Time-Varying Image Processing and
Moving Object Recognition, 3 – V. Cappellini (Ed.)

Real-Time Computation of Statistical Moments on Binary Images Using Block Representation

Iraklis M. Spiliotis and Basil G. Mertzios

Automatic Control Systems Laboratory, Department of Electrical Engineering, Democritus University of Thrace, Xanthi 67100, HELLAS.
Fax: +30-541-26947,
e-mail: iraklis%xanthi@ics.forth.gr or mertzios%xanthi@ics.forth.gr

This paper presents a new approach and an algorithm for binary image representation, which is applied for the fast and efficient computation of moments in binary images. In the terminology of this paper the binary image representation scheme is called block representation, since it represents the image as a set of nonoverlapping rectangular areas. The fast computation of moments in block represented images, is achieved exploiting the rectangular structure of the blocks.

1. INTRODUCTION

This paper presents a new advantageous representation for binary images, which in the sequel will be called *block representation.* In the block representation process the whole binary image is decomposed in a set of rectangular areas with object level. The block representation exploits the fact that many compact areas of a given binary image have the same value.

Various sets of two-dimensional (2-D) statistical moments constitute a well-known image analysis and pattern recognition tool [1]-[5]. In pattern recognition applications a small set of the lower order moments is used to discriminate among different patterns. The most common moments are the geometrical moments, the central moments, the normalized central moments and the moments invariants [1]. Other sets of moments are the Zernike moments and the Legendre moments which are based on the theory of orthogonal polynomials [4], [6] and the complex moments [5].

One main difficulty concerning the use of moments as features in pattern recognition applications is the implied high computational time. In the proposed approach, which is based on block represented images, the computational complexity of the moments calculation is in the general case independent of the size of the image and depends only on the image content and on the number of the required moments.

2. BLOCK REPRESENTATION

A bilevel digital image is represented by a 2-D array. Without loss of generality, we suppose that the object pixels are assigned to level 1 and background pixels to level 0. Due to this kind of representation, there are rectangular areas of object value 1 with edges parallel to the image axes. At the extreme case one pixel is the minimum rectangular area of the image. These rectangulars will be called *blocks* in the terminology of this paper.

Consider a set that contains as members all the nonoverlapping blocks of a specific binary image, in such a way that no other block can be extracted from the image (or equivalently each pixel with object level belongs to only one block). This set represents the image without loss of information. It is always feasible to represent a binary image with such a set, of all the nonoverlapping blocks with object level. We call this representation of the binary image, *block representation*. Figure 1 illustrates an image of the character d and the blocks.

The block representation concept leads to a simple and fast algorithm, which requires just one pass of the image and simple bookkeeping process. Consider a binary image $f(x,y)$, $x=0,1, .. ,N_1-1$, $y=0,1, .. , N_2-1$. The block extraction process requires a pass from each line y of the image. In this pass all object level intervals are extracted and compared with the previous extracted blocks.

In the following, a uniquely determined block representation algorithm, which is based on the row by row processing, is given.

Block representation algorithm

Step 1: For each line y of the image f,

Step 2: Find the object level intervals in line y

Step 3: Compare intervals and blocks that have pixels in line y-1,

Step 4: If an interval does not matches with any block, this is the beginning of a new block.

Step 5: If a block matches with an interval, the end of the block is in the line y.

As a result of the application of the above algorithm, we obtain a set of all the rectangular areas with level 1, that form the object. The block extraction process is implemented easily with low computational complexity, since it is a pixel checking process without numerical operations. In Figure 2, the application of the block representation algorithm in a binary image, is illustrated.

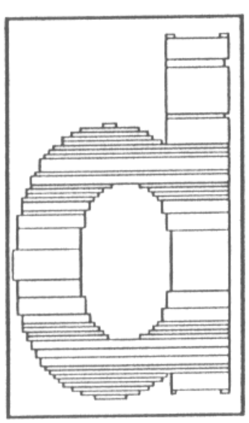

Figure 1. Image of the character d and the blocks.

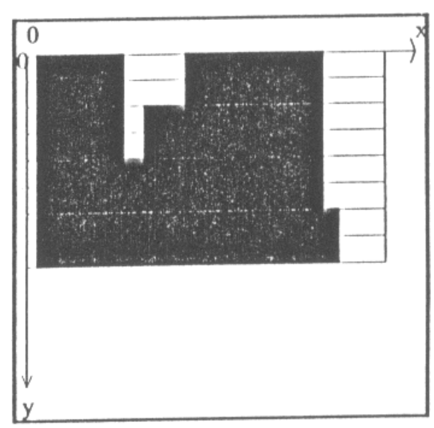

Figure 2. Application of the block representation algorithm.

It is possible to implement various processing and analysis algorithms with block represented binary images, since the most of these algorithms operate on image regions (blocks). The most important characteristic of the block representation is that a perception of image parts greater than a pixel, is provided to the machine. Therefore, all the operations on the pixels belonging to a block are substituted by a simple operation on the block; thus the block representation is advantageous for image modelling.

The block representation may be seen as a physical way for the representation of binary images. Each block is represented with four integers, the coordinates of the upper left and down right corner in vertical and horizontal axes. The block representation is an information lossless representation. It is noticeable, that the total amount of information required for the block representation of an image remains unchanged, under the scale of the image. We conclude that the block representation of a binary image is dependent only on the image content and is independent of the image size.

3. GEOMETRICAL MOMENTS

Consider a binary digital image *f(x,y)*, with N_1 pixels in horizontal axis and N_2 pixels in vertical axis. The 2-D geometrical moments of order *(p,q)* of the image are defined by the relation:

$$m_{pq} = \sum_{x=0}^{N_1-1} \sum_{y=0}^{N_2-1} x^p y^q f(x,y), \qquad p,q = 0,1,2,... \tag{1}$$

If the image is block represented, it is represented as the set of the nonoverlapping blocks that constitute the image, as follows:

$$f(x,y) = \{ b_1, b_2, \dots, b_k \} \tag{2}$$

where k is the number of the blocks. Since the background level is 0, only the pixels with level 1 are taken into account for the computation of the moments. Thus the 2-D geometrical moments of order *(p,q)* of the image *f(x,y)* are defined by the relation:

$$m_{pq} = \sum_{x} \sum_{y} x^p y^q \qquad \forall\, x,y: \quad f(x,y) = 1 \tag{3}$$

Since the image pixels with level 1 belong to the image blocks, equation (3) is rewritten as:

$$m_{pq} = \sum_{i=1}^{k} \sum_{x=x_{1,b_i}}^{x_{2,b_i}} \sum_{y=y_{1,b_i}}^{y_{2,b_i}} x^p y^q \tag{4}$$

where x_{1,b_i}, x_{2,b_i} and y_{1,b_i}, y_{2,b_i} are the coordinates of the *i*-th block with respect to the horizontal axis and to the vertical axis, respectively.

3.1 Computation of the geometrical moments of one block

Consider the block b, with x_{1b}, x_{2b} coordinates with respect to the horizontal axis and y_{1b}, y_{2b} with respect to the vertical axis. Then the 2-D geometrical moments m_{pq}^{b} for the block b are given by

$$m_{pq}^{b}=\sum_{x=x_{1b}}^{x_{2b}}\sum_{y=y_{1b}}^{y_{2b}}x^{p}y^{q}=\;x_{1b}{}^{p}\sum_{y=y_{1b}}^{y_{2b}}y^{q}\;+\;(x_{1b}+1)^{p}\sum_{y=y_{1b}}^{y_{2b}}y^{q}\;+\ldots+x_{2b}{}^{p}\sum_{y=y_{1b}}^{y_{2b}}y^{q}=$$

$$=\sum_{x=x_{1b}}^{x_{2b}}x^{p}\sum_{y=y_{1b}}^{y_{2b}}y^{q} \qquad (5)$$

Using the rectangular form appeared within the block, the computation effort for equation (4) is reduced from $O(N^2)$ to $O(N)$ in equation (5). For the computation of (5), it is enough to calculate the summations of the powers of x and y.

Moreover, taking into account that the sum $1^m+2^m+\ \ldots\ +n^m$, is provided for every $m\in Z$, by the following analytical formulae:

$$\sum_{i=1}^{n}i=\frac{n(n+1)}{2}\;,\quad \sum_{i=1}^{n}i^{2}=\frac{n(n+1)(n+2)}{6}\;,\quad \sum_{i=1}^{n}i^{3}=\frac{n^{2}(n+1)^{2}}{4}$$

$$\binom{m+1}{1}\sum_{i=1}^{n}i+\binom{m+1}{2}\sum_{i=1}^{n}i^{2}+\ \ldots\ +\binom{m+1}{m}\sum_{i=1}^{n}i^{m}=(n+1)^{m+1}-(n+1) \qquad (6)$$

the sum of the powers of x is computed using (6) and the following formula:

$$\sum_{x=x_{1b}}^{x_{2b}}x^{p}=\sum_{x=1}^{x_{2b}}x^{p}-\sum_{x=1}^{x_{1b}-1}x^{p} \qquad (7)$$

The sum of the powers of y in (5) is computed in a similar manner. Fast computation of the 2-D geometrical moments of one block, according to (5), is achieved with the above simple and analytical formulae (6) and (7).

3.2 Computation of the geometrical moments of the whole image

According to the equation (5), the 2-D geometrical moments of the whole image are computed as the summation of the 2-D geometrical moments of all the individual blocks of the binary image.

4. COMPUTATIONAL COMPLEXITY

The following analysis refers to the computational complexity of the geometrical moments. The other sets of moments can be computed in a similar manner. Consider that a binary image contains one rectangular block with level 1. For simplicity and without loss of generality, suppose a square block with MxM points. In the sequel we estimate the computational complexity required for the geometrical moments computation up to the order $(L-1,L-1)$.

The direct computation from equation (1) of one geometrical moment of that block requires M^2 power computations, M^2 multiplications and M^2 additions. For the computation of L^2 moments, it results that L^2M^2 power computations, L^2M^2 multiplications and L^2M^2 additions are required.

Consider equation (5), which exploits the rectangular form appearing within the block. For the computation of the sum of the powers of x, LM power calculations and LM additions are required. The same number of operations are required for the computation of the sum of the powers of y. Therefore, for the computation of L^2 geometrical moments from equation (5), $2LM$ power calculations, L^2 multiplications and $2LM$ additions, are required for one block.

Now, consider the analytical formula (6). The binomial coefficients are appeared in the computation formula for any specific geometrical moment of every block of the image; therefore the computational effort is reduced by the number of the blocks. Moreover the factorials that have to be computed, for the determination of the binomial coefficients, require least effort, i.e. one multiplication for the calculation of $m!$ in terms of $(m-1)!$. Therefore the complexity for the computation of the binomial coefficients of (6) is minor.

The sum x^i, $i = 1,2,...,p-1$, that have been computed previously and their values are stored, are used for the computation of the sum of x^p. Thus the computation of the sum of x^p from equations (6) and (7) requires 2 power calculations, p multiplications and p additions. The computation of the sum of y^q requires 2 power calculations, q multiplications and q additions. For L^2 moments, $4L$ power calculations, $3L^2-L$ multiplications and L^2-L additions, are required, for the computation of all the L^2 square moments.

Table 1 demonstrates the above results. The complexity is reduced from 2-D form from (1) to 1-D by the use of block representation and equation (5). Moreover, the complexity is independent to the size with the use of the analytical formulae (6) and (7). The required number of power calculations, of multiplications and of additions for the computation of the geometrical moments up to the order (4,4) of a block with MxM pixels, where M varies from 1 to 100, is illustrated in Figure 3.

Lemma 1

Assuming that the complexity of raising a number to a power is the same as one multiplication, the computation of (5) and (6) requires L^2+2LM and $3L^2+2L$ multiplications respectively. Comparing the above number of multiplications it is concluded that (5) has less computational complexity than (6) when

$$L^2 + 2LM \leq 3L^2 + 2L \Rightarrow M \leq L + 1 \qquad (8)$$

However, in typical pattern recognition applications, the higher order moments are not used since are very sensitive to noise; usually they are calculated up to the order (4,4). Usually, in images with a high entropy value, like images of text, where a significant number of small blocks appears, the time for moments computation is reduced by a factor among 10 and 50, using block representation. In images with large areas of object level, like images of industrial parts, aircrafts, ships e.t.a. the factor of time reduction is much greater. Consider the aircraft image, illustrated in Figure 4. The image size is 287x207 points. Using the algorithm described in Section II, 195 blocks extracted from the image. The total number of points with object level is 19922. The geometrical moments of that image have been computed up to the order (4,4), using the different methods described in this paper. For the direct computation from (1), 4.77 sec are required. Using the block represented image and equation (5), 42 msec are required. The use of the analytical formulae (6) and (7) increases the computation time to 55 msec, since in most of the extracted blocks one edge has width 1 or 2 points. Using the criterion, provided by Lemma 1 for L=5, the computational time is decreased to 15 msec; that implies a factor of time reduction equals to 318 times less, in comparison with the time required for the computation by (1). The above results are shown in Table 2. From Table 2, it results that an average rate greater than 40 frames/sec is achieved for the block representation and the computation of the moments. Therefore, even with the software implementation, the method is characterized as real-time.

Table 1.

The required number of operations for the computation of geometrical moments up to the order (L-1,L-1), of one block with MxM pixels.

Operations number	**Direct computation from equation (1)**	**Computation from equation (5)**	**Computation from equations (6), (7)**
L^2M^2	power calculations	$2LM$	$4L$
L^2M^2	multiplications	L^2	$3L^2$-$2L$
L^2M^2	additions	$2LM$	L^2-$2L$

5. CONCLUSIONS

In this paper the block representation idea and the associated algorithm are presented. Block representation is a useful binary image representation that allows the development of more efficient algorithms for various image processing and analysis tasks. Owing to the nature of the digital image, only rectangular areas with the same level are present. Block representation uses these rectangular similarities and offers advantages in image handling and computational cost. Also the block representation provides a perception about image pieces greater than a pixel. Two dimensional moments is a classical image analysis tool and the use of block represented binary images decrease dramatically the computation effort. The complexity of the algorithm for the computation of moments in block represented images, is independent of the image size. The use of the block representation permits the real-time computation of moments. Other image processing and analysis tasks can be implemented in block represented images. At this time this is a research subject.

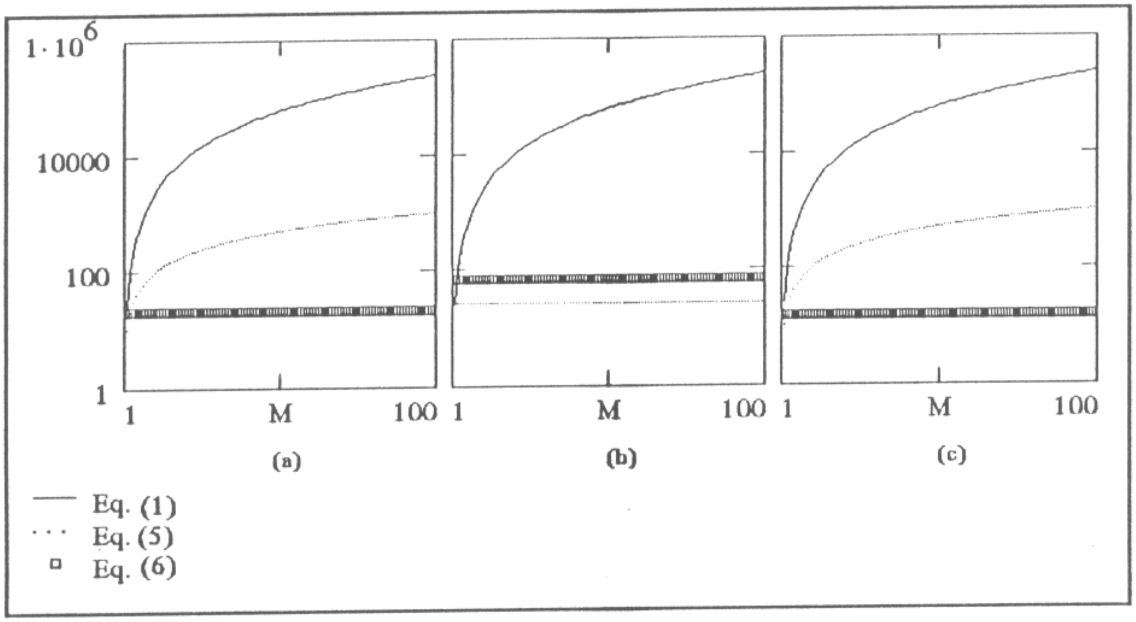

Figure 3. Number of operations for the geometrical moments computation, of a MxM block, from equations (1), (5) and (6) with L=5 and M=1,2, ... , 100. (a) Number of power calculations. (b) Number of multiplications. (c) Number of additions.

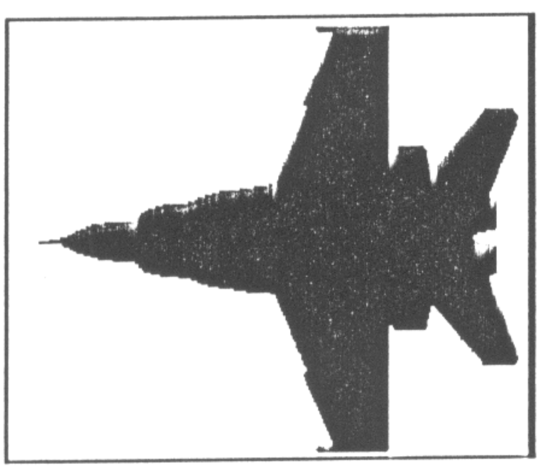

Figure 4. Aircraft image.

Table 2.
Computation of the geometrical moments up to the order (4,4) of the aircraft image, using different methods. The use of block representation, results to a great factor of time reduction.

Computation of the geometrical moments of the aircraft image	**Time in seconds**	**Reduction factor with respect to (1)**
from equation (1)	4.771	
from equation (5)	0.042	114
from equation (6)	0.055	87
using the Lemma 1	0.015	318

REFERENCES

1 M.-K. Hu, "Visual pattern recognition by moment invariants", *IRE Trans. Inform. Theory*, vol. IT-8, pp. 179-187, Feb. 1962.

2 G.L. Cash and M. Hatamian, "Optical character recognition by the method of moments", *Computer Vision, Graphics and Image Processing,* vol. 39, pp. 291-310, 1987.

3 K. Tsirikolias and B.G. Mertzios, "Statistical pattern recognition using efficient 2-D moments with applications to character recognition", *Pattern Recognition.* To appear.

4 M.R. Teague, "Image analysis via the general theory of moments", *J. Opt. Soc. Amer.*, vol. 70, pp. 920-930, Aug. 1980.

5 Y.S. Abu-Mostafa and D. Psaltis, "Image normalization by complex moments", *IEEE Trans. Pattern Anal., Machine Intel.*, vol. PAMI-7, No. 1, pp. 46-55, Jan. 1985.

6 C.-H. Teh and R.T. Chin, "On image analysis by the methods of moments", *IEEE Trans. Pattern Anal., Machine Intel.*, vol. 10, No. 4, pp. 496-513, 1980.

7 A. Papoulis, *Propability, Random Variables, and Stochastic Processes*, McGraw-Hill, New York, 1965.