December 2005

A CMOS image processing sensor for the detection of image features

Masatoshi Nishimura
Sankyo Co., Ltd.

Jan Van der Spiegel
University of Pennsylvania, jan@seas.upenn.edu

Postprint version. Published in *Analog Integrated Circuits and Signal Processing*, Volume 45, Issue 3, December 2005, pages 263-279. The original publication is available at www.springerlink.com.
Publisher URL: http://dx.doi.org/10.1007/s10470-005-4955-x

A CMOS image processing sensor for the detection of image features

Abstract
A compact CMOS vision sensor for the detection of higher level image features, such as corners, junctions (T-, X-, Y-type) and linestops, is presented. The on-chip detection of these features significantly reduces the data amount and hence facilitates the subsequent processing of pattern recognition. The sensor performs a series of template matching operations in an analog/digital mixed mode for various kinds of image filtering operations including thinning, orientation decomposition, error correction, set operations, and others. The analog operations are done in the current domain. A design procedure, based on the formulation of the transistor mismatch, is applied to fulfill both accuracy and speed requirements. The architecture resembles a CNN-UM that can be programmed by a 30-bit word. The results of an experimental 16x16 pixel chip demonstrate that the sensor is able to detect features at high speed due to the pixel-parallel operation. Over 270 individual processing operations are performed in about 54 μsec.

Keywords
feature detection, vision sensor, smart sensor, computational sensor, template-matching, transistor mismatch

Comments
Postprint version. Published in *Analog Integrated Circuits and Signal Processing*, Volume 45, Issue 3, December 2005, pages 263-279. The original publication is available at www.springerlink.com.
Publisher URL: http://dx.doi.org/10.1007/s10470-005-4955-x

Published in *Analog Integrated Circuits and Signal Processing*, Vol. 43, pp. 1-17, 2005

A CMOS image processing sensor for the detection of image features

Masatoshi Nishimura* and Jan Van der Spiegel

University of Pennsylvania, Department of Electrical and Systems Engineering, Philadelphia, PA, USA

*Sankyo Co, Ltd, 1-2-58, Hiromachi 1-Chome, Shinagawa-K, Tokyo, Japan.

Abstract — A compact CMOS vision sensor for the detection of higher level image features, such as corners, junctions (T-, X-, Y-type) and linestops, is presented. The on-chip detection of these features significantly reduces the data amount and hence facilitates the subsequent processing of pattern recognition. The sensor performs a series of template matching operations in an analog/digital mixed mode for various kinds of image filtering operations including thinning, orientation decomposition, error correction, set operations, and others. The analog operations are done in the current domain. A design procedure, based on the formulation of the transistor mismatch, is applied to fulfill both accuracy and speed requirements. The architecture resembles a CNN-UM that can be programmed by a 30-bit word. The results of an experimental 16x16 pixel chip demonstrate that the sensor is able to detect features at high speed due to the pixel-parallel operation. Over 270 individual processing operations are performed in about 54 μsec.

Index terms—feature detection, vision sensor, smart sensor, computational sensor, template-matching, transistor mismatch

1 Introduction

The conventional approach for pattern recognition, consisting of image capture and subsequent analysis by a host CPU, has two time consuming steps: the time required for image transfer and the time required for analysis. The former is determined by the video rate (1/33 msec) while the latter is determined by the amount of data and the algorithm employed. These two steps make it difficult for the conventional approach to be used for real-time pattern recognition such as high-speed product inspection on an assembly line, character recognition, autonomous navigation, and so on. The time-consuming steps originate from the inherent separation between image capture and image analysis.

The above observations have motivated new type of image sensors that incorporate a processing element at each pixel. The on-chip processing element extracts relevant information for subsequent pattern recognition, resulting in a reduced amount of data. Implementations have been done in both analog and digital domains using CMOS technology. The analog approach performs computation by exploiting the analog interaction between pixels [1][2][3][4]. No external clock is required for computation. The detected features have been primarily edges and orientations, which are relatively low level features corresponding to those detected at the early stages in the pathway of the visual system. A chip that has more general image processing capability has been also proposed [5].

On the other hand, the digital implementation provides programming flexibility [6][7][8]. The sensor operates in an iterative fashion by updating the memory content based on a specified program. In Ref [8], Ishikawa has presented a sensor in which pixels have both compactness and programmability by the incorporation of an ALU (algorithmic logical unit) at each pixel. This approach, however, has limitations for image processing that is based on neighborhood interaction, e.g. 3 x 3 window, since the instruction takes only two inputs in a

single instruction.

The concept of cellular neural network (CNN) has been also extensively employed [9][10][11], in which computation is performed in an analog mode. The CNN concept has been further extended to CNN universal machine (CNN-UM) architecture to include programming capabilities [12]. Domínguez-Castro has built a sensor based on this principle and demonstrated its functionality for image processing applications [13]. However, the operation is not very fast, usually in the order of microsecond, due to the settling behavior of the analog interaction between pixels.

Despite all these research activities, no attempt has been made so far to detect higher-level features such as corners and junctions in a discriminative fashion. Detection of these features is very important for the purpose of object recognition [14]. Even in the digital implementation, which has programming flexibility, existing software is too complicated to be mapped into hardware implementations. With final hardware implementation in mind, we have developed an efficient feature detection algorithm that is partly inspired by biology [15]. The algorithm first decomposes an input image into four orientations and performs an iterative computation resulting in the detection of corners, three types of junctions (T-, X-, Y-type), and linestops. Each operation is carried out based on template matching.

The paper presents the hardware implementation of the above algorithm. Template matching is mapped onto configurable hardware by an analog current-mode operation. An operation can be iteratively executed as many times as required according to an external digital control. Thus, the proposed sensor is a simplified form of a CNN-UM architecture, optimized for high-speed operation.

A key aspect of the design of current-mode circuits is the transistor mismatch. We will present a systematic procedure to determine design parameters including transistor dimensions and currents. Limited information is available in the literature for the design procedure based on

transistor mismatch. In a paper dealing with transistor mismatch [16], Lakshmikumar et al. have presented a method for determining the required accuracy for current sources in order to satisfy a specific yield. A procedure for the determination of sensor's design parameters for the required accuracy is presented in this paper.

The paper is organized as follows. Section 2 describes the overall design of the sensor, including a brief discussion of the algorithm. Section 3 presents the design procedure based on transistor mismatch analysis. Section 4 explains the details of the implementation, focusing on mechanisms to improve the speed. The experimental results are given in section 5, followed by a discussion and conclusion.

2 Overall sensor design

2.1 Algorithm

Fig. 1 shows the processing flow of the feature detection algorithm for a letter 'A' [15]. The image is decomposed into four orientation planes. Then a series of operations are performed for each orientation plane. The results of processing in each orientation plane are combined to generate the final five features (T-, X-, Y- type junctions, true corners, and true linestops). All these operations are carried out by template matching in a 3 x 3 window by using a set of weights specific for each operation:

$$x_{ij}(n+1) = f\left(\sum_{l=-1}^{1}\sum_{m=-1}^{1} x_{i+l,j+m}(n)\, r_{lm}; I\right) \tag{1}$$

where $x_{ij}(n)$ is the binary status of the pixel at the position (i,j) at a discrete instant n, r_{ij} is the element of the template, f is the function to generate a binary output with the threshold I given in the form below:

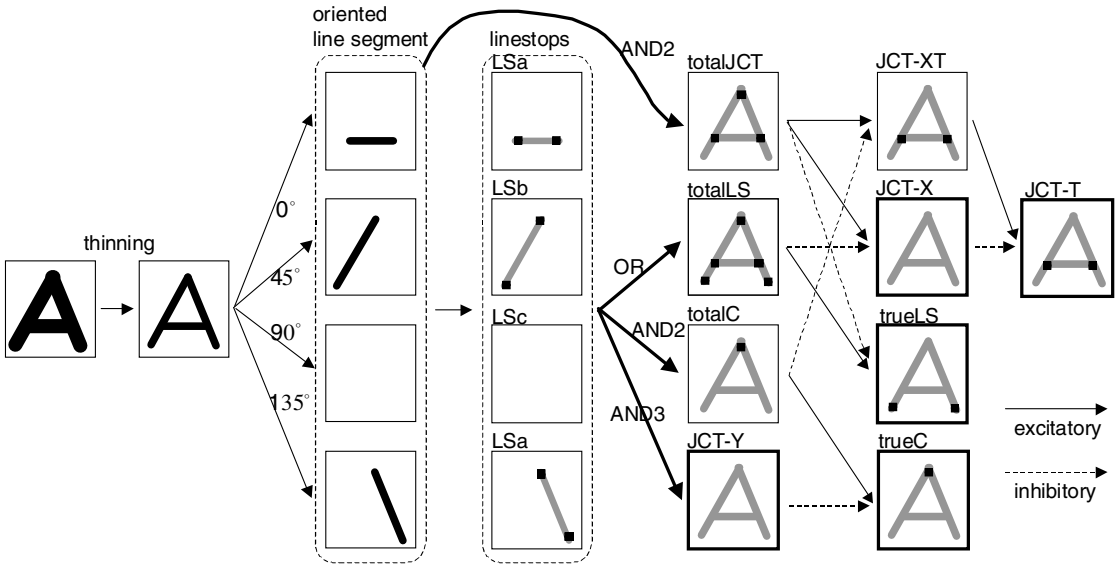

Fig. 1 Overall processing flow of the feature detection algorithm. The detected features are: JCT-T (T-type junction), JCT-X (X-type junction), JCT-Y (Y-type junction), trueC (corners), and trueLS (linestops), shown in a bold rectangle (Adapted from Ref. [15])

$$f(x;I)=\begin{cases}1 & \text{for} \quad x \geq I \\ 0 & \text{for} \quad x < I.\end{cases} \tag{2}$$

As discussed in Ref [15] the template matching can be implemented as a convolution, using current distribution and summation, where the convolution kernel (distribution pattern) is the flipped version of the template.

2.2 Chip overall architecture

Fig. 2 shows the chip architecture of the proposed sensor. The sensor consists of an array of pixels and peripheral circuitry. Each pixel further consists of a photosensor and a processing circuit. The pixel is connected to its eight neighbors to perform the operations described above. The peripheral circuit controls the sequence of operations and scanning of the sensor. Signal lines runs both horizontally and vertically to distributed control signals to each pixel and transfer the signal from each pixel to the output.

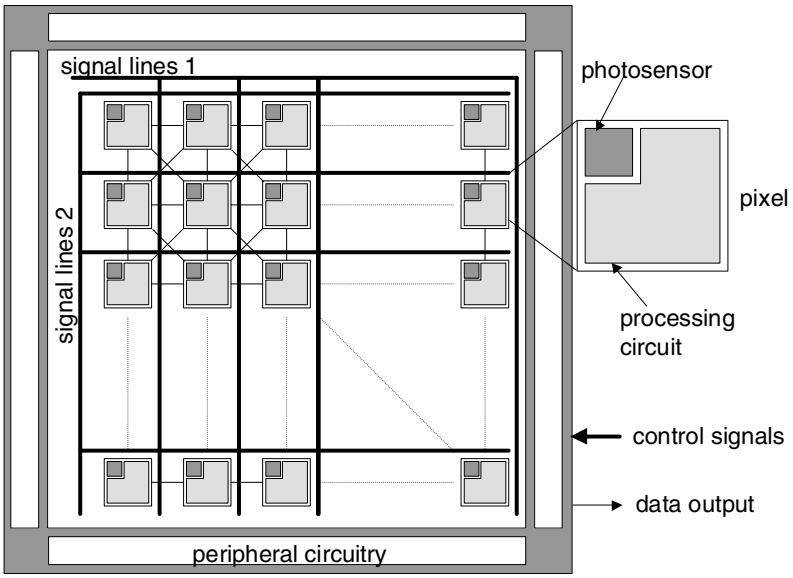

Fig. 2 Chip overall architecture.

2.3 Chip pixel architecture

The conceptual architecture of the pixel circuit is shown in Fig. 3. The sensor consists of a phototransduction (shown on the right-hand side) and a processing circuit. In the phototransduction stage, the output current of the phototransistor I_{ph} is compared to a threshold current I_{th1} to produce a binary output V_{ph}, which is transferred to the memory when signal *photo* is set to logic High.

The processing circuit is constructed in a mixed-mode fashion in the sense that the internal operation is performed in the analog domain while the control of the operation is performed in the digital domain. The circuit consists of the memory and the convolution unit with programmable kernels (distribution pattern). The basic function of this circuit is to distribute a current to 3 x 3 neighborhood pixels if the memory content is logic High. The distribution pattern is specified by setting switches along the connection path to neighbors, which are not shown in Fig. 3. The currents from neighboring pixels, in turn, are summed to generate I_{sum} and compared with I_{th2} to generate a binary output V_o. This binary voltage is then transferred to the memory by a two-phase clock (ϕ_1 and ϕ_2) and a parasitic capacitor.

A typical processing sequence is as follows. Signal *photo* is set to logic High to store the result of phototransduction V_{ph} into the memory. Next, the signal *photo* is set to a logic Low

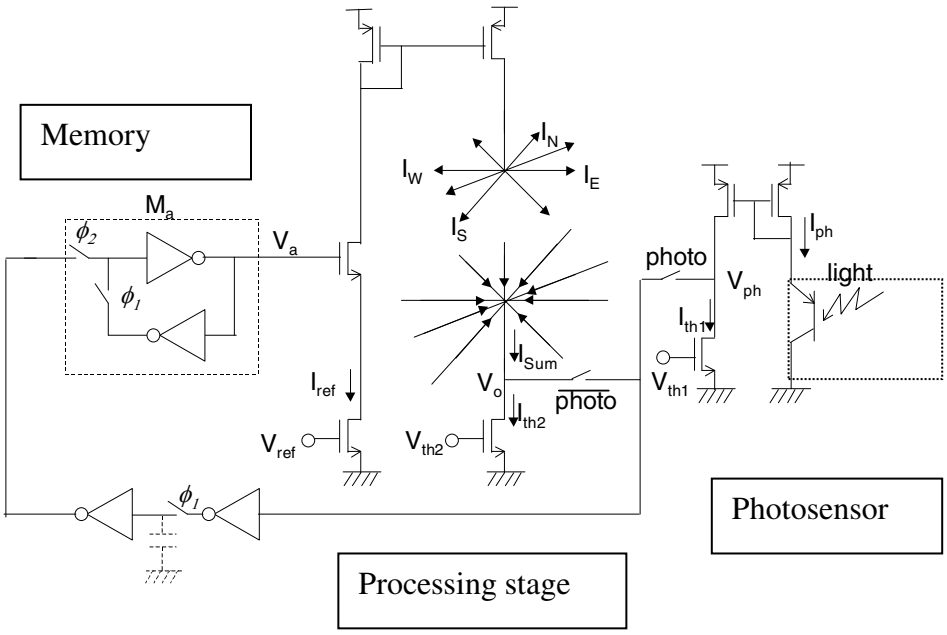

Fig. 3 Conceptual architecture of the pixel circuit

to start a series of operations. Different kernels and threshold values are specified for each operation to perform various types of processing functions.

2.4 Processing circuit

The complete schematic of the processing circuit is shown in Fig. 4, which is a detailed representation of the conceptual architecture shown in Fig. 3. Note that there are six memories. Four of them (M_a, M_b, M_c, M_d) are used to store the image in four orientation planes while the other two memories are used as working memories. Associated with each memory are six reference current sources, whose amount is determined by the bias voltage V_{ref}. A set of signals (a1, b1, c1, d1, x1, y1) determines which memory will be accessed. When one of these signals is on, and the corresponding memory stores a logic High, and signal RE is logic High , the reference current I_{ref} will pull down the voltage of node Nc, resulting in current spreading to the neighboring pixels.

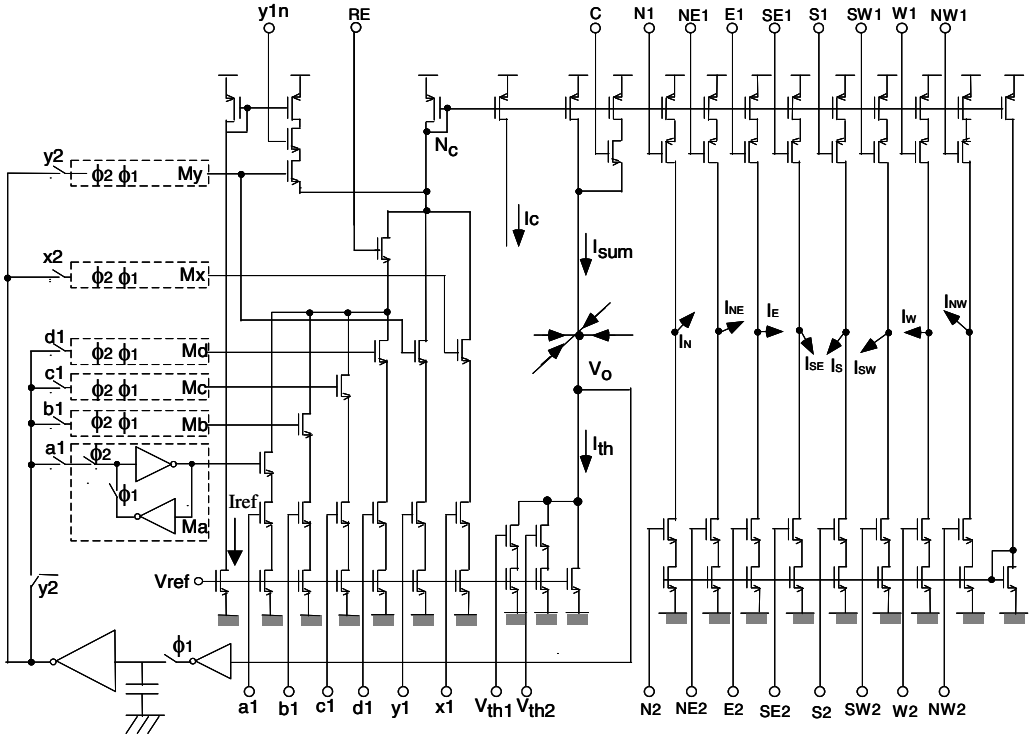

Fig. 4 Schematic of the processing circuit.

The current distribution pattern is determined by the convolution kernel, which is controlled by a set of switches (C, N1, N2, NE1, NE1, … , NW1, NW2). Note that switch C controls the presence/absence of an additional current to the central pixel, indicating that the center value of the kernel is set to either one or two. A set of eight switches (N1, NE1, … , NW1), each corresponding to eight neighbors, determines the outward (additive) current flow through the PMOS transistors. Another set of eight switches (N2, NE2, … , NW2) determines the inward current (subtractive) flow through the NMOS transistors. Only one of the two switches is allowed to be ON at one time, i.e., both N1 and N2 cannot be logic High at the same time. Thus, the convolution kernel except the center pixel can take a value of 1, 0, or –1.

While the current is distributed to neighboring pixels, the currents generated at neighboring pixels are accumulated on a thresholding node Nc. The accumulated current I_{sum} is compared to the threshold current I_{th}, specified by signals V_{th1} and V_{th2}. With these switches set

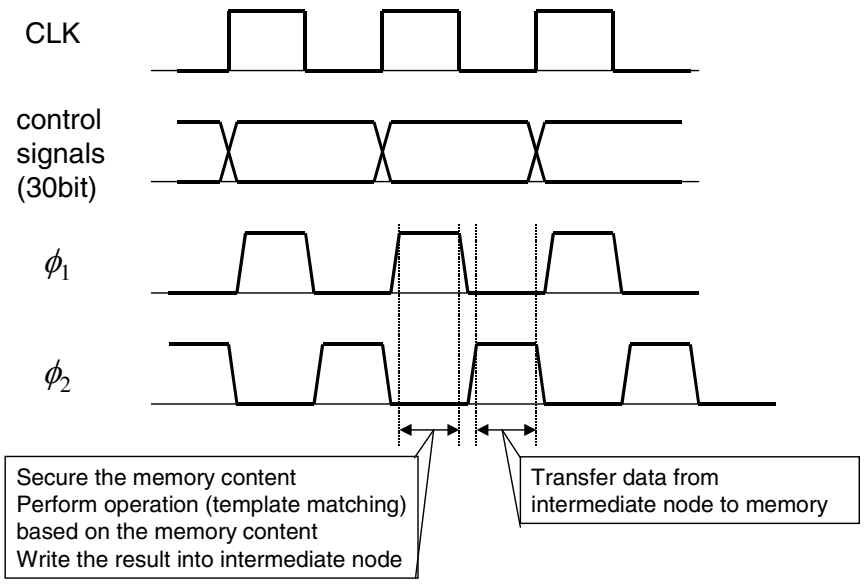

Fig. 5 Timing diagram. CLK: fundamental clock to generate necessary clocks; control signals: 30-bit signals shown in Fig. 4 (C, N1, N2, NE1, NE2, E1, E2, SE1, SE2, S1, S2, SW1, SW2, W1, W2, NW1, NW2, a1, b1, c1, d1, x1, x2, y1, y1n, y2, $\overline{y2}$, Vth1, Vth2, RE).

to logic Low, the threshold current is $0.5I_{ref}$. It can be increased to $1.5I_{ref}$ or $2.5I_{ref}$ depending on the setting of these switches.

Fig. 5 shows the clock pattern and the timing diagram of the sensor operation. Node voltage V_0, which is the result of current comparison, is transferred to the parasitic capacitor at the output of the inverter when clock ϕ_1 is logic High for temporal information storage. The stored charge is then transferred to the memory by setting ϕ_2 to logic High. Then by bringing down ϕ_2 and raising ϕ_1, the information is secured in the memory. This processing initiates a new operation cycle since control signals are already set to logic High or Low before ϕ_1 goes High. The operation can be repeated as many times as needed.

3 Design procedure

A procedure to design the channel length and width, and the amount of the reference

current is given below to satisfy a given requirement for accuracy and speed. The specification for the present sensor has been defined as the operational clock frequency larger than 5 MHz and the error rate smaller than 0.1 %.

3.1 Formulation of the current variation

The analysis starts from the following basic relationship:

$$I = \frac{\beta}{2}\left(V_{GS} - V_T\right)^2 = \frac{\mu C_{ox}(W/L)}{2}\left(V_{GS} - V_T\right)^2 \tag{3}$$

where I is the drain current, V_{GS} is the gate-source voltage, V_T is the threshold voltage of a transistor, β is a current factor, μ is the carrier mobility, C_{ox} is the oxide capacitance per unit area, W and L is the transistor width and length, respectively. For a set of transistors that are biased with a constatn gate voltage V_{GS}, the variance of I is expressed as a function of β and V_T. One can easily prove that

$$\left(\frac{\sigma_I}{\overline{I}}\right)^2 = \left(\frac{\sigma_\beta}{\overline{\beta}}\right)^2 + \left(\frac{2\sigma_{V_T}}{V_{GS} - \overline{V_T}}\right)^2. \tag{4}$$

in which $\overline{I}$, $\overline{\beta}$, and $\overline{V_T}$ are the mean value of I β, and V_T, respectively.

The variance of β and V_T are known to be inversely proportional to the transistor area [17]:

$$\sigma_{V_T}^2 = \frac{A_{V_T}^2}{WL} \tag{5}$$

$$\sigma_\beta^2 = \frac{A_\beta^2}{WL} \tag{6}$$

where A_{V_T} and A_β are mismatch proportionality constants for V_T and β, respectively, W and L represent the width and length of the transistor channel, respectively. Substituting equations (5) and (6) into (4) yields

$$\left(\frac{\sigma_I}{\overline{I}}\right)^2 = \frac{1}{WL}\left(A_\beta^2 + \left(\frac{2A_{V_T}}{V_{GS} - \overline{V_T}}\right)^2\right). \tag{7}$$

For reported values A_{V_T} and A_β, the effect of the second term is much larger than the first term unless the transistor is too heavily biased. Hence, by dropping the first term, the above equation is simplified as

$$\left(\frac{\sigma_I}{\overline{I}}\right)^2 = \frac{1}{WL}\left(\frac{2A_{V_T}}{V_{GS} - \overline{V_T}}\right)^2. \tag{8}$$

Using the relationship in equation (3), the following formula is obtained:

$$s^2(I_{ref}, L) = \left(\frac{\sigma_I}{I_{ref}}\right)^2 = \frac{2A_{V_T}^2 \mu C_{ox}}{L^2 I_{ref}}, \tag{9}$$

in which s^2 is defined as the relative variance of the current when its amount is equal to I_{ref}. It should be noted that the relative current variation is inversely proportional to L^2 and is independent of W. σ_I can be also explicitly expressed as a function of I_{ref} and L for a more general case in which the current is mI_{ref} and the channel length is bL,

$$\sigma_I^2(mI_{ref}, bL) = s^2(mI_{ref}, bL)(mI_{ref})^2 = 2\frac{m}{b^2}s^2 I_{ref}^2. \tag{10}$$

The variation of the current for a set of multiple current mirror circuits is analyzed next. When V_{GS} is treated as a random variable as well as V_{T1} in equation (3), the following formula is obtained (effect of β is neglected):

$$\left(\frac{\sigma_{I2}}{\overline{I_2}}\right)^2 = \left(\frac{2\sigma_{V_{T2}}}{V_{GS} - \overline{V_{T2}}}\right)^2 + \left(\frac{2\sigma_{V_{GS}}}{V_{GS} - \overline{V_{T2}}}\right)^2. \tag{11}$$

The gate voltage V_{GS} is expressed as

$$V_{GS} = V_{T1} + \sqrt{\frac{2I_1}{\beta}}. \tag{12}$$

The above formula can be used to obtain

$$\sigma_{V_{GS}}^2 = \sigma_{V_T}^2 + \frac{1}{2\beta I}\sigma_I^2, \tag{13}$$

or equivalently

$$\left(\frac{2\sigma_{V_{GS}}}{\overline{V_{GS}} - \overline{V_{T1}}}\right)^2 = \left(\frac{\sigma_{I_1}}{\overline{I_1}}\right)^2 + \left(\frac{2\sigma_{V_{T1}}}{\overline{V_{GS}} - \overline{V_{T1}}}\right)^2. \tag{14}$$

By substituting equation (14) into (11), we get

$$\left(\frac{\sigma_{I_2}}{\overline{I_2}}\right)^2 = \left(\frac{\sigma_{I_1}}{\overline{I_1}}\right)^2 + \left(\frac{2\sigma_{V_{T1}}}{\overline{V_{GS}} - \overline{V_{T1}}}\right)^2 + \left(\frac{2\sigma_{V_{T2}}}{\overline{V_{GS}} - \overline{V_{T2}}}\right)^2. \tag{15}$$

Since $\overline{I_1} = \overline{I_2} = I_{ref}$, $\overline{V_{T1}} = \overline{V_{T2}} = \overline{V_T}$, and the second and the third terms on the right-hand side is equal to s^2, equation (15) can be written as

$$\sigma_{I_2}^2 = \sigma_{I_1}^2 + 2s^2 I_{ref}^2. \tag{16}$$

Note that the current mirror operation increases the variance of the current by $2s^2 I_{ref}^2$. When the current is mI_{ref} and the channel length is bL, the above formula is more generally represented as

$$\sigma_{I_2}^2 = \sigma_{I_1}^2 + 2\frac{m}{b^2}s^2 I_{ref}^2 \tag{17}$$

3.2 Formulation of accuracy requirement

Based on the formulations obtained above equations (10) and (17), the relationship between the design parameters and the accuracy of the operation is given below for the detection of 45° line segments. This is the most difficult operation in the feature detection algorithm because the number of neighbors used in the computation is larger than other operations. The template consists of three 1's at the right diagonal position, three –1's at the upper triangle or the lower triangle, and three 0's at the other side of triangle [15]. When the

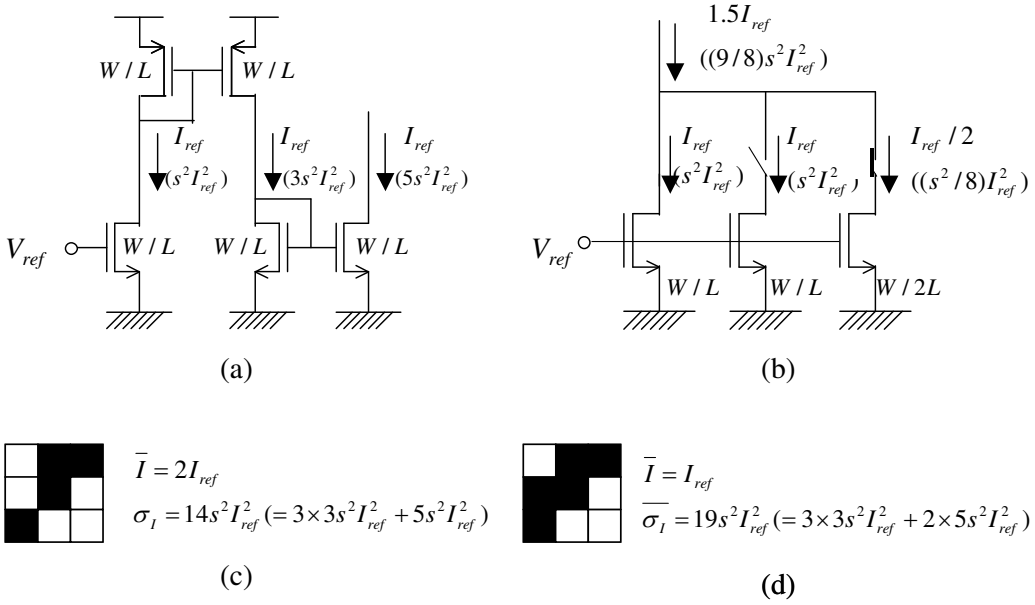

Fig. 6 Schematic explaining the analysis of the current variation for the 45° orientation detection operation. (a) the current mirror circuit, (b) the thresholding circuit. The variance for each current is shown in parenthesis. (c) image pattern for producing the current amount of $2I_{ref}$, (d) image pattern for producing the current amount of I_{ref}. The variance is also given for each of them.

current output of I_5 is larger than the threshold current of I_{th}=$1.5I_{ref}$, the local orientation is determined to be 45°.

Fig. 6 explains the analysis procedure for the variation of the current. Each current-mirroring operation increases the variance of the current by $2s^2 I_{ref}^2$, which results in the variance of $3s^2 I_{ref}^2$ from current-sourcing PMOS transistors and the variance of $5s^2 I_{ref}^2$ from current-sinking NMOS transistors (Fig. 6 (a)). The variance of the threshold current is calculated as $9/8s^2 (= s^2 + (1/8)s^2)$ (Fig. 6 (b)). Note that the variance of the current whose amount is $I_{ref}/2$ equals to $(1/8)s^2 I_{ref}^2$ by setting $m = 1/2, b = 2$ in equation (10). When the pixel is categorized as 45° by the presence of three 1's and one –1 as shown in Fig. 6 (c),

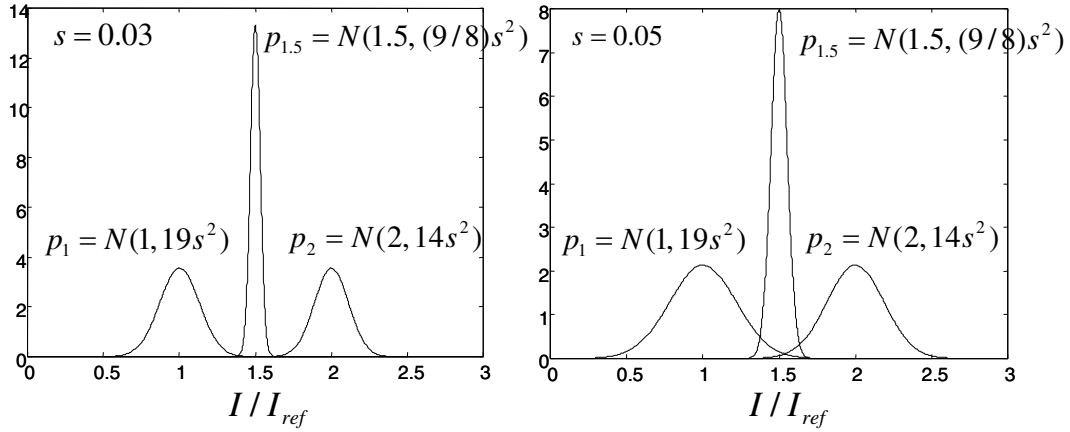

Fig. 7 Probability distribution function of the summation current and the threshold current for different relative current variations (left: s=0.03; right: s=0.05).

corresponding to the output current of $2I_{ref}$, the variance of current I_{sum} (represented in Fig. 4) is $14s^2 (= 3 \times 3s^2 + 5s^2)$. When the pixel is not categorized as 45° by the presence of three 1's and two –1's as shown in Fig. 6 (d), corresponding to the output current of I_{ref}, the variance of current I_{sum} is $19s^2 (= 3 \times 3s^2 + 2 \times 5s^2)$.

For further analysis, a Gaussian distribution is assumed for each random variable. Hence, the summation current I_{sum} (in the case of $2I_{ref}$ and I_{ref}) and the thresholding current ($1.5I_{ref}$) are represented as

$$p_2(x) = N(2I_{ref},\ 14s^2 I_{ref}^2), \tag{18}$$

$$p_1(x) = N(I_{ref},\ 19s^2 I_{ref}^2), \tag{19}$$

and

$$p_{1.5}(x) = N(1.5I_{ref},\ 9/8s^2 I_{ref}^2), \tag{20}$$

respectively, where $N(m, \sigma^2)$ is the Gaussian distribution function with a mean value m and a variance σ^2. The distributions for different values of s (s=0.03 and s=0.05) are graphically shown in Fig. 7 where the x-axis is scaled to I_{ref}. For s=0.03, which is shown on the left-hand side, the probability distribution is not very broad and almost no overlap exists between $p_1(x)$

Table.1 Error rate (%) for 45° orientation detection as a function of the relative current variation.

s	0.03	0.04	0.05	0.06
p_{right}	0.00	0.07	0.50	1.59
p_{left}	0.01	0.27	1.29	3.19

and $p_{1.5}(x)$, and between $p_2(x)$ and $p_{1.5}(x)$, On the other hand, for s=0.05, which is shown on the right-hand side, the distribution is broader, resulting in a considerable overlap between $p_1(x)$ and $p_{1.5}(x)$, and the overlap between $p_2(x)$ and $p_{1.5}(x)$.

The generation of these overlap regions indicates a classification error. Table. 1 shows the error rate broken down in two components resulting from overlap of the two right curves (p_{right}) and from the overlap of the two left curves (p_{left}), as estimated by Monte-Carlo simulations for different relative current variations. As expected, the error rate is lower for smaller values of s. The error occurrence on the left-hand side is higher than that on the right-hand because the distribution $p_1(x)$ has broader distribution than $p_2(x)$. The data in the table indicates that for an error rate to be smaller than 0.1%, s should be smaller than 0.03. The selected value of s can be obtained by various combinations of the reference current I_{ref} and the channel length L for a given parameter of μC_{ox} and A_{V_T}. The HP 0.5 μm technology provided by MOSIS, which is chosen for the fabrication of the feature detection sensor, has typical values for μC_{ox} of 104 [μA/V^2] and 35 [μA/V^2] for the NMOS and PMOS, respectively. The proportionality constant A_{V_T} is estimated as 15 [mVμm] for both NMOS and PMOS transistors based on the previously published values listed in Ref. [18].

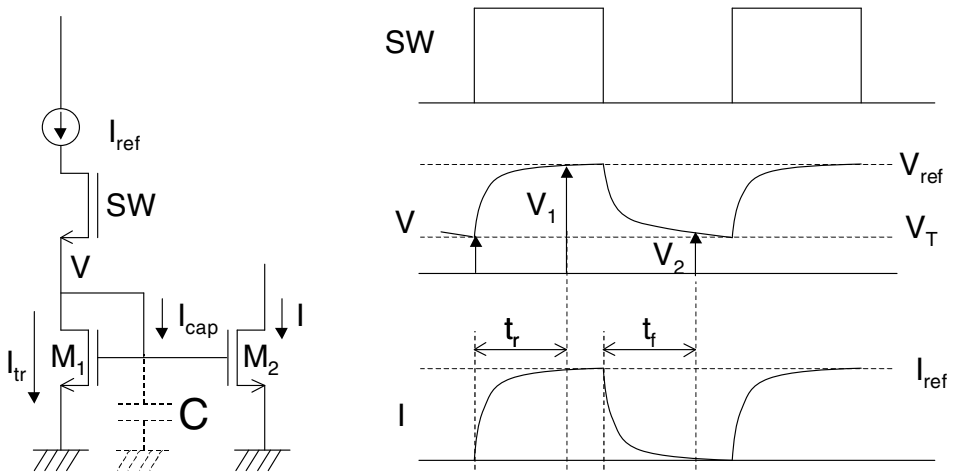

Fig. 8 Schematic for explaining the charging and discharging of the capacitance associated with the gate.

3.3 Formulation of the operational speed requirement

In addition to accuracy, the operational speed is another important specification for the sensor circuits. The response is largely determined by charging and discharging of the capacitance associated with the gate of the current mirror transistors as is schematically shown in Fig. 8. The capacitance drawn as a dashed line represents the parasitic capacitance connected to that node, which consists of the gate and drain capacitance of M_1 and the gate capacitance of M_2. Analysis of the circuit of Fig. 8 gives the rise time t_r and fall time t_f as

$$t_r = \frac{C}{\sqrt{2\beta I_{ref}}} \ln \frac{1+\sqrt{\alpha}}{1-\sqrt{\alpha}} \tag{21}$$

and

$$t_f = \frac{C}{\sqrt{2\beta I_{ref}}} \frac{2(1-\sqrt{\varepsilon})}{\sqrt{\varepsilon}}, \tag{22}$$

where t_r is defined as the time required for the current to rise to $I_1 = \alpha I_{\text{ref}}$, and t_f is defined as the

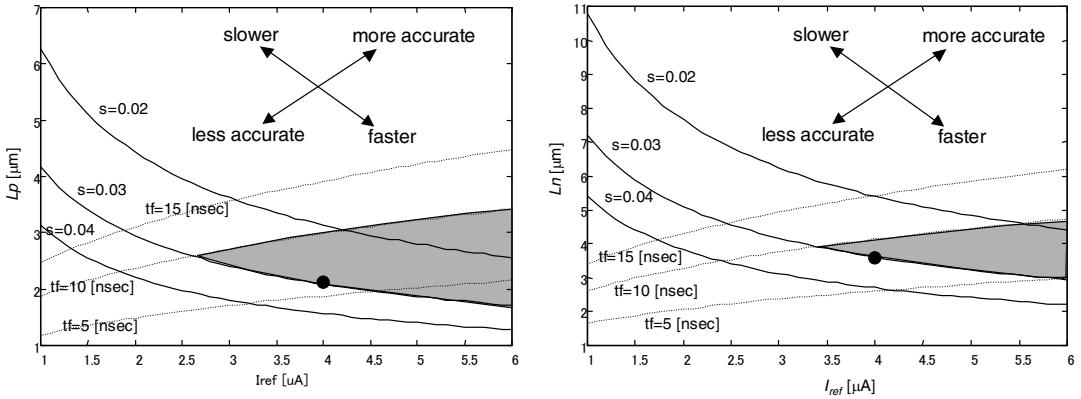

Fig. 9 Relationship between the channel length and the reference current to satisfy requirements for accuracy and speed. The left graph and the right graph represent the relationship for PMOS current mirrors and NMOS current mirrors, respectively. The three solid lines represent the relationship between L and I_{ref} to achieve accuracy given by the value of s. The three dotted lines represent the relationship to achieve a speed specified by the fall time.

time required for the current to decrease to $I_2=\varepsilon I_{\text{ref}}$. Both equations (21) and (22) have the following term in common:

$$\frac{C}{\sqrt{2\beta I_{ref}}} = \frac{2\times\frac{2}{3}C_{ox}WL}{\sqrt{2\mu C_{ox}(W/L)I_{ref}}} = \sqrt{\frac{8C_{ox}WL}{9\mu I_{ref}}}L, \tag{23}$$

which demonstrates the effect of the design parameters on the rise time and the fall time. Larger values of W and L contribute to larger values of the rise time and the fall time. Smaller values of W are preferable for faster operation. However, L should be chosen carefully since smaller values of L lead to lower accuracy (see equation (9)). The speed requirement is specified using the fall time since the fall time is larger than the rise time when α=0.95 and ε=0.05, conventionally used values for the rise time and the fall time.

3.4 Determination of the design parameters

The relationship between the reference current and the channel length to satisfy various

accuracy and speed requirements is plotted in Fig. 9. The channel width is chosen to be 1.5 μm instead of the minimum dimension of 0.9 μm for the 0.5 μm technology. The minimum width is avoided since the current sinking NMOS transistor operates close to the edge of the saturation region. It is clear from the graph in Fig. 9 that for higher accuracy, larger values of L and I_{ref} are required which corresponds to the top right region in the graph. On the other hand, to achieve faster speed, larger values of I_{ref} and smaller values of W and L are required, which corresponds to the bottom right region in the graph. It should be noted that larger values of W shifts these dashed lines downward, demanding even smaller values of L to achieve the same speed. To satisfy the specification of accuracy ($s < 0.03$) and speed ($t_f < 10$ nsec), the channel length and the reference current should be in the shaded region.

In the present design, the current level was chosen as 4 μA to satisfy both speed and accuracy requirements without consuming too much power. For the current amount of 4 μA, the channel length for the NMOS and PMOS transistors are chosen as 3.7 μm and 2.2μm, respectively, to satisfy the accuracy requirement (shown as closed circles on the graphs).

4 Implementation

4.1 Mechanisms for high speed operation

It is clear from the circuit shown in Fig. 4 that the operational speed is mainly determined by the time required for charging and discharging node Nc, which has a large capacitance due to many PMOS gates connected to this node. In order to decrease the time for the charging and discharging of this node, two additional circuits have been implemented as shown in Fig. 10. The first circuit shown in the lightly shaded block keeps the critical node voltage around 1.6 V, corresponding to $I_{ref} = 4$ μA. This is achieved by the discharging of node Nc during phase ϕ_2 by the current I_{ref} through switch M_4. To further accelerate the process of discharging, an

additional current of $2I_{ref}$ is used until the current I_c reaches $0.75I_{ref}$ (= 3 μA). This process of conditional discharging is controlled by properly turning transistor M_6 on or off: M_6 is turned on when I_c is smaller than $0.75I_{ref}$ ($\overline{V_{comp}}$ is High) and turned off when I_c is larger than $0.75I_{ref}$ ($\overline{V_{comp}}$ is Low).

The action explained above keeps the node voltage V_c around 1.6 V, and puts transistor M_1 in the "ready" state for starting current distribution. If the content of the memory specified for the operation is logic High, the voltage V_c does not have to change and the current distribution immediately occurs when ϕ_2 goes Low and the switches in the distribution path get turned on. However, if the content of the memory is logic Low, transistor M_1 has to be quickly turned off so that current distribution does not happen. The previously explained current comparison scheme for conditional discharging works also for this purpose. As the node voltage V_c goes up, the current I_c decreases, slowing down the process of turning transistor M_1 off. When the current I_C becomes smaller than $0.75I_{ref}$ (=3 μA), the voltage at the comparison node is brought down (V_{comp}: Low), which turns PMOS transistor M_{10} on. This action connects node N_c to V_{DD}, since M_{11} is turned on during the operation phase (ϕ_2 is low), leading to the quick voltage rise at N_c, which turns off transistor M_1. The same action takes place for turning off transistor M_{12}, which controls the voltage of transistors for generating the current sinks. Transistor M_{13} is turned on when I_c becomes smaller than $0.75I_{ref}$ to quickly discharge the gate of transistor M_{12}.

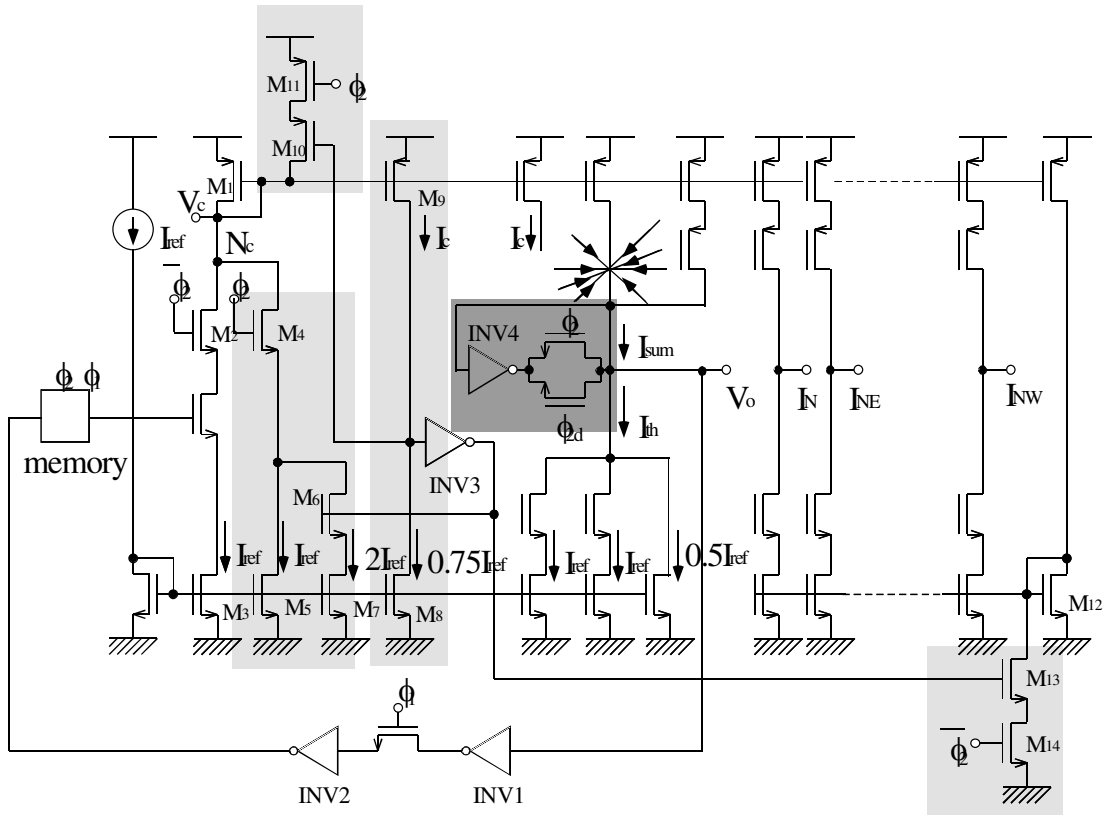

Fig. 10 Pixel circuit incorporating two mechanisms for high-speed operation.

The second mechanism to improve the speed is shown in the darkly shaded block in Fig. 10. This circuit is a simple self-feedback inverter with two switches to keep the voltage at the thresholding node constant. The inverter is designed to have a switching voltage of 1.15V. The voltage is one half of 2.3V, which appears at the output of an NMOS switch after transfer of logic High (=V_{DD}) signal. The voltage is kept constant during ϕ_2 phase and an extended time period by an additional control signal ϕ_{2d}, which is a delayed version of ϕ_2. By this mechanism, the initial voltage from which the voltage starts changing can always be set to 1.15 V. Even a small change in the voltage drives the next inverter INV1 to either logic High or Low to achieve high-speed operation.

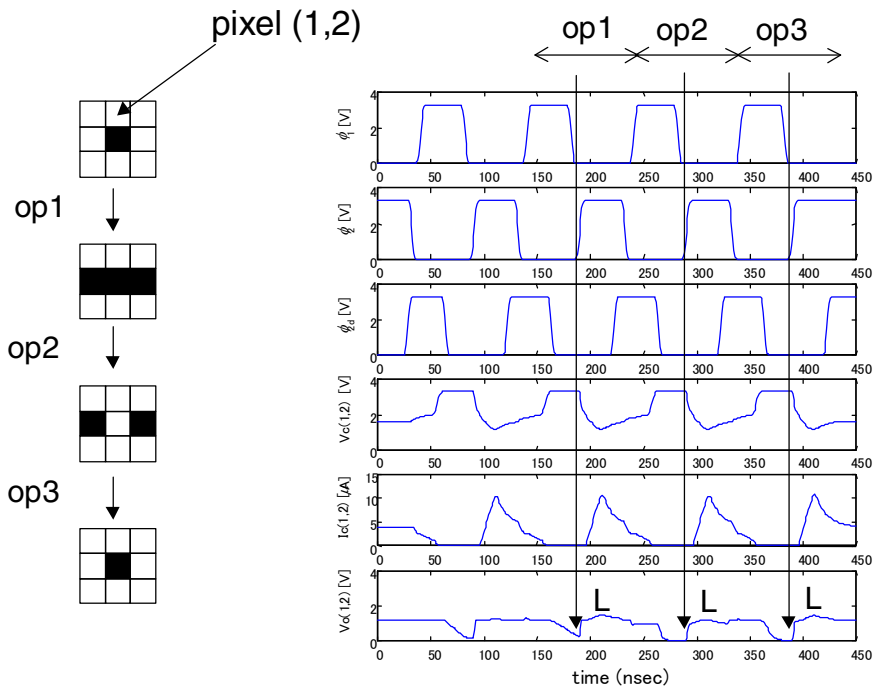

Fig. 11 Simulation results at the clock frequency of 10 MHz using the circuit shown in Fig. 10 with the delay between ϕ_{2d} and ϕ_2 set to 30 nsec. The voltages and the current shown on the right represent those at pixel (1,2) (below the center pixel) for the line completion operation consisting of three steps (op1: line elongation; op2: linestop detection; op3: removal of the linestops). The operation result is indicated by the voltage sampled at the time point specified by downward arrows .

The additional duration control by signal ϕ_{2d}, which is a delayed version of the signal ϕ_2, is to ensure that the state of all transistors sourcing a current towards the thresholding node or those sinking a current from the thresholding node are completely settled at the end of ϕ_{2d}. This indicates that the result of the current comparison is immediately reflected as the change of the voltage of the thresholding node. Fig. 11 shows the simulation result with the modified circuit at the frequency of 10 MHz. The delay between ϕ_{2d} and ϕ_2 was chosen as 30 nsec, which is the required time for transistor M_1 to be turned off completely. All the operations are executed correctly. The node voltage at the thresholding node V_o is kept at 1.15 V until ϕ_{2d} gets low. Immediately after ϕ_{2d} goes low, the voltage V_o starts decreasing for *op1, op2,* and *op3*

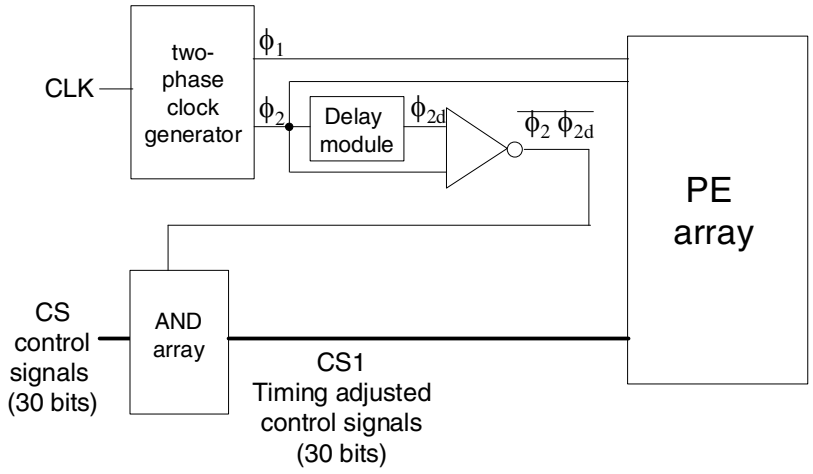

Fig. 12 Schematic of the timing control circuit.

operations to produce the correct output.

4.2 Control Circuits

Fig. 12 shows the schematic of a timing control circuit. The inputs for the circuit are a master clock (CLK) and 30-bit control signals (CS). The circuit generates two non-overlapping clocks (ϕ_1 and ϕ_2). ϕ_{2d} is then generated by introducing some delay for ϕ_2. The amount of delay is externally controlled by setting the number of inverters ϕ_1 has to go through. The control signals CS, whose status changes when CLK rises, are gated with $\overline{\phi_2 \phi_{2d}}$ to produce truncated control signals (CS1). The new control signals CS1 are generated so that: (1) the onset of the signals CS1 is synchronized with the time when the operation phase is initiated (ϕ_2 goes Low), (2) when the processing result stored at the intermediate node is transferred to the memory (ϕ_1 goes Low ϕ_1 goes High), the control signal is still maintained (until ϕ_{2d} goes High).

4.3 Layout and Fabrication

Fig. 13 (a) shows the layout of the pixel. Care was taken to make the pixel shape

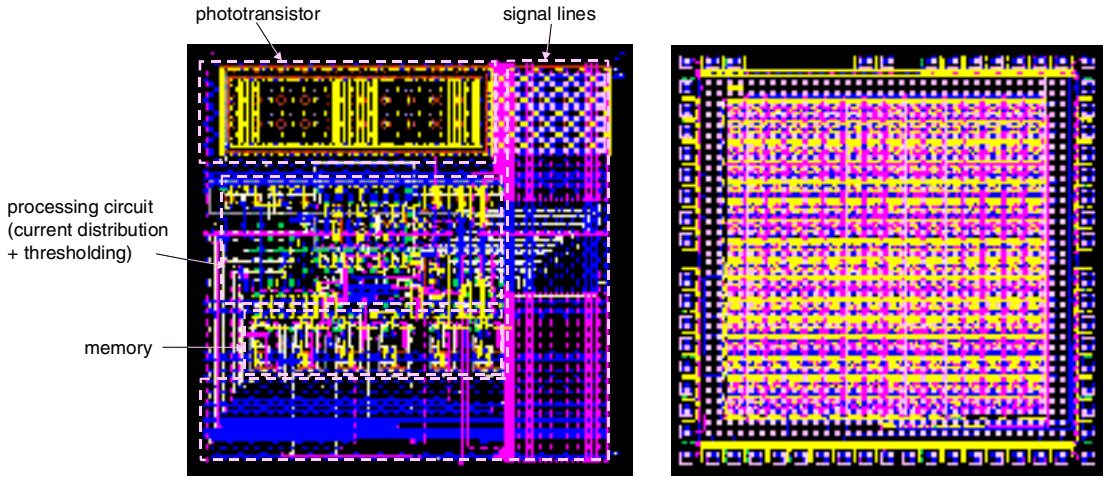

Fig. 13 Layout of the chip. (a) pixel layout (154.5μm x 153.3μm) (b) entire chip (3.2mm x 3.2mm).

square-like to have an equal number of pixels in both *x*- and *y*- directions on the chip. The pixel size measures 154.5 μm × 153.3 μm. The phototransistor, which is implemented at the top left corner of the pixel, measures 99.6 μm × 29.7 μm (base area) and accounts for 12.5% of the entire pixel area. The transistor accounts for 32.4 % of the pixel area. The rest of the pixel, which consumes more than half the entire area, was used for the common signal lines, which were laid out both horizontally and vertically. The first and the second metal layers were used for signal lines while the third metal was used for shielding the circuit from light illumination.

An array of 16 × 16 pixels were placed on a chip area of 3.2 mm × 3.2mm. Fig. 13 (b) shows the layout of the entire chip. Based on this layout, the sensor was fabricaed using HP 0.5 μm technology through MOSIS.

5 Experimental Results

The fabricated sensor was mounted on a test board. Various images were projected on the sensor through a lens. An additional light source was used to clearly produce the binary image.

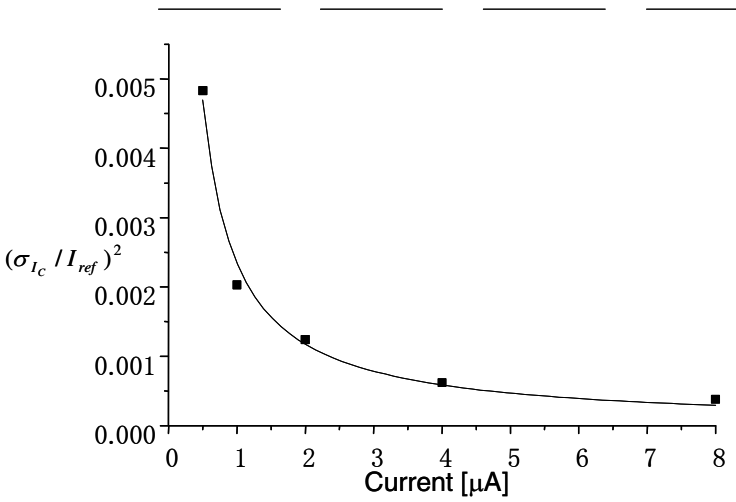

Fig. 14 Relative variation of the current as a function of the reference current.

The power supply voltage of 4V was used for experiments.

5.1 Mismatch measurement

The degree of transistor mismatch was estimated by measuring the current output (I_C in Fig. 4) from the entire pixels, which should be ideally equal to the reference current I_{ref}. However, the output varies from pixel to pixel due to transistor mismatch. The measurement was performed for different values of I_{ref}, i.e., 0.5 μA, 1 μA, 2 μA, 4 μA, and 8 μA. For each measurement, the mean and the variance of the 256 output currents were calculated. Based on the measurement result, the relative variation of the output current was plotted as a function of the reference current I_{ref}, which is shown in Fig. 14. The measurement result demonstrates that the relative current variation decreases as the reference current becomes larger. This behavior is nicely fit by the following equation:

$$\left(\frac{\sigma_{I_C}}{I_{ref}}\right)^2 = \frac{0.00235}{I_{ref}}. \tag{24}$$

Since I_C is obtained as a result of PMOS current mirroring, its variance is represented as

$$\sigma_{I_C}^2 = 3s^2 I_{ref}^2 \tag{25}$$

Substituting equation (25) into (24) yields

$$3s^2 = \frac{0.00235}{I_{ref}}. \tag{26}$$

Expressing *s* in terms of design parameters (equation (9)) gives

$$3\frac{2A_{V_T}^2 \mu C_{ox}}{L^2 I_{ref}} = \frac{0.00235}{I_{ref}}. \tag{27}$$

By substituting numerical values for parameters μC_{ox}, L, I_{ref}, the above equation can be solved for A_{V_T} as

$$A_{V_T} = 7.0 \text{ [mV}\mu\text{m]}. \tag{28}$$

This value is almost one half of the value assumed in the simulation (15 mVμm). However, the assumed value was a rather conservative estimate, and the measured value of $A_{V_T} = 7.0$ [mVμm] agrees well with the reported result extrapolated toward smaller feature sizes [19]. As a result of this rather conservative estimate for A_{V_T}, the error rate should be lower than expected.

5.2 Speed measurements

The maximum operating frequency was investigated as a function of the reference current and the internal delay between ϕ_{2d} and ϕ_2. For this purpose, line stop detection operation was applied for an image that consists of a set of line segments whose length are two. If the operation is performed without error, these two pixels are both detected as linestops. The maximum operating frequency is defined as the frequency below which no detection error occurs (error rate of 0 %). Instead of finding a maximum operating frequency, a minimum required current for a given frequency, which can be set only to certain discrete values with a maximum of 5 MHz, is determined. The experiment was carried out for different settings of

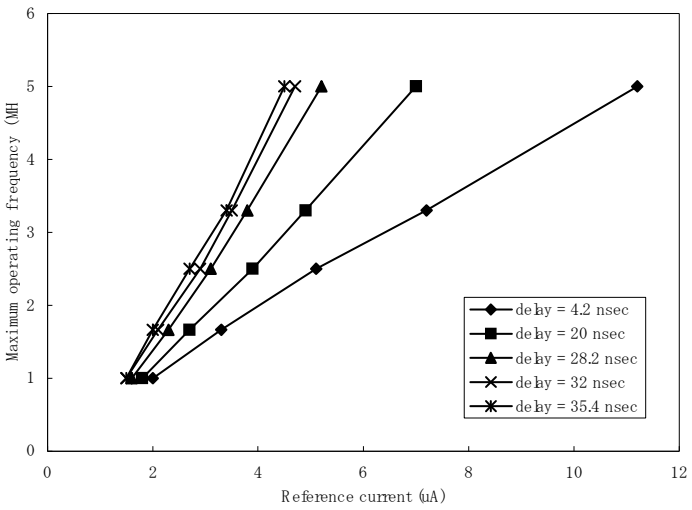

Fig. 15 Maximum operating frequency as a function of the reference current for different settings of internal delay between ϕ_2 and ϕ_{2d}.

internal delay between ϕ_2 and ϕ_{2d}.

The obtained relationship between the maximum operating frequency and the reference current for different settings of internal delay is shown in Fig. 15. It is obvious from the graph that the maximum frequency almost linearly increases as a function of the reference current. The maximum frequency also increases as the delay of ϕ_{2d} increases. It should be noted that the degree of improvement of the speed becomes less and less as the delay increases, and seems to be reaching the point close to saturation when the delay is 35.4 ns. This is the time required for the sourcing and sinking currents to become almost zero, which is close to the estimate of 30 nsec obtained in the simulation. No further improvement in the operational speed would be expected for larger delays. From this experiment, it is concluded that a frequency of 5 MHz is obtained under the following conditions: V_{DD} = 4 V, I_{ref} = 4.5 μA, the delay between ϕ_2 and ϕ_{2d} set to 35.4 nsec.

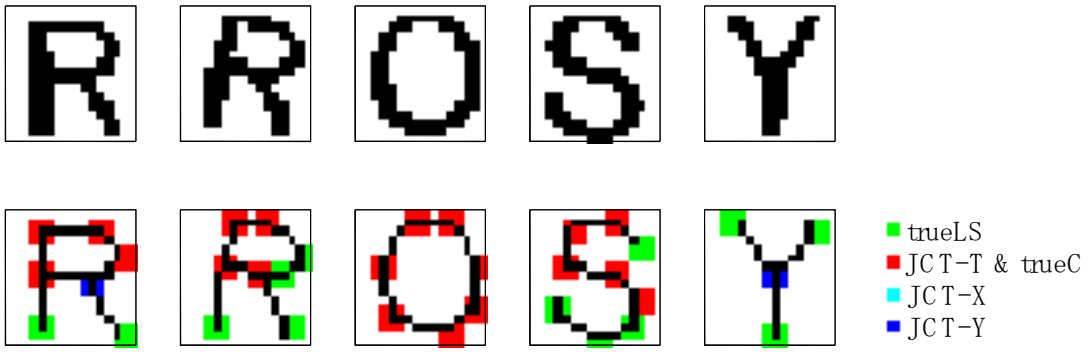

Fig. 16 Sensor responses to various letter images. For an input image shown on the top row, a set of features is obtained as shown in the bottom row. These features are superimposed on the thinned image. The center of a closed rectangle indicates the position of the detected feature.

5.3 Responses to letter images

Fig. 16 shows detected features by the sensor for five letter images under the condition described above. The images shown in the bottom row are reconstructed by superimposing each feature at the detected position on the thinned image. All the important features were detected in a discriminative fashion. These experimental results were completely identical to the result obtained by running a simulation for a digitized image.

In certain cases the sensor produces unexpected results. For example, for the slanted letter "R", two pixels are detected as linestops instead of corners. This is because the 45° line segment constituting the loop is removed by the EIP (Elimination of Isolated Points) operation due to its length being two pixels. This type of removal of short line segments also explains why only seven corners are detected instead of eight for the letter “O” and the unexpected linestops along the stroke for the letter “S”. This issue originates from the low resolution of the sensor: an array of 16×16 is not large enough for some letters to apply the present feature detection algorithm. Hence, the problem would be easily solved if the larger number of pixels were used in future.

Table 2. Execution time required for each operation. The result of each processing is shown as X_i in Fig. 1. The processing time is calculated for the operation of 5 MHz.

order	operation	resultant image	no. of steps	time (μsec)
1	binarize	X_0	1	0.2
2	thinning (1 cycle)		24	4.8
3	thinning (1 cycle)	X_1	24	4.8
4	OD	X_{2a}, X_{2b}, X_{2c}, X_{2d},	25	5
5	LC	X_{3a}, X_{3b}, X_{3c}, X_{3d}	26	5.2
6	EIP	X_{4a}, X_{4b}, X_{4c}, X_{4d}	20	4
7	LI	X_{5a}, X_{5b}, X_{5c}, X_{5d}	8	1.6
8	EIP	X_{6a}, X_{6b}, X_{6c}, X_{6d}	20	4
9	LT		2	0.4
10	LE		4	0.8
11	LT	X_{7a}, X_{7b}, X_{7c}, X_{7d}	2	0.4
12	AND2MabcdToMx	X_{10}	29	5.8
13	LSD	X_{8a}, X_{8b}, X_{8c}, X_{8d}	4	0.8
14	ORMabcdToMy		8	1.6
15	ORMxyToMy	X_{13}	1	0.2
16	rectangleMy		4	0.8
17	shrinkMy		4	0.8
18	connectMy		12	2.4
19	rectangleMy		4	0.8
20	rectangleMy	X_{14}	4	0.8
21	shrinkMy	X_{15}	4	0.8
22	LSD1	X_{9a}, X_{9b}, X_{9c}, X_{9d}	8	1.6
23	ORMabcdToMy	X_{16}	8	1.6
24	AND2MabcdToMabcd	X_{17}	1	0.2
25	rectangleMx	X_{11}	4	0.8
26	shrinkMx	X_{12}	4	0.8
27	feature_calc	X_{18}, X_{19}, X_{20}, X_{21}, X_{22}	16	3.2
	total		271	54.2

Table 2 shows the execution time for each operation used for the detection of the following four features: JCT-T, JCT-X, totalC (union of trueC and JCT-Y), and trueLS. This is because the current implementation of six memories does not allow the computation of the five features in one sequence of operations. The number of steps required for each operation is converted to the execution time by multiplying 200 ns/operation for the operational frequency of 5 MHz. The thinning operation is performed twice although in reality it depends on the original thickness of the line. Some operations were implemented by combinations of other (basic) operations (AND2MabcdToMx, ORMabcdToMy), which resulted in extra processing time. The total execution time to perform over 270 individual processing steps is about 50 μs,

which is short considering the complexity of the operations involved. By increasing the functionality of the memory the above-mentioned operations could be performed also in a single step, further reducing the overall processing time.

6 Discussion

The fabricated feature detection sensor operated successfully at high speed. The sensor has considerably more computational power than the one obtained by the sensor architecture using a single conventional signal processor. Suppose that the processor operates at the clock frequency of 1 GHz and that each arithmetic computation requires 1 ns. Also, suppose that the weighted sum is computed in one clock cycle. These assumptions lead to the computation time for template matching for 3×3 neighbors of 9 ns, which further lead to the total computation time of $9N^2$ ns for a pixel array of size $N \times N$. If N is equal to 16, which is the size of the present sensor, the computation time is 2.3 μs. This is ten times larger than the operational speed of 200 ns obtained from the sensor described in the paper. The high speed of the present sensor comes from the parallel computing of the processing elements at each pixel. The speed advantage of the sensor scales as N^2 since the time required for single processor architecture is proportional to N^2, while that required for the present sensor is independent of N.

Table 3 summarizes the specifications of the sensor. Since the sensor is a prototype to demonstrate the principle of on-chip feature detection, the pixel size is still large and the performance can be further improved by several modifications. First, the channel length can be made shorter since the experimentally derived value of A_{V_T} is about a half of the value used in the present design. The reduction of L leads to the reduced size of the pixel and faster operation. Second, the communication between pixels may be simplified. For example, Instead of directly distributing a current to diagonal neighbors, it may be possible to distribute a current to the

Table 3. Specifications of the prototype chip.

technology	HP 0.5 μm CMOS
chip area	3.2 mm × 3.2 mm
pixel size	16 × 16
pixel area	154.5 μm × 153.3 μm
number of transistors	147 / pixel
max operating freq.*	5 MHz
fill factor (photosensor)	12.5 %

(*measured for V_{DD} = 4 V and I_{ref} = 4.5 μA)

horizontal neighbors and these neighbors further distribute the current toward its vertical neighbors. Such a scheme would significantly simplify the wiring for connecting neighboring pixels and hence reduce the pixel area. Third, use of technology of smaller feature sizes enables higher pixel densities in future versions. If the 0.25 μm technology is used, the pixel size would reduce to about 80 μm × 80 μm, enabling an array of 128 x 128 pixels in a chip area of 10 mm x 10 mm. This is a practically useful resolution and is technically feasible to implement. With these modifications, the feature detection sensor may become practical for high speed feature detection applications.

7 Conclusion

The paper describes the design and implementation of a new type of VLSI computational sensor. The sensor consists of an array of 16×16 processing elements, each measuring 150 μm × 150 μm in a chip area of 3.2 mm × 3.2 mm. The sensor detects important image features including corners, three types of junctions (T-type, X-type, Y-type), and linestops for a binary image in a discriminative fashion. To realize fast operation while keeping accuracy, a design procedure based on transistor mismatch has been proposed and successfully employed. Since these features are detected on-chip in about 50 μsec, the sensor can be used for various types of applications requiring high speed feature extraction.

8 References

[1] C. Mead, *Analog VLSI and Neural Systems*. Addison Wesley, Reading, MA, 1989.

[2] D. Standley, “An object position and orientation IC with embedded imager,” *IEEE J. Solid-State Circuits*, vol. 26, no. 12, pp. 1853-1859, 1991.

[3] P. Venier, A. Mortara, X. Arreguit, and E. Vittoz, "An integrated cortical layer for orientation enhancement," *IEEE J. Solid-State Circuits*, vol. 32, no. 2, pp. 177-186, 1997.

[4] M. Barbaro, P. Burg, A. Mortara, P. Nussbaum, and F. Heitger, “A 100 x 100 pixel silicon retina for gradient extraction with steering filter capabilities and temporal output coding,” *IEEE J. Solid-State Circuits*, vol. 37, no. 2, pp. 160-172, 2002.

[5] R. Etienne-Cummings, Z. Kalayjian, and D. Cai, “A programmable focal-plane MIMD image processor chip,” *IEEE J. Solid-State Circuits*, vol. 36, no. 1, pp. 64-73, 2001.

[6] T. Bernard, B. Zavidovique, and F. Devos, “A programmable artificial retina,” *IEEE J. Solid-State Circuits*, vol. 28, no. 7, pp. 789-798, 1993.

[7] J. Eklund, C. Svensson, and A. Åström, “VLSI implementation of a focal plane image processor – A realization of the near-sensor image processing chip concept,” *IEEE Trans. VLSI Systems*, vol. 4, no. 3, pp. 322-335. 1996.

[8] M. Ishikawa, K. Ogawa, T. Komuro, and I. Ishii, "A CMOS vision chip with SIMD processing element array for 1ms image processing," *Technical Digest of the International Solid-State Circuits Conference,* pp. 206-207, 1999.

[9] L. Chua and T. Roska, “Cellular neural networks: Theory,” *IEEE Trans. Circuits Syst*., vol. 35, pp. 1257-1272, 1988.

[10] S. Espejo, A. Rodrigues-Vazquez, R. Domínguez-Castro, J. Huetas, and E. Sanchez-Sinencio, “Smart-pixel cellular neural networks in analog current-mode CMOS technology,” *IEEE J. Solid-State Circuits*, vol. 29, no. 8, pp. 895-905, 1994.

[11] P. Kinget and M. Steyaert, “A programmable analog cellular neural network CMOS chip for high speed image processing,” *IEEE J. Solid-State Circuits*, vol. 30, no. 3, pp. 235-243, 1995.

[12] L. Chua and T. Roska, “The CNN universal machine, part 1: The architecture,” in *Proc. Second IEEE Int. Workshop Cellular Neural Networks and Their Applicat.*, 1992, pp. 1-10.

[13]R. Domínguez-Castro, S. Espejo, A. Rodríguez-Vázquez, R. Carmona, P. Földesy, Á. Zarándy, P. Szolgay, T. Szirányi, and T. Roska, “A 0.8-μm CMOS two-dimensional programmable mixed-signal focal-plane array processor with on-chip binary imaging and instruction storage,” *IEEE J. Solid-State Circuits*, vol. 32, no. 7, pp. 1013-1025, 1997.

[14] F. Attneave, “Some informational aspects of visual perception,” *Psychological Review*, vol. 61, no. 3, pp. 183-193, 1954.

[15] M Nishimura, “A VLSI computational sensor for the detection of image features,” Ph. D. dissertation, University of Pennsylvania, Philadelphia, PA, 2001.

[16] K. Lakshmikumar, R. Hadaway, and M. Copeland, “Characterization and modeling of mismatch in MOS transistors for precision analog design,” *IEEE J. Solid-State Circuits*, vol. 21, no. 6, pp. 1057-1066, 1986.

[17] M. Pelgrom, “Matching properties of MOS transistors,” *IEEE J. Solid-State Circuits*, vol. 24, no. 5, pp. 1433-1440, 1989.

[18] P. Kinget and M. Steyaert, *Analog VLSI Integration of Massively Parallel Processing Systems*, Kluwer Academic Publishers, 1996.