Hardware Implementation of Moment Functions in a CMOS Retina: Application to Pattern Recognition

Olivier Aubreton[1], Lew Fock Chong Lew Yan Voon[1],
Matthieu Nongaillard[1], Guy Cathebras[3], Cédric Lemaitre[2]
and Bernard Lamalle[1]

[1] Laboratoire LE2I – UMR CNRS 5158, 12 rue de la fonderie, 71200 Le Creusot, France
[2] Laboratoire LE2I – UMR CNRS 5158, BP 47870, 21078 DIJON Cedex, France
[3] Laboratoire LIRMM - UMR 5506, 161, rue Ada, 34392 Montpellier Cedex 5, France

Abstract. We present in this paper a method for implementing moment functions in a CMOS retina for object localization, and pattern recognition and classification applications. The method is based on the use of binary patterns and it allows the computation of different moment functions such as geometric and Zernike moments of any orders by an adequate choice of the binary patterns. The advantages of the method over other methods described in the literature are that it is particularly suitable for the design of a programmable retina circuit where moment functions of different orders are obtained by simply loading the correct binary patterns into the memory devices implemented on the circuit. The moment values computed by the method are approximate values, but we have verified that in spite of the errors the approximate values are significant enough to be applied to classical shape localization and shape representation and description applications.

1 Introduction

Geometric and Zernike moments are functions that are often used in the field of image analysis and pattern recognition. Their main applications are in position and shape orientation detection for geometric moments, and in pattern recognition for both geometric and Zernike moments [1, 2]. Geometric moment functions have already been implemented in a certain number of retina chips [3, 4, 5, 6] but none of the circuits are capable of computing geometric moment values of different orders. The advantage and novelty of the method that we propose are that it can be implemented as a programmable architecture that allows the computation of not only geometric moment function values but also other moment values such as Zernike moments of different orders by simply uploading the adequate binary patterns into memory devices integrated on the circuit.

In the next section, we will describe the basic principle of binary pattern matching as well as the proposed retina architecture. In section 3, we show how the binary pattern matching method can be used to compute an approximated value of respectively geometric and Zernike moments of different orders before we conclude in section 4.

J. Martí et al. (Eds.): IbPRIA 2007, Part I, LNCS 4477, pp. 306–313, 2007.

2 Binary Pattern Matching CMOS Retina

2.1 Binary Pattern Matching

The basic principle of binary pattern matching is to compute two correlation values, denoted by S_1 and S_2, between a gray level image and a binary image (or binary pattern) according to the two following equations:

$$S_1 = \sum_{x=1}^{N} \sum_{y=1}^{M} F(x, y) \times I(x, y) \tag{1}$$

$$S_2 = \sum_{x=1}^{N} \sum_{y=1}^{M} \overline{F(x, y)} \times I(x, y) \tag{2}$$

where M and N are the size of the image, F(x, y) the binary pattern (a binary function that takes the value 0 or 1 according to the x and y coordinates of the pixel), $\overline{F(x, y)}$ the inverse binary pattern and I(x, y) the intensity of the pixel at coordinates (x, y) of the gray level image.

The reason why a binary pattern is used instead of a gray level pattern is due to hardware implementation constraints. The circuit architecture that is proposed uses memory devices to store the binary pattern on the chip and computes the correlation values between the image of the scene under observation and the stored pattern. With binary patterns only a one-bit memory device is necessary to store each pixel of the pattern whereas with gray level patterns several bits are necessary resulting in larger memory devices. Since the latters are integrated at the pixel level, large memory devices will result in low fill factor value (percentage of the pixel devoted to the collection of light) and consequently a less sensitive retina.

2.2 CMOS Retina Architecture

Fig. 1 represents the architecture of the pixel with two one-bit memory devices, $M_0(x, y)$ and $M_1(x, y)$, for the memorization of two binary patterns. The product between the two binary patterns $F_0(x, y)$ and $F_1(x, y)$ (respectively their inverses) and the intensity of the image pixel I(x, y) as expressed in Eq. 1 and Eq. 2 is electronically achieved by switching the photocurrent $I_{ph}(x, y)$ delivered by the photosensitive element to one of the four outputs, o_0, o_1, o_2 and o_3 according to the values of the patterns at coordinates (x, y) stored in the two memory devices $M_0(x, y)$ and $M_1(x, y)$. Note that there is a direct relationship between image intensity I(x, y) and the photocurrent $I_{ph}(x, y)$. For example, if both of the two memory devices contain the logic value 1 then the photocurrent due to the illumination of the photodiode will be switched to the output o_0. The advantage of storing two binary patterns is that it allows us to compute simultaneously four correlation values: two for the first pattern and two for the second pattern.

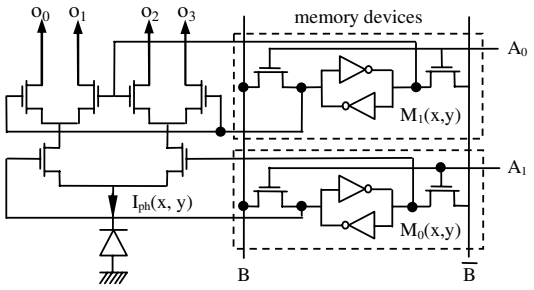

Fig. 1. Architecture of the pixel

The mathematical expressions of the four outputs are:

$$o_0(x, y) = F_0(x, y) \times F_1(x, y) \times I_{ph}(x, y) \qquad o_2(x, y) = \overline{F_0(x, y)} \times \overline{F_1(x, y)} \times I_{ph}(x, y)$$
$$o_1(x, y) = F_0(x, y) \times \overline{F_1(x, y)} \times I_{ph}(x, y) \qquad o_3(x, y) = \overline{F_0(x, y)} \times F_1(x, y) \times I_{ph}(x, y)$$

In the array of pixels, all the outputs with the same index are connected together in order to do current summations that are next converted to voltages using four current to voltage converters. At the output of the retina we obtain thus four signals that can be expressed as:

$$O_0 = K \sum_{x=1}^{N} \sum_{y=1}^{M} F_0(x, y) \times F_1(x, y) \times I_{ph}(x, y) \qquad O_2 = K \sum_{x=1}^{N} \sum_{y=1}^{M} \overline{F_0(x, y)} \times \overline{F_1(x, y)} \times I_{ph}(x, y)$$
$$O_1 = K \sum_{x=1}^{N} \sum_{y=1}^{M} F_0(x, y) \times \overline{F_1(x, y)} \times I_{ph}(x, y) \qquad O_3 = K \sum_{x=1}^{N} \sum_{y=1}^{M} \overline{F_0(x, y)} \times F_1(x, y) \times I_{ph}(x, y)$$

where K is the gain conversion factor of the current to voltage converters. These four outputs can finally be combined in order to obtain simultaneously the following four correlation products:

$$O_0 + O_1 = K \sum_{x=1}^{N} \sum_{y=1}^{M} F_0(x, y) \times I_{ph}(x, y) \qquad O_0 + O_3 = K \sum_{x=1}^{N} \sum_{y=1}^{M} F_1(x, y) \times I_{ph}(x, y)$$
$$O_2 + O_3 = K \sum_{x=1}^{N} \sum_{y=1}^{M} \overline{F_0(x, y)} \times I_{ph}(x, y) \qquad O_1 + O_2 = K \sum_{x=1}^{N} \sum_{y=1}^{M} \overline{F_1(x, y)} \times I_{ph}(x, y)$$

3 Moment Computation

3.1 General Moment Definition

The moment function m_{pq} of order (p+q) of an image is defined by Eq. 3.

$$m_{pq} = K \times \sum_{x=1}^{N} \sum_{y=1}^{M} f_{pq}(x, y) \times Im(x, y) \tag{3}$$

where K is a constant term, Im(x, y) is the value of the pixel of the image under analysis at coordinates (x, y) and $f_{pq}(x, y)$ a function that depends on the type of the calculated moment. For non-central and central geometric moments, for example, the expression of $f_{pq}(x, y)$ is given by respectively Eq. 4 and Eq. 5.

$$f_{pq}(x, y) = \left(\frac{x}{N}\right)^p \times \left(\frac{y}{M}\right)^q \quad \text{and} \quad K = N^p \times M^q \tag{4}$$

$$f_{pq}(x, y) = \left(\frac{x - x_0}{N}\right)^p \times \left(\frac{y - y_0}{M}\right)^q \quad \text{and} \quad K = N^p \times M^q \tag{5}$$

where (x_0, y_0) is the intensity centroid of the image or the coordinates of the center of mass of the object in the case of a bright object over a uniform dark background. Applications of central moments are in pattern recognition, classification and shape descriptions, and one usually needs to compute moment values of different orders. For example, in the approach proposed by Hu [7] central moments of different orders are combined in order to obtain shape descriptors that are invariant to translation or rotation. Another class of moment functions are orthogonal moments. Among the most powerful orthogonal moments used in pattern recognition are Zernike moments. The Zernike moment of order p of an image inside a circle of unit radius is defined by the following equation [1]:

$$z_{pq} = \frac{(p+1)}{\pi} \int_0^{2\pi} \int_0^1 V_{pq}^*(r, \theta) \times \mathrm{Im}(r, \theta) \times r \times dr \times d\theta \tag{6}$$

with $r \le 1$, p a positive integer and q an integer such that $p - |q|$ is even and $|q| \le p$. The functions $V_{pq}^*(r, \theta)$ denote the complex conjugate of Zernike polynomials of order p and repetition q, and Im(r, θ) denotes the image under analysis. For N×N pixels digital images, Eq. 6 can be written in the discrete form [1] as follows:

$$z_{pq} = \frac{(p+1)}{\pi(N-1)^2} \sum_{x=1}^{N} \sum_{y=1}^{N} V_{pq}^*(r, \theta)\, \mathrm{Im}(x, y) \tag{7}$$

where $r = \sqrt{(x^2 + y^2)} \Big/ N$ and $\theta = \tan^{-1}\left(y \Big/ x\right)$.

Expressions of Zernike polynomials can be found in [1] and thus will not be detailed here. The Zernike moment values are thus complex values that can be expressed as the sum of a real part and an imaginary part.

$$z_{pq} = \Re(z_{pq}) + j \times \Im(z_{pq}) \tag{8}$$

where $\Re(z_{pq}) = \frac{(p+1)}{\pi (N-1)^2} \sum\sum \Re\left(V^*_{pq}(x, y)\right) \times \mathrm{Im}(x, y)$ and

$\Im(z_{pq}) = \frac{(p+1)}{\pi (N-1)^2} \sum\sum \Im\left(V^*_{pq}(x, y)\right) \times \mathrm{Im}(x, y)$.

To conclude, moment values are always obtained by computing correlation products between the image under analysis and one (or two for Zernike moments) 2D function.

3.2 Binary Pattern Matching and Geometric Moment Function

Comparing the expression of geometric moment functions given by Eq. 3 to that of the binary pattern matching correlation expression given by Eq. 1, we notice that they are similar. The differences are in the K constant term and the $f_{pq}(x, y)$ and $F(x, y)$ functions. $f_{pq}(x, y)$ is a real function with values in the interval]-1, +1] and $F(x, y)$ is a binary one. Let

$$f_{pq}(x, y) = K_1 F_{pq}(x, y) + K_2 \overline{F_{pq}(x, y)} + \varepsilon_{pq}(x, y) \tag{9}$$

where $F_{pq}(x, y)$ is the binary function to determine, $\varepsilon_{pq}(x, y)$ an error function, K_1 the maximum value of $f_{pq}(x, y)$ and K_2 a value that depends on the values of the function $f_{pq}(x, y)$. K_2 is equal to 0 if $f_{pq}(x, y)$ is a real and positive function and the minimum value of $f_{pq}(x, y)$ if the latter is a real function with both positive and negative values. K_1 and K_2 are scaling or normalization factors to account for the fact that $f_{pq}(x, y)$ is first normalized to the interval [0, 1] before the binary pattern $F_{pq}(x, y)$ is determined by a dithering operation as we will describe later. Replacing $f_{pq}(x, y)$ by its expression given by Eq. 9 in Eq. 3 yields

$$\begin{aligned} m_{pq} = N^p \times M^q \times \Bigg[& K_1 \sum_{x=1}^{N} \sum_{y=1}^{M} F_{pq}(x, y) \times \mathrm{Im}(x, y) + \\ & K_2 \sum_{x=1}^{N} \sum_{y=1}^{M} \overline{F_{pq}(x, y)} \times \mathrm{Im}(x, y) + \sum_{x=1}^{N} \sum_{y=1}^{M} \varepsilon_{pq}(x, y) \times \mathrm{Im}(x, y) \Bigg] \end{aligned} \tag{10}$$

that can be expressed in terms of the two correlation values S_1 and S_2 given in Eq. 1 and Eq. 2 as follows:

$$m_{pq} = N^p \times M^q \times [K_1 S_1 + K_2 S_2] + err_{pq} = M_{pq} + err_{pq} \tag{11}$$

with M_{pq} the approximated moment value obtained by binary pattern matching and err_{pq} an error term.

$$M_{pq} = N^p \times M^q \times [K_1 S_1 + K_2 S_2] \tag{12}$$

$$err_{pq} = N^p \times M^q \times \sum_{x=1}^{N} \sum_{y=1}^{M} \varepsilon_{pq}(x, y) \times Im(x, y) \tag{13}$$

The problem is to find a binary function $F_{pq}(x, y)$ such that the absolute relative error term given by Eq. 14 is minimal.

$$Re\,rr_{pq} = \left| \frac{err_{pq}}{m_{pq}} \right| \tag{14}$$

This problem is similar to the problem of rendering gray level images on a printer and one can find in the literature several techniques to achieve this [8, 9]. We have studied some of the techniques and, for all the images of the Columbia COIL object image database [10], computed the absolute relative error given by Eq. 14 for non-central geometric moment values of order less than or equal to 3 using 128×128 pixels images. Among all the dithering algorithms that we have studied, the Floyd-Steinberg error diffusion algorithm [9] is the one that give the best trade off between computation time and error value. Using this algorithm the absolute relative error has been found to be less than 5%.

Fig. 2. Example of binary pattern

Fig.2 is an example of the binary pattern obtained by applying the Floyd-Steinberg dithering algorithm to the function $f_{01}(x, y)$ for calculating non-central geometric moment of order 1. To conclude, binary pattern matching can be used to compute approximate values of geometric moments using Eq. 12. In the case of geometric moments where $f_{pq}(x, y) \in]0, 1]$, such as non-central geometric moments, $K_1 = 1$ and $K_2 = 0$, and the equation simplifies to

$$M_{pq} = N^p \times M^q \times K_1\, S_1 = N^p \times M^q \times S_1 \tag{15}$$

The significance of the approximated geometric moment values have been verified on shape localization applications where the position and orientation of several shapes have been computed using both the exact and the approximated moment values [11]. The error in the position along the x and y coordinates have been found to be less than 1 pixel. Concerning the orientation the error is less than 3 degrees.

3.3 Zernike Moment Computation

The binary pattern matching method can also be used to compute approximate values of Zernike moments that find applications in pattern recognition and image reconstruction [12]. Zernike moments are complex values that can be expressed as the

sum of a real part and an imaginary part (see Eq. 8). Comparing the real and imaginary parts with the binary pattern matching correlation equation we notice again that, as for geometric moments, there is a similarity. Thus, the real and imaginary parts can be determined by binary pattern matching in the same way as for geometric moments. The binary patterns here are obtained by applying a dithering operation to the gray level image representation of the real and imaginary parts of the Zernike polynomials. Since our method gives approximate moment values, the influence of the error on the value obtained must be studied with respect to pattern analysis, pattern recognition or image compression applications. This is done in the same way as described in [13] on several images. For each image, the approximate values of the Zernike moments are first computed. Then, images reconstructed using both the approximated and the exact values are compared to the original image and the relative error calculated. The relative error is higher when using approximated Zernike moment values however it is less than 2% above the relative error value obtained using exact Zernike moment values. Details of the experiments and more complete results are reported in [11]. Fig.3 shows an example of the reconstructed images using both the approximated and the exact values.

Original image | Reconstruction using exact values | Reconstruction using approximated values

Fig. 3. Reconstruction of an image using exact and approximated values of Zernike moments

The influence of the error on the values of the Zernike moments has also been studied in an application of pattern recognition. Objects are recognized using two types of classifiers: Support Vector Machine (SVM) and a geometric classifier based on the use of 20 values of Zernike invariants. For our experiment we have used images excerpted from the COIL-100 image database [10] of four different objects: mugs, cars, boxes and soda cans. The database contains 72 different views of each object, a fraction of which is used to train the classifiers and the remaining used together with the images of other objects as the test set. An example of the percentage classification error (for both exact and approximated Zernike moment values) for a given object among the four types of objects considered is given in Table 1. The classification error due to the use of an approximated value of the Zernike moment is higher but it is less than 1% above the classification error value using exact Zernike moment values.

Table 1. Example of percentage classification error

	Exact values of Zernike invariants	*Appoximated values of Zernike invariants*
Geometric classifier (Stress polytope)	3,9%	4,5%
SVM	2,7%	3,3%

4 Conclusion

We have presented a binary pattern matching method that allows the design of a programmable CMOS retina capable of computing approximated values of geometric and Zernike moment functions of different orders. To study the influence of the error in the moment value with respect to object localization, and pattern recognition and classification applications, we have conducted several experiments and quantify the localization and classification errors. We have found that the use of approximated moment values results in a higher percentage localization and classification errors with respect to the exact values. However, the increase in the percentage localization and classification errors is not significant and our method is a good hardware implementable solution.

References

1. Mukundan, R., Ramakrishnan, K.R.: Moment functions in image analysis – Theory and applications. World Scientific, Singapore (1998)
2. Zhang, D., Lu, G.: Review of shape representation and description techniques. In: Pattern Recognition, vol. 37, pp. 1–19. Elsevier, North-Holland, Amsterdam (2004)
3. Standley, D.L.: An Object Position and Orientation IC With Embedded Imager. IEEE Journal Of. Solid State Circuits 26(12), 1853–1859 (1991)
4. Yu, N.M., Shibata, T., Ohmi, T., real, A.: time center of mass tracker circuit implemented by neuron MOS technology, IEEE Trans. on Circuits and Systems, vol. 45 (1998)
5. Cummings, R.E., Gruev, V., Abdel, M.G.: VLSI implementation, of motion centroid localization for autonomous navigation, Adv. Neural Inf. Process. Syst., vol. 45 (1998)
6. Deweerth, S.P.: Analog VLSI Circuits For Stimulus Localization and Centroid. International Journal of Computer Vision 8, 191–202 (1992)
7. Hu, M.K.: Visual pattern recognition by moment invariants. IRE Trans. on Inf. Theory 8(1), 179–187 (1962)
8. Pappas, T.N., Allebach, J.P., Neuhoff, D.L.: Model Based Digital Halftonning. IEEE Signal Processing Magazine 20(4), 14–27 (2003)
9. Floyd, R.W., Steinberg, L.: An adaptative algorithm for spatial grey scale. Proc. of Society for Information Display 17(2), 75–77 (1976)
10. Nene, S.A., Nayar, S.K., Murase, H.: Columbia Object Image Library (COIL-100), Technical Report CUCS-006-96, University of Columbia, http://www1.cs.columbia.edu/CAVE/databases/ (1996)
11. Aubreton, O.: Rétines à masques: Utilisation de masques binaires pour l'implantation d'un opérateur de reconnaissance de forme dans une rétine CMOS, Ph.D. thesis, Université de Bourgogne, France (December 2004)
12. Zhang, D., Lu, G.: Review of shape representation and description techniques. Pattern Recognition 37, 1–19 (2004)
13. The C.H.,, Chin, R.T.: On image analysis by the methods of moments. IEEE Trans. Pattern Analysis Machine Intelligence 10(4), 496–513 (1988)