Fast computation of Hahn moments on gray images using block representation

Iraklis M. Spiliotis,* Nikolaos D. Karampasis, and Yiannis S. Boutalis
Democritus University of Thrace, Department of Electrical and Computer Engineering,
Xanthi, Greece

Abstract. The moments constitute a well-known tool for image analysis and recognition tasks. The family of moments that has the most advantages is the discrete orthogonal moments. A set of these moments is the Hahn moments, as they have a great number of advantages in comparison with other sets of moments. The main disadvantage of moments, including Hahn moments, is the high computational cost, which is increased as higher order moments are involved in the computations, so the real-time analysis is hard to be done. We propose an effective approach for the computation of Hahn moments. The gray image is decomposed in a set of binary images that are named as bitplanes. The most significant bitplanes are represented using image block representation and their moments are computed fast using block techniques. The least significant binary images are substituted by a constant ideal image called "half-intensity" image, which has known Hahn moment values. The proposed method has low computational error, low computational complexity, and under certain conditions is able to achieve real-time processing rates.

Keywords: Hahn moments; image block representation; fast and real-time computation; image analysis; gray images.

1 Introduction

The moments and moments functions are the main tools in image analysis,[1–4] in image matching[5] and retrieval,[6] in image and object recognition,[7,8] and in image watermarking applications,[9] as they can be used as descriptors of features. Also 3-D images have been introduced recently, so the fast computation of moments on them is another research field.[10]

The first moments that were used are based on geometric moments and their variations, which are the central, normalized central, and moment invariants sets.[11] The problem of these moments was the large variations on the dynamic range of values and the numerical instabilities due to the approximation of the integrals in digital images.[11]

For these reasons, the continuous orthogonal functions as basis set of moments are the most suitable to be used, as they allow the representation of the image with no redundancy or information overlap between the moment values. Well-known continuous orthogonal moment sets are Zernike,[12,13] Legendre,[12] Fourier–Mellin.[14]

The discrete orthogonal moments are based on discrete orthogonal polynomials, and they have better image representation ability than the continuous orthogonal moments. The most familiar discrete orthogonal moment sets are the Tchebichef,[15] the Krawtchouk,[16] and the Hahn moments.[17,18] Also the reconstruction of the image from a finite number of moment values (Refs. 15 and 16) is indicative of the discriminative power of the moments; for this reason, moments have been used extensively as features for image description and analysis.

The most significant problem of the moments is that the computation of them requires high computational effort, especially as the moment order increases. The order of the moments determines the type of image information described. The low-order moments represent the coarse features of the image, whereas the high-order moments represent more detailed features and

require a high computational time. As a result, a number of approaches have been proposed for the reduction of this computational effort.

There were many tries to decrease this computational cost. Spiliotis and Mertzios[19] proposed a real-time method of geometric moment computation on binary images, which is based in an innovative binary image representation called image block representation (IBR).[20] Spiliotis and Boutalis[21,22] proposed an extension of the IBR for gray images, which permits the real-time computation of geometric moments in gray images. On the other hand, Papakostas et al.[23] proposed the image slice representation (ISR) for gray images, which is used for the computation of discrete orthogonal moments.

In this paper, the method of Spiliotis and Boutalis is employed for the computation of Hahn moments on grayscale images. In the proposed method, the gray image is decomposed in a set of binary images and the most significant binary images are represented using IBR. The least significant binary images can be substituted by a constant ideal image called "half-intensity" image, which has known Hahn moment values. The proposed method has very low computational complexity with low computational error, as it will be shown.

The remainder of this paper is organized as follows. Section 2 reviews the Hahn moments and their applications; in Sec. 3, the block representation of binary and gray images is presented; Sec. 4 presents the proposed method for the computation of Hahn moments on grayscale images; Sec. 5 presents the ISR method for the computation of Hahn moments in gray images; in Sec. 6, the experimental results are demonstrated and discussed; and finally in Sec. 7, there is the conclusion.

2 Hahn Moments

The 2-D Hahn moment of order pq of an image intensity function $f(x, y)$ with size $N \times M$ is defined as

$$H_{pq} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} h_p^{(\mu,\nu)}(x, N) h_q^{(\mu,\nu)}(y, M) f(x, y), \tag{1}$$

where $p = 0, 1, 2, \ldots, N - 1$, $q = 0, 1, 2, \ldots, M - 1$. The adjustable parameters μ, ν ($\mu > -1$, $\nu > -1$) can control the shape of polynomials. More specifically, if the parameters $\mu, \nu = 0$, then there will be global extraction of the features of the image. In any other case, there will be local extraction of the features of the image, it depends on the values of the parameters μ, ν.[17,18] For example, if $\mu = 25$ and $\nu = 75$, then features will be extracted from the top left of the image. In this paper, the values of the parameters that are used in experimental evaluation are $\mu, \nu = 0$, as the global extraction of the features of the image is wanted. The $h_p^{(\mu,\nu)}(x, N)$ is the p'th order orthogonal Hahn polynomial, defined by the following recursive relation:

$$A h_p^{(\mu,\nu)}(x, N) = B \sqrt{\frac{d_{p-1}^2}{d_p^2}} h_{p-1}^{(\mu,\nu)}(x, N) + C \sqrt{\frac{d_{p-2}^2}{d_p^2}} h_{p-2}^{(\mu,\nu)}(x, N), \quad p = 2, 3, \ldots, N - 1, \tag{2}$$

where

$$h_0^{(\mu,\nu)}(x, N) = \sqrt{\frac{\rho(x)}{d_0^2}} \quad h_1^{(\mu,\nu)}(x, N) = [(N + \nu - 1)(N - 1) - (2N + \mu + \nu - 2)x]\sqrt{\frac{\rho(x)}{d_1^2}}. \tag{3}$$

Also the parameters A, B, and C are defined as

$$A = \frac{p(2N + \mu + \nu - p)}{(2N + \mu + \nu - 2p + 1)(2N + \mu + \nu - 2p)},$$

$$B = x - \frac{2(N - 1) + \nu - \mu}{4} - \frac{(\mu^2 - \nu^2)(2N + \mu + \nu)}{4(2N + \mu + \nu - 2p + 2)(2N + \mu + \nu - 2p)},$$

$$C = \frac{(N - p + 1)(N - p + \mu + 1)(N - p + \nu + 1)(N - p + \mu + \nu + 1)}{(2N + \mu + \nu - 2p + 2)(2N + \mu + \nu - 2p + 1)}. \tag{4}$$

The weighting function $\rho(x)$ can be solved using the recursive relation with respect to x as

$$\rho(x) = \frac{(N-x)(N-\nu-x)}{x(x+\mu)}\rho(x-1), \quad x = 1,2,\ldots,N-1, \tag{5}$$

with $\rho(0) = \frac{1}{\Gamma(\mu+1)\Gamma(N+\nu)\Gamma(N-p)}$.

The square norm d_p^2 has the following expression:

$$d_p^2 = \frac{\Gamma(2N+\mu+\nu-p)}{(2N+\mu+\nu-2p-1)\Gamma(N+\mu+\nu-p)}$$
$$\times \frac{1}{\Gamma(N+\mu-p)\Gamma(N+\nu-p)\Gamma(p+1)\Gamma(N-p)}. \tag{6}$$

It should be mentioned that the method that is used for the computation of Gamma distribution is $\Gamma(x) = \int_0^\infty t^{x-1}e^{-t}dt$, as the parameters μ and ν may not be integers. However, as has been said before, in this paper, the $\mu, \nu = 0$ and all the other parameters that are used in Eqs. (7) and (8) are integers. So the function that is used to calculate the Gamma distribution is $\Gamma(x) = (x-1)!$, which is computed recursively.

The polynomial $h_q^{(\mu,\nu)}(y,M)$ is calculated in the same way as the $h_p^{(\mu,\nu)}(x,N)$. For a square image $N \times N$, the two polynomials have identical values, $h_p^{(\mu,\nu)}(x,N) = h_q^{(\mu,\nu)}(y,N)$, $\forall\ x = y$.

It is worth noting that the values of the polynomials are computed fast without significant complexity, and also they can be precalculated, stored, and used in the computation of moments in Eq. (1).

2.1 Image Reconstruction

An image with size $N \times M$ can be reconstructed from a set of Hahn moments up to the order $P \times Q$, using the following relation:

$$f(x,y) = \sum_{p=0}^{P-1}\sum_{q=0}^{Q-1} Q_{pq}K_p(x;s_1,N)K_q(y;s_2,M), \tag{7}$$

where $x = 0,1,2,\ldots,N-1$, $y = 0,1,2,\ldots,M-1$, and $P \leq N$, $Q \leq M$. If the number of Hahn moments used for reconstruction is equal to the number of image pixels, i.e., $P = N$, $Q = M$, then the reconstructed image is identical to the original image.

2.2 Application of Hahn Moments

The Hahn moments are considered as a generalization of Tchebichef and Krawtchouk moments. The Tchebichef moments are able to capture the global features of an image, whereas the Krawtchouk moments capture local features. So the usage of Hahn moments is very useful since the selection of the local or global descriptors can be chosen.

Since the introduction of Hahn moments from Zhou et al.[18] in 2005 and from Yap et al.[17] in 2007, a significant number of research papers concerning the applications of the Hahn moments and their variations have been published; this fact indicates their ability for image description. The Hahn moments and their variations have been used for face recognition,[24] the description and reconstruction of 3-D images,[25,26] and feature extraction of encrypted images.[27] The radial Hahn moments have been used for 2-D and 3-D image recognition.[28] The Hahn moment invariants have been used for image watermarking[29] and improved image classification.[30,31]

Hahn moments outperform Tchebichef and Krawtchouk moments in image description tasks; this is implied by lower error on images reconstructed from Hahn moments in comparison with the other moment sets as reported in Ref. 18. A similar study[32] demonstrated that Hahn moments have better image reconstruction quality in comparison with Tchebichef, Krawtchouk, Charlier, Meixner, and Racah moments.

3 Block Representation of Binary and Gray Images

3.1 *Block Representation of Binary Images*

In a binary image, the object level pixels have intensity 1, whereas the background pixels have intensity 0. The pixels with object level are represented by a set of nonoverlapping rectangles, whose edges are parallel to the axes, in such a way that every object pixel belongs to only one rectangle. These formed rectangles are called blocks and this representation is called IBR. The following definitions clarify the IBR.[19]

Definition 1 A block is called a rectangular area of the image with edges parallel to the axes of the image, containing pixels with value 1, i.e., object level luminance. ▯

Definition 2 A binary image is block represented if it is represented by a set of nonoverlapping blocks, and each image pixel with value 1 belongs to one and only one block. ▯

The IBR process as described in Algorithm 1 is a fast process without numerical computations and requires one image scan and simple pixel checking operations.

A binary image represented by blocks described as $f(x, y) = \{b_i : i = 0, 1, \ldots, k-1\}$, where k is the number of the blocks and b_i is the i'th block that is described by the coordinates of two opposite diagonal angular points as

$$b_i = (x_{1,b_i}, x_{2,b_i}, y_{1,b_i}, y_{2,b_i}). \tag{8}$$

3.2 *Image Block Representation of Gray Images*

A gray image of dimensions $N \times M$ has an intensity function $g(x, y)$ and 2^n gray levels. This gray image can be decomposed into n binary images. Each of these binary images is a bitplane of the original gray image derived from the bits of the same significance of the values of the corresponding pixel of the gray image. The first binary image is composed of the most significant bits (MSB) of the pixel values of the gray image g and is defined as p_{n-1}, the second most important is defined as p_{n-2}, and the bitplane with the least significant bits (LSB) is defined as p_0. The relation between the gray image $g(x, y)$ and the n binary images is

$$g(x, y) = 2^{n-1} p_{n-1}(x, y) + \ldots + 2^1 p_1(x, y) + 2^0 p_0(x, y). \tag{9}$$

An example of the decomposition of an image is presented in Fig. 1, where the initial image of Fig. 1(a) with 256 gray levels is decomposed to the 8 corresponding binary images of Figs. 1(b)–1(i).

It is easy to notice that the binary images of lower order are noisy. Particularly, the reduction of the computational cost of moment calculation is based on this observation and in the fact that the n binary images resulting from the decomposition of the gray image can be represented by blocks.

Algorithm 1 Image block representation.

Step 1: Consider each line y of the image f and find the object level intervals in line y.

Step 2: Compare intervals of line y with blocks of line $y - 1$.

Step 3: If an interval does not match with any block, this is the beginning of a new block.

Step 4: If a block matches with an interval, the end of the block is in the line y.
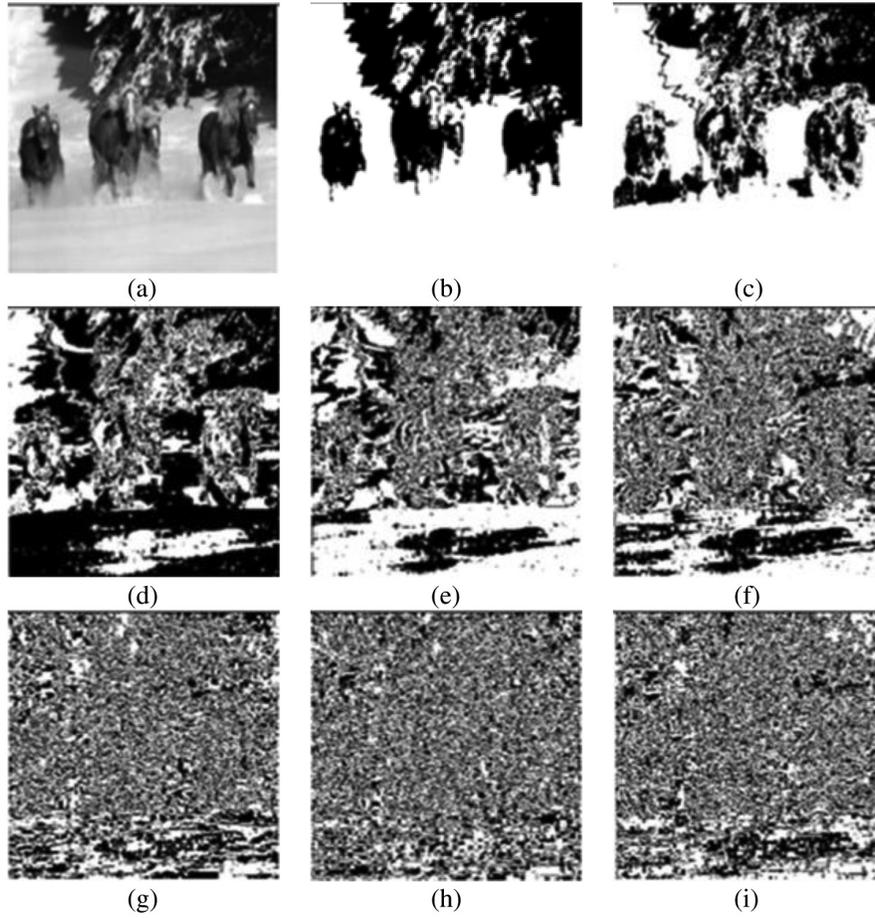
Fig. 1 Decomposition of the gray image Horses with size 1024 × 1024 and 256 gray levels into 8 binary images. (a) The original gray image. (b)–(i) The binary images p_7 at (b) derived from the MSB, and p_0 at (i) derived from the LSB.

4 Proposed Method for Fast Computation of Hahn Moments on Gray Images

4.1 Fast Computation of Hahn Moments on Block Represented Binary Images

Since the background pixels have intensity 0, only the pixels that describing the object will take part in the calculation of the moments. Thus the Hahn moments of a binary image can be defined as

$$H_{pq} = \sum_x \sum_y h_p^{(\mu,\nu)}(x,N) h_q^{(\mu,\nu)}(y,M) \quad \forall\, x,y: f(x,y) = 1. \tag{10}$$

Since all the pixels of image $f(x,y)$ with value 1 belong to the k blocks, Eq. (10) rewritten as

$$H_{pq} = \sum_{i=0}^{k-1} \sum_{x=x_{1,b_i}}^{x_{2,b_i}} \sum_{y=y_{1,b_i}}^{y_{2,b_i}} h_p^{(\mu,\nu)}(x,N) h_q^{(\mu,\nu)}(y,M). \tag{11}$$

Exploiting the rectangular form of the blocks with edges parallel to the image axes, the Hahn moments of a block b are calculated as follows:

$$H_{pq}^b = \sum_{x=x_{1,b}}^{x_{2,b}} \sum_{y=y_{1,b}}^{y_{2,b}} h_p^{(\mu,\nu)}(x, N) h_q^{(\mu,\nu)}(y, M)$$

$$= h_p^{(\mu,\nu)}(x_{1,b}, N) \sum_{y=y_{1,b}}^{y_{2,b}} h_q^{(\mu,\nu)}(y, M) + \cdots + h_p^{(\mu,\nu)}(x_{2,b}, N) \sum_{y=y_{1,b}}^{y_{2,b}} h_q^{(\mu,\nu)}(y, M)$$

$$= \sum_{x=x_{1,b}}^{x_{2,b}} h_p^{(\mu,\nu)}(x, N) \sum_{y=y_{1,b}}^{y_{2,b}} h_q^{(\mu,\nu)}(y, M). \tag{12}$$

Exploiting the rectangular shape of the block, the double sum of Hahn polynomials rewritten as the product of two separate sums, each sum contains the polynomial terms for the horizontal and vertical axis of the block, respectively. Using Eqs. (11) and (12) rewritten as

$$H_{pq} = \sum_{i=0}^{k-1} \sum_{x_k=x_{1,b_i}}^{x_{2,b_i}} h_p^{(\mu,\nu)}(x, N) \sum_{y_k=y_{1,b_i}}^{y_{2,b_i}} h_q^{(\mu,\nu)}(y, M). \tag{13}$$

4.2 *Fast Computation of Hahn Moments on Block Represented Gray Images*

Suppose that the pixel values of a gray image are in range $[0, 2^n - 1]$ and the gray image consisted of n binary images. Substituting Eq. (9) into Eq. (1), the following relation that connects the calculation of the Hahn moments of the gray image and the n binary images is obtained by

$$H_{pq} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} h_p^{(\mu,\nu)}(x, N) h_q^{(\mu,\nu)}(y, M) g(x, y)$$

$$= \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} h_p^{(\mu,\nu)}(x, N) h_q^{(\mu,\nu)}(y, M) [2^{n-1} p_{n-1}(x, y) + \cdots + 2^1 p_1(x, y) + 2^0 p_0(x, y)]$$

$$= \left[2^{n-1} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} h_p^{(\mu,\nu)}(x, N) h_q^{(\mu,\nu)}(y, M) p_{n-1}(x, y) + \cdots \right.$$

$$\left. + 2^0 \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} h_p^{(\mu,\nu)}(x, N) h_q^{(\mu,\nu)}(y, M) p_0(x, y) \right]$$

$$= (2^{n-1} H p_{(n-1)pq} + \cdots + 2^1 H p_{1pq} + 2^0 H p_{0pq})$$

$$= \sum_{i=0}^{n-1} 2^i H p_{ipq}, \tag{14}$$

where $p_{n-1}(x, y), \ldots, p_1(x, y), p_0(x, y)$ are the binary images that compose the gray image $g(x, y)$ and $H p_{(n-1)pq}, \ldots, H p_{1pq}, H p_{0pq}$ are the Hahn moments of these binary images, which are calculated using IBR and Eq. (13).

It is easy to notice from Eq. (14) that due to the weight factors 2^i, the moments of the bitplanes p^i do not contribute equally to the gray image. Thus the less important binary images contribute less to the moments of the gray image. Moreover, the lower order bitplanes look noisy with continuous black-and-white transitions and are similar to a chessboard image or simply with a half-intensity image h with constant intensity 1/2. This can be evaluated by the next Lemma.

Lemma 1. The moment values of an image with intensity 1/2 are the half of the moment values of an image with intensity 1.

Proof. Considering the "full-intensity" image $f = 1 \ \forall \ x, y$ as one block and exploiting its rectangular form, then according to Eq. (12), the Hahn moment values of the full-intensity image are $Hf_{pq} = \sum_{\forall x} h_p^{(\mu,\nu)}(x, N) \cdot \sum_{\forall y} h_q^{(\mu,\nu)}(y, M)$. Correspondingly, the moment values of the half-intensity image $h = 1/2, \ \forall \ x, y$ are

$$Hh_{pq} = \frac{1}{2} \sum_{\forall x} h_p^{(\mu,\nu)}(x,N) \cdot \sum_{\forall y} h_q^{(\mu,\nu)}(y,M) = \frac{Hf_{pq}}{2}. \qquad \square$$

The approximated Hahn moments $H_{m,pq}$, by replacing the m least significant biplanes with the half-intensity image $h(x,y)$ is

$$H_{m,pq} = \sum_{i=m}^{n-1} 2^i Hp_{ipq} + \sum_{j=0}^{m-1} 2^j Hh_{pq} = \sum_{i=m}^{n-1} 2^i Hp_{ipq} + Hh_{pq} \sum_{j=0}^{m-1} 2^j. \qquad (15)$$

In Eq. (15), the Hahn moments Hp_{ipq} of the $(n-m)$ higher order bitplanes are computed fast using IBR and Eq. (12). The Hahn moments Hh_{pq} of a half-intensity plane are computed fast using Lemma 1. Additional computational time gains can be obtained if the Hahn moments Hh_{pq} are precalculated, stored, and used in Eq. (15).

4.3 Representation Performance Evaluation on Gray Images

As expected, there is an approximation error from the replacement of the lower order bitplanes with the half-intensity image. The moments of different orders do not have the same significance as features for image description, therefore, the direct measure of the error between the exact $H_{E,pq}$ and the approximated moment values $H_{m,pq}$, using m half-intensity images, is not an appropriate criterion for the evaluation of the proposed method. Since the moments are image descriptors, a more suitable consideration of the moment error is the indirect measure of the differences among the images described by the moments. Thus for the input gray image $g(x,y)$, the exact moment values $H_{E,pq}$ are computed from order 0×0 up to the order $P \times Q$ and are used to reconstruct the image $\hat{g}_{E,PQ}(x,y)$; whereas for the input image $g_m = p_7 \ldots p_m h_{m-1} \ldots h_0$ composed from $(8-m)$ high-order bitplanes and m half-intensity planes the approximated moment values $H_{m,PQ}$ are computed up to the order $P \times Q$ and are used to reconstruct the image $\hat{g}_{m,PQ}(x,y)$.

In order to evaluate the quality of the approximated reconstructed image $\hat{g}_{m,PQ}$ in comparison with the exact reconstructed image $\hat{g}_{E,PQ}$, a number of metrics was utilized is this research. Initially, an intuitive indication is given from the observation of the two images by the human visual system. A well-known image quality metric is the normalized image reconstruction error (NIRE),[33,34] which is the normalized square error between the exact f and the approximated image g function and defined as

$$\text{NIRE}(f,g) = \frac{\sum_x \sum_y [f(x,y) - g(x,y)]^2}{\sum_x \sum_y f^2(x,y)}. \qquad (16)$$

The structural similarity index (SSIM)[35] is an image quality metric that is related to known characteristics of the human visual system and defined as

$$\text{SSIM}(f,g) = \frac{(2\mu_f \mu_g + C_1)(2\sigma_{fg} + C2)}{(\mu_f^2 + \mu_g^2 + C_1)(\sigma_f^2 + \sigma_g^2 + C2)}, \qquad (17)$$

where f and g are the original and the distorted image; μ_f, μ_g, σ_f, σ_g, σ_{fg} are the local means, standard deviations, and cross-covariance for images f, g; and the constants $C_1 = (0.01L)^2$, $C_2 = (0.03L)^2$, where L is the dynamic range of the pixel values ($L = 255$ for 8-bit gray images). Experimental results from these metrics are demonstrated in Sec. 6.

5 ISR Method for Computation of Hahn Moments on Gray Images

The well-known ISR method for fast computation of Hahn moments is presented here for comparison with our method. Papakostas et al.[23] proposed the ISR method, for the fast computation of Hahn moments on the grayscale images. The main idea behind the ISR is that a grayscale
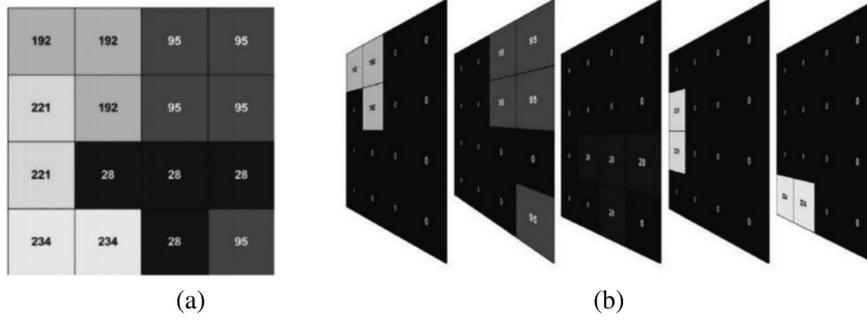
Fig. 2 (a) Grayscale image with intensity [0, 255] and (b) the ISR decomposition in 256 binary intensity slices.

image consists of pixels with different intensities with values in the range of [0, 255], as it is shown in Fig. 2(a). The method aims at the decomposition of the gray image into L binary slices as well as the maximum intensity value of the image's pixels, as it is described in Fig. 2(b). For example, if a grayscale image has 256 brightness levels, the number of slices that will be extracted is 256. Each slice is a binary image that has only the values 0 or $f_i \in [1,255]$.

As the grayscale image has been decomposed into L binary slices, the IBR method can be used on each of them. The Hahn moments of a grayscale image $f(x, y)$ are computed by

$$H_{\text{ISR},pq} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} h_p^{(\mu,\nu)}(x, N)h_q^{(\mu,\nu)}(y, M) \sum_{i=1}^{L} f_i(x, y)$$

$$= \sum_{i=1}^{L}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} h_p^{(\mu,\nu)}(x, N)h_q^{(\mu,\nu)}(y, M)f_i(x, y)$$

$$= \sum_{i=1}^{L} f_i H_{\text{ISR},pq}(i), \tag{18}$$

where $H_{\text{ISR},pq}(i)$ is the $(p + q)$'th order Hahn moment of the i'th binary slice and can be computed with the IBR.

Where $f_i(x, y)$ is

$$f(x, y) = \{f_i(x, y), i = 1,2,\ldots, L\}, \quad f_i(x, y) = \{b_{ij}, j = 0,1,\ldots, K_i - 1\}, \tag{19}$$

where b_{ij} is the j'th block of slice i and K_i is the number of image blocks having intensity f_i. Each block is described by the coordinates of the upper left and down right corner in vertical and horizontal axes.

Nevertheless, this method's computational time of Hahn moments is similar with the 2-D method, as will be shown in the next section.

6 Experimental Results

For the experimental evaluation, a Tyan computer with total 8 AMD Opteron cores at 2.2 GHz and 16 GB of memory was used. The operating system was Linux, all the programs implemented in C programming language, compiled with gcc for serial execution using one CPU core.

In this section, the experimental results for the computation of Hahn moments for binary and gray images using the direct method, the ISR method and the proposed IBR method are demonstrated. In the conducted experiments, the values of the Hahn polynomials $h_p^{(\mu,\nu)}(x, N)$ $h_q^{(\mu,\nu)}(y, M)$ are precalculated, stored, and used in the computations. The computational complexity for the computation of the Hahn polynomials is presented in Table 1. It is concluded that the required execution time is negligible in comparison with the required execution time for the computation of Hahn moments.

Table 1 The required execution time in seconds, for the computation of Hahn polynomial for size N from 512 to 4096 of signal samples and for different polynomial orders from $N/8$ to N.

	Polynomial order			
Size N	$N/8$	$N/4$	$N/2$	N
$N = 512$	0.00	0.01	0.01	0.03
$N = 1024$	0.01	0.02	0.06	0.11
$N = 2048$	0.02	0.06	0.18	0.39
$N = 4096$	0.20	0.25	0.47	0.76

(a) (b) (c)

Fig. 3 A sample of test binary images: (a) shapes, (b) text page, and (c) chessboard with 10×10 pixel squares.

6.1 Time Complexities for Binary Images

For the case of binary images, the Hahn moment values computed using the proposed method are identical with the moment of the direct method. In Fig. 3, some test binary images that have been used in the experiments in different sizes from $(1K \times 1K)$ to $(30,000 \times 30,000)$ pixels are demonstrated.

In Table 2, the execution times and the speedup values for the computation of Hahn moment from order 0×0 up to the different maximum orders, using the direct method of Eq. (1) and the proposed method of Eq. (13) are presented. The execution time of the proposed method includes the time for the IBR and the time for moments computation. In Fig. 4, the speedup values achieved using the proposed method are demonstrated. The execution time of the proposed method depends on the number k of the blocks of the image and the maximum moment order. It is observed that the speedup values are very significant.

6.2 Gray Image Representation Quality

In this section, both subjective (optical) and objective evaluations of the proposed representation quality are provided. A number of test gray images are presented in Fig. 5. As discussed earlier for the evaluation of the representation quality, the images $\hat{g}_E$ and $\hat{g}_m$ are used, where $\hat{g}_E$ reconstructed from the exact Hahn moment values and $\hat{g}_m$ reconstructed from Hahn moment values computed with the proposed method with m half-intensity planes. The experimental results presented in this section are for moment values used up to the order $N \times M$, where N, M are the image size; therefore, the image $\hat{g}_E$ is exactly the same with the original image and the image $\hat{g}_m$ is exactly the same with the image of $8 - m$ higher bitplanes and m half-intensity planes.

The representation of images from $(8 - m)$ high-order bitplanes and m half-intensity planes are demonstrated in Figs. 6–8. Figure 6(a) shows the input gray image g, whereas in Figs. 6(b)–6(h), the images $\hat{g}_m$ represented from the $(8 - m)$ most significant image bitplanes and m half-intensity planes of the input image are demonstrated. The image $\hat{g}_0$ is identical to g and is not demonstrated. The images $\hat{g}_4$, $\hat{g}_5$ are the result of the substitution of the 4 or 5 least significant

Table 2 Time complexities in seconds, for the computation of the Hahn moments of order $P \times Q$ for the test binary images using the direct and the proposed method with IBR and the achieved speedup.

Image size	1K × 1K					
Order $P \times Q$	64 × 64			1024 × 1024		
Time (s)	Direct	IBR	Speedup	Direct	IBR	Speedup
Shapes	92.49	0.45	207.10	23589.30	38.90	606.39
Text page	92.40	1.75	52.84	23556.95	392.51	60.02
Chessboard	92.32	0.61	151.40	23580.45	113.16	208.38

Image size	4K × 4K					
Order $P \times Q$	64 × 64			128 × 128		
Time (s)	Direct	IBR	Speedup	Direct	IBR	Speedup
Shapes	1478.83	3.11	474.96	5924.44	7.53	786.47
Text page	1477.17	6.49	227.71	5907.91	23.48	251.67
Chessboard	1477.54	9.63	153.40	5911.06	34.45	171.57

Image size	16K × 16K					
Order $P \times Q$	32 × 32			64 × 64		
Time (s)	Direct	IBR	Speedup	Direct	IBR	Speedup
Shapes	5918.63	19.28	306.92	23679.50	34.91	678.36
Text page	5917.67	7.57	781.62	23681.73	13.94	1698.35
Chessboard	5930.49	59.23	100.13	23717.81	171.75	138.10

Image size	30,000 × 30,000					
Order $P \times Q$	16 × 16			32 × 32		
Time (s)	Direct	IBR	Speedup	Direct	IBR	Speedup
Shapes	4972.63	34.40	144.56	19930.08	57.66	345.63
Text page	4968.09	17.29	287.42	19887.24	23.48	847.02
Chessboard	4981.47	68.41	72.82	19902.00	174.68	113.93
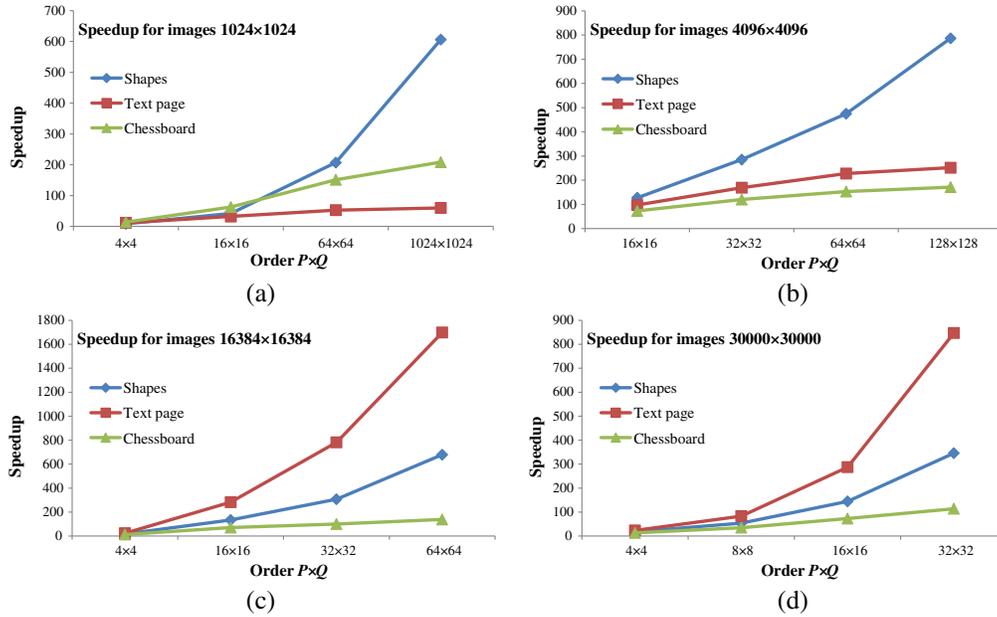
Fig. 4 Speedup values achieved using the proposed method for the Hahn moment computation from order 0×0 up to order $P \times Q$, for the test binary images of Fig. 3 for sizes (a) $1K \times 1K$, (b) $4K \times 4K$, (c) $16K \times 16K$, and (d) $30,000 \times 30,000$ pixels.
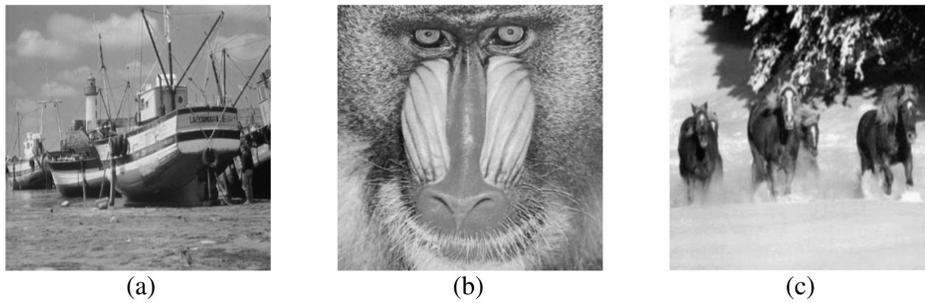
Fig. 5 A sample of test gray images used in experiments: (a) boat with size 256×256, (b) baboon with size 512×512, and (c) horses with size 1024×1024.

bitplanes with half-intensity images; the human visual system cannot distinguish particular differences between these images $\hat{g}_4$, $\hat{g}_5$ and the original input image g, and this is a strong qualitative indication for the acceptance of the proposed method. Also it is expected that an identification system that uses the moments as features will classify patterns from the two images in the same class. Thus it is proposed to preserve only the first three or four bitplanes, whereas the rest of the others can be replaced with half-intensity images.

Last but not least, Table 3 presents the quality measures of the representation between the images $\hat{g}_E$ and $\hat{g}_m$, using the NIRE and SSIM metrics that discussed earlier and described by Eqs. (16) and (17). The NIRE is an error metric, thus smaller values indicate better quality, whereas the SSIM is a similarity metric and values closer to 1 indicate greater similarity. For $m = 4$ or $m = 5$, the NIRE values are small and the SSIM values are high, and this substantiates the validity of the proposed method. Similar results have been obtained for all test gray images used in experiments.

6.3 *Object Classification Experiment*

In order to validate the ability of the proposed method for image classification, an experiment has been conducted. The Columbia University Image Library (COIL-20)[36] has been used in the experiment. The COIL-20 consisted of gray images of 20 objects, where each object is captured
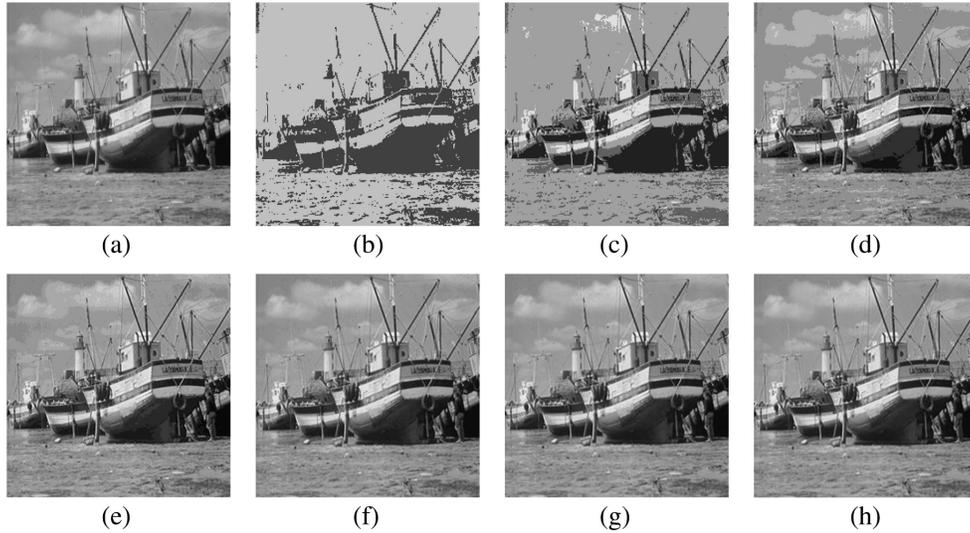
Fig. 6 (a) Original test image boat with size 256×256 and (b)–(h) the reconstructed images from the $(8 - m)$ most significant image bitplanes and m half-intensity planes, where (b) $m = 7$, (c) $m = 6$, etc. (h) $m = 1$. The case of $m = 0$ results in reconstructed image identical to (a).
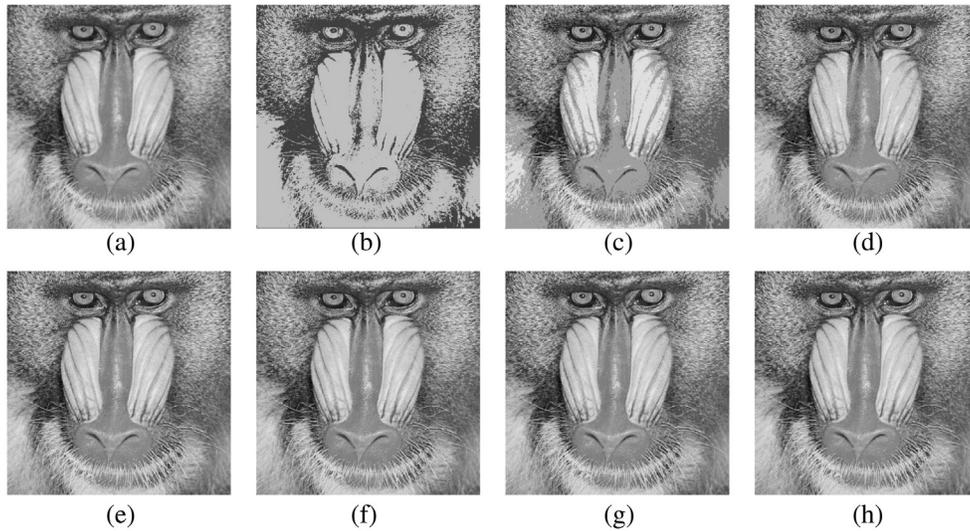
Fig. 7 (a) Original test image baboon with size 512×512 and (b)–(h) the reconstructed images from the $(8 - m)$ most significant image bitplanes and m half-intensity planes, where (b) $m = 7$, (c) $m = 6$, etc. (h) $m = 1$. The case of $m = 0$ results in reconstructed image identical to (a).

from 72 different directions and the total number is 1440 images with size 128×128 pixels. In Fig. 9, the collection of the 20 objects of the COIL-20 is demonstrated.

In pattern recognition applications, a small number of features are used to discriminate among the patterns. The Hahn moments from order $(0, 0)$ up to the order $(3, 3)$ comprise a 16-dimensional feature vector that describes each object, where the vector V_k represents the k'th class, and the vector V_i represents the input object. A Euclidean distance-based classifier used and the input object is classified to the class with the minimum distance in the feature space. The Euclidean distance[37] between the class k and the input object is defined as

$$d = \sqrt{\sum_{j=0}^{15} (v_{kj} - v_{ij})^2}.$$ (20)
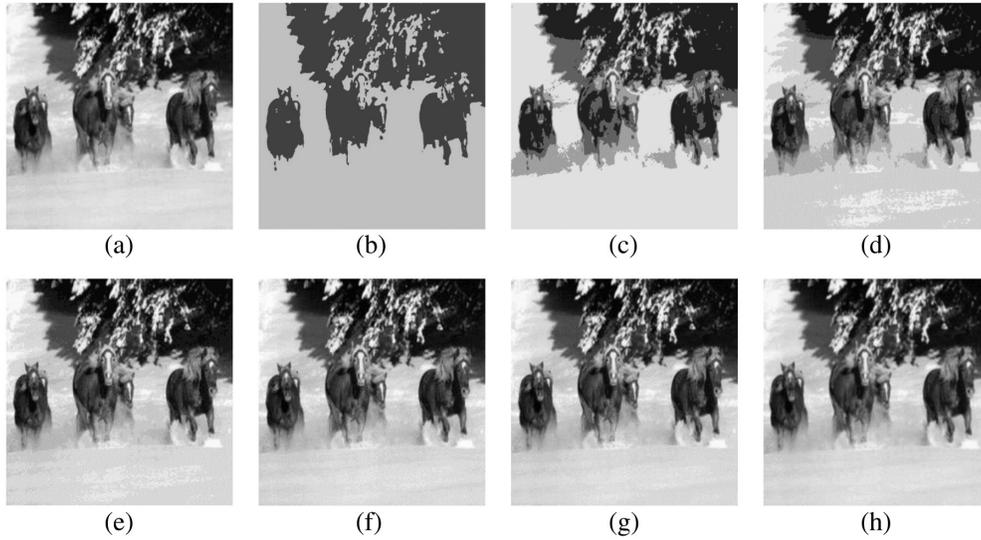
Fig. 8 (a) Original test image Horses with size 1024×1024 and (b)–(h) the reconstructed images from the $(8 - m)$ most significant image bitplanes and m half-intensity planes, where (b) $m = 7$, (c) $m = 6$, etc. (h) $m = 1$. The case of $m = 0$ results in reconstructed image identical to (a).

Table 3 The NIRE and SSIM metrics between the original test image and the corresponding reconstructed images from $(8 - m)$ most significant image bitplanes and m half-intensity planes.

Image	Boat 256×256		Baboon 512×512		Horses 1024×1024	
Metric	NIRE	SSIM	NIRE	SSIM	NIRE	SSIM
$m = 7$	8.45×10^{-02}	0.572	5.71×10^{-02}	0.538	3.63×10^{-02}	0.711
$m = 6$	1.45×10^{-02}	0.740	1.68×10^{-02}	0.708	9.85×10^{-03}	0.767
$m = 5$	4.26×10^{-03}	0.851	3.84×10^{-03}	0.871	3.06×10^{-03}	0.812
$m = 4$	1.13×10^{-03}	0.935	9.57×10^{-04}	0.958	7.00×10^{-04}	0.897
$m = 3$	2.89×10^{-04}	0.979	2.44×10^{-04}	0.989	1.70×10^{-04}	0.963
$m = 2$	7.88×10^{-05}	0.994	6.65×10^{-05}	0.997	5.39×10^{-05}	0.988
$m = 1$	2.62×10^{-05}	0.999	2.23×10^{-05}	0.999	1.65×10^{-05}	0.998

The moments of the input objects are computed using the direct method (M1), the proposed method for four higher bitplanes and $m = 4$ half-intensity bitplanes (M2), and the proposed method for three higher bitplanes and $m = 5$ half-intensity bitplanes (M3). The input objects are presented to the classification system in their original form, with 2% salt and pepper noise and with 4% salt and pepper noise.

The recognition rate R is defined as

$$R = \frac{\text{hits}}{\text{total}}, \tag{21}$$

where hits is the number of correctly classified input objects and total is the number of input objects used in the experiment.

Table 4 demonstrates the experimental results, it is noted that the recognition rates of all Hahn moment computing methods M1, M2, and M3 decrease with increasing noise. Also it is observed that for noise-free input objects the recognition rate of the proposed method M2 is 1% less than the recognition rate of direct method M1, whereas recognition rate of method M3 is 3% less than R of M1. Similar recognition rates achieved for input objects with 2% salt and

Fig. 9 The collection of the COIL-20 objects.

Table 4 The recognition rates R of the classification experiment, using noise free, noise 2%, and noise 4% input objects. The Hahn moments computed using the direct method (M1), the proposed method for $m = 4$ (M2), and the proposed method for $m = 5$ (M3).

Input	M1 (%)	M2 (%)	M3 (%)
Noise free	100	99.16	97.87
Noise 2%	90.51	89.68	87.42
Noise 4%	78.31	77.24	75.19

pepper noise and with 4% salt and pepper noise. Therefore, the recognition rates of proposed methods M2 and M3 are close to the recognition rate of the exact method M3. The results show the efficiency of our proposed method for pattern recognition applications.

6.4 *Time Complexities for Grayscale Images*

Table 5 presents the number of blocks at each bitplane of some grayscale test images that extracted with IBR and ISR. It is concluded that the number of blocks increases as the significance of the bitplane decreases in IBR; thus the substitution of 4 or 5 lower order bitplanes by half-intensity planes dramatically decreases the total number of blocks. On the other hand, in ISR the total number of blocks that extracted is comparable with the number of pixels. Moreover, in ISR the number L of binary images is 256, whereas in the proposed method the number of bitplanes used is 3 or 4.

Table 5 presents the computational time of Hahn moments for a number of test images with different sizes, using four different methods: the direct method using Eq. (1); the ISR method using Eq. (18); the proposed IBR method for $m = 4$ half-intensity planes and four higher order real bitplanes; and the proposed IBR method for $m = 5$ half-intensity planes and three higher order real bitplanes.

Table 5 The number of blocks at each bitplane for some of the test images. p_7 is defined the most important bitplane and p_0 the less important bitplane.

		IBR								ISR	
Image	Size	p_7	p_6	p_5	p_4	p_3	p_2	p_1	p_0	Total number of blocks	Total number of pixels
Boat	256×256	2517	3655	6614	9927	12,380	14,262	15,058	15,062	57,291	65,536
Baboon	512×512	17,241	31,882	44,577	55,208	59,715	60,079	60,049	59,619	247,619	262,144
Horses	1024×1024	3896	11,814	27,430	52,806	86,740	12,2731	16,5026	18,4534	570,666	1,048,576

In Table 6 the execution times for the computation of Hahn moment, using the direct method, the ISR method, the IBR method for $m = 4$ and the IBR method for $m = 5$ are presented. Also the achieved speedup in relation to the direct method is demonstrated in Table 6. The execution time of the proposed IBR method and the ISR method includes also the time for the block representation of the binary image. It is observed that the proposed method achieves better performance than the direct and the ISR methods, due to the substitution of lower order bitplanes with half-intensity bitplanes. Also from Table 6, it is concluded that the proposed method in the occasion of pattern recognition applications, where the images that are used have small dimensions and a small number of moments are required, operates fast in rates that are near real time. The real time is defined by video rate at 24 frames/s.

Table 6 Time complexities in seconds and the corresponding speedup values, for the computation of Hahn moments of grayscale images for different orders, using the direct method, the ISR method, the proposed IBR method for $m = 4$, and the proposed IBR method for $m = 5$. The proposed method used for $(8 - m)$ most significant bitplanes and m half-intensity images, $m = 4$ and $m = 5$. The time for IBR of the bitplanes included in the execution times of the proposed method.

Order of moments	Direct method Time	ISR method Time	Speedup	IBR method for $m = 4$ Time	Speedup	IBR method for $m = 5$ Time	Speedup
		Image boat with size 256×256					
16×16	0.72	0.59	1.22	0.33	2.18	0.20	3.68
64×64	6.61	5.57	1.19	3.67	1.80	2.27	2.91
128×128	23.21	21.90	1.06	9.81	2.37	6.47	3.58
256×256	93.01	77.99	1.19	30.05	3.10	17.19	5.41
		Image baboon with size 512×512					
16×16	1.44	2.90	0.50	1.07	1.35	0.69	2.09
64×64	23.10	29.26	0.79	13.35	1.73	8.47	2.72
256×256	369.68	317.53	1.16	201.35	1.84	126.24	2.92
512×512	1470.93	1335.40	1.10	790.13	1.86	498.41	2.95
		Image horses with size 1024×1024					
16×16	5.82	8.30	0.70	1.21	4.83	0.68	8.60
64×64	92.32	65.47	1.41	9.73	9.49	4.63	19.94
256×256	1475.75	793.73	1.86	135.19	10.92	61.32	24.07
512×512	5909.74	3102.82	1.90	522.18	11.32	234.62	25.19
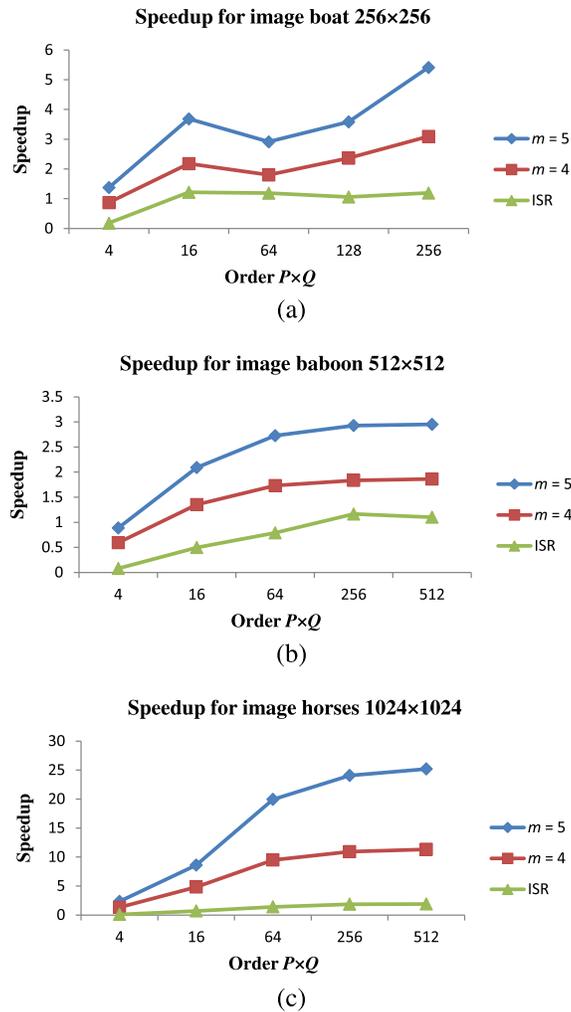
Fig. 10 The speedup value achieved for the computation of Hahn moments from order 4×4 up to order $P \times Q$, using the proposed method for a number of gray test images of different sizes and for using $(8 - m)$ bitplanes and m half-intensity images, $(m = 5)$ and $(m = 4)$ and ISR method.

Figure 10 demonstrates the achieved speedup values for the computation of Hahn moments for the ISR, IBR for $m = 4$ and IBR for $m = 5$ methods in relation to the direct method, for different gray images.

To conclude, it is qualitatively acceptable to use the images g_5, g_4 with three and four real bitplanes and five and four half-intensity images, respectively, since for these images it is observed that our proposed method is superior. This statement can be supported by the experimental results for the quality of the representation and the classification recognition rates discussed in this section.

7 Conclusion

In this paper, a fast computation method of Hahn moments in grayscale images is presented. The proposed method is based on the decomposition of the input image to the corresponding bitplanes and the representation of binary images with blocks. The lower order bitplanes can be substituted by a half-intensity image with moment values equal to the half of full-intensity image. The computation of the Hahn moments on higher order bitplanes is performed fast using the blocks. The proposed method characterized by significant acceleration and low approximation

error. The magnitude of acceleration depends mainly on the image size, and a secondary parameter is the content of the image and the number of blocks.

The representation of an image with blocks allows concurrent machine perception and processing of image areas greater than a pixel. This intrinsic parallelism of the IBR permits the acceleration of sequential algorithms on serial computers. Moreover, the implementation of the IBR and the related image analysis algorithms on various parallel computing platforms is expected to achieve additional speedup values.

A popular parallel computer is the shared memory parallel machine, which is a multicore, shared memory computer that usually uses the OpenMP (Open Multi Processing) parallel programming API.[38] Recently, a parallel IBR algorithm using OpenMP has been published by our research team.[39]

The parallel implementation of the IBR algorithm and the computation of moments, on distributed memory parallel machines using MPI, GPGPUs, and FPGAs is another quite interesting direction for our future research efforts.

References

1. M. K. Hu, "Visual pattern recognition by moment invariants," *IRE Trans. Inf. Theory* **8**, 179–187 (1962).
2. M. R. Teague, "Image analysis via the general theory of moments," *J. Opt. Soc. Am.* **70**, 920–930 (1980).
3. C.-H. The and R. T. Chin, "On image analysis by the method of moments," *IEEE Trans. Pattern Anal. Mach. Intell.* **10**, 496–513 (1988).
4. J. Flusser and T. Suk, "Rotation moment invariants for recognition of symmetric objects," *IEEE Trans. Image Process.* **15**, 3784–3790 (2006).
5. D. Bhattacharya and S. Sinha, "Invariance of stereo images via the theory of complex moments," *Pattern Recognit.* **30**, 1373–1386 (1997).
6. K. K. T. Chueng and H. S. Ip, "Image retrieval in digital library based on symmetry detection," in *Proc. Comput. Graphics Int.*, Germany (1998).
7. J. S. Wu et al., "Moments net: a simple learning-free method for binary image recognition," in *IEEE Int. Conf. Image Process.* (2017).
8. Z. H. Shao et al., "Quaternion Bessel–Fourier moments and their invariant descriptors for object reconstruction and recognition," *Pattern Recognit.* **47**, 603–611 (2014).
9. Y. Xin, S. Liao, and M. Pawlak, "Circularly orthogonal moments for geometrically robust image watermarking," *Pattern Recognit.* **40**, 3740–3752 (2007).
10. H. Karmouni et al., "Fast 3D image reconstruction by cuboids and 3D Charlier's moments," *J. Real-Time Image Process.* (2019).
11. W. G. Lin and S. S. Wang, "A note on the calculation of moments," *Pattern Recognit. Lett.* **15**, 1065–1070 (1994).
12. A. Khotanzad and Y. H. Hong, "Invariant image recognition by Zernike moments," *IEEE Trans. Pattern Anal. Mach. Intell.* **12**, 489–497 (1990).
13. K. M. Hosny, "Fast computation of accurate Zernike moments," *J. Real-Time Image Process.* **3**, 97–107 (2008).
14. C. Kan and M. D. Srinath, "Invariant character recognition with Zernike and orthogonal Fourier–Mellin moments," *Pattern Recognit.* **35**, 143–154 (2002).
15. R. Mukundan, "Image analysis by Tchebichef moments," *IEEE Trans. Image Process.* **10**, 1357–1364 (2001).
16. P. T. Yap et al., "Image analysis by Krawtchouk moments," *IEEE Trans. Image Process.* **12**, 1367–1377 (2003).
17. P. T. Yap et al., "Image analysis using Hahn moments," *IEEE Trans. Pattern Anal. Mach. Intell.* **29**, 2057–2062 (2007).
18. J. Zhou et al., "Image analysis by discrete orthogonal Hahn moments," *Lect. Notes Comput. Sci.* **3656**, 524–531 (2005).
19. I. M. Spiliotis and B. G. Mertzios, "Real-time computation of two-dimensional moments on binary images using image block representation," *IEEE Trans. Image Process.* **7**, 1609–1615 (1998).

20. I. M. Spiliotis and B. G. Mertzios, "Fast algorithms for basic processing and analysis operations on block represented binary images," *Pattern Recognit. Lett.* **17**, 1437–1450 (1996).
21. I. M. Spiliotis and Y. S. Boutalis, "Parameterized real-time moment computation on gray images using block techniques," *J. Real-Time Image Process.* **6**(2), 81–91 (2011).
22. I. M. Spiliotis and Y. Boutalis, "Fast and real-time moment computation methods of gray images using image block representation," in *Proc. 5th IASTED Int. Conf. Signal Process., Pattern Recognit. and Appl.*, Innsbruck, Austria, pp. 323–328 (2008).
23. G. A. Papakostas et al., "A unified methodology for the efficient computation of discrete orthogonal image moments," *Inf. Sci.* **179**, 3619–3633 (2009).
24. F. Akhmedova and S. Liao, "Face recognition using discrete orthogonal Hahn moments," *Int. J. Comput. Electr. Autom. Control Inf. Eng.* **9**, 1550–1556 (2015).
25. A. Mesbah et al., "Robust reconstruction and generalized dual Hahn moments invariants extraction for 3D images," *3D Res.* **8**, 7 (2017).
26. A. Mesbah et al., "Fast and efficient computation of three-dimensional Hahn moments," *J. Electronic Imaging* **25**(6), 061621 (2016).
27. T. Yang et al., "Image feature extraction in encrypted domain with privacy-preserving Hahn moments," *IEEE Access* **6**, 47521–47534 (2018).
28. M. El Mallahi et al., "Radial Hahn moment invariants for 2D and 3D image recognition," *Int. J. Autom. Comput.* **15**(3), 277–289 (2018).
29. S. Ahmad and Z.-M. Lu, "Geometric distortions-invariant digital watermarking using scale-invariant feature transform and discrete orthogonal image moments," in *Digital Rights Management: Concepts, Methodologies, Tools, and Applications*, S. Ahmad, Ed., pp. 237–289, IGI Global (2013).
30. R. Benouini et al., "Efficient image classification by using improved dual Hahn moment invariants," in *Int. Conf. Intell. Syst. and Comput. Vision* (2018).
31. M. Sayyouri et al., "Improving the performance of image classification by Hahn moment invariants," *J. Opt. Soc. Am. A* **30**(11), 2381–2394 (2013).
32. J. S. Rivera-López and C. J. Camacho-Bello, "Color image reconstruction by discrete orthogonal moment," *J. Data Anal. Inf. Process.* **5**, 156–166 (2017).
33. C. Camacho-Bello et al., "Reconstruction of color biomedical images by means of quaternion generic Jacobi–Fourier moments in the framework of polar pixels," *J. Med. Imaging* **3**(1), 014004 (2016).
34. K. M. Hosny and M. M. Darwish, "Feature extraction of color images using quaternion moments," in *Recent Advances in Computer Vision: Theories and Applications*, M. Hassaballah and K. Hosny, Eds., pp. 141–167, Springer, Cham (2019).
35. Z. Wang et al., "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.* **13**, 600–612 (2004).
36. S. A. Nene, S. K. Nayar, and H. Murase, "Columbia University Image Library (COIL-20)," http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php (1996).
37. R. Mukundan and K. R. Ramakrishnan, *Moment Functions in Image Analysis*, World Scientific Publisher, Singapore (1998).
38. R. Chandra et al., *Parallel Programming in OpenMP*, Academic Press, San Diego (2001).
39. I. M. Spiliotis, M. P. Bekakos, and Y. S. Boutalis, "Parallel implementation of the image block representation using OpenMP," *J. Parallel Distrib. Comput.* **137**, 134–147 (2020).