Refined Moment Calculation Using Image Block Representation

Jan Flusser

***Abstract*—This paper deals with effective calculation of moments of binary images. Recently, Spiliotis and Mertzios [1] published a method based on image block representation (IBR). We propose a refinement of their method which yields exact results and performs even more effectively.**

***Index Terms*—Binary image, image block representation, moment calculation.**

I. Introduction

The problem of effective calculation of two-dimensional (2-D) moments of binary images has attracted attention of many researchers because of its importance in numerous object recognition applications. Many methods have been published in the last decade (see, e.g., [2] for a survey).

Zakaria [3], Dai [4], Li [5], and Flusser [6] proposed various approaches based on the decomposition of the object into rows or row segments. Another group of methods is based on Green's theorem, which evaluates the double integral over the object by means of single integration along the object boundary [2], [7]–[9]. Recently, Spiliotis and Mertzios [1] proposed a novel method which employs image representation by nonoverlapping rectangular homogeneous blocks. An image moment is then calculated as a sum of moments of all blocks.

In this correspondence, we propose a refinement of [1]. We use the same block-wise image representation but we present a different scheme to calculate the block moments. We show that our method yields more accurate results and performs even faster than the original one.

II. Original Method by Spiliotis and Mertzios

In [1], the following approach to block moment calculation is described.

Manuscript received January 20, 1999; revised July 23, 1999. This work was supported by Grants 106/97/0827 and 102/00/1711 from the Grant Agency of the Czech Republic. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Eric L. Miller.

The author is with the Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, 18208 Prague 8, Czech Republic (e-mail: flusser@utia.cas.cz).

Publisher Item Identifier S 1057-7149(00)09386-6.

Having an $N \times N$ image block consisting of pixels with values 1, its moment m_{pq} is calculated as

$$m_{pq} = \sum_{i=1}^{N} \sum_{j=1}^{N} i^p j^q \tag{1}$$

(without loss of generality, the block is assumed to have its lower-left-corner pixel centered in $(1, 1)$). Equation (1) can be rewritten as

$$m_{pq} = \left(\sum_{i=1}^{N} i^p\right) \cdot \left(\sum_{j=1}^{N} j^q\right) \equiv S_N^p \cdot S_N^q. \tag{2}$$

In [1], (2) is used for block moment calculation. To evaluate S_N^p for low p, one can use the well-known formulae

$$S_N^1 = \frac{1}{2} N(N+1),$$
$$S_N^2 = \frac{1}{6} N(N+1)(2N+1).$$

For $p > 4$, Spiliotis and Mertzios employed the following recursive formula

$$\sum_{k=1}^{m} \binom{m+1}{k} S_N^k = (N+1)^{m+1} - (N+1). \tag{3}$$

III. Refined Method

In this Section, we propose another method for block moment calculation. To explain its idea, we recall the original definition of moments in the continuous domain

$$M_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y)\, dx\, dy \tag{4}$$

where $f(x, y)$ is, in our case, the characteristic function of the block. Clearly, (1) is only an approximation of (4). An error $|M_{pq} - m_{pq}|$ is introduced due to zero-order approximation and numeric integration of $x^p y^q$ over each pixel. We show how to calculate the block moments (4) in the discrete domain exactly without any approximation.

Let us consider a square block with its corner pixels centered in $(1, 1), (1, N), (N, 1)$ and (N, N). The characteristic function of this block is

$$f(x, y) = 1, \quad \text{if } (x, y) \in (0.5, N+0.5) \times (0.5, N+0.5),$$
$$f(x, y) = 0, \quad \text{otherwise.}$$

According to (4), the exact block moment is given as

$$\begin{aligned} M_{pq} &= \int_{0.5}^{N+0.5} \int_{0.5}^{N+0.5} x^p y^q \, dx\, dy \\ &= \frac{1}{(p+1)(q+1)} [(N+0.5)^{p+1} - 0.5^{p+1}] \\ &\quad \cdot [(N+0.5)^{q+1} - 0.5^{q+1}]. \end{aligned} \tag{5}$$

IV. Comparison of the Two Methods

Let us compare the accuracy of the above methods first. While (5) yields exact results, (2) calculates some moments with errors. There is always $M_{pq} \geq m_{pq}$; the error $M_{pq} - m_{pq}$ depends on the given p and q. Some examples are as follows:

$$M_{20} - m_{20} = \frac{m_{00}}{12}$$
$$M_{31} - m_{31} = \frac{m_{11}}{4}$$
$$M_{40} - m_{40} = \frac{m_{20}}{2} + \frac{m_{00}}{80}.$$

Spiliotis and Mertzios analyzed the computing complexity of moment calculation by means of (2) and (3), respectively. To compute all moments of order up to $(L-1, L-1)$ using (2), one needs $2LN$ power calculations, L^2 multiplications and $2LN$ additions. The usage of (3) is more effective for large blocks because its complexity does not depend on the block size. It requires $4L$ power calculations, $2L^2 - L$ multiplications and $L^2 - L$ additions.

When using our refined method (5), the terms

$$\frac{(N+0.5)^{p+1} - 0.5^{p+1}}{p+1}$$

can be precalculated in advance for $p = 0, 1, \ldots, L-1$. Thanks to this, the method requires only $2L$ power calculations, $L^2/2 + L$ multiplications and $2L$ additions, or, equivalently, $L^2/2 + 3L$ multiplications and $2L$ additions. Clearly, this is more effective than the original method.

The difference between the both methods is even more marked when one wants to calculate only one moment of order (p, q). In such a case, (2) requires $2N$ powers and $2N$ additions. The use of the recursive formula (3) is not a good choice: To find S_N^p, we have to evaluate all lower-order sums $S_N^1, \ldots, S_N^{p-1}$, that takes $O(p^2)$ operations. On the other hand, the complexity of moment calculation by means of (5) does not depend on the moment order and the block size either. M_{pq} is obtained after ten elementary operations only.

V. Conclusion

We have presented a refinement of the recently published method for moment calculation based on image block representation. The new technique has been shown to outperform the original one, both in accuracy and computational complexity.

References

[1] I. M. Spiliotis and B. G. Mertzios, "Real-time computation of two-dimensional moments on binary images using image block representation," *IEEE Trans. Image Processing*, vol. 7, pp. 1609–1615, Nov. 1998.

[2] L. Yang and F. Albregtsen, "Fast and exact computation of cartesian geometric moments using discrete Green's theorem," *Pattern Recognit.*, vol. 29, no. 7, pp. 1061–1073, 1996.

[3] M. F. Zakaria, L. J. Vroomen, P. Zsombor-Murray, and J. M. van Kessel, "Fast algorithm for the computation of moment invariants," *Pattern Recognit.*, vol. 20, no. 6, pp. 639–643, 1987.

[4] M. Dai, P. Baylou, and M. Najim, "An efficient algorithm for computation of shape moments from run-length codes or chain codes," *Pattern Recognit.*, vol. 25, no. 10, pp. 1119–1128, 1992.

[5] B. C. Li, "A new computation of geometric moments," *Pattern Recognit.*, vol. 26, no. 1, pp. 109–113, 1993.

[6] J. Flusser, "Fast calculation of geometric moments of binary images," in *Proc. 22nd OAGM'98 Workshop Pattern Recognition Medical Computer Vision*, Illmitz, Austria, 1998, pp. 265–274.

[7] B. C. Li and J. Shen, "Fast computation of moment invariants," *Pattern Recognit.*, vol. 24, no. 8, pp. 807–813, 1991.

[8] X. Y. Jiang and H. Bunke, "Simple and fast computation of moments," *Pattern Recognit.*, vol. 24, no. 8, pp. 801–806, 1991.

[9] W. Philips, "A new fast algorithm for moment computation," *Pattern Recognit.*, vol. 26, no. 11, pp. 1619–1621, 1993.