Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition 40 (2007) 3597–3605
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/pr

Exact Legendre moment computation for gray level images

Khalid M. Hosny*

Department of Computer Science, Faculty of Computers and Informatics, Zagazig University, Zagazig, Egypt

Received 15 March 2006; received in revised form 14 April 2007; accepted 19 April 2007

Abstract

A novel method is proposed for exact Legendre moment computation for gray level images. A recurrence formula is used to compute exact values of moments by mathematically integrating the Legendre polynomials over digital image pixels. This method removes the numerical approximation errors involved in conventional methods. A fast algorithm is proposed to accelerate the moment's computations. A comparison with other conventional methods is performed. The obtained results explain the superiority of the proposed method.

Keywords: Legendre moments; Fast algorithm; Gray level images

1. Introduction

Since Hu introduces the moment invariants [1], moments and moment functions have been widely used in the field of image processing. Teague [2] introduces the set of orthogonal moments (e.g. Legendre moment and Zernike moment), where orthogonal moments can be used to represent an image with the minimum amount of information redundancy [3]. Legendre moments are used in many applications such as pattern recognition [4], face recognition [5], and line fitting [6]. It is well known that, the difficulty in the use of Legendre moments is due to their high computational complexity, especially when a higher order of moments is used. There are two goals: the issue of accuracy and the computational complexity. Many works have been proposed to improve the accuracy and efficiency of moment calculations [7–10], but those methods mainly focus on two-dimensional (2D) geometric moments. Those methods are relatively efficient, but not accurate enough, since the computation of Legendre moments is based on an approximate formula. Liao and Pawlak [11] propose more accurate approximation formula for computing the 2D Legendre moments of a digital image when an analog original image is digitized. Then they uses an alternative extended Simpson's rule (ASER) to numerically calculate a double integral function for a higher order of Legendre moments in each pixel. These orthogonal moments have been successfully used to reconstruct some Chinese characters. The method proposed by Liao and Pawlak is relatively accurate, but it needs much more modification. Recently, Yap and Paramesran [12] propose an exact method to compute 2D Legendre moments. They explain that Legendre moments are continuous moments, hence, when they are applied to discrete-space image, a numerical approximation involved and error occurs, where the error due to approximation generally increases as the order of the moment increases. Their method is accurate, but it is time consuming. They achieved one goal and failed in the other.

This paper proposes a novel method for accurate and fast computation of Legendre moments for both binary and gray level images. A set of 2D Legendre moments are computed exactly by using a mathematical integration of Legendre polynomials. Then, a fast algorithm is applied for computation complexity reduction. The idea of this method is similar to that of Yap and Paramesran [12], but the implementation is completely different. The proposed method is completely independent of geometric moments, and easily extended to compute 3D Legendre moments. Experimental studies and the complexity analysis clearly show the superiority of the proposed method over the conventional ones.

The rest of the paper is organized as follows: In Section 2, an overview of Legendre moments is given. The proposed method

* Corresponding author at: Department of Computer Science, Nejran Community College, Nejran, P.O. Box 1988, Saudi Arabia. Tel.: +966 050 8896412; fax: +966 07 5440357.

E-mail address: k_hosny@yahoo.com.

0031-3203/$30.00
doi:10.1016/j.patcog.2007.04.014

is described in Section 3. Section 4 is devoted to give detailed analysis of computational complexity and some experimental results. Conclusion and concluding remarks are presented in Section 5.

2. Legendre moments

Legendre moments of order $(p+q)$ for an image with intensity function $f(x,y)$ are defined as

$$L_{pq} = \frac{(2p+1)(2q+1)}{4} \int_{-1}^{1} \int_{-1}^{1} P_p(x) P_q(y) f(x,y)\, \mathrm{d}x\, \mathrm{d}y, \tag{1}$$

where $P_p(x)$ is the pth-order Legendre polynomial defined as [13]

$$P_p(x) = \sum_{k=0}^{p} a_{k,p} x^k = \frac{1}{2^p p!} \left(\frac{\mathrm{d}}{\mathrm{d}x}\right)^p [(x^2-1)^p], \tag{2}$$

where $x \in [-1, 1]$, and the Legendre polynomial $P_p(x)$ obeys the following recursive relation:

$$P_{p+1}(x) = \frac{(2p+1)}{(p+1)} x P_p(x) - \frac{p}{(p+1)} P_{p-1}(x), \tag{3}$$

with $P_0(x) = 1$, $P_1(x) = x$ and $p > 1$. The set of Legendre polynomials $\{P_p(x)\}$ forms a complete orthogonal basis set on the interval $[-1, 1]$. The orthogonality property is defined as

$$\int_{-1}^{1} P_p(x) P_q(x)\, \mathrm{d}x = \begin{cases} 0, & p \neq q, \\ \dfrac{2}{(2p+1)}, & p = q. \end{cases} \tag{4}$$

A digital image of size $M \times N$ is an array of pixels. Centers of these pixels are the points (x_i, y_j), where the image intensity function is defined only for this discrete set of points $(x_i, y_j) \in [-1, 1] \times [-1, 1]$. $\Delta x_i = x_{i+1} - x_i$, $\Delta y_j = y_{j+1} - y_j$ are sampling intervals in the x- and y-directions, respectively. In the literature of digital image processing, the intervals Δx_i and Δy_j are fixed at constant values $\Delta x_i = 2/M$, and $\Delta y_j = 2/N$, respectively. Therefore, the points (x_i, y_j) will be defined as follows:

$$x_i = -1 + (i - \tfrac{1}{2})\Delta x, \tag{5.1}$$

$$y_j = -1 + (j - \tfrac{1}{2})\Delta y, \tag{5.2}$$

with $i = 1, 2, 3, \ldots, M$ and $j = 1, 2, 3, \ldots, N$. For the discrete-space version of the image, Eq. (1) is usually approximated by

$$\tilde{L}_{pq} = \frac{(2p+1)(2q+1)}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} P_p(x_i) P_q(y_j) f(x_i, y_j). \tag{6}$$

Eq. (6) is so-called direct method for Legendre moments computations, which is the approximated version using zeroth-order approximation (ZOA). As indicated by Liao and Pawlak [11], Eq. (6) is not a very accurate approximation of Eq. (1). To improve the accuracy, they propose to use the following approximated form:

$$L_{pq} = \frac{(2p+1)(2q+1)}{4} \sum_{i=1}^{M} \sum_{j=1}^{N} h_{pq}(x_i, y_j) f(x_i, y_j), \tag{7}$$

where

$$h_{pq}(x_i, y_j) = \int_{x_i-(\Delta x_i/2)}^{x_i+(\Delta x_i/2)} \int_{y_j-(\Delta y_j/2)}^{y_j+(\Delta y_j/2)} \times P_p(x_i) P_q(y_j)\, \mathrm{d}x\, \mathrm{d}y. \tag{8}$$

Liao and Pawlak propose (AESR) method to evaluate the double integral defined by Eq. (8), and then they use it to calculate the Legendre moments defined by Eq. (7).

2.1. Image reconstruction using Legendre moments

Liao and Pawlak [11] shows that the reconstruction from orthogonal moments only adds the individual components of each order to generate the reconstructed image. Since, Legendre polynomial $\{P_p(x)\}$ forms a complete orthogonal basis set on the interval $[-1, 1]$ and obeys the orthogonal property. The image function $f(x,y)$ can be written as an infinite series expansion in terms of the Legendre polynomials over the square $[-1, 1] \times [-1, 1]$:

$$f(x,y) = \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} L_{pq} P_p(x) P_q(y), \tag{9}$$

where the Legendre moments L_{pq} are computed over the same square. If only Legendre moments of order smaller than or equal to Max are given, then the function $f(x,y)$ in Eq. (9) can be approximated as follows:

$$\hat{f}_{Max}(x,y) = \sum_{p=0}^{Max} \sum_{q=0}^{p} L_{p-q,q} P_{p-q}(x) P_q(y), \tag{10}$$

where the number of moments used in this form for image reconstruction is defined by

$$N_{total} = \frac{(\mathrm{Max}+1)(\mathrm{Max}+2)}{2}. \tag{11}$$

3. The proposed method

The approximation of the integral terms in Eq. (8) is responsible for the approximation error of Legendre moments [12]. These integrals need to be evaluated exactly to remove the approximation error of Legendre moments computation. To achieve this, a new accurate and fast method will be showed for exact Legendre moments computation.

3.1. Exact computation of Legendre moments

One of the special results involving Legendre polynomial is that,

$$\int P_p(x)\, \mathrm{d}x = \frac{P_{p+1}(x) - P_{p-1}(x)}{2p+1}, \tag{12}$$

where $p \geqslant 1$. For simplicity, upper and lower limits of the integration in Eq. (8) will be expressed as follows:

$$U_{i+1} = x_i + \frac{\Delta x_i}{2} = -1 + i\Delta x, \tag{13.1}$$

$$U_i = x_i - \frac{\Delta x_i}{2} = -1 + (i-1)\Delta x, \tag{13.2}$$

similarly,

$$V_{j+1} = y_j + \frac{\Delta y_j}{2} = -1 + j\Delta y, \tag{14.1}$$

$$V_j = y_j - \frac{\Delta y_j}{2} = -1 + (j-1)\Delta y. \tag{14.2}$$

Using Eqs. (7), (8), and (12), the integral parts will be written as follows:

$$\int_{U_i}^{U_{i+1}} P_p(x)\,\mathrm{d}x = \left[\frac{P_{p+1}(x) - P_{p-1}(x)}{2p+1}\right]_{U_i}^{U_{i+1}}, \tag{15.1}$$

$$\int_{V_j}^{V_{j+1}} P_q(y)\,\mathrm{d}y = \left[\frac{P_{q+1}(y) - P_{q-1}(y)}{2q+1}\right]_{V_j}^{V_{j+1}}. \tag{15.2}$$

Substitute $P_{p+1}(x)$ from Eq. (3) into (15.1), (15.2), yields Eqs. (16.1) and (16.2),

$$\int_{U_i}^{U_{i+1}} P_p(x)\,\mathrm{d}x = \frac{1}{(p+1)}[xP_p(x) - P_{p-1}(x)]_{U_i}^{U_{i+1}}, \tag{16.1}$$

$$\int_{V_j}^{V_{j+1}} P_q(y)\,\mathrm{d}y = \frac{1}{(q+1)}[yP_q(y) - P_{q-1}(y)]_{V_j}^{V_{j+1}}. \tag{16.2}$$

The set of Legendre moment can thus be computed exactly by

$$\hat{L}_{pq} = \sum_{i=1}^{M}\sum_{j=1}^{N} I_p(x_i) I_q(y_j) f(x_i, y_j), \tag{17}$$

where

$$I_p(x_i) = \frac{(2p+1)}{(2p+2)}[xP_p(x) - P_{p-1}(x)]_{U_i}^{U_{i+1}}, \tag{18.1}$$

$$I_q(y_j) = \frac{(2q+1)}{(2q+2)}[yP_q(y) - P_{q-1}(y)]_{V_j}^{V_{j+1}}. \tag{18.2}$$

Eq. (17) is valid only for $p \geqslant 1$, and $q \geqslant 1$.

Special cases:

(i) *First row*

$p = 0$; $q = 0, 1, 2, 3, \ldots, Max$:

$$\hat{L}_{0q} = \frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{N} I_q(y_j) f(x_i, y_j). \tag{19.1}$$

(ii) *First column*

$q = 0$; $p = 0, 1, 2, 3, \ldots, Max$:

$$\hat{L}_{p0} = \frac{1}{N}\sum_{i=1}^{M}\sum_{j=1}^{N} I_p(x_i) f(x_i, y_j). \tag{19.2}$$

The moment kernel of exact 2D Legendre moments is defined by Eq. (17). This kernel is independent of the image. Therefore, this kernel can be pre-computed, stored, recalled whenever it is needed to avoid repetitive computation.

3.2. Moment kernel generation

Eqs. (18.1) and (18.2) will be rewritten as follows:

$$I_p(x_i) = \frac{(2p+1)}{(2p+2)}(U_{i+1}P_p(U_{i+1}) - P_{p-1}(U_{i+1}) - U_iP_p(U_i) + P_{p-1}(U_i)), \tag{20.1}$$

$$I_q(y_j) = \frac{(2q+1)}{(2q+2)}(V_{j+1}P_q(V_{j+1}) - P_{q-1}(V_{j+1}) - V_jP_q(V_j) + P_{q-1}(V_j)). \tag{20.2}$$

Eqs. (13) and (14) are used to generate the columns U and V, respectively. The recurrence relation (3) is used to generate Legendre polynomial $P_p(x_i)$. In order to generate $P_p(U_{i+1})$, U_{i+1} is used instead of x_i. The circulation property of U_{i+1} and U_i is implemented to avoid the duplication of kernel generation time. The polynomial $P_p(U_i)$ will be generated from $P_p(U_{i+1})$ using the following algorithm:

```
for i = 1 to N
        g3(i, 0) = 1.0
endfor
for k = 1 to Max
        g3(1, k) = (−1.0)^k*g2(N, k)
        for i = 2 to N
                g3(i, k) = g2(i − 1, k)
        endfor
endfor
```

where **g2**, **g3** are matrix representations of $P_p(U_{i+1})$ and $P_p(U_i)$, respectively, N is the image size, and Max is the maximum moment order.

3.3. Fast algorithm

Computation of exact Legendre moments using Eq. (17) is similar to the direct method, which is very time consuming. Similar to the method of Fourier transform, the principle advantage of separability property is that: the 2D $(p+q)$-order Legendre moment can be obtained in two steps by successive computation of the 1D qth order moment for each row. A fast method for exact Legendre moments computation will be presented. Eq. (17) will be rewritten in a separable form as follows:

$$\hat{L}_{pq} = \sum_{i=1}^{M} I_p(x_i) Y_{iq}, \tag{21}$$

where

$$Y_{iq} = \sum_{j=1}^{N} I_q(y_j) f(x_i, y_j). \tag{22}$$

Y_{iq} in Eq. (22) is the qth order moment of row i. Since,

$$I_0(x_i) = 1/M, \tag{23}$$

substitutes Eq. (23) into Eq. (21), yields:

$$\hat{L}_{0q} = \frac{1}{M}\sum_{i=1}^{M} Y_{iq}. \tag{24}$$

4. Computational complexity and experimental results

In this section, the validity proof of the proposed method will be presented. The performance for the proposed method is evaluated and compared with the other methods. This section is divided into three subsections. The first subsection is devoted to prove the validity of the proposed method where the computed values are compared with theoretical ones. As in Ref. [12], the images used are artificially generated and are deliberately made relatively small in size so that hand calculations can be employed to obtain the theoretical values. In the second subsection, the image reconstruction aspect for real and randomly generated images is considered. In the third subsection, a complexity analysis, the computation times of the proposed method and the method of Yap and Paramesran [12] are compared. The computation time of generating kernels as well as Legendre moment computation will be considered.

4.1. Artificial images

4.1.1. First image

As mentioned above, artificial images are used to prove validity of the proposed methods. A special image whose function $f(x, y)$ has the same constant value 1 for all points (x,y) is considered. In such case, theoretical values of Legendre moments will be calculated by the following equation:

$$L_{pq} = \frac{(2p+1)(2q+1)}{4}\int_{-1}^{1}\int_{-1}^{1} P_p(x)P_q(y)\,\mathrm{d}x\,\mathrm{d}y. \tag{25}$$

Using Eq. (4) with Eq. (25) yields:

$$L_{pq} = \begin{cases} 1, & p = q = 0, \\ 0 & \text{otherwise.} \end{cases} \tag{26}$$

It is clear that, Legendre moments that are computed with Eqs. (17), (19.1), (19.2) are equal to zero. The only non-zero value is obtained from Eq. (19.1) for $p = q = 0$. These exact values are identical to the theoretical ones. The theoretical values of Legendre moments (L_{pq}, Eq. (26)), exact values ($\hat{L}_{pq}$, Eqs. (17)–(19)), and the ZOA approximated values ($\tilde{L}_{pq}$, Eq. (7)) are shown in Table 1.

4.1.2. Second image

Consider an artificial image $f(x_i, y_j)$, which is represented by the matrix $A = [3, 2, 1, 5; 6, 1, 7, 3; 2, 8, 4, 6; 5, 1, 4, 2]$. Legendre moments for this image are shown in Table 2. It is obvious that the exact values ($\hat{L}_{pq}$, Eqs. (17)–(19)) match the theoretical values (L_{pq}, Eq. (1)) while that of ZOA ($\tilde{L}_{pq}$, Eq. (7)) deviates from the theoretical values especially when the order increases.

4.2. Image reconstruction

In this section, rating the performance of the reconstructed images using approximated and exact Legendre moments will be performed using error analysis and some criteria commonly used for measuring image quality. These criteria are mean-square error (MSE) and peak signal-to-noise ratio (PSNR).

MSE is used as a measure of reconstruction error. For an n-bit image of size $M \times N$ pixels, MSE and PSNR are defined as

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(\hat{f}_{Max}(x_i, y_j) - f(x_i, y_j))^2, \tag{27}$$

$$PSNR = 10 \times \log_{10}\left(\frac{(2^n - 1)}{MSE}\right). \tag{28}$$

Eq. (27) can be rewritten as follows [11]:

$$MSE = \sum_{p=0}^{Max}\sum_{q=0}^{p}(\hat{L}_{p-q,q} - L_{p-q,q})^2 + \sum_{p=Max+1}^{\infty}\sum_{q=0}^{p}\frac{4}{(2p-2q+1)(2q+1)}\hat{L}_{p-q,q}. \tag{29}$$

The first term of Eq. (29) is the discrete approximation error, while the second one is the result of using a finite number of moments. The first error increases as Max tends to infinity. On the other hand, the second error decreases as Max tends to infinity.

The proposed method completely removes the first error. The second error is a common error in all methods that are used to compute continuous orthogonal moments. The reconstructed

Table 1
Comparison of theoretical, L_{pq}, exact, $\hat{L}_{pq}$, and ZOA, $\tilde{L}_{pq}$ for $f(x_i, y_j) = 1$

n	Theoretical, L_{pq}				Exact, $\hat{L}_{pq}$				ZOA, $\tilde{L}_{pq}$			
	Max				Max				Max			
	0	1	2	3	0	1	2	3	0	1	2	3
0	1	0	0	0	1	0	0	0	1.0000	0.0000	−0.1563	0.0000
1	0	0	0	0	0	0	0	0	0.0000	0.0000	0.0000	0.0000
2	0	0	0	0	0	0	0	0	−0.1563	0.0000	0.0244	0.0000
3	0	0	0	0	0	0	0	0	0.0000	0.0000	0.0000	0.0000

Table 2
Comparison of theoretical, L_{pq}, exact, $\hat{L}_{pq}$, and ZOA, $\tilde{L}_{pq}$ for $f(x_i, y_j) = A$

n	Max			
	0	1	2	3
Theoretical, L_{pq}				
0	3.7500	0.1875	0.4688	−0.5195
1	0.2813	−0.7734	−1.2305	−0.3179
2	−1.6406	−0.5273	2.1973	−0.0769
3	−0.3691	−1.2407	1.1023	−3.2016
Exact, $\hat{L}_{pq}$				
0	3.7500	0.1875	0.4688	−0.5195
1	0.2813	−0.7734	−1.2305	−0.3179
2	−1.6406	−0.5273	2.1973	−0.0769
3	−0.3691	−1.2407	1.1023	−3.2016
ZOA, $\tilde{L}_{pq}$				
0	3.7500	0.1875	−0.1172	−0.5879
1	0.2813	−0.7734	−1.2744	−0.0359
2	−2.2266	−0.5566	2.4719	0.2072
3	−0.4717	−0.9587	1.6246	−2.7361

image will be very close to the original one when the maximum moment order reaches a certain value.

A randomly image $f(x_i, y_j)$ is generated using MatLab7 as follows:

$$f(x_i, y_j) = rand(M, N), \quad 0 \leqslant f(x_i, y_j) \leqslant 1 \ \forall i, j. \tag{30}$$

Both the proposed and the approximated methods are used to reconstruct the random image defined by Eq. (30). Image dimensions are selected to be $M = N = 64$, and the maximum moment order ranging from 10 to 60. Fig. 1(a) shows MSE for both the proposed method (Exact) and the approximated method (ZOA). It is clear that, MSE for the exact method decreases as the moment order increases, while, it increases as the moment order increases for the approximated method. This result clearly shows the efficiency of the proposed method. Fig. 1(b) shows PSNR for both methods. PSNR for both methods are relatively equal for low order moments. As moment order increases the PSNR values are strongly deviated, where the values of the exact method monotonically increases. On the other hand, the values of the approximated method monotonically decrease.

Fig. 2(b) and (c) shows the curves of MSE and PSNR of the real gray level image in Fig. 2(a). The first figure shows that, the estimated MSE of the proposed method tends to zero as the moment order increase, while second shows the big difference between the PSNR of the proposed method and the approximated one. The same conclusion is obtained from Fig. 3. It is clear that, the obtained results confirm the accuracy of the proposed method.

4.3. Computation time

Any set of parameters obtained by projecting an image onto a 2D polynomial basis are called moments. Therefore, computation of Legendre moments basically consists of two stages. In the first stage, the moment kernels are generated, while in the second stage the moment kernels are multiplied with the image

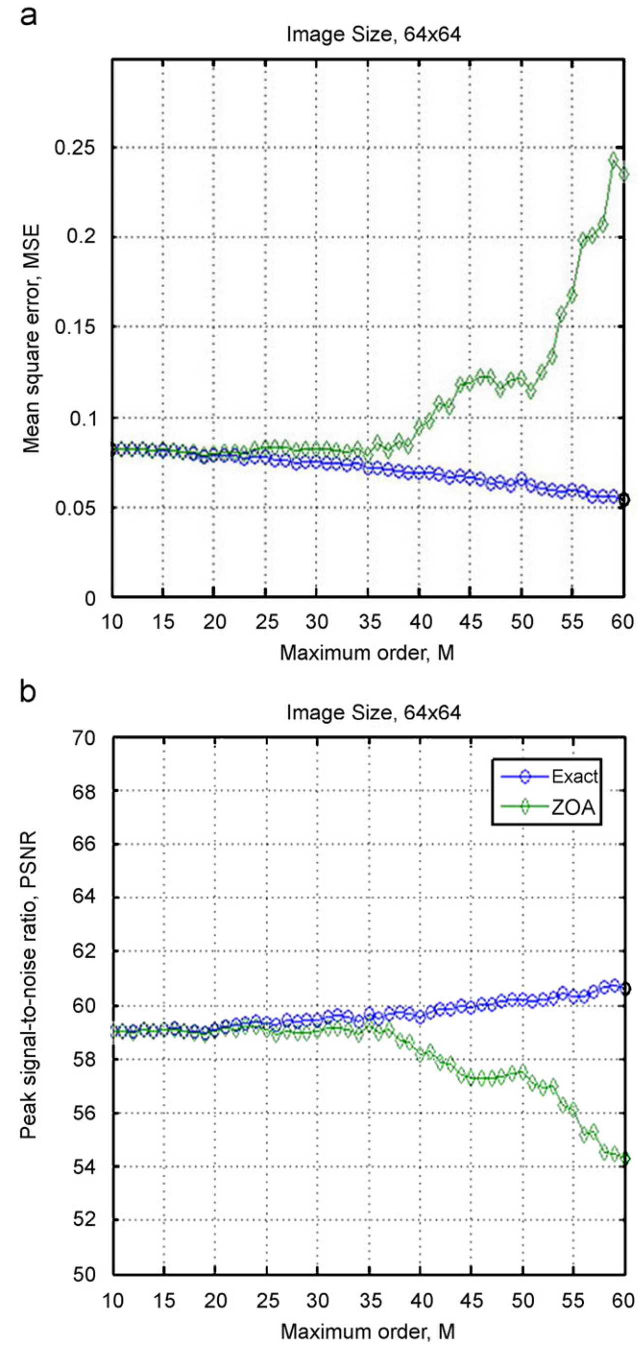

Fig. 1. Random generated image of size 64 × 64: (a) MSE and (b) PSNR.

function and resulted in the set of Legendre moments. Moment kernels are independent of images; therefore, they can be computed in advance, stored and retrieved whenever necessary. The computation time for the first stage is much less important than that of the second stage. Consequently, we concentrate on the reduction of the moment commutation time by minimization of the total addition and multiplication operations.

Computation time of the moment kernels for exact method of Yap and Paramesran [12], and the proposed method will be presented. For 1000 polynomial points, Table 3 shows the computation time of the kernels for different values of moment order. It is clear that, the time required by the proposed method

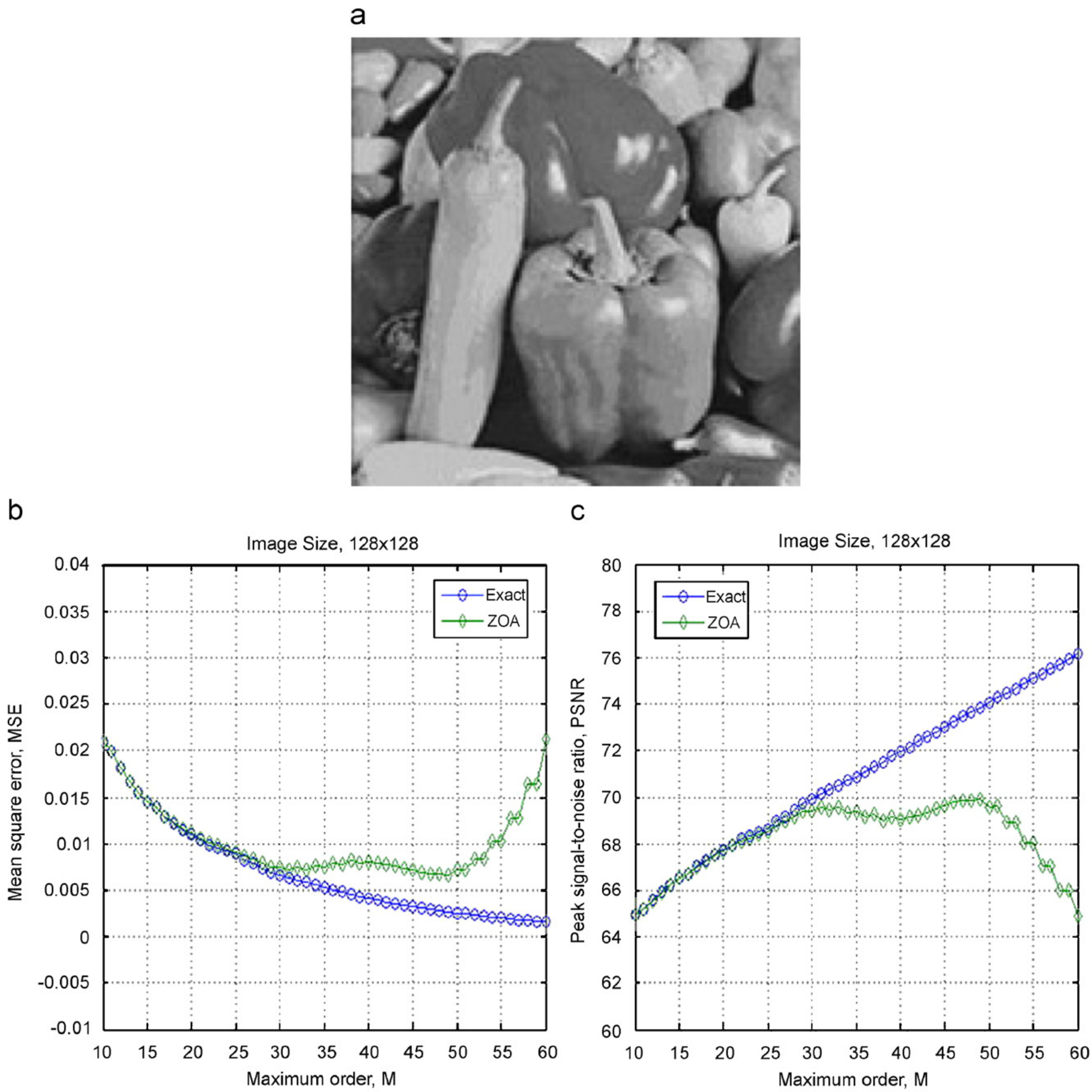

Fig. 2. Real image of size 128 × 128: (a) peppers original image, (b) MSE, and (c) PSNR.

to generate the moment kernel is very small compared to that time of Yap and Paramesran [12] especially for higher order moments.

For a digital gray level image of size $N \times N$, and Max is moment order, Yap and Paramesran [12], reported that, the total number of operations required by their method and the ZOA one for Legendre moment's computation are identical. Yang et al. [10] reported that, the ZOA required $(\mathrm{Max}+1)^2N^2/2$ additions and $(\mathrm{Max}+1)^2N^2$ multiplications. Based on Eq. (8), $(\mathrm{Max}+1)(\mathrm{Max}+2)/2$ is used instead of $(\mathrm{Max}+1)^2$, therefore, the numbers of operations are

$$\frac{(\mathrm{Max}+1)(\mathrm{Max}+2)}{2}N^2 \quad \text{additions}, \tag{31.1}$$

$$(\mathrm{Max}+1)(\mathrm{Max}+2)N^2 \quad \text{multiplications}. \tag{31.2}$$

The computational complexity for the proposed method will be discussed in detail. Legendre moment's computation using the proposed method consists of two main steps. Each step will be discussed individually; then the whole computational complexity will be evaluated easily. Step 1, the creation of the matrix Y_{iq} requires $N(N-1)(\mathrm{Max}+1)$ additions and $N^2(\mathrm{Max}+1)$ multiplications. The matrix of Legendre moments is an upper triangle square matrix of dimensions $(\mathrm{Max}+1)$. The total number of Legendre moments is $N_{total} = (\mathrm{Max}+1)(\mathrm{Max}+2)/2$. The computation of the Legendre moment matrix is divided to three steps namely; the first row, the first column and the rest of the moments.

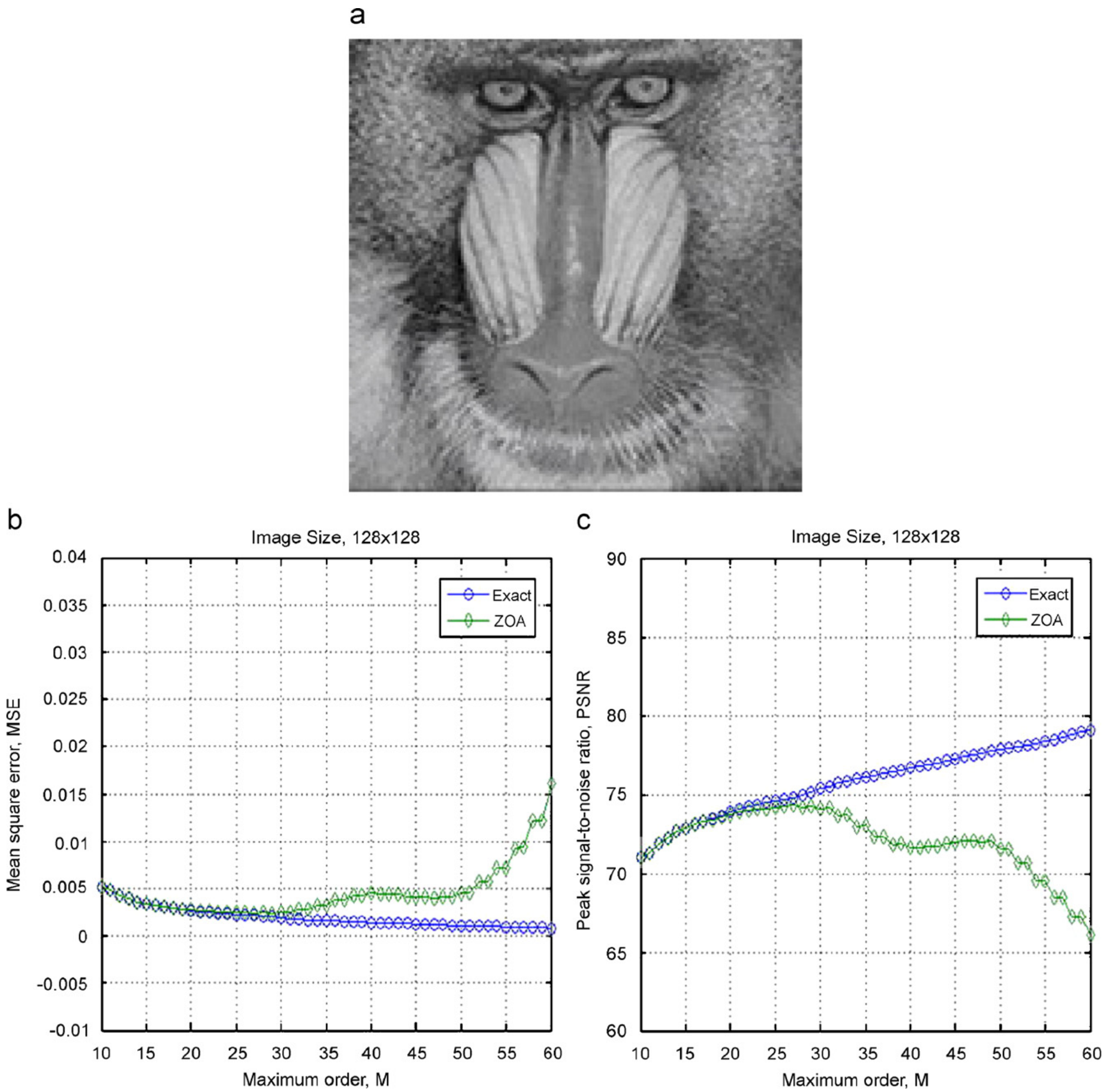

Fig. 3. Real image of size 128 × 128: (a) baboon original image, (b) MSE, and (c) PSNR.

Table 3
Comparison between kernels's generating times (in seconds)

Moment order	Yap and Paramesran [12]	Proposed method
10	0.0630	0.0470
20	0.2190	0.0940
30	0.4680	0.1250
40	0.7970	0.1870
50	1.2500	0.2350
100	4.9840	0.4850

According to Eq. (19.1), the computation of the first row needs only the addition process of the elements of Y_{iq}. This process requires $(N-1)(\mathrm{Max}+1)$, where $(\mathrm{Max}+1)$ refers to the number of elements. Similarly, by using Eq. (19.2) the computation of the first column requires $\mathrm{Max}(N-1)$ addition process, where Max refers to the number of elements. The rest of the non-zero matrix elements is $\mathrm{Max}(\mathrm{Max}-1)/2$. The computation of these moments requires $\mathrm{Max}(\mathrm{Max}-1)(N-1)/2$ additions and $\mathrm{Max}(\mathrm{Max}-1)N/2$ multiplications. So computing all the required exact Legendre moments needs

$$\frac{(\mathrm{Max}+1)(N-1)}{2}(2N+\mathrm{Max}+2) \quad \text{additions}, \tag{32.1}$$

$$\frac{N\,\mathrm{Max}}{2}(2N+\mathrm{Max}-1)+N^2 \quad \text{multiplications}. \tag{32.2}$$

Table 4
Computational complexity of Yap's exact method and the proposed method

	Yap [12]		Proposed method	
	No. of +	No. of ∗	No. of +	No. of ∗
Max = 30, $N = 64$	2 031 616	4 063 232	156 240	154 816
Max = 40, $N = 128$	14 106 624	28 213 248	757 843	771 584
Max = 40, $N = 512$	255 705 984	451 411 968	11 166 883	11 147 264
Max = 50, $N = 512$	347 602 944	695 205 888	14 020 818	13 996 544
Max = 60, $N = 512$	495 714 304	991 428 608	16 925 853	16 897 024

Table 5
CPU elapsed time of Yap's exact method and the proposed method (in seconds)

	ZOA	Yap [12]	Proposed method
	Time (s)	Time (s)	Time (s)
Max = 30, $N = 64$	0.2350	0.2970	0.0470
Max = 40, $N = 128$	1.2660	1.5000	0.1100
Max = 40, $N = 512$	55.0000	55.7340	1.6250
Max = 50, $N = 512$	84.4530	86.2350	1.8130
Max = 60, $N = 512$	127.7030	129.8430	2.3910

The total number of additions and multiplications required by Yap's method [12], and the proposed are compared. Table 4 shows the number of arithmetic operations for some values of N, and Max. It is clear that, the proposed method tremendously reduced the total number of arithmetic operations. Consequently, the CPU time required to compute Legendre moments is reduced tremendously.

The CPU elapsed times (the program is coded in Matlab7, and implemented on P4 1.8 GHz with 512 MB RAM) for the ZOA, Yap's method and the proposed method are showed in Table 5.

Despite of, ZOA and exact method of Yap and Paramesran, required the same total number of arithmetic operations, the CPU elapsed time of the latest is higher than the first. This is according to the higher time required to generate the moment kernel.

Since the moment kernel is pre-computed and stored and for fair comparison, we performed the experiment for moment's computation only. A baboon gray level image of size 512×512 is used in this experiment. The obtained results are plotted in Fig. 4. Based on this comparison, it is easy to say that the proposed method for 2D Legendre moment computation is exact and fast method.

To confirm the superiority of the proposed method, a quick comparison with the result of the recent method of Yang and his co-authors [10] will be presented. Yang and his colleagues propose an approximated method to compute 2D Legendre moment for gray level images. Their method reduces the number of multiplication operations. On the other hand, unfortunately, tremendously increased the number of addition operations. To compute 2D Legendre moment for a gray level image of size equal $N = 512$, and the order of moment is Max = 50, Yang's method [10] requires 2 643 333 multiplication

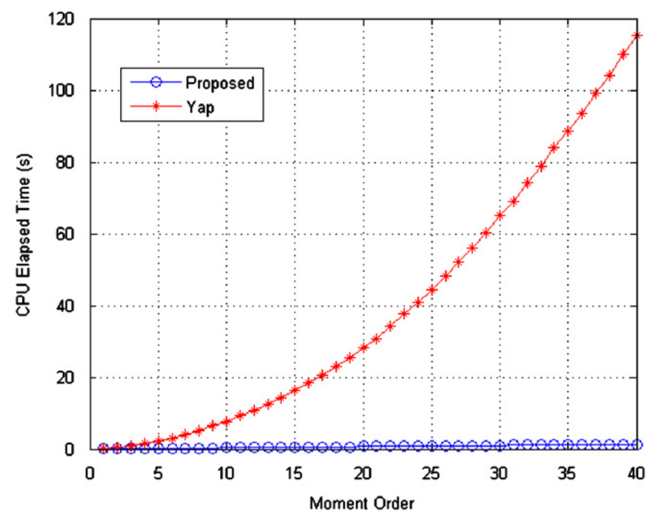

Fig. 4. CPU elapsed time (in seconds).

operations, and 666 026 666 addition operations. This comparison ensures the superiority of our method.

5. Conclusion

This paper proposes a new exact and fast method for computing 2D Legendre moments for gray level images. The Legendre moment values calculated by using the approximated method are deviated from those theoretical values. The error steadily increases as the moment order increases. On the other hand, Legendre moments calculated using proposed method are identical to those obtained by theoretical calculations. Image reconstruction using the proposed method shows improvement over that of the approximated method, where the reconstruction error increases as the moment order increases. The computation time of the proposed method is extremely smaller than that of the approximated method. The proposed method is extended easily to calculate 3D Legendre moments. It is obvious that, the proposed method is outperformed over than all available methods for Legendre moment computations.

References

[1] M.K. Hu, Visual pattern recognition by moment invariants, IRE Trans. Inf. Theory 8 (1) (1962) 179–187.
[2] M.R. Teague, Image analysis via the general theory of moments, J. Opt. Soc. Am. 70 (8) (1980) 920–930.

[3] C.H. The, R.T. Chin, On image analysis by the method of moments, IEEE Trans. Pattern Anal. Mach. Intell. 10 (4) (1988) 496–513.
[4] C.-W. Chong, R. Paramesran, R. Mukundan, Translation and scale invariants of Legendre moments, Pattern Recognition 37 (1) (2004) 119–129.
[5] J. Haddadnia, K. Faez, P. Moallem, Neural network based face recognition with moment invariants, Proceedings of the International Conference on Image Processing, vol. 1, October 2001, pp. 1018–1021.
[6] H. Qjidaa, L. Radouane, Robust line fitting in a noisy image by the method of moments, IEEE Trans. Pattern Anal. Mach. Intell. 21 (11) (1999) 1216–1223.
[7] R. Mukundan, K.R. Ramakrishnan, Fast computation of Legendre and Zernike moments, Pattern Recognition 28 (9) (1995) 1433–1442.
[8] H.Z. Shu, L.M. Luo, W.X. Yu, Y. Fu, A new fast method for computing Legendre moments, Pattern Recognition 33 (2) (2000) 341–348.
[9] J.D. Zhou, H.Z. Shu, L.M. Luo, W.X. Yu, Two new algorithms for efficient computation of Legendre moments, Pattern Recognition 35 (5) (2002) 1143–1152.
[10] G.Y. Yang, H.Z. Shu, C. Toumoulin, G.N. Han, L.M. Luo, Efficient Legendre moments computation for grey level images, Pattern Recognition 39 (1) (2006) 74–80.
[11] S.X. Liao, M. Pawlak, On image analysis by moments, IEEE Trans. Pattern Anal. Mach. Intell. 18 (3) (1996) 254–266.
[12] P.-T. Yap, R. Paramesran, An efficient method for the computation of Legendre moments, IEEE Trans. Pattern Anal. Mach. Intell. 27 (12) (2005) 1996–2002.
[13] M.R. Spiegel, Schaum's Handbook of Formulas and Tables, McGraw-Hill, New York, 1968.

About the Author—KHALID M. HOSNY received the B.Sc., M.Sc. and Ph.D. from Zagazig University, Zagazig, Egypt in 1988, 1994, and 2000, respectively. From 1997 to 1999 he was a Visiting Scholar, University of Michigan, Ann Arbor and University of Cincinnati, Cincinnati, USA. He joined the Faculty of Computers and Informatics at Zagazig University, where he held the position of Assistant Professor. His research interests include mathematical modeling, image processing, and pattern recognition.