work of Indiveri *et al.* [47], who have published a multichip motion system that employs three stages of processing: a photosensitive sender with nonlinear temporal differentiation, a programmable interconnect processor (the Silicon Cortex [48]) that allows for arbitrary address remapping, and a motion processing receiver. The sender utilizes the same photoreceptor and nonlinear differentiator used here, but splits the edge information from rising and falling edges into two output channels, allowing for independent sensitivity adjustment. The motion receiver chip is based on the FS algorithm [49] which computes local image velocity, unlike the direction-of-motion sensor used in the present work. Again, this work does not address wide-field spatial integration of motion.

Portions of the present work have been previously published in thesis form [50].

III. DESCRIPTION OF THE ARCHITECTURE

The first processing stage of the architecture, diagrammed in Fig. 1, is a 2-D focal plane array of phototransduction elements. Each element in this stage transduces local light intensity from an image focused onto it into a signal which can then be transmitted to further stages. At each element, this stage includes adaptation to the mean local light intensity. Only changes from the local mean illumination are transmitted to the next stage.

The second stage includes multiple 2-D arrays of EMDs, each of which receives the local contrast change information from the first stage and computes a measure of local image

motion. In the most trivial case, the multiple parallel motion detector arrays are distinguished by differences in orientation of the EMDs; however, these arrays could potentially differ in speed tuning, spatial frequency tuning, or other properties that could be spatially integrated to produce useful wide-field measures of image motion.

The reason for duplication of motion arrays at this stage is twofold. First, the motion computations carried out in this stage need not interact and thus may be carried out efficiently in parallel. Second, identical motion processors may be used in this stage by appropriate manipulation of the image information which they receive.

The final stage spatially sums the output of each EMD array after multiplying with weighting matrices (called here *innervation matrices* after Douglass and Strausfeld [25]) to synthesize units which are tuned to specific position/orientation patterns of wide-field optical flow. The innervation matrices may be (and in general are) different for each EMD array due to the particular properties of each. As Perrone *et al.* [18] have shown, these sets of innervation matrices may be considered as templates for patterns of desired wide-field motion, and the multiplication and summing operation a correlation with these templates. Multiple simultaneous sets of innervation matrices are supported, each summed into a separate unit in the final stage. Thus, the final stage unit with the largest value may be considered to represent the template having the highest correlation with the input optical flow pattern.

An additional function of the final stage is a soft thresholding operation so that outputs above a certain numerical threshold are emphasized, and outputs below this threshold are diminished. This will allow simple discrimination of flow field types based on template matching. For each set of innervation matrices, the final stage produces a scalar number which indicates the thresholded correlation with that template.

IV. THEORETICAL PREDICTIONS

In this section, we analyze the performance of the motion processing architecture in distinguishing patterns of visual motion using templates. Crucially, this analysis shows that optical flow patterns which match the current template can be distinguished from other patterns on the basis of the thresholded scalar system output.

In order to simplify the following discussion, let us trivialize the operation of the first and second stages by assuming that the inputs to the system, rather than visual images that change over time, are *optical flow fields*: vector fields indicating local 2-D image motion direction and speed. We will compute the output of each element of the motion processor arrays as a function of the local motion vector at its location. Let us also assume a static flow field, that is, whatever pattern of wide-field motion is present (e.g., expansion, contraction, rotation, or translation) is maintained throughout the experiment.

Fig. 2 shows examples of the types of flow fields relevant for self-motion. Flow fields such as expansion, contraction, and rotation will be referred to as *generalized spiral* stimuli because spiral optical flow patterns are made up of linear combinations

of these. Unlike pure translational flow patterns, generalized spiral stimuli all have a *singular point* defined as a point where flow field vectors pass through zero.

We will define the response of an elementary motion detector (see Fig. 3) in terms of two parameters: the preferred direction θ_p and the angular “bandwidth” θ_b . θ_p is the motion direction at which the EMD responds maximally and also corresponds to the angular centroid of motion tuning. θ_b is the angular extent around the preferred direction to which the EMD responds, which we will take to have a maximum of 180 degrees. This simple EMD model takes into account only the optical flow vector angle, without regard to the speed of local image motion.¹ If the local optical flow vector angle is θ_{stim} , then the EMD output may be expressed as

$$O_{EMD} = \begin{cases} 1, & \theta_p - \frac{\theta_b}{2} < \theta_{stim} < \theta_p + \frac{\theta_b}{2} \\ 0, & \text{otherwise.} \end{cases}$$

Leaving aside speed tuning, this simple EMD model captures the essentials both of the hardware motion detector used in the current implementation and of more advanced implementations currently being developed.

In the discussion to follow, each of the four motion processor arrays will be considered to have a different preferred direction 90° apart.

A. Centered Optical Flow Patterns

Let us now consider a template for centered expanding motion such as might be encountered with a fixed translating camera looking in the direction of heading [Fig. 2(a)].

Fig. 4 shows a simple binary set of innervation matrices which would most closely match the response of this system to centered expanding motion; Fig. 5 shows the response of the system with these innervation matrices to various flow patterns. Though we explore only binary innervation matrices both for simplicity of theoretical discussion and for emulation of the hardware system, the innervation matrices in general can be signed real numbers.

Let there be N by N elements in each motion processor array. If centered expanding motion is presented, each EMD with nonzero innervation matrix entry will be activated, resulting in a maximal response at the output of the final stage of approximately

$$O_{ce,ce} = 4 \frac{\theta_b}{360} N^2.$$

This quantity represents the element-by-element product of each EMD array with its corresponding innervation matrix followed by a spatial sum over all arrays to produce a single scalar which will appear at the output of the system. The above quantity is an approximation derived by assuming that the visual field is circular and that the number of elements N is very large. The first subscript (ce for *centered expansion*) indicates the tuning of the set of innervation matrices and the second subscript (again,

¹Douglass and Strausfeld [25], [26] extensively address more complex EMD models in a similar architecture.

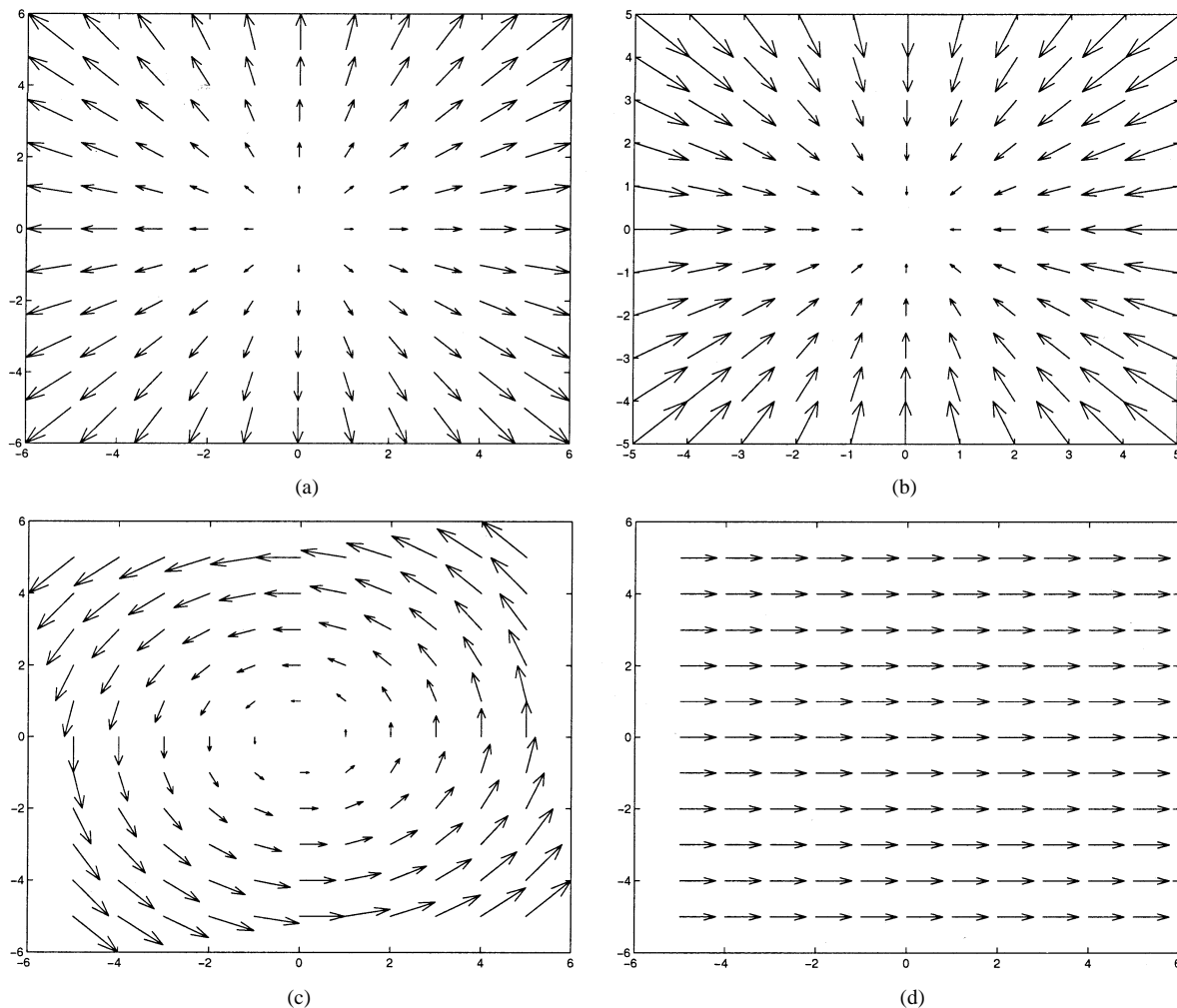


Fig. 2. Optical flow fields relevant for self-motion estimation. (a) Expansion, corresponding to translation of the camera in the direction the camera is pointing. (b) Contraction, corresponding to translation away from the direction the camera is pointing. (c) Rotation, corresponding to rotation of the camera about an axis normal to the focal plane. (d) Translation, corresponding to rotation of the camera about an axis parallel to the focal plane.

ce for *centered expansion*) indicates the type of pattern presented. For the purpose of comparison, let us define a baseline quantity

$$S_{im} = \frac{\theta_b}{360} N^2$$

which is the sum of a single innervation matrix. This value comes from the fraction of all possible innervation matrix entries which are unity. The output in response to a centered expanding pattern may now be expressed as $O_{ce,ce} = 4S_{im}$.

If centered contracting motion is presented (second subscript cc), motion occurs *only* in areas which have zero innervation matrix entries [see Fig. 5(b)], resulting in zero response

$$O_{ce,cc} = 0.$$

A centered rotational flow field (second subscript cr), in either rotational direction, will activate areas of EMDs which are 90° out of alignment with the expansion innervation matrices [Fig. 5(c)], resulting in zero response if θ_b is less than 90° , and a response of approximately

$$O_{ce,cr} = 4 \frac{\theta_b - 90}{360} N^2 \leq 2S_{im}$$

if θ_b is greater than 90° .

Finally, pure translational motion (second subscript t1, t2) in any given direction will activate all EMDs oriented within θ_b degrees of the stimulus angle [Fig. 5(d)]. If $\theta_b < 90^\circ$, either one or none of the EMD arrays will be activated. The response will be zero for stimulus angles more than $\theta_b/2$ from any of the four preferred directions or

$$O_{ce,t1} = \frac{\theta_b}{360} N^2 = S_{im}$$

for directions within 90° of a preferred direction. If, however, $\theta_b > 90^\circ$, more than one EMD array can be simultaneously activated, resulting in a maximum response (at $\theta_b = 180^\circ$) of

$$O_{ce,t2} = 2 \frac{\theta_b}{360} N^2 = 2S_{im}.$$

The results of this section are summarized in Table I. If we compare the responses to these four flow field types with the centered expansion template, we find the worst-case situation for discrimination of these four flow field types occurs with $\theta_b = 180^\circ$ and results in the response to rotation and translation being half that of expansion, with the contraction response being identically zero. In the best case, if $\theta_b < 90^\circ$, the response to contraction and rotation are zero, and translation yields a response of at most one quarter that of expansion. Thus, to

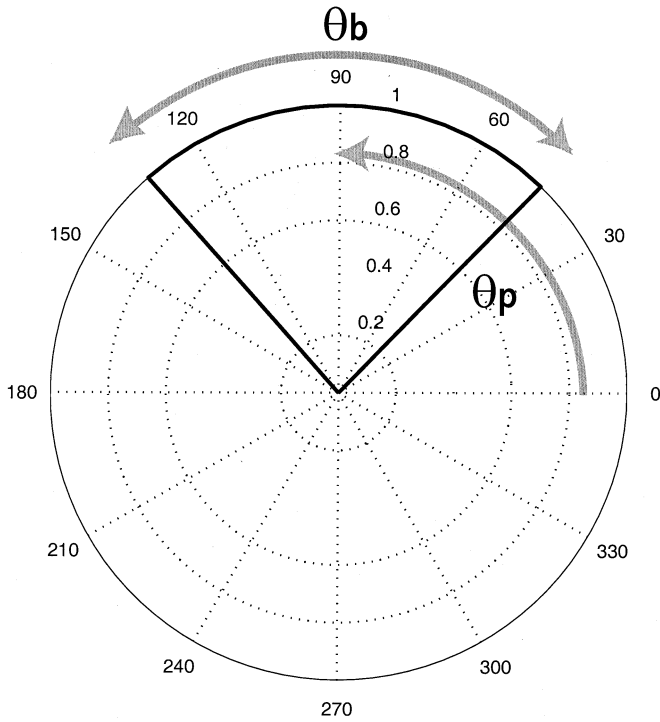


Fig. 3. EMD tuning curve. The response of the EMD is plotted against stimulus angle θ_{stim} . The response is unity in the range of local motion vector angles $\theta_p - \theta_b/2 < \theta_{stim} < \theta_p + \theta_b/2$ and zero otherwise. In the example shown, θ_p and θ_b are both 90 degrees.

discriminate expansion from other centered pattern types, the threshold in the final stage must be set in the worst case between $2S_{im}$ and $4S_{im}$.

B. Optical Flow Patterns With Variable Singular Points

Let us now generalize our set of flow field types to allow non-centered patterns. As mentioned earlier, our generalized spiral stimuli (expansion, contraction, and rotation) each have a singular point, defined as a point where flow field vectors pass through zero. In the case of expansion, this point is called the focus of expansion (FOE). For rotatory flow fields, the singular point is called the axis of rotation (AOR). No such singular point exists in pure translatory fields, and thus our previous discussion has handled those flow field types. If we continue to use our centered expansion innervation matrix set and vary the FOE and AOR of our flow fields, how will the system respond?

Let us first consider an expanding flow field pattern. As illustrated in Fig. 6(a), the maximal response occurs when the flow field is centered, and thus aligned with the template

$$O_{ce,ce} = 4S_{im}.$$

If the FOE moves to the far left edge of the visual field [Fig. 6(b)], the left-oriented array is completely inactivated, while the right-oriented array is fully activated. It can be shown geometrically that, if $\theta_b < 60^\circ$, the up and down arrays are also inactivated. As θ_b increases beyond 60° , the activation of the up and down arrays increases, and in the worst case the up and down arrays are fully activated. Thus,

$$O_{ce,e1} = \begin{cases} S_{im}, & \theta_b < 60^\circ \\ 3S_{im}, & \theta_b = 180^\circ \text{ (worst case)}. \end{cases}$$

If the FOE is positioned in the upper left corner of the visual field [Fig. 6(c)], the up and left arrays are completely inactivated, and the activation of the down and right arrays depends upon θ_b . It can be shown geometrically that if $\theta_b < 37^\circ$, no EMDs with nonzero innervation matrix entries will be activated. In the worst case ($\theta_b = 180^\circ$), the down and right arrays are fully activated and thus

$$O_{ce,e2} = \begin{cases} 0, & \theta_b < 37^\circ \\ 2S_{im}, & \theta_b = 180^\circ \text{ (worst case)}. \end{cases}$$

The response at the other edges and corners are the same by symmetry. A plot of the worst-case output as the FOE is varied over the entire visual field of the system is shown in Fig. 6(d). The peak is clearly at the center where the flow field best matches the template. In the best case, the corners of this plot will reach zero. In the worst case, to match only a centered expanding pattern the threshold of the final stage must be set between $2S_{im}$ and $4S_{im}$.

It is important in this case to consider a contracting stimulus (Fig. 7) because the response with the centered expansion template actually *increases* as the focus of contraction (FOC) moves away from the center. If the FOC is centered [Fig. 7(a)], there is no overlap between activated EMDs and nonzero innervation matrix entries, resulting in a zero system response

$$O_{ce,cc} = 0.$$

However, if the FOC moves to the left edge of the visual field [Fig. 7(b)], the left-oriented EMD array is activated in areas with nonzero innervation matrix entry, and the system response is proportional to θ_b . In the worst case ($\theta_b = 180^\circ$), we have

$$O_{ce,c1} = S_{im}.$$

With the FOC in the upper left corner of the visual field [Fig. 7(c)], both up and left-oriented innervation matrices may have activated EMDs. If $\theta_b < 90^\circ$, there is no response in this case. In the worst case ($\theta_b = 180^\circ$), these two innervation matrices are fully activated and thus

$$O_{ce,c2} = \begin{cases} 0, & \theta_b < 90^\circ \\ 2S_{im}, & \theta_b = 180^\circ. \end{cases}$$

A plot of the worst case output as the FOC is varied over the entire visual field of the system is shown in Fig. 7(d). The lowest value is in the center where the flow field is orthogonal to the template. Values rise away from the center, with a maximum of $2S_{im}$ in the worst case. Thus, to discriminate a centered expanding pattern from all other contracting and expanding patterns, the threshold of the final stage must be set between $2S_{im}$ and $4S_{im}$.

Let us now consider rotatory flow fields. With the AOR aligned with the expansion template [Fig. 8(a)], zero response is obtained as long as θ_b is less than 90 degrees. In the worst case ($\theta_b = 180^\circ$), two innervation matrices are fully activated and thus

$$O_{ce,cr} = \begin{cases} 0, & \theta_b < 90^\circ \\ 2S_{im}, & \theta_b = 180^\circ. \end{cases}$$

With the AOR at the left edge of the visual field, the up-oriented innervation matrix is inactivated, the down- and left-oriented innervation matrices are activated proportional to θ_b , and the

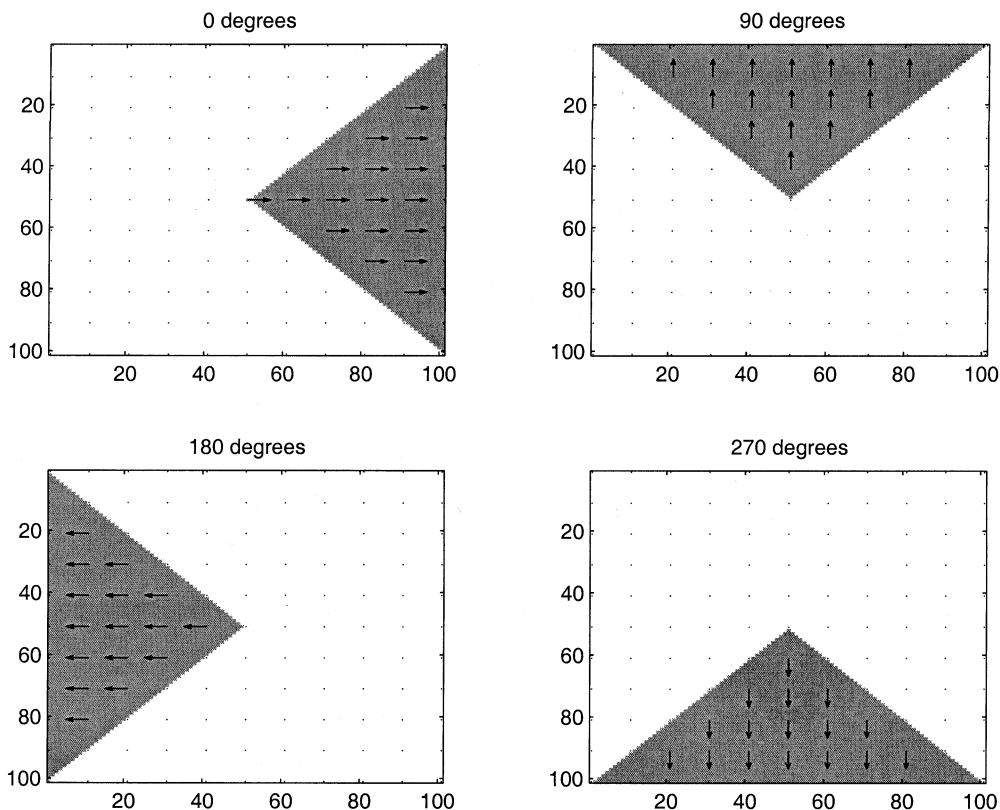


Fig. 4. Template for expanding motion, $\theta_b = 90^\circ$. This set of binary “innervation matrices” yields a maximal response when the pattern of motion spatially correlates. Above each panel is indicated the preferred direction θ_p of the EMD array: 0° , 90° , 180° , and 270° corresponding, respectively, to rightward, upward, leftward, and downward motion. The 2-D extent of each panel represents the visual field of the system. Gray indicates unity innervation matrix value; all other entries are zero. The arrows indicate the flow field pattern which would maximally stimulate this innervation matrix.

right-oriented innervation matrix is activated only for $\theta_b > 116^\circ$. In the worst case ($\theta_b = 180$), we have

$$O_{ce,r1} = 2S_{im}.$$

Finally, with the AOR in the upper left corner of the visual field, the up- and right-oriented innervation matrices are inactivated, the left-oriented matrix is activated in proportion to θ_b , and the down-oriented matrix is activated for $\theta_b > 90^\circ$. In the worst case ($\theta_b = 180$), we have

$$O_{ce,r2} = 2S_{im}.$$

Therefore, the maximum response for rotatory flow fields using the centered expansion template is $2S_{im}$. In fact, for $\theta_b = 180^\circ$, the output as the AOR is varied over the entire visual field is constant at $2S_{im}$.

Comparing the results for all three flow field types in this section, summarized in Table II, it is clear that we may set our threshold between $2S_{im}$ and $4S_{im}$ in order to discriminate centered expanding flow from other flow field types even if the singular points are allowed to vary. This is the same threshold value suggested for centered optical flow patterns.

C. Other Templates

Our entire discussion so far has assumed an innervation matrix set tuned for centered expansion. With the exception

of translation, it is very easy to extend our results for other template types.

Templates for noncentered expansion provide much the same response as that for centered expansion. The same thresholds and values apply, allowing us to use multiple simultaneous templates for expansion and take the maximum to estimate a crude focus of expansion. Templates for contraction are orthogonal to those for expansion, but otherwise provide very similar results with contraction and expansion responses reversed. The output for templates tuned to rotation are again very similar, but the results cited for expansion and contraction now apply to rotation and vice versa. In all of the above cases, the same threshold may be used to discriminate the optical flow field pattern to which the innervation matrix set is tuned from other patterns.

In the case of a template for translation, however, some special consideration is required. Let us consider the particular case of a leftward-tuned innervation matrix set. This innervation matrix is unity over the entire visual field of the array of EMDs oriented in the leftward direction and zero elsewhere. Thus, for translatory optical flow patterns, the response is

$$O_{t,t} = N^2 \geq 2S_{im} \text{ (with equality for } \theta_b = 180^\circ \text{)}$$

for stimulus angles within $\theta_b/2$ of the leftward direction, and zero otherwise.

However, in the case of expanding optical flow the sum of the leftward-oriented EMD array reflects how many EMDs are activated, which is monotonically dependent on the position of

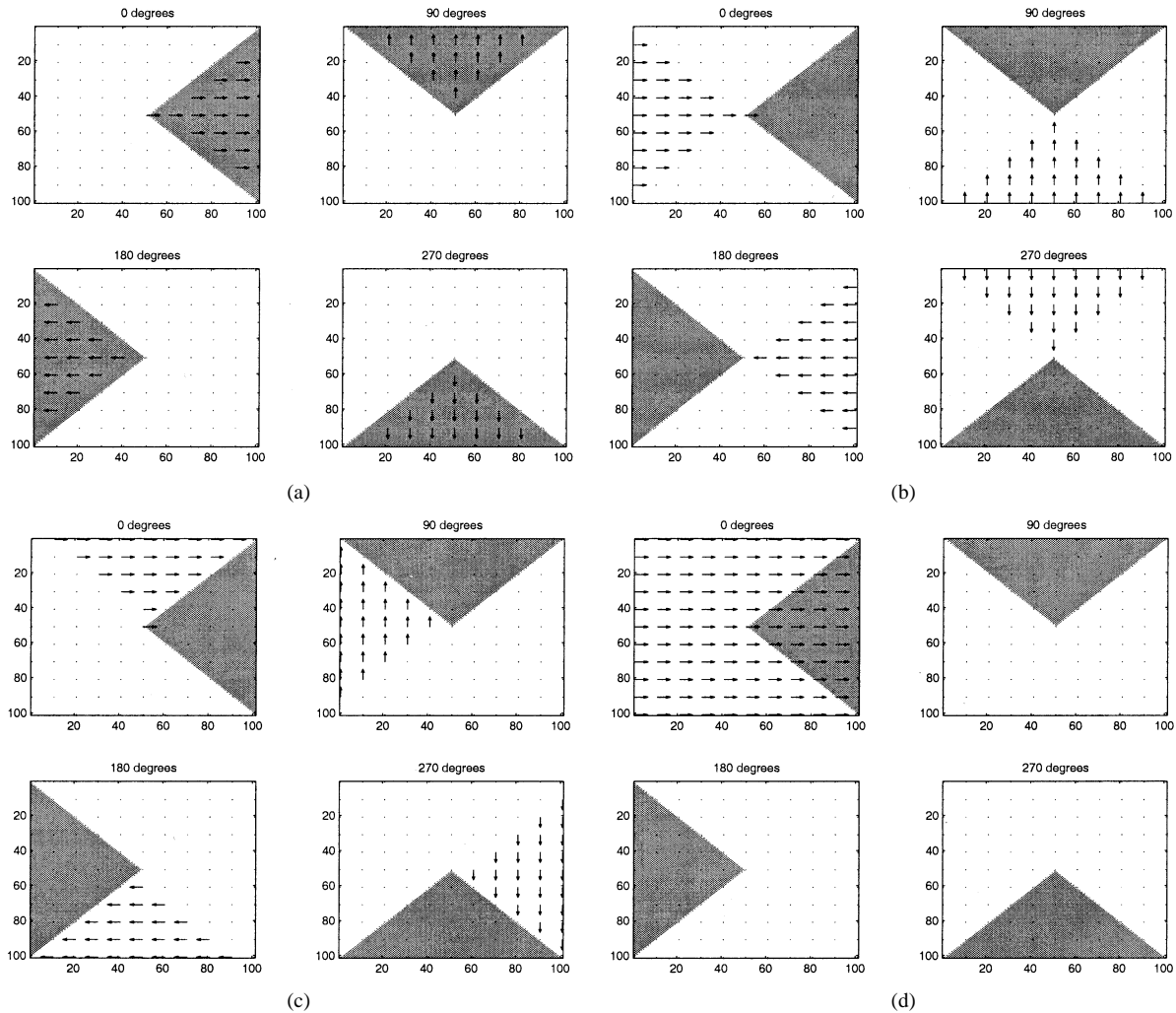


Fig. 5. Response of centered expansion template to four elementary types of motion, $\theta_b = 90^\circ$. The 2-D extent of each panel represents the visual field of the system. Gray areas indicate nonzero innervation matrix values. Arrows indicate the response of the EMD array to the given flow field pattern. Above each panel is indicated the preferred direction θ_p of the EMD array. (a) Expansion: maximal response; each responsive EMD has a corresponding nonzero matrix entry. (b) Contraction: zero response; no correlation with pattern. (c) Rotation: zero or weak response. (d) Translation: weak response.

TABLE I
SUMMARY OF FINAL STAGE OUTPUT WITH A CENTERED EXPANSION
TEMPLATE IN RESPONSE TO CENTERED OPTICAL FLOW PATTERNS.
A THRESHOLD BETWEEN $2S_{im}$ AND $4S_{im}$ IS SUFFICIENT TO
DISTINGUISH EXPANDING PATTERNS FROM ALL OTHERS

Type of optical flow presented	Final stage output (scaled to S_{im})	Comment
Centered expansion	4	Maximal response (best match to template)
Centered contraction	0	Minimal response (orthogonal to template)
Centered rotation	0	$0^\circ < \theta_b < 90^\circ$
	≤ 2	$90^\circ < \theta_b < 180^\circ$
Translation	≤ 1	$0^\circ < \theta_b < 90^\circ$
	≤ 2	$90^\circ < \theta_b < 180^\circ$

the FOE (see Fig. 9). If the FOE is to the far left, no leftward motion will be observed and thus the response is zero. If the FOE is to the far right, approximately

$$O_{t,e3} = \frac{\theta_b}{180} N^2 = 2S_{im}.$$

EMDs will be activated. Between these two FOE positions, the number of EMDs activated varies in proportion to the distance of the FOE from the left side. If the FOE moves up or down in the visual field, the modulation of the output is much less, decreases as θ_b increases, and the output does not vary at all for vertical FOE movements if $\theta_b = 180^\circ$.

The similarity in response between leftward translation and expansion with the FOE at the far right is not artificial. In fact, with $\theta_b = 180^\circ$, the leftward EMD array response to these two flow field types is identical.

Thus, for a leftward translatory template presented with an expanding flow field pattern, the output of the system monotonically reflects the position of the FOE. In the special case of $\theta_b = 180^\circ$, the output is linearly proportional to the horizontal FOE position and is independent of the vertical FOE position. An upward-tuned template can likewise compute the vertical FOE position. We have written earlier about this fact [40] and used it to build a monolithic hardware system to compute optical flow singular points. With a simple set of calculations, the output of this system can also compute the singular point of rotating flow fields, as well as spiral flow fields made from a combination of expansion, contraction, and rotation.

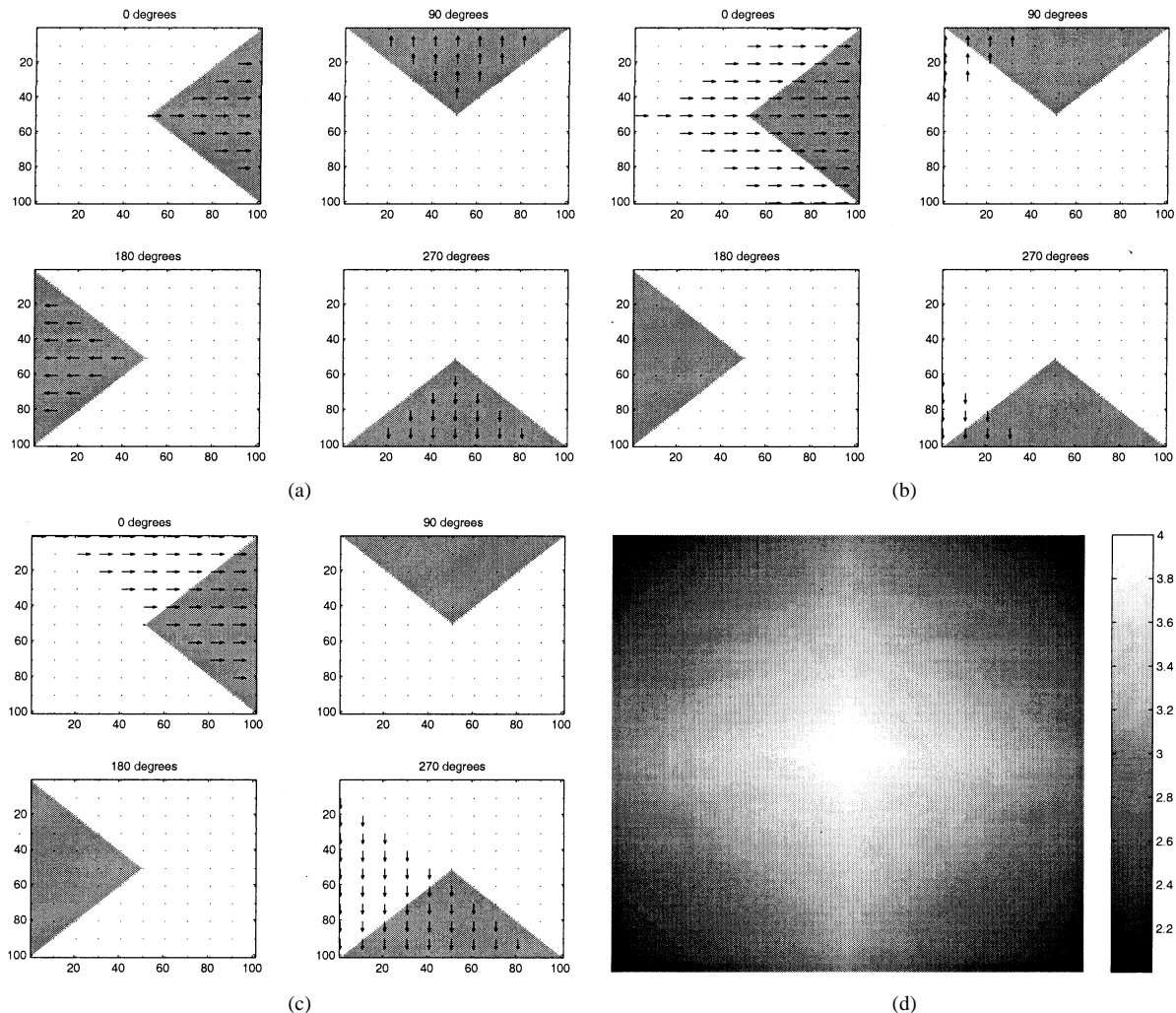


Fig. 6. Response of centered expansion template to expanding motion with variable FOE, $\theta_b = 90^\circ$. The 2-D extent of panels (a)-(c) represents the visual field of the system. Gray areas indicate nonzero innervation matrix values. Arrows indicate the response of the EMD array to the given flow field pattern. (a) $O_{cc,e1}$: FOE aligned with template. (b) $O_{cc,e1}$: FOE at left edge. (c) $O_{cc,e2}$: FOE at upper left corner. (d) Plot of worst case response with $\theta_b = 180^\circ$ as the FOE is varied over the entire visual field. The 2-D extent of this panel represents the position of the FOE in the visual field. The plot has been scaled to S_{im} .

The results of this section are summarized in Table III. In order to fulfill the purpose of discriminating a translatory flow field type from other types, it is necessary in general to set the final stage threshold lower than $2S_{im}$. Because this threshold is different from that required by other flow field types, in practical application we will *double* the response from each EMD in this template, allowing us to set our threshold at a level comparable to the other template outputs: between $2S_{im}$ and $4S_{im}$. This is mathematically equivalent to setting each nonzero innervation matrix entry in the set tuned for translation to *two* instead of one.

V. IMPLEMENTATION DETAILS

The hardware implementation of the motion processing architecture is based on modular mixed-signal VLSI building blocks connected with a high-speed asynchronous digital communications bus. In direct contrast to a monolithic VLSI system, the individual components of this multichip system are meant to be reusable in different configurations. In addition, manipulation of communications on the bus can be used to

achieve “virtual wiring” [51], [42] which cannot be practically achieved in conventional VLSI technology.

The interchip communications protocol used in this work is known as the address-event representation (AER). The original and most basic form of AER utilizes two digital control lines and several digital address lines to interface a sender chip to a receiver chip, as shown in Fig. 10. The protocol is used to communicate the occurrence of a binary “event” from sender to receiver in continuous time. A four-phase asynchronous handshake between sender and receiver guarantees reliable communication between chips; the address lines communicate the spatial position of a requesting sender pixel to the receiver chip, which forward the event to the receiver pixel with the same spatial position.

This protocol effectively allows any sender pixel to communicate digital events to the corresponding receiver pixel. Because requests can come at any time from any pixel in the array, it is necessary to use an arbitration scheme on the sender to serialize simultaneous events onto the single communications bus. Because the asynchronous protocol operates so quickly (on nanosecond scales) relative to the timescale of visual stimuli

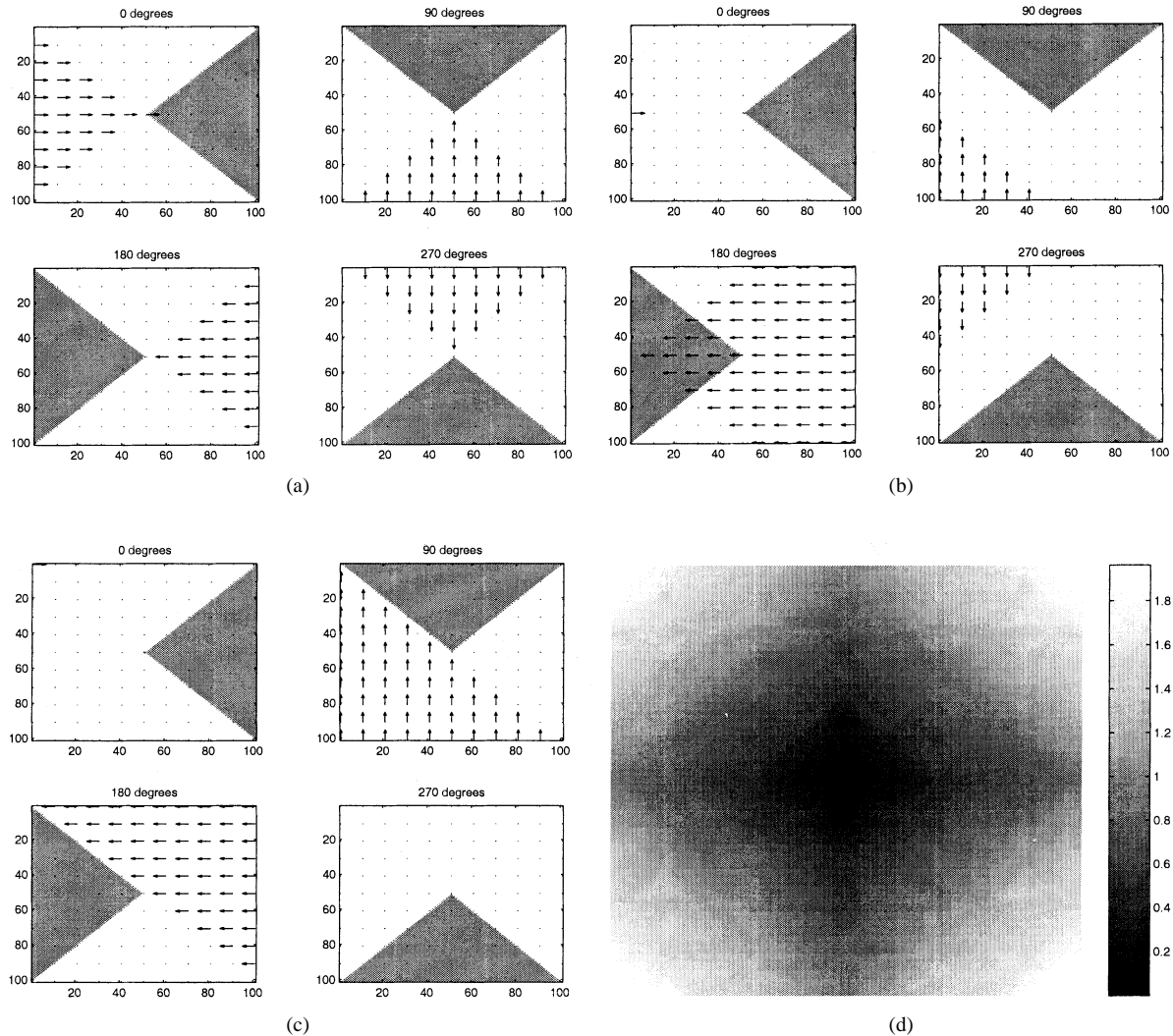


Fig. 7. Response of centered expansion template to contracting motion with variable FOC, $\theta_b = 90^\circ$. The 2-D extent of panels (a)–(c) represents the visual field of the system. Gray areas indicate nonzero innervation matrix values. Arrows indicate the response of the EMD array to the given flow field pattern. (a) $O_{cc,cc}$: FOC aligned with template. (b) $O_{cc,c1}$: FOC at left edge. (c) $O_{cc,c2}$: FOC at upper left corner. (d) Plot of worst case response with $\theta_b = 180^\circ$ as FOC is varied over the entire visual field. Plot has been scaled to S_{im} .

(on millisecond scales), the serialization caused by sharing of a single digital bus is usually benign for sensor applications. Various schemes exist for deciding which of several simultaneously requesting sender pixels is allowed to use the bus first [43], [52]–[54]. The scheme used in this paper is a binary tree arbiter [45], which yields a quick decision and scales well to large array sizes.

The circuitry necessary to implement the protocol varies from scheme to scheme. The particular hardware implementation of AER used in this chipset has been devised by Boahen; refer to [45] for further details. The AER bus in this implementation has a maximum bandwidth around 2.2 MHz, with a request-acknowledge cycle occurring about every 450 ns.

The first stage of the implementation (see Fig. 11) is a photosensitive sender chip, which transduces changes in light intensity into moving edge information. This edge information is sent through a bank of EPROMs which rotate the address space so that each of the identical second-stage motion transceiver chips has a different rotated view of the edges generated. Because each transceiver chip computes 1-D motion at a

fixed orientation in its address space, this address rotation gives each a different orientation in visual space. The motion computed by the transceivers is sent out to the third stage. Motion information from all of the transceivers is combined in a routing processor which sends this information to a final integrating receiver chip which simply converts its input into a voltage which can be read out of the system. This system combines the programmability implied by the EPROMs and the routing processor mapping function with the high speed and low power consumption of custom VLSI chips.

A. Photosensitive Sender Chip

The sender chip provides a visual front end for all further processing. This chip detects moving contrast edges in an image focused directly upon it. Edge locations are communicated off-chip via the AER bus. Qualitatively, the output of this chip looks like an image filtered in real-time with an spatial edge-enhancing operator; however, the edges disappear when no motion is present. A very similar chip has been described and characterized in detail in an earlier paper [42].

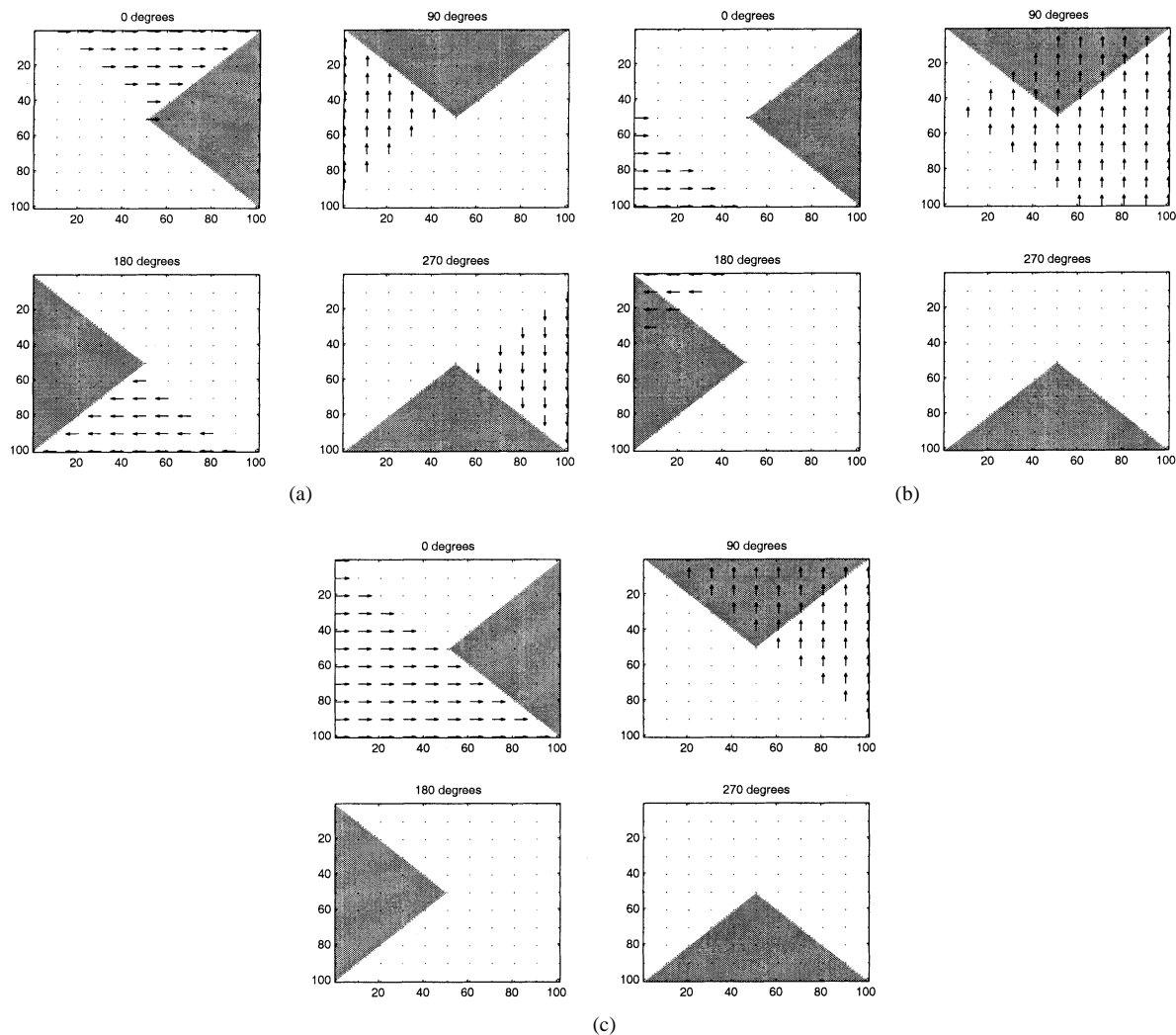


Fig. 8. Response of centered expansion template to rotatory motion with variable AOR, $\theta_b = 90^\circ$. The 2-D extent of each panel represents the visual field of the system. Gray areas indicate nonzero innervation matrix values. Arrows indicate the response of the EMD array to the given flow field pattern. (a) $O_{ce,cr}$: AOR aligned with template. (b) $O_{ce,r1}$: AOR at left edge. (c) $O_{ce,r2}$: AOR at upper left corner. Worst-case response with $\theta_b = 180^\circ$ yields a constant output of $2S_{im}$ as the AOR is varied over the entire visual field.

TABLE II
SUMMARY OF FINAL STAGE OUTPUT WITH A CENTERED EXPANSION TEMPLATE IN RESPONSE TO OPTICAL FLOW PATTERNS WITH VARIABLE SINGULAR POINTS. A THRESHOLD BETWEEN $2S_{im}$ AND $4S_{im}$ IS SUFFICIENT TO DISTINGUISH CENTERED EXPANDING PATTERNS FROM ALL OTHERS

Type of optical flow presented	Maximal response (scaled to S_{im})	Minimal response (scaled to S_{im})
Expansion	4 (centered)	0 $\theta_b < 37^\circ$ 2 $\theta_b = 180^\circ$
Contraction	2 ($\theta_b = 180$)	0 (centered)
Rotation	2 ($\theta_b = 180$)	0 $\theta_b < 90^\circ$ 2 $\theta_b = 180^\circ$

The core of the sender chip is a 14×12 array of sender pixels. See Fig. 12 for a layout diagram. Each sender pixel contains an adaptive photoreceptor [55] and a nonlinear differentiator circuit [49] interfaced to the interchip communication circuitry. This combination of adaptive photoreceptor and nonlinear differentiator is sensitive only to sudden changes in light intensity, and is referred to as a *temporal edge detector*, with

the assumption that a sudden change in local intensity is due to a passing spatial edge. When an illumination edge passes over the pixel, the event is communicated to the receiver. In this implementation, events are communicated on the bus only when the illumination changes, resulting in an efficient use of bus bandwidth. Arbitration, address encoding, and other interface circuitry to support the protocol are located in the periphery and described in [45]. The chip also incorporates a serial scanner for readout of the raw photoreceptor image.

The photoreceptor circuit (shown in Fig. 13(a) and analyzed in detail by Delbrück and Mead [55]) adapts to the local light intensity on slow time scales (a few seconds), allowing high sensitivity to transient changes over a wide range of illumination without a change in bias settings. The nonlinear differentiator circuit (shown in Fig. 13(b) and analyzed in detail by Kramer *et al.* [49]) produces a current pulse whenever the photoreceptor output changes suddenly. This circuit is nonlinear in the sense that it produces a fairly narrow (1–10 ms) current pulse at the change of the derivative sign [56] both for sharp and smooth inputs, due to the nonlinear feedback. The amplitude of the current pulse from this circuit can be shown to be proportional

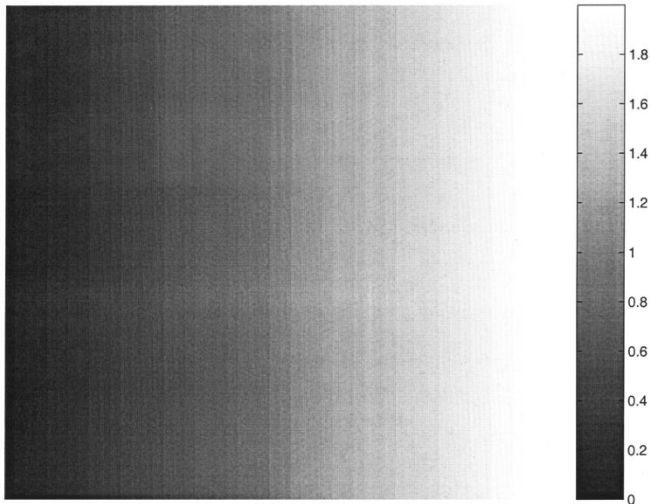


Fig. 9. Response of leftward translatory template to expanding motion with variable FOE, $\theta_b = 180^\circ$. The 2-D extent of the figure represents the position of the FOE in the visual field. The response of the system linearly indicates the position of the FOE. If $\theta_b = 180^\circ$ (as in this example), the response linearly indicates the horizontal FOE position independent of the vertical FOE position. Plot has been scaled to S_{im} .

TABLE III

SUMMARY OF FINAL STAGE OUTPUT WITH A LEFTWARD TRANSLATION TEMPLATE. A THRESHOLD OF $2S_{im}$ IS SUFFICIENT IN MOST CASES TO DISTINGUISH TRANSLATING PATTERNS FROM ALL OTHERS

Type of optical flow presented	Response (scaled to S_{im})	Comment
Translation	≥ 2	Equality for $\theta_b = 180$
Expansion, contraction, and rotation	$0 - 2$	Monotonically dependent on singular point position

to *temporal contrast*, the product of stimulus speed and spatial contrast [33].

The sender pixel communications interface circuit (modified from [45]) is shown in Fig. 14. Its input current I_{in} is taken from the nonlinear differentiator circuit output. Under normal AER bus load conditions, this circuit generates a event rate linearly proportional to the current input I_{in} .

The input current from the differentiator circuit will only exceed the threshold set by V_{leak} for a few milliseconds after a sudden illumination change. If the pixel request is not serviced within this time, the request will be withdrawn. For this reason, a slowdown in bus activity will not cause a large buildup of unserved events. During the time that the interface circuit input current is sufficiently large, the pixel will communicate a burst of events to the corresponding receiver pixel the rate and duration of which is dependent upon the temporal contrast of the stimulus.

The average power consumption of the sender chip is only 3.8 mW even at high bus usage due to the short duration of each request.

B. Motion Transceiver

The purpose of the motion transceiver chip is to receive edge information, compute local 1-D motion, and transmit this information in the form of a train of events to the next stage. Both

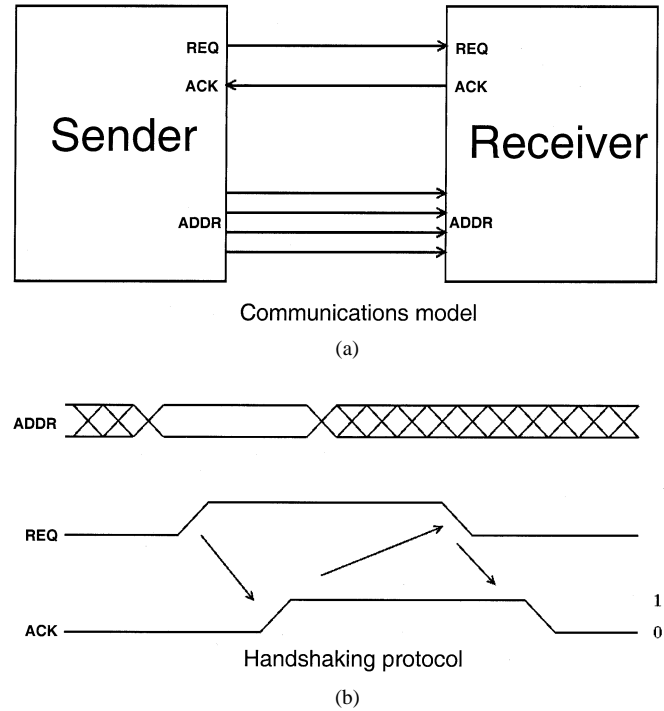


Fig. 10. AER protocol summary. (a) The model for AER transmission is shown: a sender chip communicates with a receiver chip via request, acknowledge and address lines. (b) The handshaking protocol for transmission using the above control and address lines is shown: a request with a valid address leads to an acknowledgment, which in turn leads to falling request and falling acknowledge.

receiver and sender peripheral circuitry are present on this chip (Fig. 15), taking up a significant proportion of the available area.

The core of the transceiver chip is a 12×12 array of pixels. Each pixel contains both receiver and sender communications interface circuitry, and a motion circuit implementing the inhibit-trigger-inhibit (ITI) direction-of-motion algorithm [33].

The receiver pixel communications interface circuit (modified from [45]) is shown in Fig. 16. A current, the magnitude of which is controlled by the bias V_{qua} , is produced to charge the capacitor only when both X_{sel} and Y_{sel} are active high. The biases V_{qua} and V_{tau} allow adjustable low-pass filtering of the input current to produce the complementary voltage signals V_{out} and $\overline{V_{out}}$ after amplification by a series of CMOS inverters.

The ITI motion algorithm (shown in Fig. 17 and analyzed in detail in [33]) computes the direction of a moving edge by detection of the order of edge arrival at neighboring pixels. It requires the V_{out} inputs from both left and right transceiver chip neighbors, as well as the $\overline{V_{out}}$ from the current pixel. The voltage signal V_{right} is "triggered" (raised to the upper voltage supply) by the arrival of a local edge, and "inhibited" (lowered to ground) by the arrival of an edge to the left. The signal V_{left} is also triggered by local edge arrival, but inhibited by an edge to the right. Both signals have an adjustable leak to ground (V_{leak}) so that the motion signals detected have a variable persistence time. These two binary voltage signals together unambiguously encode the direction of edge motion in one dimension across the sensor: when they are the same, no motion has been detected. When they are *different*, the direction of motion is indicated by which one is high. Thus the numerical difference of these two

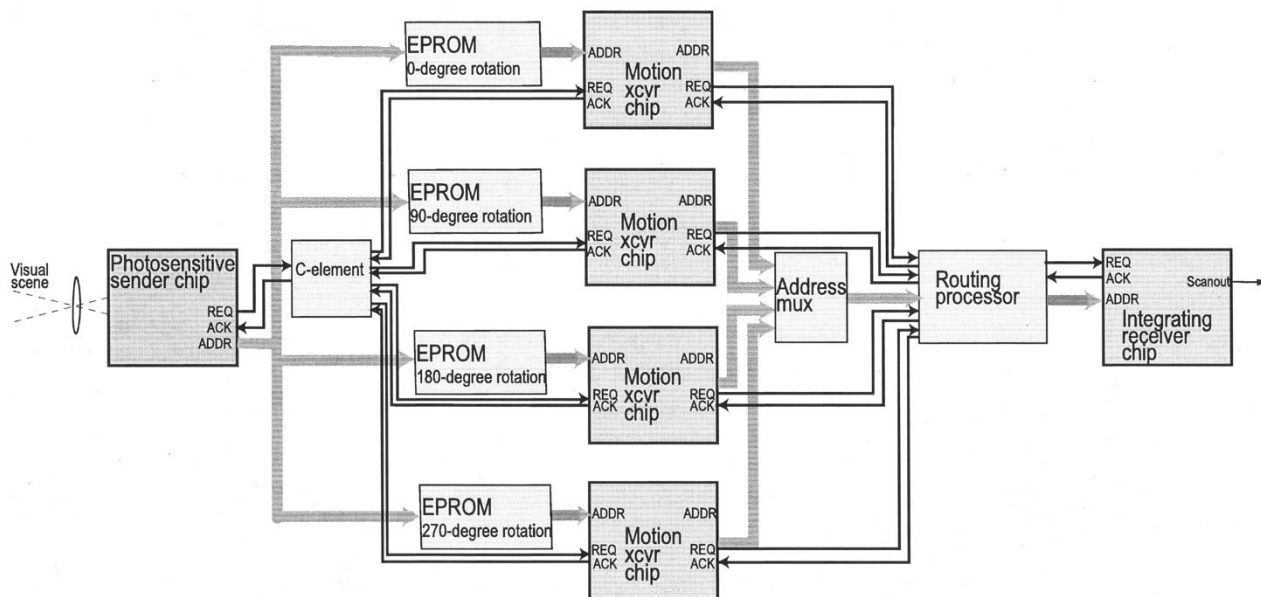


Fig. 11. Overall hardware architecture of motion processor system. Gray arrows indicate AER address buses. The EPROM for zero-degree rotation is not required in this configuration and is included for generality only. The C-element is a device built from discrete logic to provide synchronization of all four transceivers to the single sender. The address multiplexer is used to reduce the pin count into the routing processor; connections from the processor to the multiplexer are not shown for simplicity. The routing processor is currently implemented with a microcontroller, but can be replaced by a much more efficient FPGA (see text).

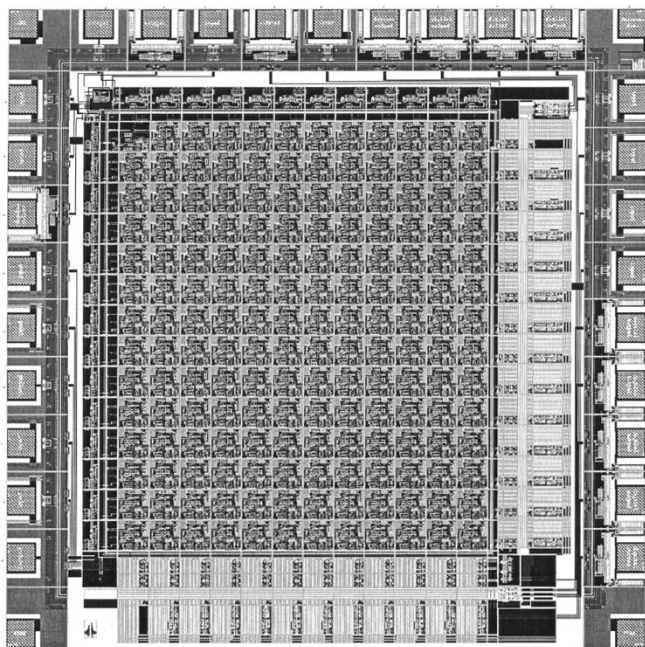


Fig. 12. Layout of the sender chip, as fabricated on a MOSIS tiny chip in a 1.2- μm standard CMOS process. Total area of the chip is $2.1 \times 2.1 \text{ mm}^2$. Inside the padframe at the periphery to the top and left is scanner circuitry for observation of pixel activity. At the periphery to the bottom and right is arbitration and address encoding circuitry to support the AER protocol. The core of the chip contains a 14×12 array of pixels.

voltage signals is computed by further circuitry in the form of a bidirectional current, which goes on to the sender interface circuitry.

Because this circuit implements a “direction-of-motion” algorithm, it responds with a nearly constant positive current to 1-D motion having a component in the direction of its orientation for nearly 180° of stimulus angle. Thus, its output closely

resembles the output of the prototypical EMD from Fig. 3 with $\theta_b = 180^\circ$.

The sender communications interface circuit is shown in Fig. 18 and contains one additional element from that shown for the sender chip. The input current I_{in} from the ITI motion circuit is added to a constant current controlled by V_{spon} so that both positive and negative excursions of the ITI circuit output current can be represented in event frequency. This “spontaneous” activity of the spiking circuit continues when no motion is present and serves as a baseline around which increases and decreases can occur.

Under medium bus load conditions, each transceiver chip consumes a static power of less than 5 mW at 5 V.

C. Integrating Receiver

The integrating receiver chip (Fig. 19) contains a 27×29 array of pixels, each of which serves the purpose of converting event frequency into dc voltage. The periphery of the chip contains AER interface circuitry for decoding addresses and distributing events to individual pixels, as well as serial scanner circuitry which allows individual access to the output voltage of each pixel.

The entire circuit for the receiver pixel is shown in Fig. 20. The receiver interface is identical to that shown earlier for the transceiver chip, integrating the train of events into a voltage V_{mem} . With the use of the V_{rowsel} transistor, this value can be scanned out of the chip as a current. The bias V_{offset} is provided to allow multiplication of the output current to a convenient level.

In order to analyze the behavior of this circuit, let us assume that it has a constant input from the AER bus with event frequency f and pulse width T . Each time an event is received, a “quantum” of charge is delivered to the V_{mem} node. If the V_{qua}

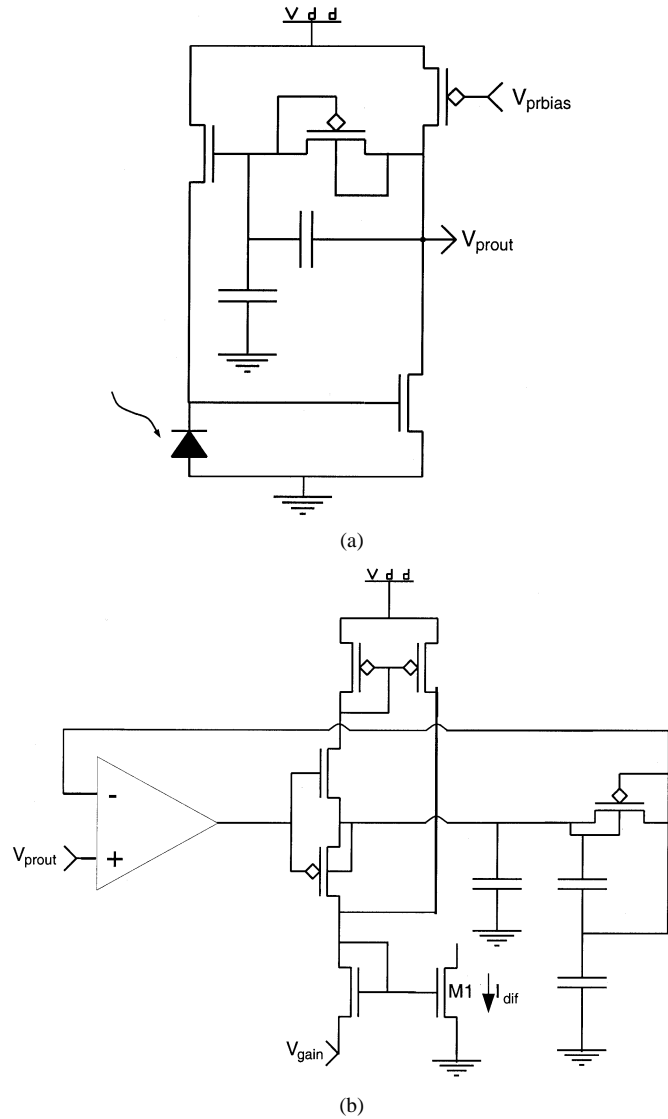


Fig. 13. Components of the temporal edge detector. (a) Adaptive photoreceptor which allows sensitivity to transient contrast changes over a wide range of illuminations. (b) Nonlinear differentiator circuit which responds only to sharp changes in illumination.

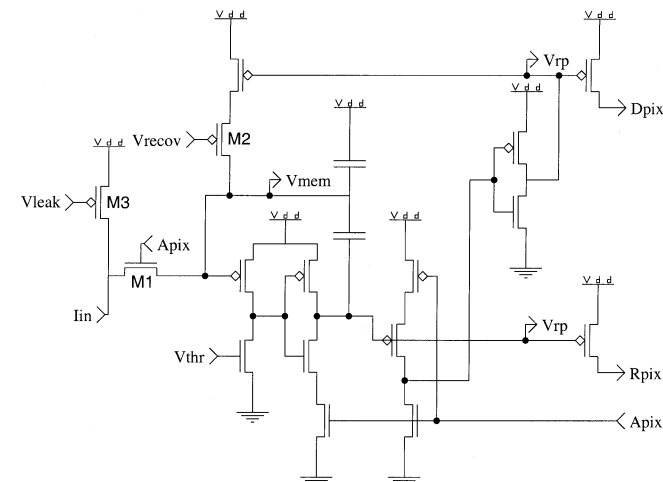


Fig. 14. Sender pixel communications interface circuitry. R_{pix} and A_{pix} are the interface with the row arbiter, and D_{pix} goes to the column circuitry.

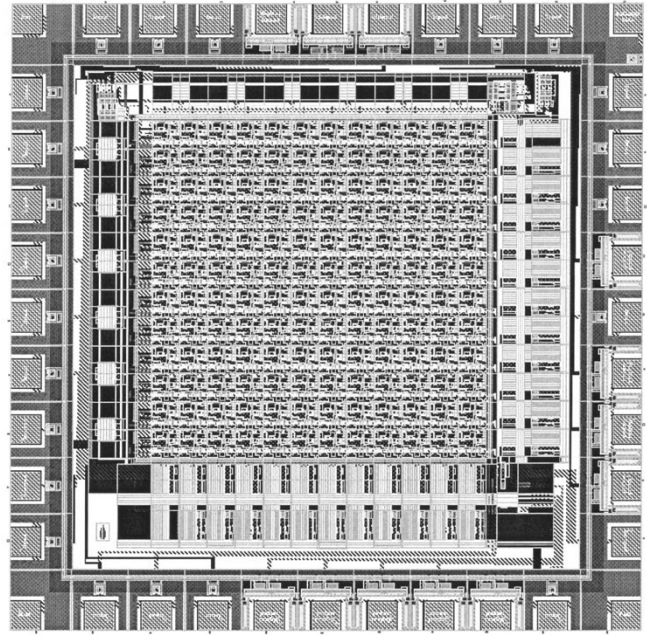


Fig. 15. Layout of the transceiver chip, as fabricated on a MOSIS tiny chip in a 1.2- μm standard CMOS process. Total area of the chip is $2.1 \times 2.1 \text{ mm}^2$. Inside the padframe at the periphery to the top and left are address decoding and other circuits implementing an AER receiver interface. At the periphery to the bottom and right are address encoding and arbiter circuits implementing an AER sender interface. The core of the chip is a 12×12 array of pixels.

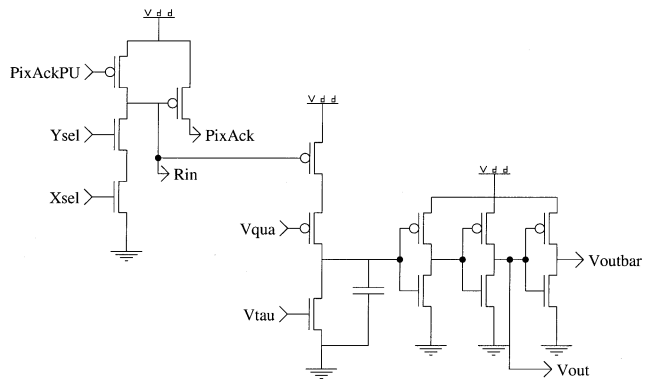


Fig. 16. Transceiver pixel receiver communications interface circuitry. When inputs X_{sel} and Y_{sel} are both active high, digital signals V_{out} and \bar{V}_{out} are activated.

transistor is operating in the subthreshold regime, the current during the time that V_{spk} is low (assuming V_{mem} is much less than V_{dd}) is

$$I_{up} = I_{0p} e^{-\kappa V_{qua} + V_{dd}}$$

(where κ is the subthreshold model parameter relating changes in gate voltage to changes in channel surface potential) and the total charge delivered is

$$Q_{spk} = I_{up} T.$$

With a fixed event frequency, we can compute the average current tending to raise the potential of V_{mem} as

$$\hat{I}_{up} = f Q_{spk}.$$

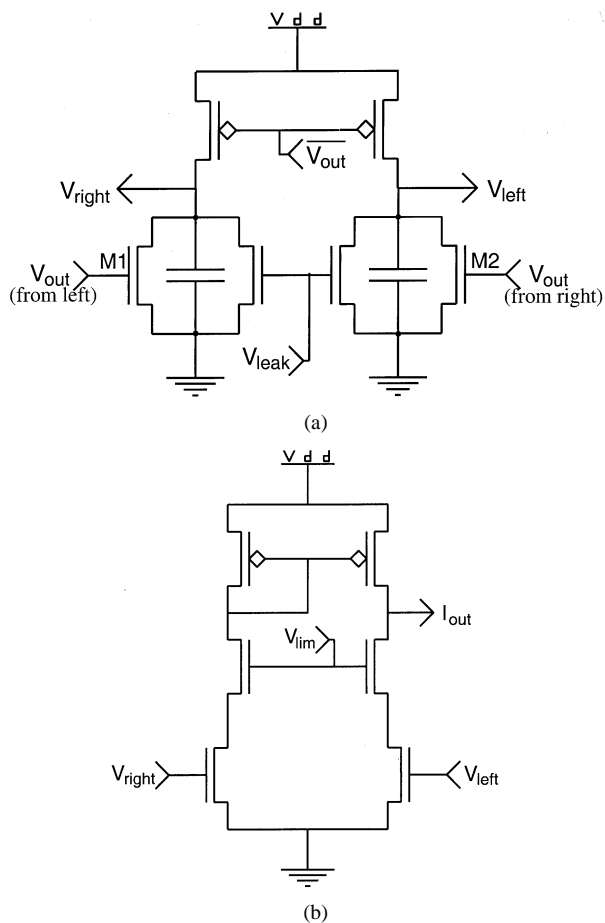


Fig. 17. Transceiver pixel circuits implementing the ITI (Inhibit-Trigger-Inhibit) motion algorithm. (a) Computation of the direction of motion by order of edge arrival. (b) Production of a bidirectional output current from the direction information.

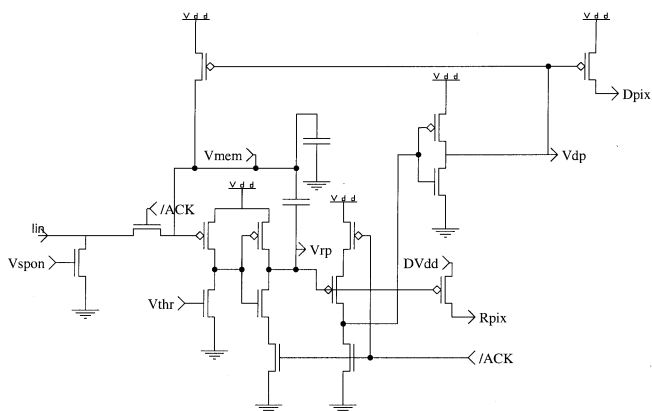


Fig. 18. Transceiver pixel sender communications interface circuitry. R_{pix} and $/ACK$ are the interface with the row arbiter, and D_{pix} goes to the column circuitry.

The dc current tending to lower the potential of V_{mem} is provided by the V_{tau} transistor and can be computed in sub-threshold as

$$I_{down} = I_{0n} e^{\kappa V_{tau} - V_{offset}}$$

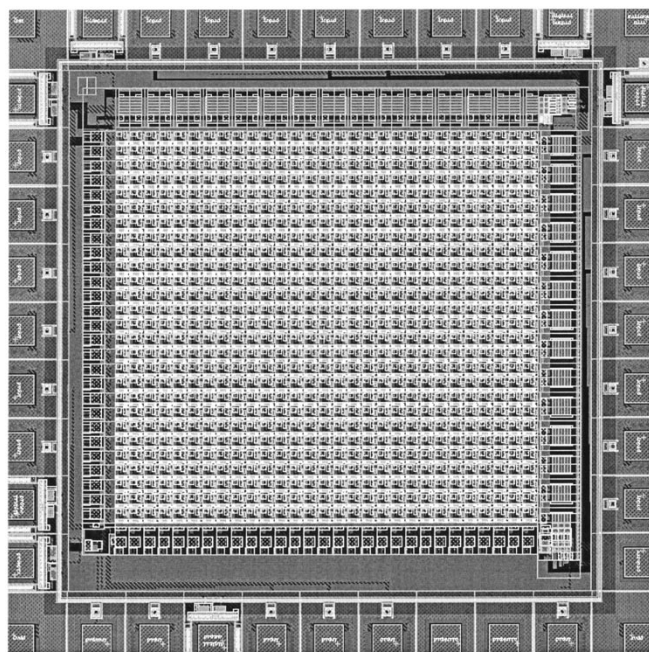


Fig. 19. Layout of the receiver chip, as fabricated on a MOSIS tiny chip in a 1.2- μm standard CMOS process. Total area of the chip is $2.1 \times 2.1 \text{ mm}^2$. Inside the padframe at the periphery the bottom and left is scanner circuitry for readout of pixel activity. At the periphery to the top and right is address decoding circuitry to support the AER protocol. The core of the chip contains a 27×29 array of pixels.

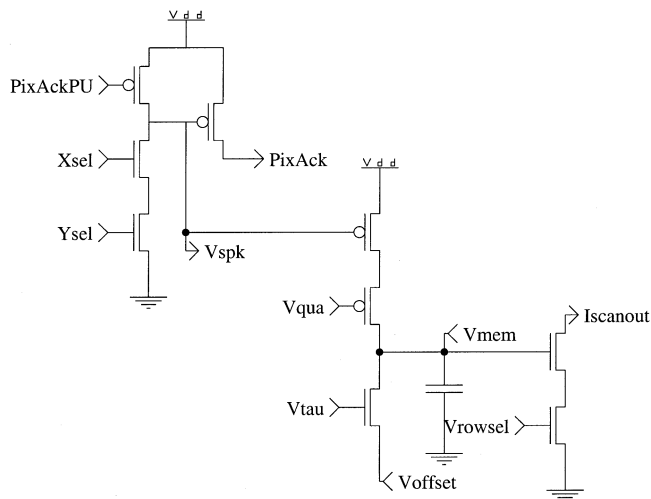


Fig. 20. Receiver pixel circuitry. When inputs X_{sel} and Y_{sel} are both active high, current is provided to the node V_{mem} . Based on the biases V_{qua} and V_{tau} an adjustable threshold may be set so that V_{mem} goes high when the input event frequency is greater than the threshold.

Thus, the net average current to the V_{mem} node is

$$\hat{I}_{up} - I_{down} = fQ_{spk} - I_{down}$$

where Q_{spk} and I_{down} are constants for fixed pulse width and bias conditions. If this net current is positive, the potential of

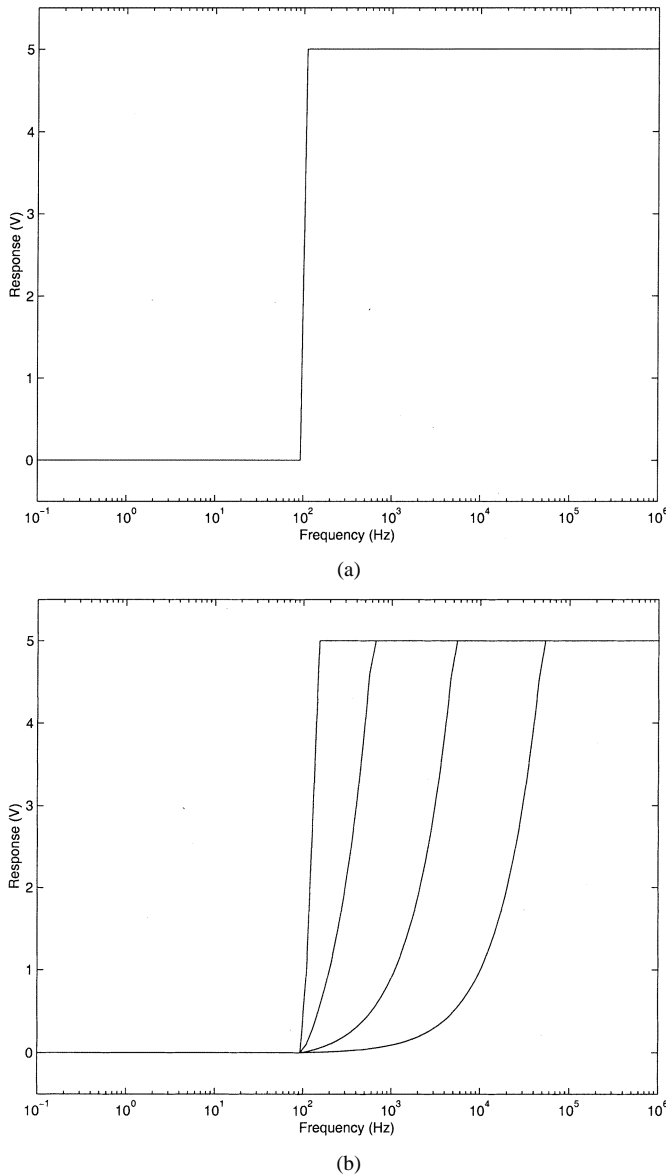


Fig. 21. Theoretical receiver pixel performance. (a) Infinite number of events at each frequency. (b) Finite number of events at each frequency; curves for four different event counts are plotted. The sharpness of the threshold increases with the number of events received.

V_{mem} will rise with each event and eventually saturate near V_{dd} . For a net negative current, the potential will fall with each event and saturate near V_{offset} . Thus given an infinite number of events, the output of the circuit [Fig. 21(a)] is a binary comparison of the input event frequency with a threshold event frequency

$$f_{\text{thr}} = \frac{I_{\text{down}}}{Q_{\text{spk}}}$$

which would make the net current exactly zero. This frequency is affected both by biases V_{tau} and V_{qua} .

Let us now consider a more realistic case [Fig. 21(b)] in which a finite number N of events arrives at a given event frequency. If the event frequency is very far from the threshold, V_{mem} will still saturate at one of the rails. However, if the event frequency is close to the threshold, the potential will change less. If we

assume that V_{mem} is allowed to leak down to V_{offset} between each stimulus presentation, we can expect a significantly softer thresholding operation to be performed for a finite number of events. In fact, the sharpness of the thresholding operation is proportional to N .

Under medium bus load conditions, the receiver chip consumes a static power of less than 3 mW at 5 V.

D. System-Level Hardware Details

The board on which all of the above components were combined is shown in Fig. 22. The sender chip is at left underneath a camera lens which serves to focus an image of the stimulus upon it. The rest of the board is spatially layed out much like Fig. 11.

To reduce power consumption, the EPROMS are held in standby mode until a request is generated by the sender chip. Because they are only active for the period that the request line is high, their power consumption depends upon the amount of activity on the AER bus, and thus average power consumption is quite low. Because of the use of CMOS components for the C-element and address multiplexer, the discrete logic circuitry on the board draws power only when changing states and thus also has a low average power consumption.

In order to implement the desired spatial motion integration algorithm through binary innervation matrices, the routing processor (currently a microcontroller running at 33 MHz) is programmed to receive events from each of the transceivers and transmit only selected ones from a preprogrammed map to specified units on the receiver chip. The summing operation is performed implicitly by integration of the combined train of events arriving at any receiver pixel. For example, to implement a given set of binary innervation matrices, any event arriving from a transceiver pixel with nonzero innervation matrix entry would be sent to a given receiver pixel. This pixel converts the total event rate at its input into a voltage, with no knowledge of the origin of each individual event, thus reflecting the desired sum. No sequential logic is required to perform this task, and thus an EPROM might implement the routing processor to support a *single set* of innervation matrices.

However, if multiple sets of innervation matrices are to be simultaneously supported, a sequential processor is required. For example, to implement simultaneous expansion and rotation innervation matrices, a single transceiver event might be sent on to as many as two different receiver pixels. In general, to implement M sets of innervation matrices, the AER bus at the output of the routing processor may send out as many as M events for a single transceiver event. While a sequential processor is required, the full complexity of a microcontroller is not. As Häfliger [57] has shown, an asynchronous FPGA may be used to implement this function quite efficiently. However, in whatever manner the routing processor is implemented, it represents a significant bottleneck in the system.

By clever selection of the wide-field motion templates that are simultaneously implemented, it is possible to minimize the bus slowdown out of the routing processor. For example, innervation matrices for expansion and contraction are completely nonoverlapping, yielding no bus slowdown when implemented simultaneously. Clockwise and counterclockwise rotation are similarly nonoverlapping. When expansion, contraction, and both direc-

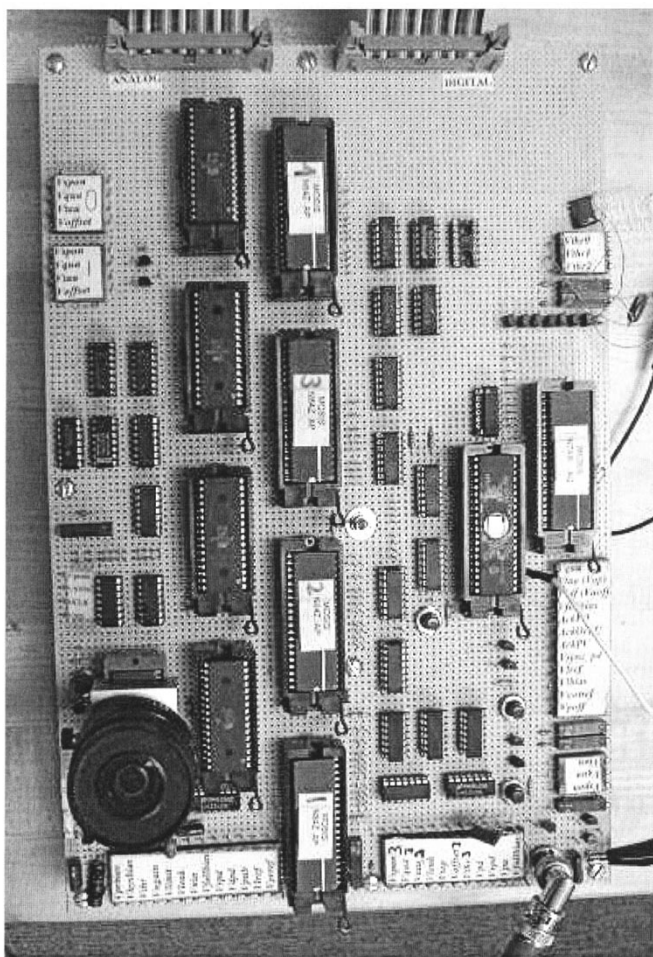


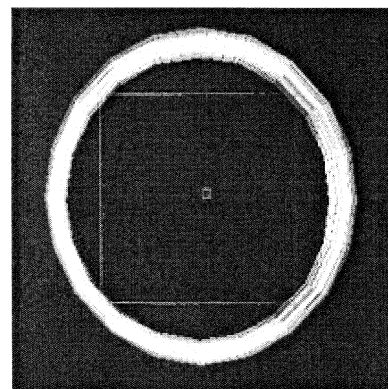
Fig. 22. Wirewrap board implementation of the spatial motion integration system. The sender chip is to the lower left covered by a camera lens. The rest of the board is laid out like Fig. 11 with, from left to right, columns of EPROMs and transceivers, the microcontroller, and the receiver chip.

tions of rotation are implemented simultaneously, there are only twice as many events out of the routing processor as come in.

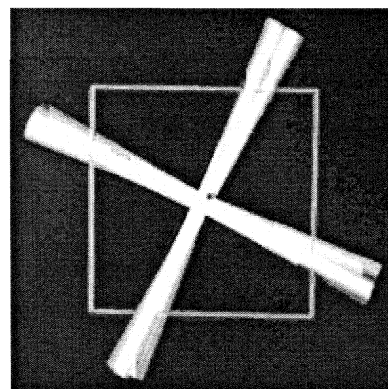
VI. EXPERIMENTAL RESULTS

Unlike the theoretical EMDs analyzed in Section IV, the hardware implementation requires moving images to be visually presented in order to produce motion outputs. In order to demonstrate the performance of our hardware system most clearly, we presented sequences of images such as expanding circles, rotating wagon wheels, and moving bars which generate optical flow patterns similar to those which might be generated in simple self-motion situations.

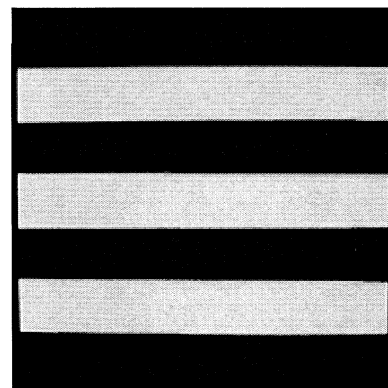
Computer-generated image sequences were presented on an LCD screen for reduced flicker. A compound lens was used to focus an image of the LCD screen onto the photosensitive sender chip. Stimuli presented include those shown in Fig. 23. The singular point (axis of rotation or focus of expansion) of each generalized spiral stimulus (expansion, contraction, rotation) could be moved around a rectangular grid covering the entire visual field of the chip. The output of the system was obtained from an appropriate number of receiver chip pixels through serial scanners.



(a)



(b)



(c)

Fig. 23. Visual stimuli presented to the self-motion system. (a) Expanding (or contracting) circles. (b) Rotating “wagon wheel.” (c) Translating bar grating. The large box in (a) and (b) indicates the visual field of the chip, around which the singular point could be moved.

Because full characterization of even one configuration of this system requires quite a lot of data, we focus on a single rather complex configuration. We support eight sets of innervation matrices, simultaneously synthesizing units on the receiver chip tuned for expansion, contraction, both directions of rotation, and four directions of translation. In this experiment, the bias V_{sp0n} is set to zero so that the motion transceiver EMDs produce no output when not sensing motion and respond positively for motion in their preferred direction. This makes the EMD tuning curve look very much like our prototypical EMD from Fig. 3 with $\theta_b = 180^\circ$. For this reason, each

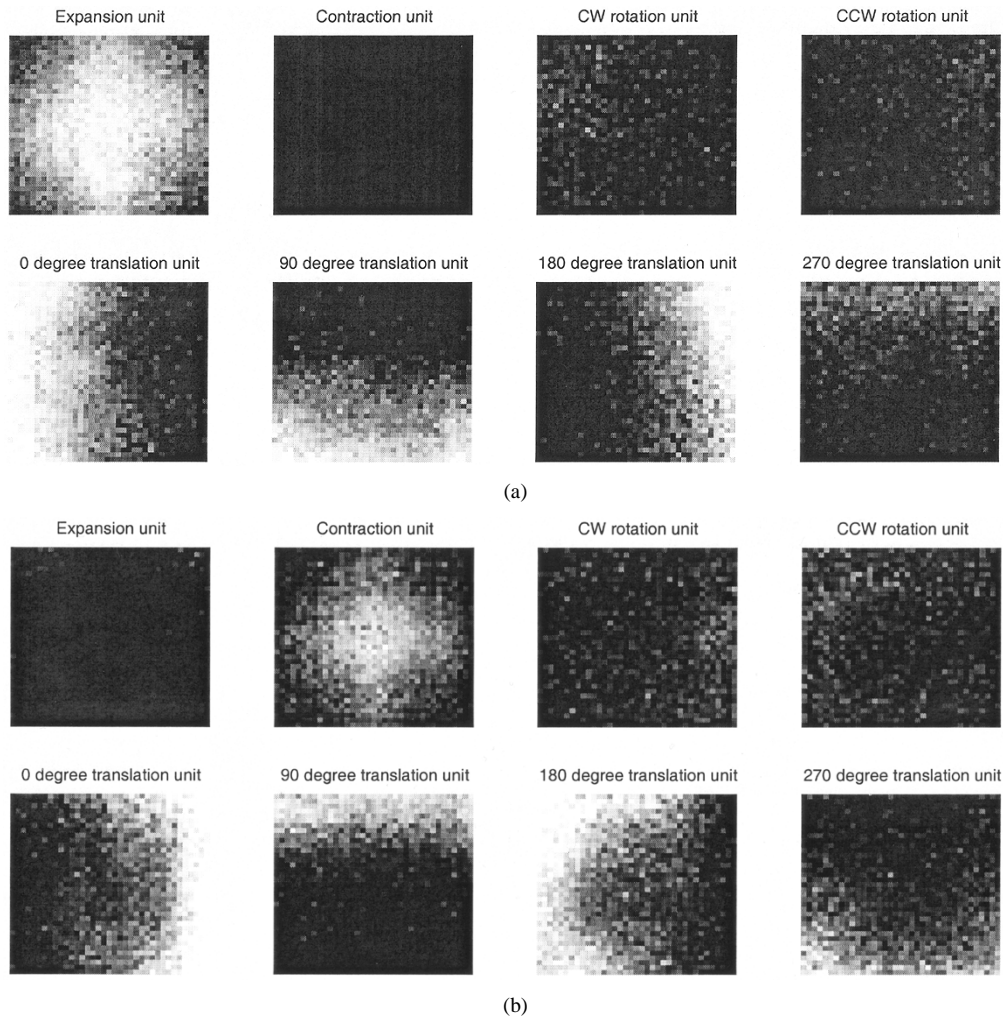


Fig. 24. Experimental results from board-level hardware system. Eight templates are simultaneously implemented: expansion, contraction, clockwise (CW) rotation, counterclockwise (CCW) rotation, and the four cardinal directions of translation. Shown are the responses of all eight templates to generalized spiral motion patterns with variable singular points. The spatial extent of each data figure corresponds to the position of the singular point in the visual field of the system. Brighter shading indicates stronger responses. (a) Expanding stimulus. (b) Contracting stimulus. (Continued in Fig. 25.)

generalized spiral innervation matrix covers exactly half of its associated transceiver chip. Translational units sum the outputs from one entire transceiver chip. To make the threshold for the translation-tuned units comparable to that of the generalized spiral-tuned units (as explained in Section IV), it is necessary to send two events to each receiver chip translational unit for every one received from the corresponding transceiver chip. When all eight types of pattern are simultaneously implemented, the routing processor produces four times as many events as the transceivers transmit.

The output of each of the eight receiver chip units is shown in Figs. 24 and 25 for a full set of generalized spiral stimuli with singular points located in a regular grid around the visual field. The receiver chip threshold is set so that the each unit can clearly distinguish the flow field type for which it is tuned not only from other flow field types, but also from the same flow field type with a significantly displaced singular point. Note that, as predicted in Section IV-C, the translation-tuned units respond to the generalized spiral stimuli at specific extreme singular point locations. This is due to the fact that, as the singular point of any of the generalized spiral patterns approaches the edge of

the visual field, it becomes indistinguishable (to sensors with $\theta_b = 180^\circ$) from a pure translational pattern. For example, in an expanding pattern with FOE on the left edge of the visual field, every motion vector has a component of motion within 180° of pure rightward translation. This fact yields the response on the left edge of the zero-degree-tuned unit in Fig. 24(a). The effects of receiver pixel mismatch are visible in the differences between the shape of the various responses.

Fig. 26 shows responses of all eight units to translating patterns. The generalized spiral stimuli show no significant response to these patterns. The translation-tuned units respond in the direction of their tuning over nearly 180° of stimulus angle.

VII. DISCUSSION

We have presented a computational architecture which has the ability to simultaneously correlate the visually presented pattern of wide-field motion with a number of templates expressed in the form of sets of innervation matrices.

The purely feedforward model presented has been shown to discriminate wide-field optical flow patterns of expansion, con-

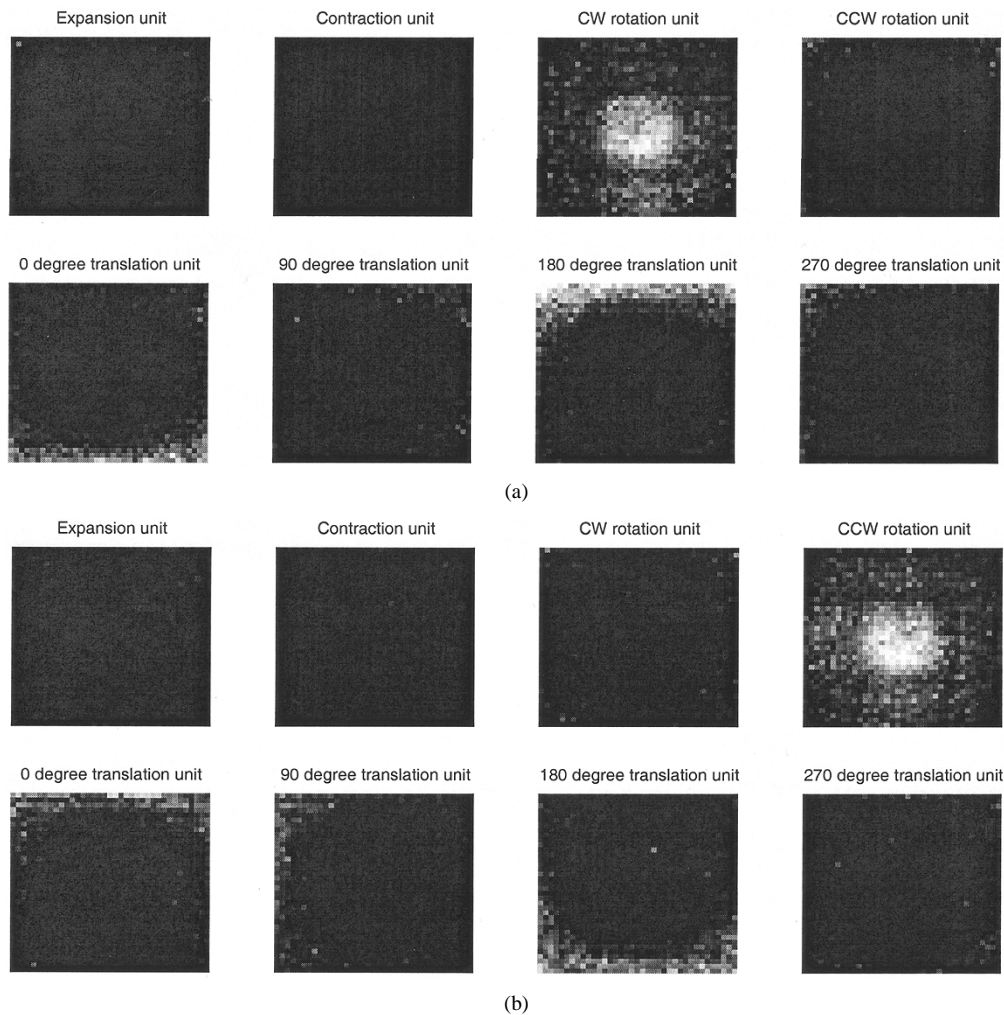


Fig. 25. Experimental results from board-level hardware system. Eight templates are simultaneously implemented: expansion, contraction, clockwise (CW) rotation, counterclockwise (CCW) rotation, and the four cardinal directions of translation. Shown are the responses of all eight templates to generalized spiral motion patterns with variable singular points. The spatial extent of each data figure corresponds to the position of the singular point in the visual field of the system. Brighter shading indicates stronger responses. (a) CW rotating stimulus. (b) CCW rotating stimulus. (Continued from Fig. 24.)

traction, translation, and rotation with a fixed threshold. For smaller EMD bandwidths (particularly $\theta_b < 90^\circ$), discrimination is made considerably easier by the fact that many patterns which do not match the template produce no response at all. In the worst case ($\theta_b = 180^\circ$), there is only a factor of two between outputs corresponding to matching and nonmatching stimuli.

Because the presented architecture is linear up to the point of the thresholding operation, these optical flow patterns can also be detected when presented in linear combination, as in more realistic self-motion scenarios. This ability would allow a moving camera system to identify the kind of motion it is currently undergoing. However, our discussion in this paper does not address the speed of such motion because speed tuning in our EMDs was left out for simplicity. In any implementation, the speed of the moving pattern does of course play a role in the output of the system. The response of this system would be graded in speed in the same way that the EMDs themselves are, allowing a less binary estimation of self-motion parameters.

In a self-motion application, a sensor of this type would produce multiple simultaneous outputs continuously indicating the correlation of the current wide-field motion pattern with the prescribed set of templates. The choice of template set is clearly

key to the functional usefulness of this system, and must be made based both on the *specific platform* on which the sensor will reside and the *types of self-motion* which are important to detect. These choices would be quite different, for example, between terrestrial and airborne platforms. With a modest set of carefully chosen templates, a simple maximum operation would allow a computation of the most likely self-motion within the given possibilities; the value of this maximum is proportional to the certainty of the self-motion classification. A more complex set of templates together with a linear combination of template outputs could also estimate continuous parameters of platform motion such as heading direction and translation velocity.

The spatial resolution required for practical application of the proposed system to self-motion estimation would depend on the spatial environment of the platform and the speed at which it must move. High spatial resolution is clearly required to resolve small moving objects at large distances from the imager, which would be required for a high-speed airborne platform. However, a terrestrial robot moving relatively slowly in an indoor environment would have much less stringent requirements. The number of motion templates required and their complexity would depend similarly on the specific problem.

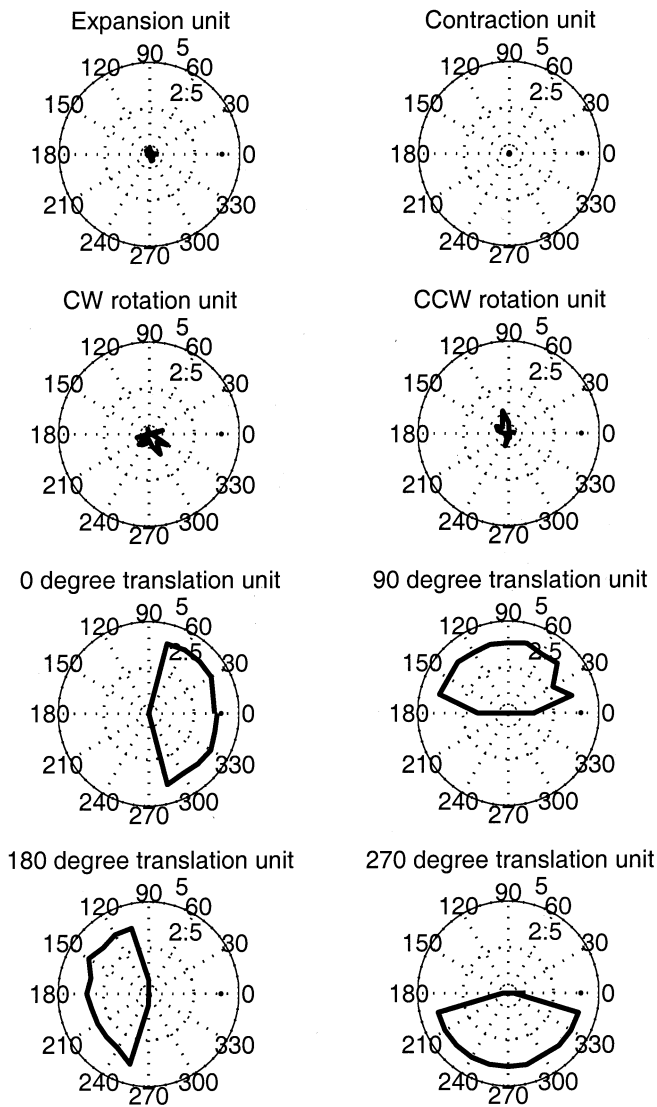


Fig. 26. Experimental results from board-level hardware system. Eight templates are simultaneously implemented: expansion, contraction, clockwise (CW) rotation, counterclockwise (CCW) rotation, and four cardinal directions of translation. Shown are the responses of all eight templates to translational motion patterns with the direction of translation varied 360° . Plot has been scaled to S_{im} .

The possible applications of this system extend well beyond self-motion estimation. Spatial integration of smaller regions of motion with inhibition from wide-field units could be used to implement localized detectors that respond only to small moving targets [58]. By making specialized patterns of connections between small-field motion units, small moving targets may be acquired and tracked [59]. By addition of subtractive as well as additive inputs to the integration stage, the signal-to-noise ratio could be increased by adding to each innervation matrix negative entries where there are presently zeros [26]. In addition, it would be possible to implement a “center-surround” motion field by subtracting from each motion output the activation of its neighbors such that only motion *discontinuities* are highlighted in the final stage. Further, if the innervation matrix pattern were allowed to change dynamically, it would be possible to tune the pattern of desired motion to a specific target as it moves across the visual field.

We have also shown a mixed-signal custom VLSI hardware implementation of the computational architecture presented above, and demonstrated its ability to discriminate wide-field spatial patterns of visual motion including expansion, contraction, rotation, and translation which are relevant to self-motion. The precision with which the implementation can discriminate flow fields is limited by the nearly 180° width of the direction tuning of the EMDs used. As the theoretical discussion showed, an EMD with narrower θ_b would allow a larger difference in system output between matching and nonmatching patterns.

This system allows the real-time processing of visual motion with modest requirements for power, weight, and physical size. Because the custom VLSI building blocks are employed in a multichip architecture, the system possesses reconfigurability (embodied in the EPROMs and routing processor) superior to a monolithic implementation. While a significant number of individual components make up the system, each operates with very low average power consumption largely because electrical operation is asynchronously *data driven*, not synchronously clock driven like conventional serial computer implementations. The physical size and weight of the present board is driven by standard 40-pin DIP packages on the VLSI components which could be easily replaced with more space-efficient packages. Currently, commercial EPROMs are employed, but an EPROM could simply be integrated into each motion transceiver chip. Similarly, the address multiplexing logic could be integrated into a custom routing processor, resulting in a much more physically compact system. Further, even better performance could be achieved at the cost of some flexibility by implementing the entire system on a single chip [60]. This would improve speed, reduce power consumption, and attenuate many noise issues. In such a monolithic implementation, the AER bus could still be used to achieve “virtual wiring.”

Because of the fully parallel implementation strategy used in all of the custom VLSI elements, increasing the pixel resolution to more practical values requires only duplication of existing processor elements. Power consumption will scale sublinearly from the values given as the number of elements increases due to power consumption by peripheral circuitry which does not grow, or grows slowly, as resolution increases. The resolution of the current system is driven only by what can be fit onto a MOSIS “tiny chip.” With the current pixel design, systems with a spatial resolution of more than 128 by 128 pixels are realizable through MOSIS.

The implementation of eight templates for generalized spiral and translational spatial patterns of motion is meant to clearly demonstrate the capabilities of the present system, but will not lead to the best possible performance in estimation of self motion in real-world scenes. Better performance can certainly be achieved with the more involved “optimal” template set derived by Franz *et al.* [22] or by using sets of patterns as suggested by the work Duffy and Wurtz [8].

The primary limitation of the current implementation is the routing processor, implemented at present with a microcontroller. It is the only clocked component of the system and draws considerably more power than the nonclocked components. Together with its clock generation circuitry, it draws 208 mW at 5 V. Because its operation is rather slow

compared to the operation of the asynchronous digital elements in the system, the speed at which it can produce events on the receiver bus limits the number of simultaneous units that can be synthesized by the system. The event rate at this juncture is not very high relative to the AER bus bandwidth, and thus an FPGA implementation [57] would certainly improve the speed at this bottleneck and allow system performance to be improved.

ACKNOWLEDGMENT

The authors would like to gratefully acknowledge the assistance of K. Boahen in providing and maintaining his implementation of the AER protocol and would also like to thank G. Indiveri, J. K. Douglass, and the anonymous reviewers for thoughtful comments on the manuscript. Thanks also to N. J. Strausfeld, Regents Professor of Neurobiology, Entomology, Ecology and Evolutionary Biology, and Anatomy at the University of Arizona for technical correspondence. This research was begun while one of the authors, C. M. Higgins, was a post-doctoral research fellow in the laboratory of C. Koch at the California Institute of Technology, where he was supported by the Caltech Center for Neuromorphic Systems Engineering as a part of the National Science Foundation's Engineering Research Center program, as well as by the Office of Naval Research.

REFERENCES

- [1] S. Ullman, "Analysis of visual motion by biological and computer systems," *Computer*, vol. 14, pp. 57–69, 1981.
- [2] M. R. Ibbotson, "Identification of mechanisms underlying motion detection in mammals," in *Motion Vision: Computational, Neural, and Ecological Constraints*, J. M. Zanker and J. Zeil, Eds. New York: Springer, 2001, pp. 57–65.
- [3] B. Hassenstein and W. Reichardt, "Systemtheoretische analyse der Zeit-, Reihenfolgen- und Vorzeichenbewertung bei der Bewegungspertzeption des Rüsselkäfers *Chlorophanus*," *Zeits. Naturforschung*, vol. 11b, pp. 513–524, 1956.
- [4] E. Adelson and J. Bergen, "Spatiotemporal energy models for the perception of motion," *J. Opt. Soc. Amer. A*, vol. 2, no. 2, pp. 284–299, 1985.
- [5] K. Hausen, "The lobula-complex of the fly: Structure, function, and significance in visual behavior," in *Photoreception and Vision in Invertebrates*, M. Ali, Ed. New York: Plenum, 1984, pp. 523–599.
- [6] H. Krapp and R. Hengstenberg, "Estimation of self-motion by optic flow processing in single visual interneurons," *Nature*, vol. 384, pp. 463–466, 1996.
- [7] H. Saito, M. Yukie, K. Tanaka, K. Hikosaka, Y. Fukada, and E. Iwai, "Integration of direction signals of image motion in the superior temporal sulcus of the macaque monkey," *J. Neurosci.*, vol. 6, pp. 145–157, 1986.
- [8] C. Duffy and R. Wurtz, "Response of monkey MST neurons to optic flow stimuli with shifted centers of motion," *J. Neurosci.*, vol. 15, no. 7, pp. 5192–5208, 1995.
- [9] R. Jain, "Direct computation of the focus of expansion," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, pp. 58–64, 1983.
- [10] B. Horn and E. Weldon, "Direct methods for recovering motion," *Int. J. Comput. Vis.*, vol. 2, pp. 51–76, 1988.
- [11] W. Burger and B. Bhanu, "Estimating 3-D egomotion from perspective image sequences," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp. 1040–1058, Nov. 1990.
- [12] A. Verri, M. Straforini, and V. Torre, "Computational aspects of motion perception in natural and artificial vision systems," *Phil. Trans. R. Soc. Lond. B*, vol. 337, pp. 429–443, 1992.
- [13] M. J. Barth and S. Tsuji, "Egomotion determination through an intelligent gaze control strategy," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, pp. 1424–1432, Sept./Oct. 1993.
- [14] J. S. Chahl and M. V. Srinivasan, "Visual computation of egomotion using an image interpolation technique," *Biol. Cybern.*, vol. 74, pp. 405–411, 1996.
- [15] C. Fermüller and Y. Aloimonos, "On the geometry of visual correspondence," *Int. J. Comput. Vis.*, vol. 21, no. 3, pp. 233–247, 1997.
- [16] J. J. Gibson, *The Perception of the Visual World*. Boston, MA: Houghton Mifflin, 1950.
- [17] D. Bradley, M. Maxwell, R. Andersen, M. Banks, and K. Shenoy, "Mechanisms of heading perception in primate visual cortex," *Science*, vol. 273, September 1996.
- [18] J. Perrone, "Model for the computation of self-motion in biological systems," *J. Opt. Soc. Amer. A*, vol. 9, pp. 177–194, 1992.
- [19] J. Perrone and L. Stone, "A model of self-motion estimation within primate extrastriate visual cortex," *Vis. Res.*, vol. 34, no. 21, pp. 2917–2938, 1994.
- [20] —, "Emulating the visual receptive-field properties of MST neurons with a template model of heading estimation," *J. Neurosci.*, vol. 18, no. 15, pp. 5958–5975, 1998.
- [21] K. Tanaka, Y. Fukada, and H. Saito, "Underlying mechanisms of the response specificity of expansion/contraction and rotation cells in the dorsal part of the medial superior temporal area of the macaque monkey," *J. Neurophys.*, vol. 62, no. 3, pp. 642–656, 1989.
- [22] M. Franz and H. Krapp, "Wide-Field, Motion-Sensitive Neurons and Optimal Matched Filters for Optic Flow," *MPIK*, 61, 1998.
- [23] —, "Wide-field, motion-sensitive neurons and matched filters for optic flow fields," *Biol. Cybern.*, 2000.
- [24] H.-J. Dahmen, M. Franz, and H. Krapp, "Extracting egomotion from optic flow: Limits of accuracy and neural matched filters," in *Motion Vision: Computational, Neural, and Ecological Constraints*, J. M. Zanker and J. Zeil, Eds. New York: Springer, 2001, pp. 143–168.
- [25] J. K. Douglass and N. J. Strausfeld, "Optic flow representation in the optic lobes of Diptera: Modeling the role of T5 directional tuning properties," *J. Comp. Physiol. A*, vol. 186, pp. 783–797, 2000.
- [26] —, "Optic flow representation in the optic lobes of Diptera: Modeling innervation matrices onto collators and their evolutionary implications," *J. Comp. Physiol. A*, vol. 186, pp. 799–811, 2000.
- [27] J. Tanner and C. Mead, "An integrated analog optical motion sensor," in *VLSI Signal Processing, II*, N. York, Ed. New York: IEEE Press, 1986, pp. 59–76.
- [28] T. Horiuchi, J. Lazzaro, A. Moore, and C. Koch, "A delay-line based motion detection chip," in *Advances in Neural Information Processing Systems*, R. Lippman, J. Moody, and D. Touretzky, Eds. San Mateo, CA: Morgan Kaufmann, 1991, pp. 406–412.
- [29] T. Delbrück, "Silicon retina with correlation-based, velocity-tuned pixels," *IEEE Trans. Neural Networks*, vol. 4, pp. 529–541, May 1993.
- [30] A. Moini, A. Bouzerdoum, A. Yakovlev, and K. Eshraghian, "A two dimensional motion detector based on the insect vision," in *Proc. Advanced Focal Plane Arrays and Electronic Cameras*, Berlin, Germany, Oct. 10–11, 1996, pp. 146–157.
- [31] X. Arreguit, F. van Schaik, F. Bauduin, M. Bidiville, and E. Raeber, "A CMOS motion detector system for pointing devices," *IEEE J. Solid State Circuits*, vol. 31, pp. 1916–1921, Dec. 1996.
- [32] R. Etienne-Cummings, J. Van der Spiegel, and P. Mueller, "A focal plane visual motion measurement sensor," *IEEE Trans. Circuit Syst. I*, vol. 44, pp. 55–66, Jan. 1997.
- [33] C. M. Higgins, R. A. Deutschmann, and C. Koch, "Pulse-based 2D motion sensors," *IEEE Trans. Circuits Syst. II*, vol. 46, pp. 677–687, June 1999.
- [34] R. Harrison and C. Koch, "An analog VLSI model of the fly elementary motion detector," in *Advances in Neural Information Processing Systems*, M. I. Jordan, M. J. Kearns, and S. A. Solla, Eds. Cambridge, MA: MIT Press, 1998, vol. 10, pp. 880–886.
- [35] R. A. Deutschmann and C. Koch, "Compact real-time 2-D gradient based analog VLSI motion sensor," *Proc. Int. Conf. Advanced Focal Plane Arrays and Electronic Cameras*, pp. 98–108, 1998.
- [36] A. Stocker and R. Douglas, "Computation of smooth optical flow in a feedback connected analog network," in *Advances in Neural Information Processing Systems*, M. S. Kearns, S. A. Solla, and D. A. Cohn, Eds. Cambridge, MA: MIT Press, 1999, vol. 11.
- [37] R. Sarpeshkar, J. Kramer, G. Indiveri, and C. Koch, "Analog VLSI architectures for motion processing: From fundamental limits to system applications," *Proc. IEEE*, vol. 84, pp. 969–987, 1996.
- [38] G. Indiveri, J. Kramer, and C. Koch, "Parallel analog VLSI architectures for computation of heading direction and time-to-contact," in *Advances in Neural Information Processing Systems*, D. Touretzky, M. Mozer, and M. Hasselmo, Eds. Cambridge, MA: MIT Press, 1996, vol. 8, pp. 720–726.

- [39] I. McQuirk, "An Analog VLSI Chip for Estimating the Focus of Expansion," Mass. Inst. Technol., Cambridge, Artif. Intell. Lab., 1577, 1996.
- [40] C. M. Higgins and C. Koch, "An integrated vision sensor for the computation of optical flow singular points," in *Advances in Neural Information Processing Systems*, M. S. Kearns, S. A. Solla, and D. A. Cohn, Eds. Cambridge, MA: MIT Press, 1999, vol. 11.
- [41] R. A. Deutschmann and O. G. Wenisch, "Compressive computation in analog VLSI motion sensors," in *Proc. Deutsche Arbeitsgemeinschaft für Mustererkennung*, 1998, pp. 564–571.
- [42] C. M. Higgins and C. Koch, "A modular multi-chip neuromorphic architecture for real-time visual motion processing," *Analog Integr. Circuits Signal Process.*, vol. 24, no. 3, Sept. 2000.
- [43] M. Mahowald, "VLSI analogs of neuronal visual processing: A synthesis of form and function," Ph.D. dissertation, California Inst. Technol., Dept. Comput. Neural Syst., Pasadena, 1992.
- [44] K. Boahen, "Retinomorph vision systems," in *Proc. Int. Conf. Microelectronics for Neural Networks and Fuzzy Systems*, 1996, pp. 2–14.
- [45] —, "A throughput-on-demand 2-D address-event transmitter for neuromorphic chips," in *Proc. 20th Conf. Advanced Research in VLSI*, Atlanta, GA, 1999.
- [46] —, "Retinomorph chips that see quadruple images," in *Proc. 7th Int. Conf. Microelectronics for Neural, Fuzzy and Bio-Inspired Systems*, Apr. 1999.
- [47] G. Indiveri, A. M. Whatley, and J. Kramer, "A reconfigurable neuromorphic VLSI multi-chip system applied to visual motion computation," in *Proc. 7th Int. Conf. Microelectronics for Neural, Fuzzy and Bio-Inspired Systems*, Apr. pp. 37–44, 1999.
- [48] S. R. Deiss, R. J. Douglas, and A. M. Whatley, "A pulse-coded communications infrastructure for neuromorphic systems," in *Pulsed Neural Networks*, W. Maass and C. M. Bishop, Eds. Cambridge, MA: MIT Press, 1998, ch. 6, pp. 157–178.
- [49] J. Kramer, R. Sarpeshkar, and C. Koch, "Pulse-based analog VLSI velocity sensors," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 86–101, Jan. 1997.
- [50] S. Shams, "A multichip neuromorphic motion processor for extracting egomotion information," M.S. thesis, Univ. Arizona, Dept. Elect. Comput. Eng., Tucson, 2000.
- [51] J. G. Elias, "Artificial dendritic trees," *Neural Computat.*, vol. 5, pp. 648–663, 1993.
- [52] A. Mortara, E. Vittoz, and P. Venier, "A communications scheme for analog VLSI perceptive systems," *IEEE J. Solid State Circuits*, vol. 30, pp. 660–669, June 1995.
- [53] Z. Kalayjian, J. Waskiewicz, D. Yochelson, and A. Andreou, "Asynchronous sampling of 2D arrays using winner-takes-all arbitration," in *Proc. IEEE Int. Symp. Circuits and Systems*, Atlanta, GA, pp. 393–396, 1996.
- [54] T. S. Lande, Ed., *Neuromorphic Systems Engineering: Neural Networks in Silicon*. Boston, MA: Kluwer, 1998.
- [55] T. Delbrück and C. Mead, "Analog VLSI phototransduction by continuous-time, adaptive, logarithmic photoreceptor circuits," in *Vision Chips: Implementing Vision Algorithms with Analog VLSI Circuits*, C. Koch and H. Li, Eds: IEEE Computer Soc. Press, 1995, pp. 139–161.
- [56] C. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.
- [57] P. Häfliger, "Asynchronous event redirecting in bio-inspired communication," *Proc. 8th IEEE Int. Conf. Electronics, Circuits, and Systems*, 2001.
- [58] A. Warzecha, M. Egelhaaf, and A. Borst, "Neural circuit tuning fly visual interneurons to motion of small objects. I. Dissection of the circuit by pharmacological and photoinactivation techniques," *J. Neurophysiol.*, vol. 69, no. 2, pp. 329–339, 1993.
- [59] W. Gronenberg and N. J. Strausfeld, "Descending pathways connecting the male-specific visual system of flies to the neck and flight motor," *J. Comparative Physiol. A*, vol. 169, pp. 413–426, 1991.
- [60] O. Landolt, *Place Coding in Analog VLSI – A Neuromorphic Approach to Computation*. Boston, MA: Kluwer, 1998.
- [61] N. J. Strausfeld, private communication, 2002.



Charles M. Higgins received the Ph.D. degree in electrical engineering from the California Institute of Technology (Caltech), Pasadena, in 1993.

He worked in the Radar Systems Group at MIT Lincoln Laboratory, Lexington, MA, until 1996, when he returned to Caltech as a Post-Doctoral Research Fellow in the Division of Biology. In 1999, he joined the Department of Electrical and Computer Engineering, University of Arizona, Tucson, as an Assistant Professor, where he holds a joint appointment in the Division of Neurobiology. His research is in the area of biologically inspired analog/digital VLSI vision and robotic systems. His research interests include analog computation, asynchronous digital inter-chip communication, hardware emulations of biological neural systems, and autonomous systems.



Shaikh Arif Shams received the M.S. degree in electrical and computer engineering from the University of Arizona, Tucson, in 2000. During his Master's research, he worked on developing a multichip neuromorphic motion processor for extracting egomotion information.

After graduation, he joined Intel Corporation, Chandler, AZ, as a Component Design Engineer. He is working on circuit design, RTL coding, and logic verification in the Madison project developing a 64-b architecture in the Itanium processor family. His research interests include analog/digital/mixed signal VLSI, neuromorphic systems, computation, and robotics.