

# Compact Analog VLSI 2-D Velocity Sensor

Rainer A. Deutschmann and Christof Koch

**Abstract**— We present an algorithm and its analog VLSI implementation for computing the spatially resolved two dimensional velocity of a moving visual scene. The algorithm is based on detecting and tracking intensity edges and works without clock in real time. The pixel-parallel implementation of our focal-plane sensor is space-efficient because of a low transistor count and minimal inter-pixel wiring. We present experimental results of a  $14 \times 17$  prototype array fabricated in a standard  $1.2 \mu\text{m}$  CMOS process. The sensor is found to compute stimulus velocity over a velocity range of more than two orders of magnitude. The velocity output is independent of stimulus contrast down to 11% contrast. A snapshot of the sensor output is to demonstrate the sensor operation with natural 2-D visual input. Due to a maximum power consumption of below  $50 \mu\text{W}$  per pixel even high-resolution sensors can be powered off a small battery and lend themselves ideally to mobile applications.

**Keywords**— Motion estimation, velocity sensor, focal plane sensor, smart vision sensor, parallel image processing, optical flow, analog VLSI, robot vision.

## I. INTRODUCTION

In automobiles an increasing amount of sensors and subsequent electronic signal processing is used to render driving ever more comfortable and safer. Sensing and processing of *optical* information is a promising but also difficult task. On the one hand automobiles inherently are operated in a fast changing and dynamic environment, which poses strong requirements on the real time performance of the sensor system. On the other hand a wealth of information can be obtained from the analysis of moving scenes. Including the time domain into image processing allows one to tackle tasks such as obstacle avoidance, ego-motion estimation, especially determining the heading direction and the time to contact as well as slip detection, further object tracking and projection of its trajectory, figure background segmentation, and recovering the 3-D velocity and 3-D structure of the viewed scene. The first step to solving these tasks is often to compute the so called optical flow field, which is an estimate of the perspective projection on the image plane of the 3-D velocity field.

The optical flow field has traditionally been computed on serial computers. For an overview over some algorithms cf. the comparative study of Barron et. al. [1]. It has proven difficult, though, to obtain the motion information in real-time, unless powerful computers were used. Additionally computing times scale unfavourably with the image size and usually increase

R. A. Deutschmann is with the Walter Schottky Institute, Am Coulombwall, 85748 Garching GERMANY. E-mail: Rainer.Deutschmann@wsi.tu-muenchen.de

C. Koch is with the Division of Biology, California Institute of Technology, Pasadena, CA 91125 USA

with  $\mathcal{O}(N^2)$ , where  $N \times N$  is the amount of image pixels.

In the last decade a new approach to motion computation has been paid attention to. Progress in VLSI technology for the first time allowed to implement motion detection algorithms on a custom designed chip in CMOS technology [2]-[15]. Using different algorithms, these motion sensors share the following features:

- They are single-chip sensors, i.e. the photoreceptors and the motion computation circuitry sit in the focal plane. Very compact motion detection systems are therefore possible.
- They are pixel-parallel implementations, i.e. motion computation is performed in synchrony by all pixels. Parallel computation avoids the bottleneck between photoconversion and further processing and also makes the system more tolerant against failure of individual pixels.
- No clock is required for motion computation. Since image irradiance is by nature continuous, an asynchronous circuit implementation is well suited to motion computation and avoids the negative effect of temporal aliasing otherwise encountered in computer implementations.
- In contrast to digital implementations, here transistors are used as analog computing elements. This allows for compact implementation of complex functions and filters. Subthreshold operation of the transistors greatly reduces the power consumption.

Existing analog VLSI motion sensors can be classified into *gradient-based* and *correlation-based* sensors. Gradient-based sensors use local temporal and spatial derivatives of the light intensity to compute motion. An early design of Tanner and Mead [2] tried to implement the gradient constraint equation through a feedback mechanism. The chip could only solve for one global 2-D velocity vector and showed poor performance. Deutschmann et. al. [14] implemented straightforwardly the 1-D gradient model, which yields velocity independent of spatial frequency and contrast by a division of temporal and spatial derivatives. In 2-D, though, the algorithm would yield a large pixel size. A 2-D sensor tuned to a fixed velocity was reported by Benson and Delbrück [5]. Most recently Deutschmann et. al. [15] have presented a 2-D gradient based sensor of similar compactness as the present chip. Due to the multiplicative algorithm, though, the motion output is dependent on the shape of the stimulus.

On the other hand correlation-based sensors generally look for features, such as intensity edges, in the visual field and track them over time. Etienne-Cummings et. al. [9] have reported a  $5 \times 5$  sensor, the scaling of which to large arrays might be difficult be-

cause the architecture is not fully parallel. In 1996 Kramer et. al. have introduced a powerful feature detector, namely a temporal edge detector based on which they developed three different 1-D velocity sensors [8],[10]. Their FS sensor might show the best overall performance to date, but no single chip 2-D sensor has been reported, partly because of algorithmic reasons, and partly because of a too large pixel size. Based on a similar feature extraction stage Deutschmann et. al. [12] have reported two more compact sensors that compute the direction-of-motion vector field in a 2-D array, but are not sensitive to stimulus velocity.

Along these lines we have developed the *Facilitate and Compare* (FC) sensor that we present in this paper. The FC sensor is the first fully pixel-parallel, focal-plane analog VLSI velocity sensor that combines the above mentioned advantages of a monolithic CMOS integration with a robust 2-D operation and has the potential of pixel resolutions beyond  $80 \times 80$  in a  $1.2 \mu\text{m}$  CMOS process. To the knowledge of the authors this is the first velocity sensor that requires only *one* communicating wire between every pair of neighbouring pixels, thus greatly reducing space-consuming inter-pixel wiring.

The paper is organised in the following way: In the next two sections we will introduce the motion algorithm and its analog VLSI implementation. In the following section we will present experimental results characterising the elementary motion detector; first the transit time output, then the output dependence on stimulus velocity, contrast and orientation. Finally we will show a  $14 \times 17$  motion vector field of the entire 2-D prototype pixel array to give an intuitive impression of the sensor performance with real-world input.

## II. ALGORITHM

Each pixel of the 2-D array consists of a photoreceptor, a feature detector, and, separately for X and Y direction, the motion computation circuitry. Since the X and Y components of the motion vector are computed independently, it is sufficient to subsequently describe the principle of operation in one spatial dimension.

The principle idea of the FC sensor is to measure the time it takes an image feature to travel between two adjacent pixels. This time will henceforth be called transit time. A feature detector which is sensitive to intensity edges is used in the FC sensor. Any other feature detector could be used instead. Using the correlation of events at two adjacent pixels is a concept already used in earlier designs [9], [10], [12] and is a special case of a correlation based method for motion computation. The FC sensor, though, implements a much simplified way of correlating events and naturally yields the transit time of the respective feature.

In Figure 1 two adjacent elementary motion detectors of the FC sensor are schematically displayed. Consider an intensity edge moving left to right. As the edge is detected at time  $t_A$  by the feature detector in

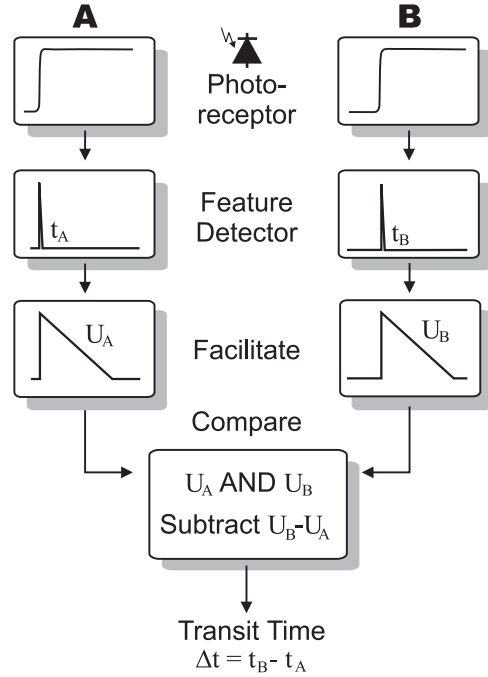


Fig. 1. Principle of operation of the FC sensor

the left pixel *A*, a signal  $U_A$  is initiated which starts at a certain value and subsequently decays over time. Similarly at the arrival of the edge at the second pixel *B*, an identical but time-delayed signal is initiated. In the comparison stage the difference  $U_B - U_A$  between those two signals is computed. Because of the linear decay of the signals, this difference is constant and a direct measure of the transit time  $\Delta t = t_B - t_A$  of the edge moving from pixel *A* to pixel *B*. For fast moving edges the transit time will be small whereas for slowly moving features the transit time will be large. In order to prevent null direction output during the time when the edge is moving between the two pixels, the transit time output is only allowed when both signals  $U_A$  and  $U_B$  are larger than zero. We therefore call these signals facilitation signals.

We will now discuss the output dependence of the FC sensor on the stimulus velocity  $v$ , c.f. Figure 2. We call the facilitation signal strength after initiation  $U_0$ , the time over which the signal decays  $\tau_{leak}$ , the time during which the transit time signal is reported (persistence time)  $\tau_{pt}$ , and the transit time itself  $\Delta t$ . Two ranges for  $\Delta t$  can be distinguished:

1. For inter-pixel transit times  $0 < \Delta t < \Delta t_{max}$  we find

$$\begin{aligned} \Delta U(t) &= \Delta U = \\ &= U_0 - U_0(1 - \Delta t/\tau_{leak}) = \\ &= U_0 \Delta t/\tau_{leak} \end{aligned} \quad (1)$$

where  $t_{max}$  represents the longest time for which the difference  $U_B - U_A$  can be computed by the comparison circuit. We find that  $\Delta U$  is constant during  $\tau_{pt}$ ,

proportional to the inter-pixel transit time  $\Delta t$  and thus inversely related to the stimulus velocity  $v$ :

$$\Delta t = \frac{\Delta U}{U_o} \tau_{leak} \quad (2)$$

$$v = a/\Delta t = \frac{1}{\Delta U} \frac{a U_o}{\tau_{leak}} \quad (3)$$

where  $a$  is the pixel spacing. The persistence time  $\tau_{pt}$  is given by

$$\tau_{pt} = \tau_{leak} - \Delta t \approx \tau_{leak} \quad (4)$$

because

$$\Delta t \ll \tau_{leak}. \quad (5)$$

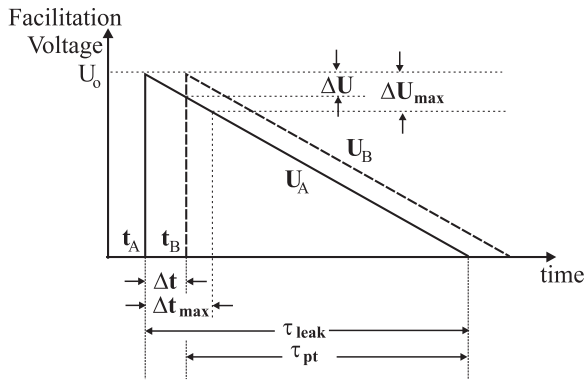


Fig. 2. The facilitation signals  $U_A$  and  $U_B$  of two neighbouring pixels decay over time resulting in a difference signal  $\Delta U$  linear in the transit time  $\Delta t$  of a moving feature

2. For inter-pixel transit times  $\Delta t > \Delta t_{max}$  differences  $\Delta U > \Delta U_{max}$  produce a constant maximal output. This is to prevent the output to grow too large. In this velocity regime the persistence time of the constant output pulse

$$\tau_{pt} = \tau_{leak} - \Delta t = \tau_{leak} - \frac{a}{v} \quad (6)$$

linearly decays with increasing transit time  $\Delta t$ . It follows that in the regime of very slow stimulus velocities the FC sensor pulse *width*  $\tau_{pt}$  codes inter pixel transit times and

$$v = \frac{a}{\tau_{leak} - \tau_{pt}} \quad (7)$$

If the stimulus moves so slow that the second pixel is reached after a time longer than the persistence time ( $\Delta t > \tau_{pt}$ ), no output will be produced.

### III. IMPLEMENTATION

One pixel consists of a photoreceptor, a temporal edge detector, a facilitation circuit and a comparison circuit. The former three circuits are required only once per pixel even in a 2-D implementation. Solely the comparison circuit receives input from neighbouring pixels and is required separately for X and Y direction.

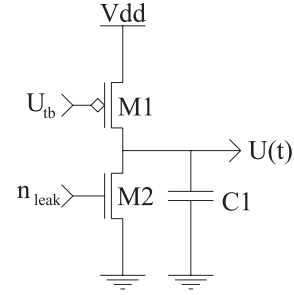


Fig. 3. The facilitation circuit is triggered by a voltage pulse  $U_{tb}$  from the temporal edge detector and generates a linearly decaying voltage signal  $U(t)$ .

We use the adaptive photoreceptor by Delbrück [16] to convert the incoming irradiance into a voltage signal. This CMOS compatible receptor works over 6 decades of illumination and has the desirable property to provide a continuous time output with low gain for static signals and high gain for dynamic signals.

The temporal edge detector (TED) [8] basically is a nonlinear temporal derivative circuit. It receives input from the photoreceptor and generates a short output voltage pulse for large enough negative illumination transients. Different from the design by Kramer et. al. we added to the output of the TED two coupled inverters in order to control the sensitivity of the TED and to obtain a digital output signal.

The facilitation circuit is shown in Figure 3. The inverted output of the TED  $U_{tb}$  is connected to transistor M1 such that capacitor C1 is charged up to  $V_{dd}$  when the TED responds with a pulse. The charge on the capacitor subsequently leaks off through transistor M2 such that the facilitation voltage  $U(t)$  linearly decays to GND within the time  $\tau_{leak}$ . The bias voltage  $n_{leak}$  allows to adjust the time  $\tau_{leak}$  and thus the time  $t_{max}$  which determines the velocity range to which the FC sensor is sensitive. A strong bias results in short leakage times. In this case the FC sensor will be maximally sensitive to fast stimulus velocities. Similarly for a weak bias the FC sensor will be able to distinguish slow velocities.

The comparison circuit (c.f. Figure 4) computes the difference between two facilitation voltages  $U_A$  and  $U_B$  and outputs a current linear in this difference only if both voltages are larger than a small threshold  $U_t \approx 0.7V$ . The circuit consists of a five transistor transconductance amplifier with additional two transistors M3 and M4 shutting it off when either one of the two facilitation voltages is below  $U_t$ , thus implementing a logical AND function. If both facilitation voltages are larger than  $U_t$  the output current of the comparison circuit is approximately given by

$$I_{out} \approx I_0 \frac{U_B(t) - U_A(t)}{U_k} \quad (8)$$

where  $U_k$  is a constant and  $I_0$  is a current defined by the bias voltage  $U_{diff}$ . The amplifier linear range of

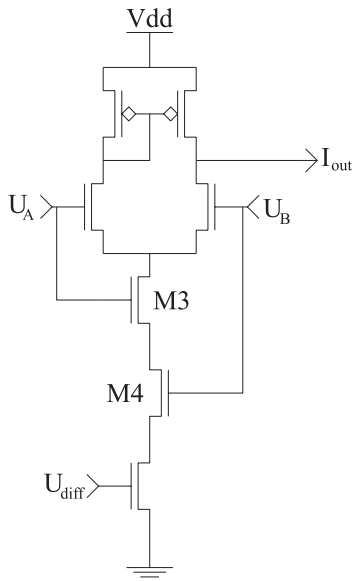


Fig. 4. The comparison circuit takes as input two facilitation signals  $U_A$  and  $U_B$  and generates the transit time output  $I_{out}$ .

about 100 mV corresponds to  $\Delta U_{max}$  defined earlier. The persistence time  $\tau_{pr}$  of the current pulse  $I_{out}$  is determined by the leakage time of the facilitation voltages. Outside of this linear range the amplifier saturates and thus generates a constant maximal output current pulse  $I_{max}$  with decreasing persistence time  $\tau_{pt}$  for increasing stimulus transit time.

The FC sensor chip was fabricated in a standard 1.2  $\mu\text{m}$  double poly double metal N-well CMOS process. The chip size of our prototype is 2.2 mm  $\times$  2.2 mm and the resolution is 14  $\times$  17 pixels. The pixel size is 109  $\mu\text{m}$   $\times$  103  $\mu\text{m}$ . A total of 35 transistors, one photodiode of area 16  $\mu\text{m}$   $\times$  15  $\mu\text{m}$  and a total capacitance of 2.57 pF are used per pixel. The total power consumption of one pixel with  $V_{dd}$  at 5 V is 20  $\mu\text{W}$  at steady state and maximal 50  $\mu\text{W}$  during the output of a transit time vector.

#### IV. EXPERIMENTAL RESULTS

We are now characterising the elementary motion detector. For that purpose one representative pixel of the array was selected and its output  $I_{out}$  for one spatial dimension was monitored. A small video lens of focal length 8 mm was placed above the open chip carrier and at a distance of 25 cm gray bars printed on a white sheet of paper were moved in front of the sensor with a motor setup. The stimulus speed, its contrast and its orientation could be well controlled. Stimulus speeds are given as on-chip speeds in units of pixels/sec. Contrast is defined by the ratio of the difference over the sum of two illuminations.

##### A. Transit time output

First we demonstrate the transit time output  $I_{out}$  generated as a response to stimuli moving in positive

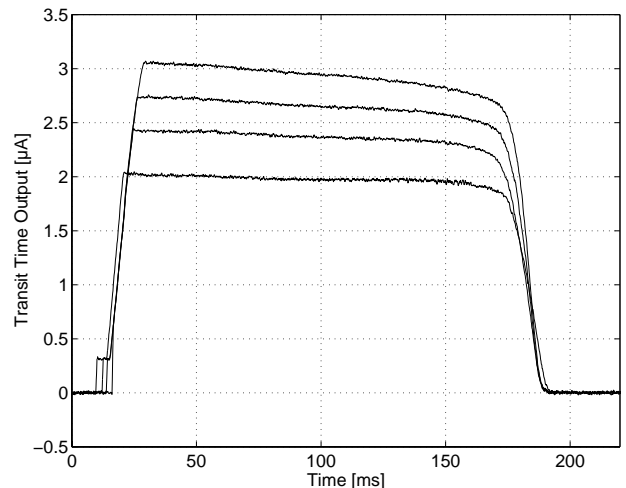


Fig. 5. Transit time output

X direction, c.f. Figure 5. As expected from the discussion in Section II the output  $I_{out}$  is generated when the stimulus passes, then stays at a constant positive value for a fixed time ( $\tau_{pt}$ ), and finally goes back to zero. For faster stimuli the pulse height of the output  $I_{out}$  is smaller, but the pulse width is always the same. In the second case for very slow velocities ( $\Delta t > t_{max}$ ) the pulse reaches a maximum height and the pulse width linearly decreases with increasing transit time (not shown here).

##### B. Velocity dependence

We now characterise the transit time output as a function of the stimulus velocity. Dark bars of 70% contrast were moved in front of the sensor in positive and negative X direction with speeds from 8 pixels/sec to 400 pixels/sec. For 20 consecutive stimulus presentations at every speed the transit time pulse height was recorded. It was found that for all speeds 100% of the stimulus presentations were detected by the sensor. In Figure 6 the mean and standard deviation of the inverse pulse height is plotted against the stimulus velocity. As predicted by Equation 3 a linear relationship between the inverse transit time output and the stimulus velocity is obtained for stimulus speeds between  $\pm 30$  pixels/sec and  $\pm 400$  pixels/sec. As expected for smaller velocities the inverse transit time pulse height becomes constant. According to Equation 7 in this regime the pulse *width* can be used to give information about the stimulus speed. In Figure 7 the data is shown together with the theoretical curve. Stimulus speeds down to 3 pixels/sec were correctly detected by the sensor.

An interesting feature of the FC sensor is that by virtue of the bias voltage  $n_{leak}$  the sensor can be optimally adjusted to the range of stimulus speeds that are expected. For example if the transit time pulse height is to be used,  $n_{leak}$  should be adjusted such

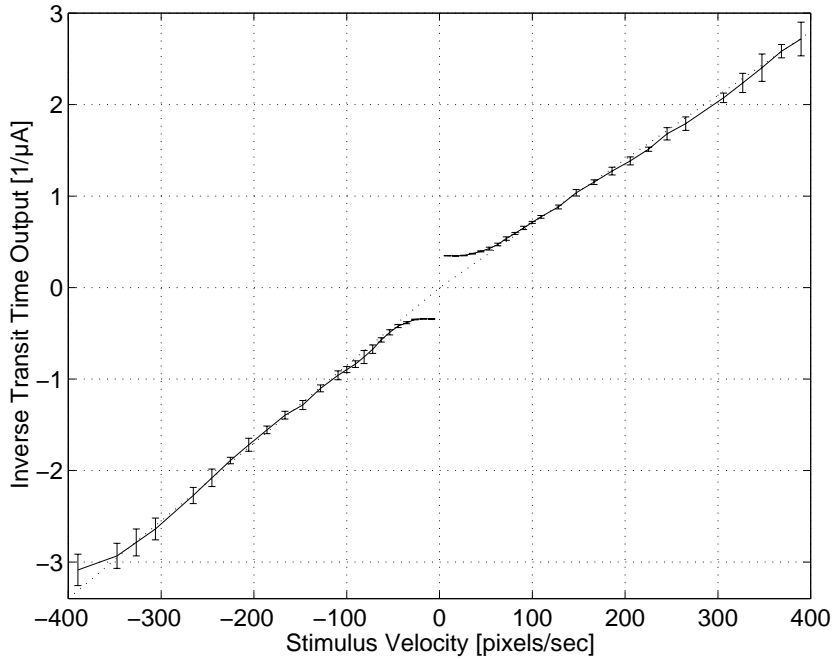


Fig. 6. Inverse transit time output for stimulus velocities between 8 and 400 pixels/sec

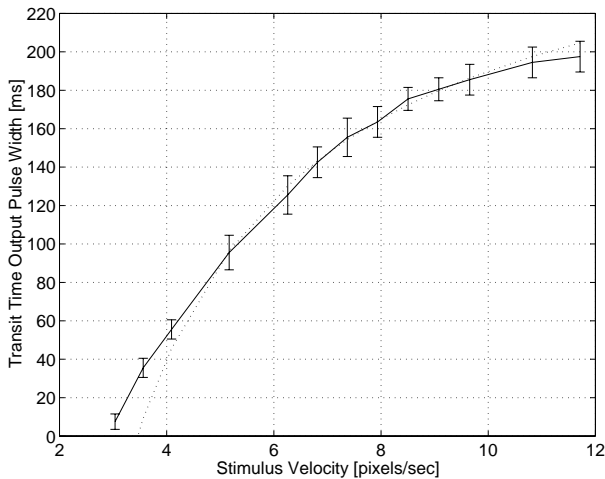


Fig. 7. Transit time pulse width output with theoretical fit

that the mean of all encountered stimulus velocities lies in the middle of the dynamic range of the FC sensor output. Automatically the persistence time is then adjusted as well. For predominantly slow stimulus velocities for example,  $n_{leak}$  should be small, resulting in a large  $\tau_{leak}$ . The sensor is then most sensitive to small stimulus speeds and the persistence time is long. For predominantly fast speeds the opposite is true.

### C. Contrast dependence

We have tested the FC sensor with stimuli of contrasts between 70% (black bar on white paper) and 4% (light gray on white) for various velocities. We have found that the FC sensor detects the stimulus with 100% reliability for contrasts down to 11%. In this range the mean transit time output is con-

stant, whereas its standard deviation increases towards smaller contrasts from 2% to 10%. The reason is that the appearance time of a low contrast edge can less reliably be detected by the temporal edge detector that unavoidably also amplifies noise, such as the 120 Hz flicker of AC lighting.

### D. Orientation tuning curve

It can be shown mathematically that two joint orthogonal pairs of pixels measuring transit time yield a vector whose length is inversely proportional to the stimulus velocity and whose orientation is normal to the local stimulus gradient. The resulting vector field thus is the inverse normal optical flow. We demonstrate that the FC sensor computes this type of flow field by using a bar stimulus of fixed velocity at orientations from 0 to 360 degrees. The transit time output for the X direction, shown in Figure 8, very well obeys the expected cosine law. The output in Y direction is phase shifted by 90 degrees.

### E. Sample flow field

The pixels of the FC sensor can be randomly accessed and the local transit time vector components as well as the local light intensity can be read out as bidirectional currents. We connected the FC sensor to a computer and displayed vector field and image as read from the chip. The frame rate of 42 Hz was limited by the speed of the display. In Figure 9 we show a snapshot as a hand was moved in front of the sensor. Most of the vectors point in the true direction of motion of the hand at an angle of 45 degrees. The so called aperture problem becomes apparent at the lo-

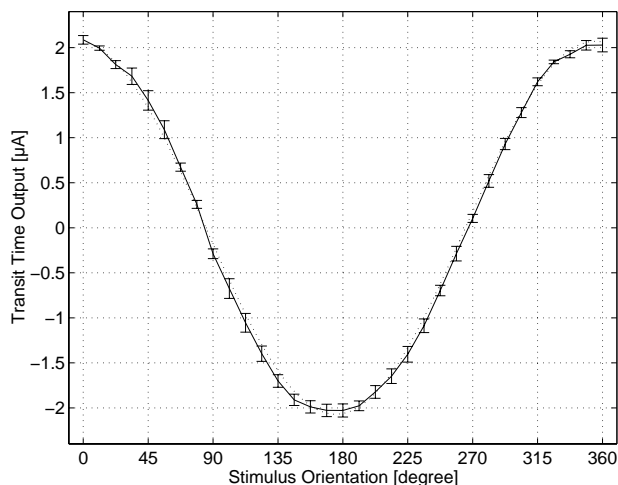


Fig. 8. Measured orientation tuning curve with theoretical fit of the thumb, where the pixels can locally only detect motion to the right. The trailing smaller vectors are only transients and are about to disappear because their persistence time is over.

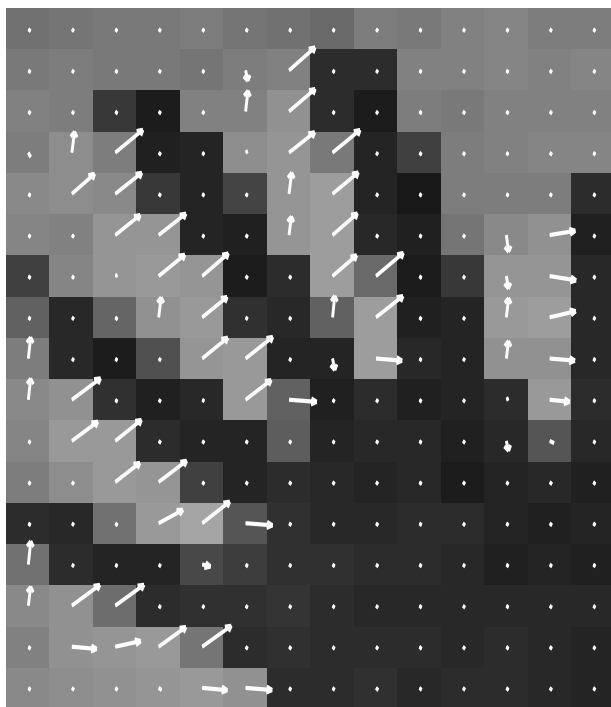


Fig. 9. Sample flow field

## V. CONCLUSIONS

We have presented a new algorithm for estimating two-dimensional image motion. We have cast this algorithm in analog VLSI CMOS circuits and built a real time focal plane velocity sensor. Experimental results demonstrate the robustness of the FC sensor under real world conditions. Our sensor is more com-

compact than previous designs with the same functionality. Using the same  $1.2 \mu\text{m}$  process on a chip size of  $9.4 \text{ mm} \times 9.7 \text{ mm}$  pixel resolutions of beyond  $80 \times 80$  pixels were possible, the design should also scale to smaller processes. As a low power, low cost integrated system the FC sensor can be used in various industrial applications and real time vision systems.

## ACKNOWLEDGEMENTS

R. A. D. wants to thank Dr. Charles Higgins for helpful discussions.

## REFERENCES

- [1] J.L. Barron, D.J. Fleet, and S.S. Beauchemin, "Systems and experiment: Performance of optical flow techniques," *Intern. J. Comp. Vis.*, vol. 12, pp. 43-77, 1994.
- [2] J. Tanner and C. Mead, "An integrated analog optical motion sensor," *VLSI Signal Processing, IEEE Press*, vol. 2, pp. 59-76, 1986.
- [3] A.G. Andreou, K. Strohhahn, and R.E. Jenkins, "Silicon retina for motion computation," *Proc. IEEE Int. Symp. on Circuits and Systems*, vol. 3, pp. 1373-1376, 1991.
- [4] A. Moore and C. Koch, "A multiplication based analog motion detection chip," *Proc. SPIE, Visual Information processing: From neurons to chips*, vol. 1473, pp. 66-75, 1991.
- [5] R. G. Benson and T. Delbrück, "Direction-selective silicon retina that uses null inhibition," in *Advances in Neural Information Processing Systems*, J. E. Moody, S. J. Hanson, and R. P. Lippmann, Eds. 1992, vol. 4, pp. 756-763, Morgan Kaufmann, San Mateo, CA.
- [6] T.K. Horiuchi, W. Bair, B. Bishofberger, A. Moore, and C. Koch, "Computing motion using analog VLSI vision chips: an experimental comparison among different approaches," *Intern. Journal of Computer Vision*, vol. 8, pp. 203-216, 1992.
- [7] T. Delbrück, "Silicon retina with correlation-based, velocity-tuned pixels," *IEEE Trans. on Neural Networks*, vol. 4, pp. 529-541, 1993.
- [8] J. Kramer, "Compact integrated motion sensor with three-pixel interaction," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, pp. 455-560, 1996.
- [9] R. Etienne-Cummings, J. Van der Spiegel, and P. Mueller, "A focal plane visual motion measurement sensor," *IEEE Trans. Circuits and Systems 1*, vol. 44, pp. 55-66, 1997.
- [10] J. Kramer, R. Sarpeshkar, and C. Koch, "Pulse-based analog VLSI velocity sensors," *IEEE Trans. Circuits and Systems II*, vol. 44, pp. 86-101, 1997.
- [11] A. Moini, A. Bouzerdoum, K. Eshraghian, A. Yakovlev, X. Nguyen, A. Blanksby, R. Beare, D. Abbott, and R. Bogner, "An insect vision-based motion detection chip," *IEEE J. of Solid-State Circuits*, vol. 32, no. 2, pp. 279-284, 1997.
- [12] R. A. Deutschmann, C. Higgins, and C. Koch, "Real-time analog VLSI sensors for 2-D direction of motion," in *Proc. Int. Conf. on Artificial Neural Networks ICANN'97. 1997*, vol. 1327 of *Lecture Notes in Computer Science*, pp. 1163-1168, Springer Verlag.
- [13] R. 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. 1998, vol. 10, MIT Press, Cambridge, MA.
- [14] R. A. Deutschmann and C. Koch, "An analog VLSI velocity sensor using the gradient method," in *Proc. IEEE International Symposium on Circuits and Systems ISCAS'98*, 1998, vol. 6, pp. 649-652.
- [15] R. A. Deutschmann and C. Koch, "Compact real-time 2-D gradient based analog VLSI motion sensor," in *Proc. International Conference on Advanced Focal Plane Arrays and Electronic Cameras AFPAC'98 Zurich*, 1998.
- [16] T. Delbrück and C. A. Mead, "Analog VLSI phototransduction," *Caltech CNS Memo*, vol. 30, pp. 139-161, 1994.