An Analog Motion Field Detection Chip for Image Segmentation

Ming-Han Lei and Tzi-Dar Chiueh, *Member, IEEE*

Abstract—In this paper, we propose an analog integrated circuit to compute the motion field of a time-varying image by means of a multiple-constraint method. The chip converts optical input to electrical form with an array of on-chip image sensors. A resistive network is used to smooth the input image. The spatial and temporal derivatives of the image are used to compute the optical-flow constraint. An array of motion cells enforce the optical-flow constraints, and two nonlinear resistive networks enforce the smoothness constraint over the optical-flow field. In order to preserve object boundaries, the smoothness constraint must be adjusted according to discontinuities in the optical-flow field and edges in the image. This is achieved by the nonlinear nature of the resistors and by adjusting the conductance of the resistors according to spatial gradients of image intensity. A 32×32 optical-flow based motion field detection chip is fabricated using a $0.5\text{-}\mu\text{m}$ CMOS process. Measurement results show that the proposed IC can compute the optical-flow field in a scene efficiently and correctly so as to facilitate segmentation of moving objects in image sequences.

Index Terms—Analog integrated circuits, image segmentation, object detection, optical flow, resistive network.

I. INTRODUCTION

TRADITIONAL video-coding standards such as MPEG-1 and MPEG-2 [1] derive much of their savings in coded bit rate from a block-based motion estimation for prediction of the next image frame from the current image frame. Therefore, much of the temporal redundancy in successive frames need not be coded, thus saving much of the bandwidth. As the drive for lower bit-rate video-coding continues, more redundancy needs to be removed. In MPEG-4 low bit-rate video-coding standard and more recent video-coding methods, object-based video coding is specified [2]–[4]. With computer-generated or synthesized video signals, objects are natural in their representations. However, for captured video signal made up of image sequences containing real scenes, segmentation of the image into regions belonging to different objects is a prerequisite for using the object-based video-coding technique.

In segmentation applications, the goal is to recognize which portion of the image corresponds to which object in the scene. Segmentation can be done from static image, but since most ob-

jects in a video image sequence are rigid objects, thus all pixels in an object most likely are moving with the same velocity, i.e., have uniform velocity. Therefore, by combining the motion field of an image and the image intensity itself, a segmentation method will be more robust in the case of texture-rich images [4], [5]. Without a motion field, texture in an image alone can be used to find the boundaries of the objects in the image. However, an object may consist of several parts, each having its own texture, and thus makes object boundary detection very difficult. On the other hand, with motion field of the image given, objects can easily be segmented since all pixels in an object will exhibit coherent motion field.

Algorithms to segment regions in an image sequence fall into two categories: 1) a correlation-based approach [6]–[8] and 2) a gradient-based approach [5], [9]. The correlation-based approach is more robust than the gradient-based approach, because it does not use differentiation that is likely to generate singular outputs from input images with noises. However, it is more computationally intensive and involves more pixels, and it is less tractable in hardware implementation since long-range wiring is usually necessary to carry the feature or intensity signals. On the contrary, the gradient-based approach has the potential of finding both the motion vector and can easily deal with 2-D images. After years of development and improvement, both types of algorithms grew quite sophisticated. Due to their complexity, most of these algorithms are only simulated in software and implemented on digital signal processors rather than special-purpose hardware.

Since motion-estimation algorithms are usually computationally expensive, techniques which can be implemented in parallel are very attractive, especially if image capturing and motion estimation can be integrated within a single chip. Because image intensity is a continuous-time vector signal, asynchronous analog circuit implementations are preferable to clocked digital implementations since temporal aliasing effects introduced by the sampling clock can significantly compromise the time-critical computations [10]. Thus, in this paper, we focus on continuous-time analog array processor implementing for motion-field detection.

There have been several analog hardware implementations of integrated image capture and motion detection chips aimed at real-time processing. Typically, these are VLSI analog array processors with on-chip image sensors. Many hardware implementations of motion estimation are based on the correlation of image intensity or of some temporal or spatial features to compute a 1-D motion field [11]–[13]. Another gradient-based motion sensor can compute only one motion vector instead of a motion field [14].

Manuscript received January 5, 2001; revised November 6, 2001. This work was supported in part by National Science Council, Taiwan, R.O.C., under Grant NSC87-2213-E-002-019. This paper was recommended by Associate Editor J. N. Hwang.

M.-H. Lei was with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan 10617, R.O.C. He is now with the Industrial Technology Research Institute, HsinChu, Taiwan, R.O.C.

T.-D. Chiueh is with the Department of Electrical Engineering and the Graduate Institute of Electronics Engineering, National Taiwan University, Taipei, Taiwan 10617, R.O.C. (e-mail: chiueh@cc.ee.ntu.edu.tw).

Publisher Item Identifier S 1051-8215(02)05262-X.

Some of the correlation-based motion field detectors can produce 2-D motion-field outputs [15], [16] but no real-scene 2-D motion fields were presented. Another gradient-based optical-flow field detector can produce 2-D optical-flow fields [17], albeit with blurring optical flow across region boundary. In this paper, we will present a 2-D optical-flow-based motion field detector with integrated image sensor array and readout circuit. The detector is based on a modified optical-flow approach that selectively smooth the 2-D optical-flow field so that optical flow discontinuity across region boundary can be preserved, which greatly facilitates further image segmentation task.

The rest of the paper is organized as follows. A description of the famous gradient-based motion field computation method—optical flow is given in Section II. Also a new modified smoothness constraint amenable to IC implementation is proposed in this section. Section III then illustrates architecture of the proposed motion field detector chip and presents detail circuit design of its main function blocks. In Section IV, we show the physical layout design of the IC and report its measurements. Finally, conclusions are given in Section V.

II. OPTICAL-FLOW COMPUTATION

A. Motion Field and Optical Flow

Motion field is a vector field formed from projecting the moving velocity of each point in a 3-D scene to the viewing plane. Therefore, the motion field is closely related to the scene and moving velocities. We can analyze and interpret the 3-D structure of the scene by means of the motion field. However, it is hard to find the motion field from the image itself. What we can find is the *optical flow*. Optical flow is the velocity of image intensity profile movement. Unlike motion field, the definition of optical flow only involves the image. Therefore, finding optical flow from the image sequence is possible. We will thus approximate the motion field by the computed optical-flow field.

B. Gradient-Based Approach

The gradient-based optical-flow-field estimation approach is based on the assumption of the conservation of local image intensity. A constraint equation is derived from this assumption. However, one constraint equation involving local image pixels is not sufficient for solving both horizontal and vertical components of the optical-flow vector at the corresponding location. More constraint equations must be included to resolve the ambiguity. Over-constrained systems are usually preferred, as redundant information enforces robustness over noises in the input [18]. Among many multiple-constraint gradient-based methods that have been proposed, we will review two methods that use the least mean square error constraint and some smoothness constraint [19], [20].

1) *Optical-Flow Constraint Equation*: Following the derivation of Horn and Schunck [19], we will briefly introduce the optical-flow derivation. Denote the image intensity at the pixel (x, y) in the image at time t as $I(x, y, t)$. Assume uniform and stationary illumination, and pure translational motion. When a particular point in the scene is moving, the intensity

of the corresponding point on the image will not change. Therefore, we can write

$$\frac{d}{dt}I(x(t), y(t), t) = 0. \quad (1)$$

This is called the *conservation of image intensity*. Applying the chain rule of differentiation to (1), one has

$$\frac{\partial I}{\partial x} \frac{dx}{dt} + \frac{\partial I}{\partial y} \frac{dy}{dt} + \frac{\partial I}{\partial t} = 0 \quad (2)$$

where dx/dt and dy/dt are the horizontal and vertical components of the optical flow at pixel (x, y) at time t , respectively. For conciseness, we can be further formulate them as $u = dx/dt$ and $v = dy/dt$. If we denote the partial derivatives of the intensity in subscript notation $I_x = \partial I/\partial x$, $I_y = \partial I/\partial y$, and $I_t = \partial I/\partial t$, (2) can be rewritten as

$$I_x u + I_y v + I_t = 0 \quad (3)$$

which is called the *optical-flow constraint equation*.

The three partial derivatives in (3) can be measured from the image in a local area around the pixel under consideration. The two components of the optical flow, u and v , are unknown variables. Thus, the optical-flow constraint provide only a linear equation in the variables u and v . As a result, these two variables can not be uniquely determined without more equations.

Another interpretation of the optical-flow constraint equation is to plot (3) on the u - v space. This will result in a straight line on the plane. For this reason, the optical-flow constraint equation is also referred to as the *intensity constraint line*. All points on the intensity constraint line are possible (u, v) solutions to the constraint equation. Again the need for extra constraint equations to determine a unique solution is obvious.

2) *Least-Mean-Square Error (LMSE) Criterion*: Once a set of local intensity derivatives are computed, the optical-flow constraint equation can be found. However, there is still ambiguity because the equation is under-determined. The ambiguity can be resolved by using another set of local intensity derivatives from a nearby pixel. For example, as shown in Fig. 1, a corner moves from the location shown in solid lines to the location shown in dash lines. We want to determine the optical flow at point A . The intensity derivatives at A defines an intensity constraint line C_A on the u - v plane. Another intensity constraint line C_B can be found from the derivatives measured at a neighboring pixel B . The intersection of these two lines uniquely determines the optical flow at A . However, if our field of view is confined to a small region, then we may very well see only points that have parallel intensity constraint lines (e.g., points A and C in Fig. 1), then a unique (u, v) solution is still impossible. This is the so-called *aperture problem* [21].

Therefore, only if the other constraint equation provided by a neighboring pixel is independent of the original one can we uniquely determine (u, v) . We generally cannot tell which point can provide an independent constraint equation and which can not, so we must ensure that all available neighboring information is included. If we define a neighborhood window for each pixel in the image, optical-flow constraint equations for all pixels in the window will form an over-determined system of equations. As in an under-determined system of

Fig. 1. Using neighboring constraint equations to determine the velocity.

equations, an over-determined equation system does not have a unique solution, hence usually the LMSE criterion can be applied to find the best-fit solution [14]. In other words, with several intensity constraint equations from the neighboring pixels, the optimal (u, v) is the one that minimizes

$$E_c = \sum_i (I_{x,i}u + I_{y,i}v + I_{t,i})^2 \quad (4)$$

which is the mean-squared error, and it is called the *MSE cost function*.

3) *Smoothness Constraint*: Apart from the LMSE criterion, the smoothness constraint is also widely used. The underlying assumption of the smoothness constraint is that the motion vectors of neighboring pixels are similar and vary smoothly. This assumption provides extra constraint equations for solving the optical flow.

Horn and Schunck used the squared magnitude of the spatial gradient of the optical flow as a measure of deviation from the smoothness constraint (*smoothness cost function*) [19]

$$E_s = \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2. \quad (5)$$

Enforcing the smoothness constraint on a optical-flow field is equivalent to minimizing the above measure. In their proposal, optical-flow-field computation is equivalent to a minimization problem of these two aforementioned cost functions simultaneously. Specifically, the function to be minimized is formulated as

$$E = \int \int (E_c + \alpha E_s) dx dy \quad (6)$$

and α controls the degree of smoothing. If α is very large, the resulting optical flow will mainly be subject to the optical-flow constraint. On the contrary, if α is close to zero, the optical flow will tend to be very smooth.

In [17], an optical-flow motion field detector based on the above smooth constraint using linear resistive networks with constant conductance was presented. The major drawback of this approach is that genuine motion boundaries are blurred if the smoothness constraint is strongly enforced while optical-flow homogeneity of pixels inside a region is impossible if the smoothness constraint is weakly enforced. In the former case, the smoothness constraint essentially propagates the optical-flow information into the area with static image intensity, where little or no velocity should be observed. In the latter case, all pixels in a region compute their motion velocities

individually and the resultant velocities often vary wildly due to texture and fluctuation in pixel intensity inside the region. This will cause problems in many situations, for example, when an object moves against a uniform background. In this case, it becomes difficult to distinguish between the velocities assigned to the object and to the background, because the optical flow is smoothed across the object boundary. In [17], the authors showed two different cases which required different smoothness strengths (one weak and one strong) for optimal optical-flow results. Therefore, it is only reasonable to modify the Horn and Schunck model to one in which the degree of smoothness constraint enforcement can be adaptively tuned according to local image intensity characteristics.

Nagel proposed one such model with a smoothness constraint that avoids the problem of blurring optical flow across region boundaries [20]. The smoothness constraint proposed by Horn and Schunck causes problems because the constraint is isotropic and homogeneous in the sense that it smooths the optical-flow field uniformly in all directions and at all places. To avoid the blurred-boundary problem, Nagel developed a smoothness constraint that does not impose uniform smoothness but selectively smooths the optical-flow field according to the image gradient. In the regions with strong second-order intensity variations, such as corners, the smoothness constraint is enforced quite weakly. In other words, the optical flow is allowed to be discontinuous. Also, in the vicinity of edges, e.g., regions with strong intensity gradient but weak second-order intensity derivative, the smoothness is enforced strongly along the direction of the underlying contour, and weakly across the contour. Because of these features, Nagel named this constraint *oriented smoothness constraint*. The oriented smoothness cost function is given by

$$E_{os} = \{(I_y u_x - I_x u_y)^2 + (I_y v_x - I_x v_y)^2\} + \sigma \{(I_{yy} u_x - I_{xy} u_y)^2 + (-I_{xy} u_x + I_{xx} u_y)^2 + (I_{yy} v_x - I_{xy} v_y)^2 + (-I_{xy} v_x + I_{xx} v_y)^2\} \quad (7)$$

where the subscripts indicate spatial partial derivatives. The first bracket corresponds to the squared magnitude of projection of the optical-flow spatial derivative along the direction that is perpendicular to the image gradient direction. The second bracket involves complicated second-order image gradients and scales with the sum of squared magnitude of the optical-flow spatial gradient component perpendicular to the curvature of the image intensity. The parameter σ determines the relative weight given to the two oriented smoothness measures. Solving a new minimization problem with E_{os} replacing E_s in (6) will give an optical-flow field that has minimal variation along directions normal to the image gradients and curvature.

4) *Selective Smoothness Constraint*: Although the problem of boundary blurring can be fixed to some extent by the oriented smoothness constraint, this method has its practical limitations. Computing the second-order spatial partial derivatives of the image intensity is inherently error-prone due to noise and quantization errors associated with any practical imaging system. Moreover, the computation complexity is much greater than that of the case using isotropic and uniform smoothness constraint. So in the proposed optical-flow based motion field

Fig. 2. Block diagram of the proposed optical-flow-based motion field detector.

detector, we adopt another circuit that simplifies the “selective smoothing” concept proposed by Nagel. This new method uses a smoothing mechanism whose strength is adaptive as in Nagel’s method. However, the strength depends only on the first-order-gradient (“edginess”) of the image intensity, not the complicated second-order image gradients. By this, we mean that whenever there is a large first-order gradient in the image intensity, it is more likely (not necessarily) a region boundary has been encountered; therefore, the optical flow across that boundary should be allowed to be dis-continuous and thus the smoothing constraint should be relaxed. The proposed selective smoothness cost function is given by

$$E = \int \int (E_c + \alpha(x, y) E_s) dx dy \quad (8)$$

where $\alpha(x, y)$ is now a variable and is a monotonic decreasing function of the local image gradient magnitude.

III. ARCHITECTURE AND CIRCUIT

The block diagram of the proposed optical-flow based motion field detector is shown in Fig. 2. For simplicity, the figure draws a 3×3 array. The chip actually implements a 32×32 array. The analog detector consists of five important functional blocks: 1) an array of photoreceptors which senses the input image; 2) a linear resistive network which smoothes the input image; 3) an array of motion-computation cells which imposes the optical-flow constraint; 4) two “edginess-adaptive” nonlinear resistive networks which impose the smoothness constraint in horizontal and vertical optical-flow components, respectively; and 5) a readout circuit for outputting the captured image and the computed optical-flow field. Note that the two nonlinear resistive networks (one for u and one for v) are connected similarly, so only one of them is shown. Furthermore, the conductance of individual nonlinear resistor in these two networks is locally

Fig. 3. Schematic of the photoreceptor and the linear resistive network.

Fig. 4. Block diagram of the proposed motion cell.

controlled by the first-order gradient of the image intensity at the corresponding location so that the degree of smoothing can be tuned properly.

A. Photoreceptor and Linear Resistive Network

The photoreceptor array and the resistive network sense the input image and smoothes the image spatially. As shown in Fig. 3, the parasitic p-n-p transistor in an N -well CMOS process, when reverse biased, can be used as a photo transistor. Incident light generates excess carriers in the depletion region of the transistor, thus producing a photo-current. Apart from the photo transistor, another bipolar transistor further amplifies the photo current, since the output signal level in normal room illumination is usually too low without further amplification. The amplified photo-current is then converted to voltage by a pair of diode-connected pMOS transistors.

The photoreceptor may introduce errors. One of the sources is the thermal noise; another is the mismatch among photoreceptors. The noise in the image will severely degrade the performance of the gradient-based motion-detection algorithms. Therefore, in many software implementations of the gradient-based algorithm, the input image is heavily smoothed by a spatial Gaussian low-pass filter to improve the robustness of the

Fig. 5. Circuit diagram of the proposed motion cell.

system. Similar effect can be achieved by connecting the photoreceptors with a linear resistive network [10]. The two pMOS transistors labeled M_x and M_y form such a network when the depicted photoreceptor is repeated to construct an image-capturing sensor array. The degree of smoothness enforced by the network can be controlled by the resistance of the resistors, which in turn is controlled by V_{sm} .

B. The Motion Cell

The motion cells follow the design in [10] and are used to enforce the optical-flow constraint at each pixel. The block diagram is shown in Fig. 4. The motion cell computes the deviation of the optical flow from the intensity constraint line and tries to minimize the MSE cost function (E_c), as defined in (4). If all optical-flow vectors fall on their respective intensity constraint lines, then E_c is exactly zero and it is minimized. If the optical flow at a pixel does not fall on its intensity constraint line, the corresponding motion cell shall adjust that optical flow so that it moves toward the line. The farther an optical-flow vector deviates from the line, the larger the adjustment should be. The following gradient-descent-based adjustment is applied:

$$(\Delta u, \Delta v) = -E_c \cdot \nabla I = -(I_x u + I_y v + I_t) \cdot (I_x, I_y). \quad (9)$$

Note from Fig. 4 that the two optical-flow components u and v are stored in the voltages of two local capacitors and the adjustments, Δu and Δv , are represented as two currents flowing into the two capacitors, respectively.

The motion computation cell circuit (see Fig. 5) is similar to the one proposed in [10] except for how the cells are connected and the extra low-gain amplifiers in the feedback loops for extending the output voltage range of the motion cell. Instead of joining all u and v nodes together and computing only one motion vector as in [10], the proposed optical-flow-based motion-field detector connects the motion cells through two nonlinear resistive networks. The motion cell is implemented with CMOS sub-threshold circuits. The multipliers are the conventional Gilbert type and the products are conveyed in the output currents, which can be easily summed up by joining output

nodes. The input range of such circuits for linear operation is only about 150 mV, peak to peak. Note that the motion cell consists of two negative feedback loops, i.e., the outputs u and v are also inputs to the circuit itself. As a result, the output range of the circuit is also limited to about 150 mV. Such a small operation range renders the circuit vulnerable to noise and mismatch. Also, in order for the nonlinear resistors to operate normally, the output range of the motion cells needs be several times larger than the linear range of the nonlinear resistors. Therefore, it is necessary to enlarge the output range of the motion cell. Using an amplifier whose gain is less than unity inserted in the feedback path, we can extend the motion cell output range to about 3 V, peak to peak.

C. Nonlinear Resistive Networks

In [17], it is shown that minimization of (6) can be implemented by two linear resistive networks connecting the motion cell array. The degree of smoothing is controlled by the conductance of the resistors in the two networks. To implement (8), we choose to adopt nonlinear locally-adaptive resistive networks over the linear globally-adaptive networks in [17]. The function of the two nonlinear resistive networks are two folds. First, they establish connections among the motion cells. Thus the motion cells can obtain information about the optical-flow constraint of their neighboring cells. This can help resolve the aperture problem. Second, they smooth the output optical-flow field by enforcing the selective smoothness constraint.

As mentioned previously, nonuniform oriented smoothing is preferable to uniform isotropic smoothing. In the proposed detector, the smoothing function is determined by the following rule: edges in the image and discontinuities in the optical flow should both weaken the smoothing strength, i.e., the smoothness constraint should be relaxed. We implemented nonuniform smoothing using two resistive networks with variable-conductance nonlinear resistors. In the networks, the underlying mechanisms for these two factors that affect the smoothness are different. Image edges influence the degree of smoothness by adjusting the conductance of the nonlinear resistors, whereas the reduced smoothness due to optical-flow discontinuity is a direct result of the nonlinearity of the resistors.

Fig. 6. Optical-flow outputs using uniform smoothing and different nonlinear resistors in the resistive networks. (a) HRes network. (b) Fuse network. (c) Tiny-tanh network.

Fig. 7. Optical-flow outputs using nonuniform smoothing. (a) HRes network. (b) Fuse network. (c) Tiny-tanh network.

Several nonlinear resistors can be found in the literature, such as horizontal resistance (HRes) [10], fuse [22], and tiny-tanh [23]. Before determining which kind of nonlinear resistor to use, we need to understand their effects on optical-flow computation. Some simulations are conducted to examine the behaviors of these three nonlinear resistors. An image with two moving objects is used as the input image in the simulation. Fig. 6 depicts the simulation results of three motion detectors using different nonlinear resistors and uniform smoothness, i.e., the resistance value of the resistors is fixed. On the other hand, Fig. 7 shows the results when nonuniform smoothing is adopted, that is, the resistance value is controlled by the image edge intensity. The performance of the HRes network and the tiny-tanh network improves with nonuniform smoothing. However, the performance of the fuse network deteriorates with nonuniform smoothing. In the latter, the optical-flow vectors near the object boundaries become very large. The reason for this phenomenon is that the discontinuity in the optical-flow field makes the fuses along object boundaries blown off and the motion cells near the boundaries are no longer connected to one another.

With the nonuniform smoothing scheme, the performance of the tiny-tanh network improves dramatically but the performance of the HRes networks is still not satisfactory. The difference between these two cases must be interpreted

carefully. Both HRes and tiny-tanh have hyperbolic tangent I - V characteristics. They differ only in the linear range of the transfer curves. The HRes resistor saturates at about 150-mV input, while tiny-tanh resistor saturates at only several millivolts. In fact, the output range of the motion cells are about 200 mV peak to peak, which is about the linear range of HRes. Therefore, the HRes networks always operate in its linear region, thus blurring object boundaries in the optical-flow field. If we extend the output range of the motion cell, the performance of the HRes network will be comparable to that of the tiny-tanh network (Fig. 8). Based on the simulation results, we know the tiny-tanh network has better performance over the other two resistive networks. However, the performance of the tiny-tanh network and that of the HRes network are similar provided that the motion cells have a large output swing. Considering the complexity of the tiny-tanh circuit, we choose HRes as the nonlinear resistor.

Another factor that needs consideration when smoothing the optical flow is edges in the image, since they too indicate possible object boundaries. In order to segment the moving objects, these boundaries must be identified, and smoothness constraint should be relaxed at the boundaries. As chip area is critical in the analog VLSI motion detector, we avoid using complicated algorithms in finding the image edges. A very simple algorithm is

Fig. 8. Optical-flow outputs using the motion cell with extended output range and different nonuniform, nonlinear resistive networks. (a) HRes network. (b) Tiny-tanh network.

used to determine the strength of smoothing in the resistive networks. In this algorithm, the spatial gradients of the image, I_x and I_y , are first computed. At a horizontal edge there is a large vertical gradient in image intensity; at a vertical edge, there is a large horizontal intensity gradient. Hence, the spatial intensity gradients can work as an indicator of the existence of image edges. If the intensity gradient is zero, there is no first-order intensity variation along the corresponding direction. Therefore, the smoothing strength along this direction shall be large. On the contrary, a large-magnitude intensity gradient, either positive or negative, should impose only weak smoothing strength. In other words, the maximal conductance of the HRes will be controlled by the image gradient while large difference in neighboring optical flow will lower the conductance owing to the saturating response of the nonlinear HRes.

The Gaussian function has precisely the property described above. The Gaussian function reaches its maximum when its argument is zero, and approaches zero when the magnitude of its argument is very large. In the proposed detector, the smoothing strength (and hence the conductance of the HRes resistor) is proportional to a Gaussian-like function of the spatial image intensity gradient. Moreover, the Gaussian-like function is implemented by the modified “bump” circuit proposed in [24]. The modified “bump” circuit (see left part of Fig. 9) has the advantage of tuning the height and width of the implemented Gaussian-like function independently through V_{bs} and (V_{bn}, V_{bp}) , respectively.

The circuit diagram of the smoothing-strength control circuit is shown in Fig. 9. In this figure, only the resistors connecting two horizontally neighboring pixels are shown. In both resistive networks (one for u and the other for v), the horizontal gradient of the image intensity controls the nonlinear resistors connecting horizontally adjacent motion cells, while the vertical gradient of the image intensity controls the nonlinear resistors connecting vertically adjacent motion cells. The intensity of the pixel is labeled $I(i, j)$, while the intensity of its horizontal neighbor is labeled $I(i+1, j)$. The Gaussian-like function output of the difference between $I(i, j)$ and $I(i+1, j)$ is conveyed in the signal V_g , which in turn controls the conductance of the nonlinear HRes resistor. When there is a large difference in intensity between two neighboring pixels ($I(i, j)$ and

Fig. 9. Circuit diagram of the smoothness control circuit.

$I(i+1, j)$), V_g will be low and the conductance of the HRes will be low, indicating weaker smoothing. On the other hand, smaller intensity gradient between $I(i, j)$ and $I(i+1, j)$ will increase the HRes conductance and make smoothing stronger.

D. Read-Out Circuit

The smoothed image and the computed optical-flow field must be read out of the chip. Fig. 10 shows the proposed readout circuit that multiplexes the outputs of each pixel to two common amplifiers (one for image and one for optical flow) that drive output pins.

Each pixel in the 32×32 array output three signals: image intensity, horizontal optical-flow component, and vertical optical-flow component. These signals are read out via the “X-Y” addressing technique by using two 32-bit shift registers to access the cells sequentially. Both 32-bit shift registers circulate a single logic high signal, but with different clock rates. Together they select a single cell from the array and the output buffers samples the output signals of the selected cell. As the content of the registers shifts, the whole array is scanned out of the chip serially.

IV. EXPERIMENTAL RESULTS

The motion field detector is fabricated using a $0.5\text{-}\mu\text{m}$, double-poly double-metal CMOS process. Each of the 32×32

Fig. 10. Circuit diagram of the readout circuit in the proposed optical-flow-based motion field detection chip.

Fig. 11. Die photograph of the optical-flow-based motion field detection chip.

cells occupies $124 \mu\text{m} \times 124 \mu\text{m}$ with a fill factor of 10%. All active areas, except for the photoreceptor, are covered with the METAL2 layer. Thus, only the photoreceptor is affected by the input light. The METAL2 layer not only works as a light screen but also is used to deliver power. The die size of the chip is $5400 \mu\text{m} \times 5400 \mu\text{m}$ and its die photo is shown in Fig. 11.

We first characterized the fabricated motion cell by disabling all motion cells except those on the topmost row. In this measurement, an image intensity profile is scanned in. In the topmost row, all vertical connections are disabled so the optical-flow constraint equation becomes

$$I_x u + I_t = 0. \quad (10)$$

(a)

(b)

Fig. 12. Characterization of the motion cell. (a) Measured horizontal optical flow versus I_t . (b) Measured horizontal optical flow versus I_x .

Setting I_x to a constant makes the optical-flow output u proportional to the temporal gradient, I_t , while setting I_t to a constant makes u inversely proportional to I_x . Fig. 12 shows the measurements of one motion cell in these two cases. Generally speaking, the motion cell correctly computes the horizontal optical-flow component in both cases. However, in the first cases, there is about 25-mV offset in I_x and 15-mV offset in I_t , while in the second case, there is about 75-mV offset in I_x .

The block diagram of the test setup is shown in Fig. 13. An optical interface focuses the image onto the chip. An incandescent lamp is used as back lighting to provide appropriate illumination for the chip. A computer controls the addressing of the read-out circuit via a computer interface. Bias voltages of the detector are provided by a biasing circuit made up of a 16-channel DAC and amplifiers. The values of the biases can be coarsely adjusted by variable resistors and fine tuned by the DAC, which in turn is controlled by the computer. The smoothed image output and the optical-flow output of the motion detector are sent to an ADC. The computer then gathers the data for calibration, display, and storage.

The sensitivity of the photoreceptors turned out to be low. It does not allow the chip to operate under normal lighting. An incandescent lamp is used as the back lighting. It produces a bright

Fig. 13. Block diagram of the test setup.

(a)

(b)

Fig. 14. Measured image output. (a) the sub-sampled original image pattern and (b) the captured and smoothed image.

background (with some mismatch) in the image output. A transparency foil with patterns printed on it is placed in front of the lamp to produce an image. Fig. 14(a) shows the sub-sampled 32×32 version of the original pattern on the foil, and Fig. 14(b) shows the smoothed image captured by the chip. Obviously, due to the small array size, a low fill factor (10%), and spatial

Fig. 15. Measured optical-flow output.

smoothing, the captured image is of low quality by itself. However, this image is good enough for computing the optical flow, as will be seen in the following measurement.

The measured optical-flow field output of a car-shaped object moving in a background is plotted in Fig. 15. The corresponding captured image is also super-imposed. One can clearly see uniform right-ward motion vectors inside the car-shape object, and no perceptible motion is observed in most of the background area. With this optical-flow field, the boundary of the moving object can be easily identified. The spurious left-ward motion vectors to the left of the object is an “after image” of the optical-flow field produced by the capacitive effect of the nodes computing I_t (I_t is computed by subtracting the temporally-averaged intensity from the instantaneous intensity). This effect can be mitigated by reducing the integration time constant of the circuit that computes the temporal average of the image intensity.

The chip can output image at a speed higher than 80 frames/s. Since the detector array itself operates in the subthreshold region, it thus consumes very little power, about 6 mW, while the readout circuit and pads consume about 24 mW. A summary of the chip is listed in Table I.

V. CONCLUSION

An analog optical-flow-based motion-field detector is proposed and implemented in this paper. The motion field detector adopts the gradient-based method. An image sensor array using parasitic p-n-p transistors in a standard CMOS technology is integrated within the chip. A linear resistive network is used to smooth the input image, and the degree of smoothness is adjustable via a bias voltage. Optical-flow constraint, LMSE constraint, and a new “edginess-controlled” smoothness constraint are combined to compute the optical flow. Two non-linear resistive networks enforce the LMSE constraint and the

TABLE I
SUMMARY OF THE PROPOSED MOTION FIELD DETECTION CHIP

Technology	0.5 μm CMOS DPDM process
Supply voltage	5V
Array size	32 $\times$ 32
Cell area	124 μm $\times$ 124 μm
Photosensor fill factor	10%
Image readout speed	> 80 frame/sec
Motion detection range	0.08 – 22 pixel/sec
Detector power consumption	6 mW
Total power consumption	30 mW
Die size	5.1 mm $\times$ 5.1 mm
Packaging	68-pin PGA

proposed smoothness constraint. The nonlinearity and image-gradient dependency of the resistors in the proposed nonlinear resistive networks make identifying the discontinuities in the optical-flow field feasible. The strength of smoothing is adjusted adaptively in order to preserve object boundaries. The chip was fabricated using a 0.5- μm double-polysilicon double-metal CMOS process. Measurements on real images show that the chip can output optical-flow field that greatly facilitates segmentation of moving objects. We believe that with the proposed motion field detector as the analog frontend, object segmentation task needed in next-generation object-based video-coding systems and computer vision systems can be completed with ease and efficiency.

ACKNOWLEDGMENT

The authors would like to thank the Chip Implementation Center (CIC) of the National Science Council in Taiwan for chip fabrication. They also thank the anonymous reviewers for their valuable suggestions that greatly improved this paper.

REFERENCES

- [1] *MPEG Handbook*, Focal Press, Woburn, MA, 2001.
- [2] M. Bober and J. Kittler, "Video coding for mobile communications—MPEG4 perspective," *Proc. Inst. Elect. Eng. Colloquium on the Future of Mobile Multimedia Communications*, pp. 3/1–3/9, 1996.
- [3] E. Francois, J.-F. Vial, and B. Chupeau, "Coding algorithm with region-based motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 97–108, Feb. 1997.
- [4] D. Zhong and S.-F. Chang, "An integrated approach for content-based video object segmentation and retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 1259–1268, Dec. 1999.
- [5] D. LaQuang, A. Zaccarin, and S. Caron, "Object-oriented coding using successive motion field segmentation and estimation," *Proc. IEEE Int. Conf. Image Processing*, vol. 1, pp. 207–210, Oct. 1995.
- [6] M. T. Orchard, "Predictive motion-field segmentation for image sequence coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, pp. 54–70, Feb. 1993.
- [7] D. S. Tweed and A. C. Calway, "Estimating dense motion fields using a local two-component model," in *Proc. Int. Conf. Image Processing and Its Application*, 1999, pp. 696–700.
- [8] D. Tzovaras and M. G. Strintzis, "Motion and disparity field estimation using rate-distortion optimization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 171–180, Apr. 1998.

- [9] A. Singh, *Optical Flow Computation—A Unified Perspective*. New York: IEEE Computer Society Press, 1991.
- [10] C. A. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.
- [11] J. Kramer, R. Sarpeshkar, and C. Koch, "Pulse-based analog VLSI velocity sensors," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 86–101, Feb. 1997.
- [12] A. Moini *et al.*, "An insect vision-based motion detection chip," *IEEE J. Solid-State Circuits*, vol. 32, pp. 279–284, Feb. 1997.
- [13] R. R. Harrison and C. Koch, "A robust analog VLSI Reichardt motion sensor," *Analog Integ. Circuits and Signal Processing*, vol. 24, no. 3, pp. 213–229, Sept. 2000.
- [14] J. Tanner and C. A. Mead, "An integrated analog optical motion sensor," in *VLSI Signal Processing 2*, S.-Y. Kung, R. Owen, and G. Nash, Eds: IEEE Press, 1986.
- [15] T. Delbrück, "Silicon retina with correlation-based velocity-tuned pixels," *IEEE Trans. Neural Networks*, vol. 4, no. 3, pp. 529–541, May 1993.
- [16] R. Etienne-Cummings, J. Van Spiegel, and P. Müller, "A focal plane visual motion measurement sensor," *IEEE Trans. Circuits Syst. I*, vol. 44, pp. 55–66, Jan. 1997.
- [17] A. Stocker and R. Douglas, "Computation of smooth optical flow in a feedback connected analog network," in *Advances in Neural Information Processing Systems 11*, M. S. Kearns, S. A. Solla, and D. A. Cohn, Eds. Cambridge, MA: MIT Press, 1999, pp. 706–712.
- [18] M. Tistarelli, "Multiple constraints to compute optical flow," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, pp. 1243–1250, Dec. 1996.
- [19] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intell.*, vol. 17, pp. 49–52, 1981.
- [20] H.-H. Nagel and W. Enkelmann, "An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 8, pp. 49–52, Sept. 1986.
- [21] D. Marr, *Vision*. San Francisco, CA: Freeman, 1982.
- [22] P. C. Yu, S. J. Decker, H. S. Lee, C. G. Sodini, and J. L. Wyatt Jr., "CMOS resistive fuses for image smoothing and segmentation," *IEEE J. Solid-State Circuits*, vol. 27, pp. 545–553, Apr. 1992.
- [23] J. G. Harris, "An analog network for continuous-time segmentation," *Int. J. Comput. Vis.*, vol. 10, no. 1, pp. 43–51, 1993.
- [24] S. Y. Lin, R. J. Huang, and T. D. Chiueh, "A tunable Gaussian/square function computation circuit for analog neural networks," *IEEE Trans. Circuits Syst. II*, vol. 45, pp. 441–446, Mar. 1998.

Ming-Han Lei was born in Taiwan, R.O.C., in 1973. He received the B.S. and M.S. degrees in electrical engineering from the Department of Electrical Engineering, National Taiwan University, Taiwan, R.O.C., in 1996 and 1998, respectively.

He is currently with SOC Technology Center, Industrial Technology Research Institute, Chu-Tung, Taiwan, R.O.C. His primary research interests include mixed-mode ASIC design and embedded system programming.

Tzi-Dar Chiueh (S'87–M'90) was born in 1960. He received the B.S. degree from the Department of Electrical Engineering, National Taiwan University, Taiwan, R.O.C., in 1983, and the Ph.D. degree from California Institute of Technology, Pasadena, in 1989.

He joined the faculty of National Taiwan University in 1994. His research interests include VLSI design for neuromorphic systems and communication systems.