J Real-Time Image Proc (2011) 6:73–80
DOI 10.1007/s11554-009-0135-z

ORIGINAL RESEARCH PAPER

Fast computation of orthogonal Fourier–Mellin moments in polar coordinates

Khalid Mohamed Hosny · Mohamed A. Shouman · Hayam M. Abdel Salam

Received: 2 May 2009 / Accepted: 3 October 2009 / Published online: 11 November 2009

Abstract Fast, accurate and memory-efficient method is proposed for computing orthogonal Fourier–Mellin moments. Since the basis polynomials are continuous orthogonal polynomials defined in polar coordinates over a unit disk, the proposed method is applied to polar coordinates where the unit disk is divided into a number of non-overlapping circular rings that are divided into circular sectors of the same area. Each sector is represented by one point in its center. The implementation of this method completely removes both approximation and geometrical errors produced by the conventional methods. Based on the symmetry property, a fast and memory-efficient algorithm is proposed to accelerate the moment's computations. A comparison to conventional methods is performed. Numerical experiments are performed to ensure the efficiency of the proposed method.

Keywords Fourier–Mellin moments · Symmetry property · Fast algorithm · Polar coordinates · Gray-level images

1 Introduction

Orthogonal Fourier–Mellin moments (OFMMs) were introduced by Sheng and Shen [1]. These moments are defined using a set of orthogonal radial polynomials $Q_n(r)$ and have better performance than orthogonal Zernike moments in terms of image description and noise sensitivity, where the orthogonal radial polynomials of the first have more zeros than that of the second in the region of small radial distance. Like other orthogonal moments [2] and based on their orthogonal property, the set of OFMMs are used to describe objects with the minimum amount of information redundancy. These moments are rotationally invariants and have the ability to describe spatial frequency components of an image. Due to their elegant characteristics, OFMMs are used in image processing and pattern recognition applications [3–7].

Direct computation of Fourier–Mellin moments through the basis orthogonal polynomials produced two types of errors. These errors are numerical and geometrical. The first error is a result of approximation process, while the second is a result of square-to-circle mapping. In addition to these produced errors, an excessive amount of time is required to compute Fourier–Mellin moments, especially when the moment's order is high. This makes it impractical for real-time applications.

To reduce the computational time, few algorithms have been proposed to speed up the computation of Fourier–Mellin moments [8, 9]. Recently, Xin et al. [10] proposed a novel method for accurate computation of Zernike moments in polar coordinates. Based on the fact that all circular moments are defined in a unit disk, the method of Xin et al. is modified to compute accurately Fourier–Mellin moments.

This study proposes a fast method for accurate Fourier–Mellin moment computation. The whole set of 2D Fourier–Mellin moments are computed accurately in polar coordinates. The symmetry property is applied where more than 87% of computational complexity is reduced. A fast algorithm is applied to accelerate the computation of Fourier–Mellin moments. The idea of this algorithm is

K. M. Hosny (✉) · M. A. Shouman · H. M. Abdel Salam
Faculty of Computers and Informatics,
Zagazig University, Zagazig, Egypt
e-mail: k_hosny@yahoo.com

K. M. Hosny
Najran Community College,
Najran University, Najran, Saudi Arabia

similar to the implemented one in [11]. Based on the significant time and computational complexity reduction, the proposed method is used for computation of real-time Fourier–Mellin moments. Numerical experiments clearly show the superiority of the proposed method.

The rest of the paper is organized as follows: In Sect. 2, an overview of OFMMs and the approximate computational method are given. The proposed method is described in Sect. 3. Section 4 is devoted to numerical experiments. Conclusion and concluding remarks are presented in Sect. 5.

2 Orthogonal Fourier–Mellin moments

The OFMMs are defined using the orthogonal polynomials $Q_p(r)$. These polynomials are defined as follows:

$$Q_p(r) = \sum_{k=0}^{p} \alpha_{pk} r^k, \tag{1}$$

where the coefficient matrix α_{pk} is defined as:

$$\alpha_{pk} = (-1)^{(p+k)} \frac{(p+k+1)!}{(p-k)!k!(k+1)!}. \tag{2}$$

The set of polynomials $Q_p(r)$ is orthogonal over the range $0 \le r \le 1$. The orthogonality property is expressed as follows:

$$\int_0^1 Q_p(r) Q_q(r) r \mathrm{d}r = a_p \delta_{pq}, \tag{3}$$

where δ_{pq} is the Kronecker symbol and a_p is the normalization constant defined as:

$$a_p = \frac{1}{2(p+1)}. \tag{4}$$

The direct computation of the coefficient matrix α_{pk} required calculation of four factorial terms plus the evaluation of an exponential function for every order p. Recurrence relations are derived and used to avoid these time and memory-consuming computational processes. These relations are derived from the definition in (2) as follows:

$$\alpha_{p0} = (-1)^p (p+1), \tag{5.1}$$

$$\alpha_{pk} = -\frac{(p+k+1)(p-k+1)}{k(k+1)} \alpha_{p(k-1)}. \tag{5.2}$$

The OFMMs of order p and repetition q are defined as:

$$A_{pq} = \frac{(p+1)}{\pi} \int_0^{2\pi} \int_0^1 f(r,\theta) Q_p(r) \mathrm{e}^{-\hat{i}q\theta} r \mathrm{d}r \mathrm{d}\theta, \tag{6}$$

where $\hat{i} = \sqrt{-1}$, $p = 0, 1, 2, 3, \ldots, \infty$ and q is any positive or negative integer and the number of independent OFMMs of order Max is $(\mathrm{Max} + 1)^2$. One of the important characteristics of Fourier–Mellin moments is the rotation invariance, where the magnitude values of Fourier–Mellin moments are rotation invariants, which means that these values are unaffected and remain identical for image functions before and after rotation. If an image function $f(r, \theta)$ having Fourier–Mellin moments, A_{pq} is rotated counterclockwise by angle α; the transformed image function is $g(r, \theta) = f(r, \theta - \alpha)$. The moments of the rotated image are:

$$A_{pq}^r = \mathrm{e}^{-\hat{i}q\theta} A_{pq}. \tag{7}$$

Equation (7) shows that the magnitude values of moments of rotated and original image are identical, because $\mathrm{e}^{-\hat{i}q\theta} = 1$.

Approximate computation of Fourier–Mellin moments is generally based on square-to-circle mapping as shown in Fig. 1 where the basis functions are deified in polar coordinates over a unit disk, and digital images are usually defined in Cartesian coordinates. Such mapping produces two sources of errors [12], namely geometrical and numerical

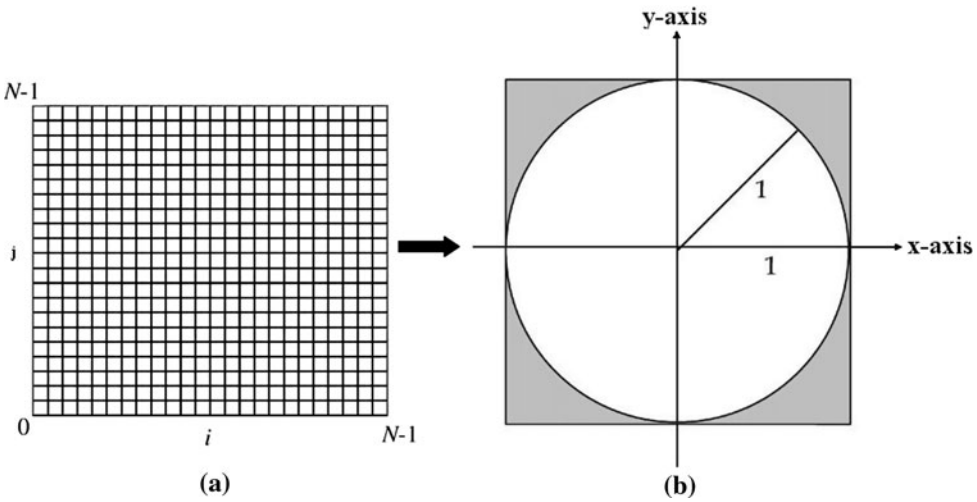

Fig. 1 **a** Image plane, **b** square-to-circle mapping of the image plane

errors. The first error is a result of approximation process, while the second one is a result of square-to-circle mapping. The propagation of these errors in numerical computations degrades the accuracy of the computed moments. The OFMMs of order p and repetition q for an image intensity function $f(x, y)$ are computed approximately using the zeroth-order approximation (ZOA), where the integrals in (6) are replaced by summations and the image is normalized inside the unit disk. The approximate OFFMs are computed using the following form of the direct ZOA method:

$$\tilde{A}_{pq} = \frac{p+1}{\lambda_p} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} Q_p(r_{xy}) e^{-\hat{i}q\theta_{xy}} f(x,y) \tag{8}$$

with $r_{xy} = \sqrt{x^2 + y^2}$ and $\theta_{xy} = \arctan(y/x)$. The number λ_p refers to the total number of pixels that achieves the condition $|r_{ij}| \leq 1$.

3 The proposed method

This section describes the proposed method. Section 3.1 is devoted to the computation of OFMMs moments in polar coordinates. In Sects 3.2 and 3.3, a fast algorithm calculating OFMMs in a separable form is proposed and symmetry property is applied to accelerate the computational process.

3.1 Accurate computation of OFMMs in polar coordinates

Computation of orthogonal moments in polar coordinates is based on the idea of splitting the unit disk into a set of non-overlapped circular sectors. Each sector is represented by one point in its center. The unit disk of area π is divided into M non-overlapped circular rings. All circular rings are divided into circular sectors, where all of these sectors must have the same area. The value of M is dependent on the constraint $N/2 \leq M \leq N$ [10], where $N \times N$ is the size of the digital image. Figure 2 present an illustration example.

The radius of the ith circular ring is $(i + 1)/M$; where $i = 0, 1, 2, \ldots, M - 1$ and $i = 0$ refers to the innermost circular ring. The area of the ith circular ring is $A_i = \pi((2i + 1)/M^2)$;

Assume S is the number of circular sectors in the innermost circular ring, then the total number of circular sectors is SM^2, where the area of each circular sector is π/SM^2. The number of circular sectors in the ith circular ring is $K_i = (2i + 1)S$. Each circular sector is represented with only one point at its center. All sectors in the same circular ring have the same radial distance. The radial distance of the ith circular ring is $R_i = (2i + 1)/2M$. The upper and lower limit of the radial integral are:

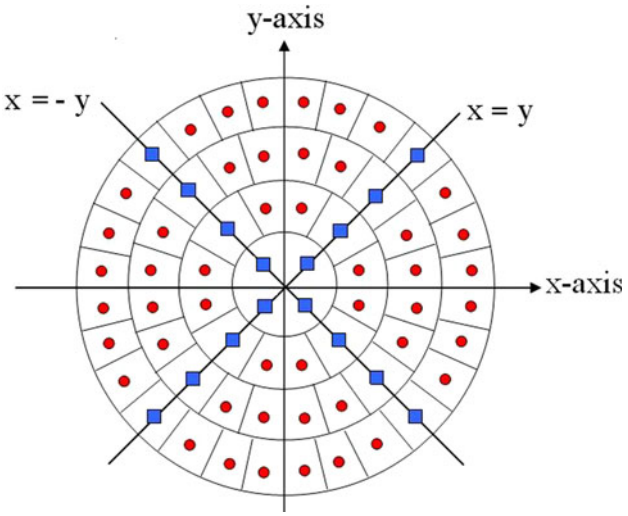

Fig. 2 A unit disk is divided into non-overlapping circular rings

$$U_{i+1} = R_i + \Delta R_i/2, \quad U_i = R_i - \Delta R_i/2, \quad \text{where } \Delta R_i = 1/M. \tag{9}$$

The distribution of angles is dependent on the circular ring, where the value of increment is different for the different circular rings. Figure 3 shows a circular sector in the $i + 1$th circular ring. The figure clearly shows the dependence of the angle θ on the order of the circular ring. The values of the angle θ can be created using the following pseudo-code.

For $i = 0$ **to** $M - 1$
  $K_i = (2i + 1)S$
  For $j = 0$ **to** $K_i - 1$
    $\theta_{i,j} = 2\pi(j + 0.5)/K_i$
  endfor
endfor

The upper and lower limits of the angle integral are:

$$V_{i,j+1} = \theta_{i,j} + \Delta\theta_{i,j}/2, \quad V_{i,j} = \theta_{i,j} - \Delta\theta_{i,j}/2, \tag{10}$$

where $\Delta\theta_{i,j} = 2\pi/K_i$. A cubic interpolation method is applied to the image intensity function $f(x, y)$ of size N^2 to produce an interpolated intensity function $\hat{f}(r_i, \theta_{i,j})$ of size

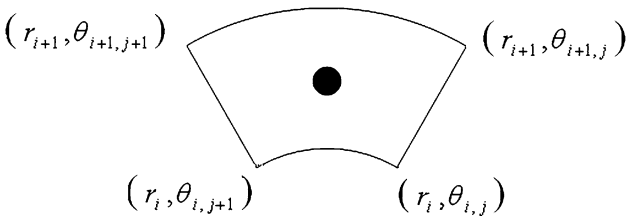

Fig. 3 A circular sector with the four corner points

SM^2. For detailed discussion of this process, we refer the readers to [10]. The 2D OFMMs of order p and repetition q are:

$$A_{pq} = \frac{p+1}{\pi} \sum_i \sum_j \hat{f}(r_i, \theta_{i,j}) H_{pq}(r_i, \theta_{i,j}) \tag{11}$$

with

$$H_{pq}(r_i, \theta_{i,j}) = I_p(r_i) I_q(\theta_{ij}), \tag{12}$$

where

$$I_p(r_i) = \int_{U_i}^{U_{i+1}} Q_p(r) r \mathrm{d}r, \tag{13}$$

$$I_q(\theta_{ij}) = \int_{V_{i,j}}^{V_{i,j+1}} \mathrm{e}^{-\hat{i}q\theta} \mathrm{d}\theta. \tag{14}$$

Applying (1) to (13), the real-valued radial-based integral may be represented as:

$$I_p(r_i) = \int_{U_i}^{U_{i+1}} \left(\sum_{k=0}^{p} \alpha_{p,k} r^k \right) r \mathrm{d}r, \tag{15}$$

$$I_p(r_i) = \sum_{k=0}^{p} \alpha_{p,k} \left(\frac{U_{i+1}^{k+2} - U_i^{k+2}}{k+2} \right). \tag{16}$$

The integral in (14) can be rewritten as:

$$I_q(\theta_{i,j}) = \begin{cases} \frac{I}{q} \left(\mathrm{e}^{-\hat{i}qV_{i,j+1}} - \mathrm{e}^{-\hat{i}qV_{i,j}} \right), & q \neq 0, \\ V_{i,j+1} - V_{i,j}, & q = 0. \end{cases} \tag{17}$$

It is clear that the kernels $I_p(r_i)$ and $I_q(\theta_{i,j})$ are independent of the image. Both kernels can be pre-computed and stored for any future use.

3.2 Fast algorithm

Implementing (11) for accurate computation of 2D OFMMs is time-consuming. Fast computation can be achieved by successive computation in angular and radial directions. Equation (11) can be rewritten in a separable form as follows:

$$A_{pq} = \frac{p+1}{\pi} \sum_i I_p(r_i) Y_{iq}, \tag{18}$$

$$Y_{iq} = \sum_j I_q(\theta_{ij}) f(r_i, \theta_{ij}). \tag{19}$$

3.3 Symmetry property

All points in the same circular ring have the same radial distance from the origin as illustrated in Fig. 4. The point $P_1 = (r_i, \theta_{ij})$ in the ith circular ring has seven similar

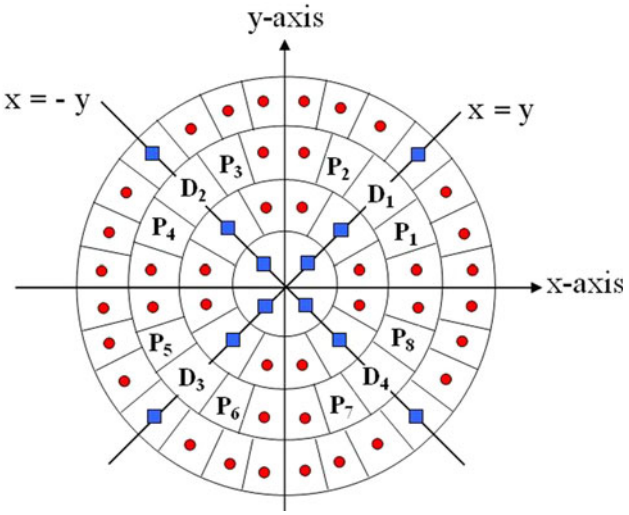

Fig. 4 Symmetrical points within the same circular ring

Table 1 The first type of symmetry points and their coordinates

P_1	(r_i, θ_{ij})
P_2	$(r_i, \pi/2 - \theta_{ij})$
P_3	$(r_i, \pi/2 + \theta_{ij})$
P_4	$(r_i, \pi - \theta_{ij})$
P_5	$(r_i, \pi + \theta_{ij})$
P_6	$(r_i, 3\pi/2 - \theta_{ij})$
P_7	$(r_i, 3\pi/2 + \theta_{ij})$
P_8	$(r_i, 2\pi - \theta_{ij})$

points, where the index one refers to the first octant. Coordinates of similar points are defined according to Table 1 with $0 < \theta_{ij} < \pi/4$.

According to Fig. 5, we conclude that there are two types of symmetrical points. The first one refers to the

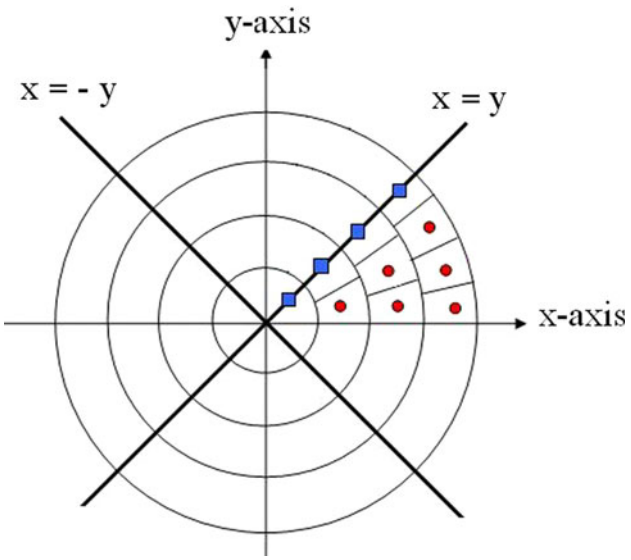

Fig. 5 Symmetrical points in the first octant

Table 2 The second type of symmetry points and their coordinates

T_1	$(r_i, \pi/4)$
T_2	$(r_i, \pi/2 + \pi/4)$
T_3	$(r_i, \pi + \pi/4)$
T_4	$(r_i, 3\pi/2 + \pi/4)$

points that fall inside the first octant. Each one of these points has similar points in the other seven octants. The coordinates of these points are defined according to Table 1. The other type of symmetrical points is those that have the angle coordinate equal to $\theta_{ij} = \pi/4$. Each one of these points has only three symmetrical ones. Their coordinates are listed in Table 2.

3.4 Memory requirements

The memory requirement is an important issue in evaluation of any computational method. In this section, the reduction due to symmetry is discussed in detail. As discussed in Sect. 3.1, the unit disk of M non-overlapping circular rings has SM^2 total number of circular sectors. According to the symmetry property, the value of $I_p(r_i)$ is the same for all circular sectors in the ith circular ring. Therefore, only one value is computed for each circular ring. On the other hand, the values of $I_q(\theta_{ij})$ are handled in a different way. The innermost circular ring ($i = 0$) has four circular sectors. Only one is computed and the rest are straightforward, inferred using the second type of symmetrical points (see Table 2). The computation process required only two points: one of the first type and another one of the second type. On going through the outer circular rings, the number of points of the first symmetrical type increases, while only one point of the second type is required for each circular ring. Consequently, the total number of computed points in the first octant is equal to

$$(0+1)+(1+1)+(2+1)+(3+1)+(4+1)+\cdots+(M-1+1). \tag{20}$$

Each pair represents the required number of points to be computed in the circular ring. The first number refers to the first type of symmetrical points, while the second number refers to the second type of points. The total number defined by (20) can be replaced by the series $M(M+1)/2$. Therefore, the reduction in the computed points according to the symmetry property is defined as follows:

$$\left(1 - \frac{M(M+1)/2}{SM^2}\right) \times 100. \tag{21}$$

A fast numerical example is used to explain the efficiency of the proposed method. In Table 3, an example with different number of circular rings is presented.

Table 3 Reduction percentage of the total number of points to be computed

No. of circular rings	No. of computed points		Percentage of reduction
	Xin's [10] method	Proposed method	
6	144	21	85.42
12	576	78	86.46
18	1,296	171	86.81
24	2,304	300	86.98
36	5,184	666	87.15
64	16,384	2,080	87.30
128	65,536	8,256	87.40
512	1,048,576	131,328	87.48

4 Numerical experiments

In this section, the efficiency of the proposed method will be presented and compared with the direct ZOA method defined by (8) and the method of Papakostas et al. [8]. In fact, Papakostas et al. derived a relation to recursively compute the set of orthogonal radial polynomials $Q_p(r)$. According to [8], the set of OFMMs are approximately computed by using the following equations:

$$O_{pq} = \frac{p+1}{\pi} \sum_x \sum_y f(x,y) Q_p(r) \mathrm{e}^{-\hat{\imath} q\theta}, \tag{22}$$

where the orthogonal radial polynomials are computed recursively using the following equations:

$$Q_p(r) = (-1)^p T_{p0} + \sum_{k=1}^{p} (-1)^{p+k} T_{pk} r^k \tag{23}$$

with

$$T_{p0} = p + 1, \tag{24.1}$$

$$T_{pk} = \frac{(p+k+1)(p-k+1)}{k(k+1)} T_{p(k-1)}. \tag{24.2}$$

The CPU elapsed times for the computation of OFMMs in polar coordinates are discussed with illustrated figures to ensure the superiority of the proposed method.

4.1 Computational time

The issue of computational time is very crucial, especially for large images. For explicit computational time, a number of numerical experiments are performed. All our numerical experiments were performed with 1.8-GHz Pentium IV PC with 512 MBYTE RAM. The code is designed using Matlab7. The elapsed CPU times of ZOA, Papakostas, and

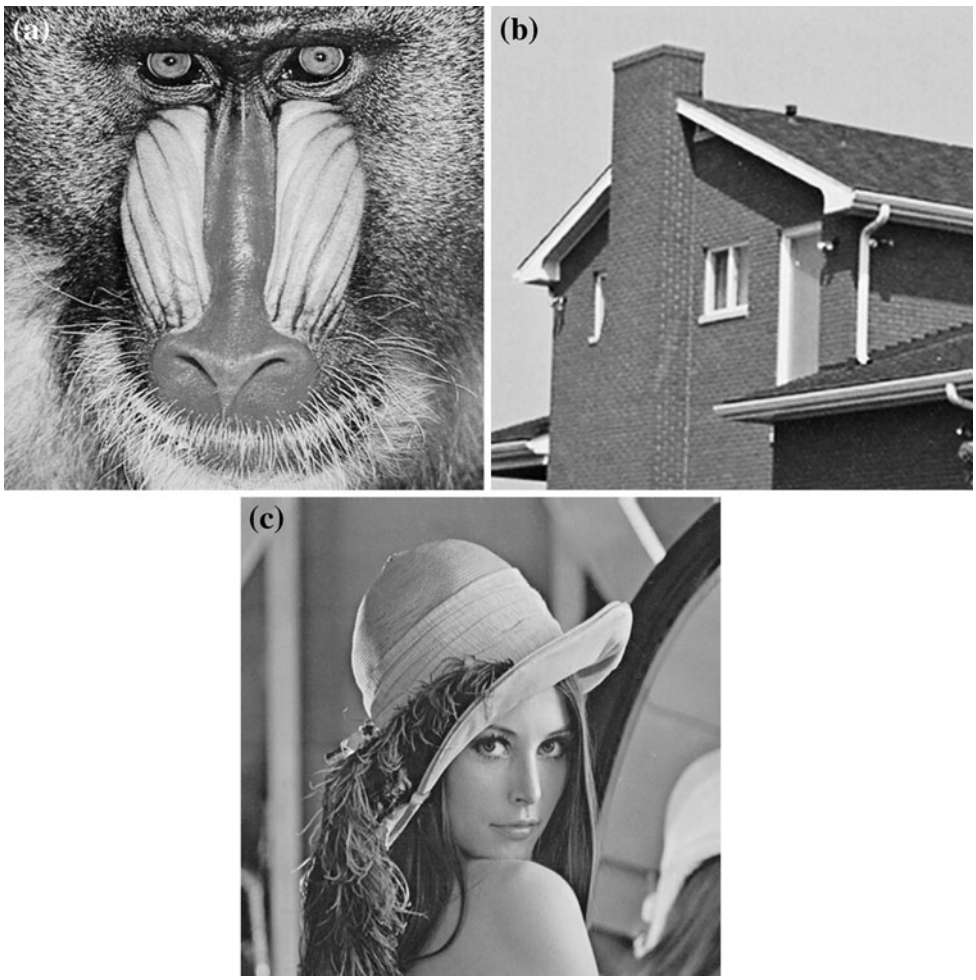

Fig. 6 Test images: **a** baboon, **b** house and **c** Lena

Table 4 CPU elapsed time (s) of the baboon grayscale test image

$N = 128$	ZOA	Papakosta [8]	Proposed method
Max = 1	0.219	0.109	0.032
Max = 5	2.625	1.188	0.125
Max = 15	43.234	9.187	0.438
Max = 25	254.109	26.906	0.954
Max = 35	722.500	57.453	1.672

proposed methods for computing the whole set of OFMMs were compared.

In the first experiment, a 128 × 128 gray-level image of baboon as depicted in Fig. 6a was used. In this experiment, the unit disk was divided into a 64 non-overlapped circular rings. Table 4 shows CPU elapsed times for selected moment orders. This table gives the reader a quick comparison using explicit real values. The complete elapsed CPU times are plotted against the moment order in Fig. 7. It is clear that the proposed method significantly reduced the execution time.

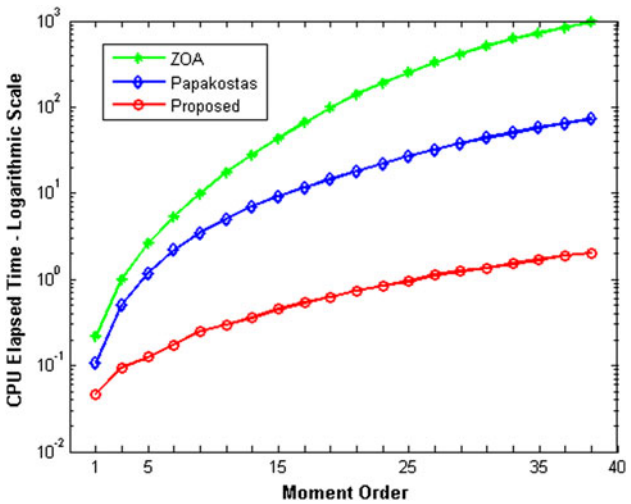

Fig. 7 Elapsed CPU time in seconds for 128 × 128 grayscale image of baboon

In the second numerical experiment, a 256 × 256 grayscale image of "house" in Fig. 6b was used to compute the full set of OFMMs. In this experiment, the elapsed

CPU times are plotted against the moment order in Fig. 8. Similar to the first experiment, the CPU elapsed times are shown in Table 5. It is clear that both direct ZOA and Papakostas methods are time consuming. On the other hand, the proposed method is very fast.

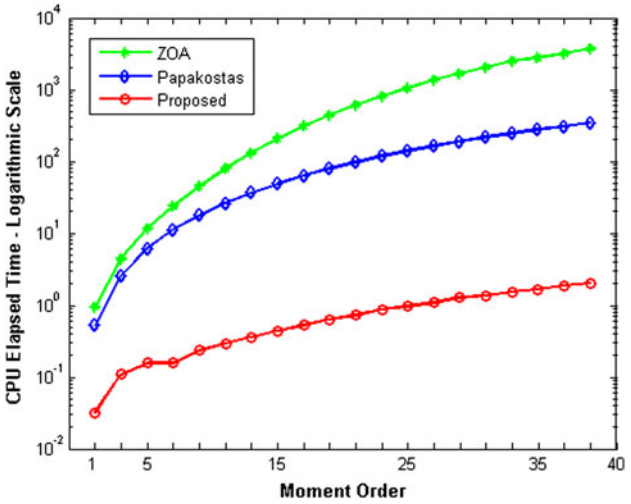

Fig. 8 Elapsed CPU time in seconds for 256 × 256 grayscale image of house

Table 5 CPU elapsed time (s) of the house grayscale test image

N = 256	ZOA	Papakosta [8]	Proposed method
Max = 1	0.953	0.531	0.047
Max = 5	11.625	6.062	0.157
Max = 15	209.015	48.969	0.457
Max = 25	1046.125	139.938	0.969
Max = 35	2778.875	274.718	1.687

4.2 Accuracy

To explain the accuracy of the proposed method, we performed an experiment to ensure this issue. The standard gray-level image of "Lena" of size 128 × 128 is used in our numerical experiment. The mean-square error (MSE) is widely used as a criteria for testing the quality of the digital image [13]. The MSE is computed twice: the first time by using the ZOA method, while the proposed method is used in the second time. Zernike moments of order ranging from 1 to 20 are considered in both computational runs. The graphical representation of the MSE for the different orders of moment are plotted and depicted in Fig. 9. It is clear that the MSE evaluated by the proposed method continually decreased and approached zero value, while the corresponding MSE evaluated by the ZOA method decreased up to a certain value and suddenly increased as the moment order increased. This experiment clearly shows the high accuracy of the proposed method.

The mapping method of Xin et al. [10] was proved to be accurate. The results of the performed experiment by using the proposed modified mapping method confirms this conclusion.

5 Conclusion

This study proposes a new fast, accurate and memory-efficient method for computing 2D OFMMs in polar coordinates. The unit disk is divided into concentric,

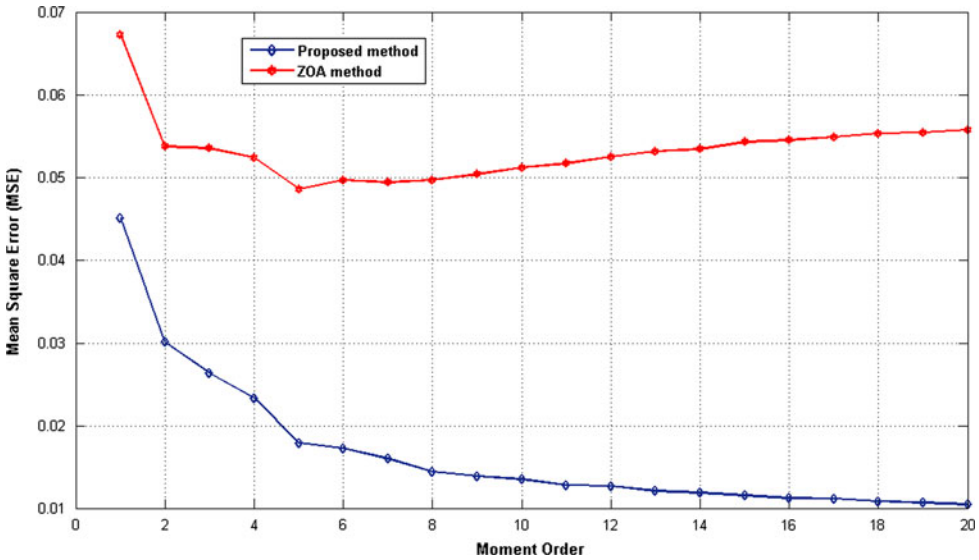

Fig. 9 MSE for 128 × 128 grayscale image of Lena

non-overlapped circular rings. All these rings are sliced to produce a set of equal-area circular sectors, where each of these sectors is represented by only one point at its center. Symmetry property is applied to reduce the computation process by approximately 87%. This formulates a novel computation algorithm. Fast algorithm of separate calculation of OFMMs is applied to accelerate the computation process. The computational time required by the proposed method is extremely small compared to the direct method.

References

1. Sheng, Y., Shen, L.: Orthogonal Fourier–Mellin moments for invariant pattern recognition. J. Opt. Soc. Am. A **11**(6), 1748–1757 (1994)
2. Teague, M.: Image analysis via the general theory of moments. J. Opt. Soc. Am. **70**(8), 920–930 (1980)
3. Terrillon, J.C., McReynolds, D., Sadek, M., Sheng, Y., Akamatsu, S.: Invariant neural-network based face detection with orthogonal Fourier–Mellin moments. In: Proceedings of the 15th IEEE International Conference on Pattern Recognition, vol. 2, pp. 993–1000 (2000)
4. Kan, C., Srinath, M.D.: Invariant character recognition with Zernike and orthogonal Fourier–Mellin moments. Pattern Recognit. **35**(1), 143–154 (2002)
5. Andrew, T.B.J., David, N.C.L.: Integrated wavelet and Fourier–Mellin invariant feature in fingerprint verification system. In: Proceedings of the 2003 ACM SIGMM Workshop on Biometrics Methods and Applications, pp. 82–88 (2003)
6. Wang, X., Xiao, B., Jian-Feng, M., Xiu-Li, B.: Scaling and rotation invariant analysis approach to object recognition based on Radon and Fourier–Mellin transforms. Pattern Recognit. **40**(12), 3503–3508 (2007)
7. Jiu-bin, T., Lei, A., Ji-wen, C., Wen-jing, K., Dan-dan, L.: Subpixel edge location based on orthogonal Fourier–Mellin moments. Image Vis. Comput. **26**(4), 563–569 (2008)
8. Papakostas, G.A., Boutalis, Y.S., Karras, D.A., Mertzios, B.G.: Fast numerically stable computation of orthogonal Fourier–Mellin moments. IET Comput. Vis. **1**(1), 11–16 (2007)
9. Papakostas, G.A., Boutalis, Y.S., Karras, D.A., Mertzios, B.G.: Fast computation of orthogonal Fourier–Mellin moments using modified direct method. In: Proceedings of 6th EURASIP Conference Focused on Speech and Image Processing, Multimedia Communications and the 14th International Workshop of Systems, Signals and Image Processing, pp. 153–156 (2007)
10. Xin, Y., Pawlak, M., Liao, S.: Accurate computation of Zernike moments in polar coordinates. IEEE Trans. Image Process. **16**(2), 581–587 (2007)
11. Hosny, K.M.: Exact and fast computation of geometric moments for gray-level images. Appl. Math. Comput. **189**(2), 1214–1222 (2007)
12. Hosny, K.M.: Fast computation of accurate Zernike moments. J. Real-Time Image Process. **3**(1–2), 97–107 (2008)
13. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 2nd edn. Prentice-Hall, Englewood Cliffs (2002)

Author Biographies

Khalid Mohamed 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 associate professor. He is a member of ACM, IEEE and the IEEE computer society. His research interests include image processing, pattern recognition and computer vision. Dr. Hosny published more than twenty papers in international journals. He is a member of the editorial board of Information Technology Journal and Applied Science Journal. Also, he works as a reviewer for the Image and Vision computing Journal, Information Science Journal, Pattern Recognition Letters, Journal of Real-Time Image Processing and Optics Letters.

Mohamed A. Shouman received his BS degree in Production Engineering, his MS degree in Production Engineering (A study of a general Time/Cost Relation in Network Analysis), and his Ph.D. degree in Production Engineering (On the Optimization of Resource Usage in the Project Network of Cyclic Nature). His research interests include Industrial engineering, Mathematical modeling and computer applications, Network analysis, Project Management, Transportation problem, Implementation and design of simulation models, Inventory models and quality control, Scheduling techniques, CAD/CAM and CAPP techniques, Flexible manufacturing systems, Experimental design techniques, Expert systems and artificial intelligent techniques, Form error evaluation, Facility layout planning and design, Neural network, Intelligent information systems, Optimization techniques, Genetic Algorithms.

Hayam M. Abdel Salam had Received the B.Sc. in computer science from Zagazig University, Zagazig, Egypt in 2005. She is working as teaching assistant in the faculty of computers and informatics. Her fields of interest are image processing, pattern recognition and computer vision.