Pattern Analysis and Applications (2022) 25:731–755
https://doi.org/10.1007/s10044-022-01071-6

THEORETICAL ADVANCES

Accurate quaternion fractional-order pseudo-Jacobi–Fourier moments

Xiangyang Wang[1] · Yuyang Zhang[1] · Jialin Tian[1] · Panpan Niu[1] · Hongying Yang[1]

Received: 10 July 2021 / Accepted: 12 March 2022 / Published online: 5 April 2022

Abstract

Pseudo-Jacobi–Fourier moments (PJFMs) are a set of orthogonal moments which have been successfully applied in the fields of image processing and pattern recognition. In this paper, we present a new set of quaternion fractional-order orthogonal moments for color images, named accurate quaternion fractional-order pseudo-Jacobi–Fourier moments (AQFPJFMs). We initially propose a fast and accurate algorithm for the PJFMs computation of an image using new recursive approach and polar pixel tiling scheme. Then, we define a new set of orthogonal moments, named accurate fractional-order pseudo-Jacobi–Fourier moments, which is characterized by the generic nature and time–frequency analysis capability. We finally extend the gray-level fractional-order PJFMs to color images and present the quaternion fractional-order pseudo-Jacobi–Fourier moments. In addition, we develop a new color image representation for enhancing simultaneously the discriminability and robustness, called mixed low-order moments feature. We conduct extensive experiments to evaluate the performance of the proposed AQFPJFMs, in which the encouraging results demonstrate the efficacy and superiority of the proposed scheme.

Keywords Pseudo-Jacobi–Fourier moments · Recursive method · Fractional-order moments · Polar pixel tiling scheme · Quaternion · Zero-watermarking

Abbreviations

AFPJFMs	Accurate fractional-order pseudo-Jacobi–Fourier moments
APJFMs	Accurate fractional-order pseudo-Jacobi–Fourier moments
AQFPJFMs	Accurate quaternion fractional-order pseudo-Jacobi–Fourier moments
AR	Accuracy rate
BER	Bit error rate
CHFMs	Chebyshev–Fourier moments
FCMs	Fractional-order Chebyshev moments
FJFMs	Fractional-order Jacobi–Fourier moments
FOFMMs	Fractional-order orthogonal Fourier–Mellin moments
FPHFMs	Fractional-order polar harmonic Fourier moments
FPHT	Fractional-order polar harmonic transforms
FPR	False positive ratio
FQEMs	Fractional-order quaternion exponential moments
FQZMs	Fractional-order orthogonal quaternion Zernike moments
JFMs	Jacobi–Fourier moments
LFMs	Legendre–Fourier moments
MLMF	Mixed low-order moments feature
MSRE	Mean square reconstruction error
OFMMs	Orthogonal Fourier–Mellin moments
OMs	Orthogonal moments
PJFMs	Pseudo-Jacobi–Fourier moments
PSNR	Peak signal-to-noise ratio
PZMs	Pseudo-Zernike moments
QRFCMs	Quaternion radial fractional Charlier moments
RST	Rotation, scaling, translation
ZMs	Zernike moments
ZOA	Zero-order approximation

✉ Xiangyang Wang
wxy37@126.com

✉ Hongying Yang
yhy_65@126.com

[1] School of Computer and Information Technology, Liaoning Normal University, Dalian 116029, Liaoning, People's Republic of China

1 Introduction

How to effectively describe an image is a vital issue in image processing. It means that a small set of descriptions represents an image. However, compared with the original image, the recognized image will have a certain degree of distortion (for example, translation, and rotation). Therefore, moments are proposed to solve this question which are not sensitive to various distortions or noises and have invariant characteristics. Because the invariant moments highly summarize the image features and have a variety of invariance, moments have been widely used in scene analysis, image classification, and image pattern recognition.

1.1 Brief review of moments

As early as 1962, Hu [1] proposed using geometric moments to describe images. However, with the deepening of research, scientists found that low-order geometric moments do not adequately describe the details of the image, while high-order geometric moments have lower robustness. Therefore, the geometric moments are not easy to reconstruct the original image. In 1980, Teague [2] proposed orthogonal moments (OMs) to solve the shortcomings of geometric moments. OMs have excellent image reconstruction ability and low sensitivity to noise invariance [3], so they are widely used in object recognition applications. The most common orthogonal moment is the Zernike moments (ZMs) proposed by Teague [2] in 1980. Although ZMs have good performance in information redundancy, noise sensitivity and image description ability [4], it is challenging to describe smaller images. In 1994, Sheng [5] proposed Orthogonal Fourier-Mellin moments (OFMMs). OFMMs are better than ZMs in describing smaller images, but the values of radial polynomials of OFMMs are large at $r = 0$, OFMMs did not solve the problem. Many experiments [5–7] have proved that the radial basis functions' zeros are related to the moments' descriptive ability. Subsequently, Ping [6] proposed Chebyshev–Fourier moments (CHFMs), but CHFMs still failed to overcome the shortcomings of OFMMs. Lately, Ping [7] proposed Jacobi–Fourier moments (JFMs) based on Jacobi polynomials. This kind of moment is a generalized orthogonal moment; ZMs [8], pseudo-Zernike moments (PZMs) [9], pseudo-Jacobi–Fourier moments (PJFMs) [10], OFMMs, CHFMs, and Legendre–Fourier moments (LFMs) [11] are all special cases of JFMs. The almost radial basis functions of JFMs have n zeros in the interval $(0, 1)$, and the zeros' distribution in this interval is practically uniform. However, PJFMs have $(n + 2)$ zeros in the special cases of JFMs. Compared with the basis functions of other moments based on Jacobi polynomials, PJFMs added two zeros. The distribution of PJFMs' zeros is more uniform, and the first zero is very close to the origin. Therefore, PJFMs are more suitable for describing and recognizing multi-distortion invariant images [12].

1.2 State of the art and motivation

The research directions of traditional orthogonal moments mainly include the following five aspects: (1) accurate calculation; (2) fast calculation; (3) robustness/ invariance optimization; (4) definition extension; (5) application [13]. The ideal moments should have outstanding performance in all aspects. However, the existing moments can only be optimized in a few aspects. There will always be shortcomings in some aspects compared with the ideal moments. Therefore, how to design the more ideal moments is still a problem to be solved.

The direct calculation method of traditional OMs will produce two kinds of errors: geometric errors and numerical integration errors [14]. These errors affect the accuracy of the image reconstruction ability. Wee [15] minimized geometric errors by circular mapping. This method mapped the entire square image to the unit disk and calculated integrals separately like precise geometric moments. However, if the moment's order is bigger than 50th, this method will appear a numerically unstable phenomenon. Also, the calculation time of the moments will increase with the order increase. To reduce the calculation time, many mathematicians have proposed some algorithms to accelerate the calculation of orthogonal moments [16–19]. The most common acceleration method is the recursive method to calculate the radial basis functions' coefficients, this method can not only reduce the time complexity, but also improve the numerical stability. However, studies have found that using the recursive method [20] to calculate high-order polynomials can also cause errors. Xin [21] proposed a high-precision numerical calculation algorithm for ZMs, which significantly improved image reconstruction ability. First, this algorithm re-divides the pixels with a specific strategy; then, the image is generated through interpolation in the polar coordinates. After that, the integral is calculated directly on the image in polar coordinates. This method eliminates geometric errors and can calculate integrals better. Inspired by this, Camacho-Bello [22] combined the polar pixel tiling scheme and recursive methods, and the proposed method has an excellent performance in terms of speed and accuracy.

In recent years, people have attached great importance to using more accurate orthogonal polynomials fractional differential equations [23–26]. They introduce a fractional-order parameter $\alpha > 0$ to extend the conception of integer-order N to fractional-order αn $(n \in N)$. The classical orthogonal polynomials can be acquired by setting $\alpha = 1$. This new kind of fractional orthogonal polynomials can control the zeros' distribution by changing the fractional parameter and solve the problem of information suppression. Compared with traditional orthogonal moments that can only reflect global information, fractional orthogonal moments can better describe the image information of the region of interest. Therefore, it is also applied to the new radial basis functions to define the fractional orthogonal moments. Zhang [27] defined fractional-order orthogonal Fourier–Mellin moments (FOFMMs). Benouini [28] defined fractional-order Chebyshev moments (FCMs). Yang [29] defined fractional-order Jacobi–Fourier moments (FJFMs), Hosny [30] defined fractional-order polar Harmonic transform (FPHT). Wang [31] defined fractional-order polar harmonic Fourier moments (FPHFMs). To make the fractional orthogonal moments not only limited to grayscale images, Chen [32] proposed fractional-order orthogonal quaternion Zernike moments (FQZMs) to calculate color images. Wang [33] proposed fractional-order quaternion exponential moments (FQEMs), Yamni [34] proposed novel quaternion radial fractional Charlier moments (QRFCMs).

1.3 Contribution of this paper

In order to design an ideal moment descriptor, we propose a new algorithm called quaternion accurate fractional-order Pseudo-Jacobi–Fourier moments (AQFPJFMs) in this paper. Compared with PJFMs, AQFPJFMs have improved accuracy, numerical stability and applicability.

The contributions of this paper are as follows:

- We propose a new recursive method of PJFMs. This new recursive relationship can increase the calculation speed and reduce the numerical instability of high-order radial basis functions.
- Based on the recursive relationship and polar pixel tiling scheme between radial basis functions, accurate Pseudo-Jacobi–Fourier moments (APJFMs) are realized. The polar pixel tiling scheme can cancel geometric and numerical integration errors, so we can reconstruct high-order images better.
- To show more details, we extended APJFMs to accurate fractional-order pseudo-Jacobi–Fourier moments (AFPJFMs). Due to introducing the fractional-order parameter, we can adjust the distribution of radial basis functions' zeros by altering the fractional-order parameter according to different application backgrounds.
- We use quaternion to calculate color images, then we get AQFPJFMs. AQFPJFMs reflect the color information and the correlation between the color channels. Therefore, AQFPJFMs have better image representation and distinguishing capabilities.
- To making full use of the time–frequency analysis capabilities of AQFPJFMs, we propose a scheme that can simultaneously enhance the discriminability and robustness of global features, called mixed low-order moments feature (MLMF).
- We apply AQFPJFMs and MLMF to zero-watermarking. Comparing our proposed algorithm with other advanced algorithms, the test results prove that our advanced algorithm has significant advantages in robustness and discriminability.

The structure of this paper is as follows. In Sect. 2, first, we will propose the recursive method of PJFMs; second, we will use the polar pixel tilling scheme to eliminate error, then we will get APJFMs. In Sect. 3, we will define AFPJFMs and discuss AFPJFMs' properties. In Sect. 4, we will use quaternion to calculate color images, this algorithm is called AQFPJFMs. In Sect. 5, we will define MLMF and discuss the time–frequency analysis capabilities of MLMF. In Sect. 6, we will show a lot of experimental results of AQFPJFMs and apply MLMF to the zero-watermarking algorithm. Finally, Sect. 7 will summarize our work and point out the direction of future work.

2 Accurate pseudo-Jacobi–Fourier moments

In this section, first, we will review PJFMs; second, we will propose the recursive method of PJFMs; finally, we will define APJFMs.

2.1 Definition

The classical PJFMs' radial basis functions $J_n(r)$ in the polar coordinate system (r, θ) are defined as:

$$\begin{aligned} J_n(r) &= \left[\frac{w(r)}{rb_n}\right]^{\frac{1}{2}} G_n(r) \\ &= (-1)^t \left[\frac{(2n+4)}{(n+3)(n+1)}(1-r)r\right]^{\frac{1}{2}} \\ &\quad \sum_{t=0}^{n} (-1)^t \frac{(n+t+3)!}{(n-t)!t!(t+2)!} r^t, \end{aligned} \tag{1}$$

where $G_n(r)$ represents the Jacobi polynomials with parameters $p = 4, q = 3$ [21], $w(r)$ is the weighting function and b_n are normalization factors:

$$G_n(r)=\sum_{t=0}^{n}(-1)^t\frac{\Gamma(4+n+t)}{(n-t)!t!\Gamma(3+t)}r^t,$$
$$w(r)=(1-r)\cdot r^2,\quad b_n=\frac{(n+3)(n+1)}{(2n+4)}. \tag{2}$$

The Jacobi polynomials $G_n(r)$ satisfies the normalization condition:

$$\int_0^1 G_n(r)G_{n'}(r)w(r)\mathrm{d}r=b_n\delta_{nn'}, \tag{3}$$

δ is the Kronecker delta. Similar:

$$\int_0^1 J_n(r)J_{n'}(r)r\mathrm{d}r=\delta_{nn'}. \tag{4}$$

Therefore, the classic PJFMs can be defined as [10] with order $n \in N$ and repetition $m \in Z$:

$$\begin{aligned}C_{nm}&=\frac{1}{2\pi}\int_0^1\int_0^{2\pi}V_{nm}^*(r,\theta)f(r,\theta)r\mathrm{d}r\mathrm{d}\theta\\&=\frac{1}{2\pi}\int_0^1\int_0^{2\pi}J_n(r)\exp(-jm\theta)f(r,\theta)r\mathrm{d}r\mathrm{d}\theta.\end{aligned} \tag{5}$$

$f(r,\theta)$ can be approximated reconstruction by classical PJFMs and basis functions:

$$\begin{aligned}f(r,\theta)&=\sum_{n=0}^{+\infty}\sum_{m=-\infty}^{+\infty}C_{nm}\cdot V_{nm}(r,\theta)\\&\approx\sum_{n=0}^{K}\sum_{m=-K}^{K}C_{nm}\cdot V_{nm}(r,\theta).\end{aligned} \tag{6}$$

Some visual illustrations of the kernels are given in Fig. 1.

2.2 Recursive computation

The Jacobi polynomials $G_n(r)$ involve multiple factorials and gamma functions; therefore, the time complexity in the calculation process is relatively large, and numerical instability may overflow when calculating high-order images. To solve this problem, referring to [35, 36], we will put forward a recursive calculation method of the PJFMs' radial basis functions $J_n(r)$. This method has the following two advantages:

2.2.1 The time complexity is low

PJFMs directly calculate K times $J_n\left(\sqrt{x_i^2+y_j^2}\right)$ with a size of $N \times N$ digital image requiring $O(K^2N^2)$ additions; PJFMs recursively calculate K times $J_n\left(\sqrt{x_i^2+y_j^2}\right)$ with a size of $N \times N$ digital image requiring $O(KN^2)$ additions.

2.2.2 Numerical stability is better

When PJFMs directly calculate double-precision numbers, the factorial term $a!$ is accurate while $a \le 21$. When $a > 21$, the direct calculation method of PJFMs will be numerically unstable. [10] However, the recursive calculation method of PJFMs we proposed does not involve the factorial (or gamma function) of large numbers, so it has better numerical stability.

Based on Eq. (2), let:

$$P_n(r)=\sum_{t=0}^{n}(-1)^t\frac{(n+t+3)!}{(n-t)!t!(t+2)!}r^t. \tag{7}$$

Using the property that $a!=a\cdot(a-1)!$ and $\Gamma(a)=a\cdot\Gamma(a-1)$, $P_n(r)$ can be computed recursively:

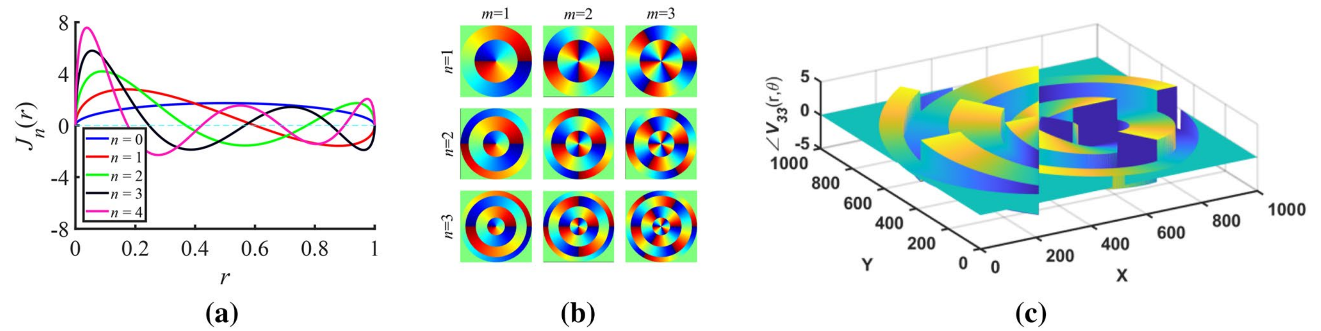


Fig. 1 Some examples of the PJFMs' kernels. **a** PJFMs' radial kernels, **b** the 2D view of PJFMs' phase angle $\angle V_{nm}(r,\theta)=\arctan(\mathrm{Im}(V_{nm}(r,\theta))/\mathrm{Re}(V_{nm}(r,\theta)))$ (The color from blue to yellow indicates that the phase angle is from small to large), **c** The 3D view of PJFMs' phase angle $\angle V_{nm}(r,\theta)=\arctan(\mathrm{Im}(V_{nm}(r,\theta))/\mathrm{Re}(V_{nm}(r,\theta)))$. $(n=3, m=3,$ X and Y are the horizontal and vertical coordinates of the image)

$$P_n(r) = (L_1 r + L_2)P_{n-1}(r) + L_3 P_{n-2}(r), \quad n \geq 2. \tag{8}$$

where

$$\begin{aligned} L_1 &= \frac{-(2n+3)(2n+2)}{n(2+n)}, \\ L_2 &= (2n+2) + \frac{(n-1)(1+n)}{2n+1} L_1, \\ L_3 &= \frac{(2n)(2n+1)}{2} + \frac{(n)(n-2)}{2} L_1 - (2n)L_2. \end{aligned} \tag{9}$$

and the initial values of $P_n(r)$ are:

$$P_0(r) = \frac{\Gamma(4)}{\Gamma(3)}, \quad P_1(r) = \frac{\Gamma(5)}{\Gamma(3)}\left(1 - \frac{5}{3}r\right). \tag{10}$$

2.3 Accurate pseudo-Jacobi–Fourier moments

2.3.1 Error analysis in Cartesian coordinates

In the traditional moment calculation method, the direct calculation method in the Cartesian coordinate system is based on the zero-order approximation (ZOA) discrete calculation method. We define the discrete digital image which size is $N \times N$ as $f[i,j]$, $(i,j) \in [1, 2, \ldots, N]^2$. This method defines the Cartesian coordinate system (x, y) by the integral variable in Eq. (5):

$$C_{nm} = \frac{1}{2\pi} \iint_{x^2+y^2 \leq 1} V^*_{nm}\left(\sqrt{x^2+y^2}, \arctan\left(\frac{y}{x}\right)\right) f(x, y)\mathrm{d}x\mathrm{d}y. \tag{11}$$

The discrete digital image cannot be used directly; therefore, we need to map the domain $[1,\ 2,\ \ldots,\ N]^2$ of the discrete digital image to the unit circle $[-1,\ 1]^2$, where $(x_i, y_j) \in [-1,\ 1]^2$:

$$\begin{cases} x_i = (i - N/2) \cdot (2/N) \\ y_j = (j - N/2) \cdot (2/N) \end{cases}, \quad \begin{cases} \Delta x = 2/N \\ \Delta y = 2/N \end{cases}. \tag{12}$$

PJFMs' discrete calculation method according to Eqs. (11) and (12) can be defined as:

$$C_{nm} = \frac{1}{2\pi} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \psi_{nm}(x_i, y_j) f(x_i, y_j). \tag{13}$$

x_i and y_j meet the condition $x_i^2 + y_j^2 \leq 1$, then:

$$\psi_{nm}(x_i, y_j) \approx \int_{x_i - \frac{\Delta}{2}}^{x_i + \frac{\Delta}{2}} \int_{y_j - \frac{\Delta}{2}}^{y_j + \frac{\Delta}{2}} V_{nm}\left(\sqrt{x_i^2 + y_j^2}, \arctan\left(\frac{y_j}{x_i}\right)\right). \tag{14}$$

The width of the integral unit area is $\Delta = 2/N$. In the Cartesian coordinate system, the exact value of Eq. (14) cannot be obtained, but the estimated value can only be obtained from Eq. (15), so it will result in a numerical integration error:

$$\psi_{nm}(x_i, y_j) \approx \Delta^2 V_{nm}\left(\sqrt{x_i^2 + y_j^2}, \arctan\left(\frac{y_j}{x_i}\right)\right). \tag{15}$$

Substituting the estimated value into Eq. (13), the discrete form of PJFMs are obtained:

$$C_{nm} = \frac{2}{\pi N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} V_{nm}\left(\sqrt{x_i^2 + y_j^2}, \arctan\left(\frac{y_j}{x_i}\right)\right) f(x_i, y_j). \tag{16}$$

As shown in Fig. 2, only when the total points are in the unit circle can they be substituted. Therefore, when using Eq. (20) to perform the discrete calculation of PJFMs in the Cartesian coordinate system, numerical integration errors will occur from the above derivation process.

2.3.2 Polar coordinate division and pixel interpolation

To eliminate geometric errors, the method we proposed should satisfy: All pixels together exactly cover the unit circle, and the pixel areas do not intersect. It can be expressed as the following equation, where each pixel can be represented by $\Omega_{u,v}$, and D represented the unit circle:

$$\bigcup_{(u,v)} \Omega_{uv} = D, \tag{17}$$

$$\Omega_{uv} \cap \Omega_{u'v'} = \Phi \ , \forall (u, v) \neq (u', v'). \tag{18}$$

The problem of eliminating geometric errors and integration errors now comes down to rearranging the polar coordinates in the calculation process. The following segmentation method can satisfy Eqs. (17) and (18) well. Figure 3a shows the result of the division method.

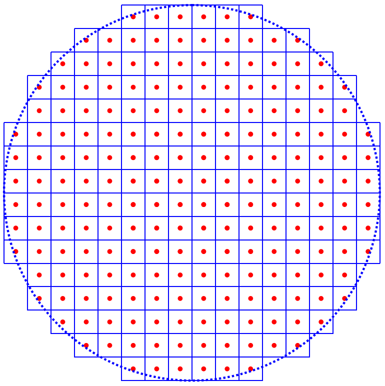


Fig. 2 Pixels in the Cartesian coordinate

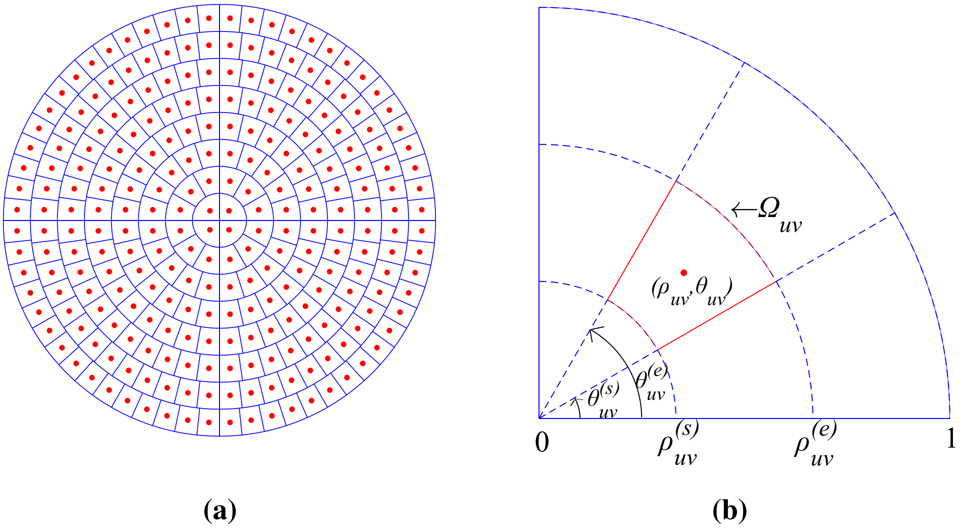


Fig. 3 **a** PJFMs' pixel distribution in the polar coordinates, **b** a pixel $\Omega_{u,v}$ of PJFMs in the polar coordinate

Assuming that the area size of PJFMs needs to be calculated as $N \times N$, the radius of the circular area of PJFMs is defined by PJFMs as $N/2$, the division method will be presented as follows:

- The unit circle is equally divided into U layer ring regions with equidistant concentric circles along the radial direction. The ring closest to the center of the circle is marked as the first layer, which increases outward in sequence.
- The unit circle is equally divided into V sectors. This process divides all the circles into V regions equally.
- Divide all the circles from the inside to the outside in sequence, and divide the uth ring equally into $S_u = (2u - 1)\ V$ pixel areas.

We can get the original image through the mentioned above division method to obtain the pixel division in polar coordinates. The unit circle image after division is divided into VU^2 sectors, the size of each sector is π/VU^2. The value of U and V will affect the amount of calculation and calculation accuracy; therefore, the number of pixels and calculation amount will increase with the increase of VU^2. Then more accurate image moments can be obtained. However, when the number of divided pixels exceeds the pixels in the original image, increasing the pixels' number will significantly increase the time and amount of calculation and thus lose meaning. We take $U = N/2$, $V = 4$, so that we can get roughly the same number of pixels as the original image. We define the divided pixel label as $\Omega_{u,v}$ $(u = 1, 2, 3, \dots, U$ $; v = 1, 2, 3, \dots, V(2u - 1))$. Figure 3b shows the schematic diagram of $\Omega_{u,v}$. The pixel point $\Omega_{u,v}$ under the polar coordinate division represents the pixel area converted to the polar coordinate (ρ_{uv}, θ_{uv}), the start and end radius of $\Omega_{u,v}$ are $\rho^{(s)}_{uv}$ and $\rho^{(e)}_{uv}$. The start and end angles of the pixel area of $\Omega_{u,v}$ are $\theta^{(s)}_{uv}$ and $\rho^{(e)}_{uv}$, so (ρ_{uv}, θ_{uv}) can be defined as:

$$\rho_{uv} = \left(\rho^{(s)}_{uv} + \rho^{(e)}_{uv}\right)/2, \quad \theta_{uv} = \left(\theta^{(s)}_{uv} + \theta^{(e)}_{uv}\right)/2. \tag{19}$$

The digital image is usually defined in Cartesian coordinates, and this pixel position cannot directly correspond to the gray value of the original image. Therefore, we use the pixel sampling method to solve this question, then the gray value of the new pixel can be estimated through the points in the original image. The sampling method we use in this article is the nearest neighbor interpolation. It is also suggested to choose other more complex nonlinear interpolation methods [14] to improve accuracy.

2.3.3 Accurate computation of fractional-order pseudo-Jacobi–Fourier moments

After completing the polar coordinate pixel sampling method, the image f' in polar coordinates is obtained. The discrete PJFMs calculation method in polar coordinates can be obtained as follows:

$$\hat{C}_{nm} = \frac{1}{2\pi} \sum_{u} \sum_{v} f'(\rho_{uv}, \theta_{uv}) \omega_{n,m}(\rho_{uv}, \theta_{uv}). \tag{20}$$

$f'(r_u, \theta_{uv})$ is the approximate interpolation of $f(r_{i,j}, \theta_{i,j})$ in the pixel block $\Omega_{u,v}$. Therefore, the new complete orthogonal basis functions can be defined as:

$$\begin{aligned} \omega_{n,m}(\rho_{uv}, \theta_{uv}) &= \iint_{\Omega_{u,v}} V^{*}_{n,m}(\rho, \theta) \rho \mathrm{d}\rho \mathrm{d}\theta \\ &= \int_{\rho^{(s)}_{uv}}^{\rho^{(e)}_{uv}} J_n(\rho) \rho \mathrm{d}\rho \int_{\theta^{(s)}_{uv}}^{\theta^{(e)}_{uv}} e^{-jm\theta} \mathrm{d}\theta, \end{aligned} \tag{21}$$

$$\int_{\theta_{uv}^{(s)}}^{\theta_{uv}^{(e)}} e^{-jm\theta}\mathrm{d}\theta = \begin{cases} \frac{j}{m}\left[e^{-jm\theta_{uv}^{(e)}} - e^{-jm\theta_{uv}^{(s)}}\right], & m \neq 0 \\ \theta_{uv}^{(e)} - \theta_{uv}^{(s)}, & m = 0 \end{cases}. \tag{22}$$

Studies have found that recursive methods [20] to calculate high-order polynomials can also produce numerical instability. Many scholars have used Gaussian integration for the accurate calculation of orthogonal moments [37, 38]. Therefore, on the basis of the recursive calculation, we use the ten-point integration [39] to calculate the integral. This method can maintain numerical stability when calculating the integral of high-order polynomials.

The compound Gauss quadrature rule of radial basis functions integration can be expressed as:

$$\begin{aligned}&\int_{\rho_{uv}^{(s)}}^{\rho_{uv}^{(e)}} J_n(\alpha,\rho)\rho\mathrm{d}\rho \\ &= \frac{\rho_{uv}^{(e)} - \rho_{uv}^{(s)}}{2}\sum_{k=1}^{10}\eta_k J_n\left(\frac{\rho_{uv}^{(e)} - \rho_{uv}^{(s)}}{2}z_k + \frac{\rho_{uv}^{(e)} + \rho_{uv}^{(s)}}{2}\right),\end{aligned} \tag{23}$$

η_k is the weight and z_k is the sampling point. Next, Table 1 will give the ten-point Gaussian integral node table. As described above, in conjunction with Eqs. (20)–(23), we obtained the accurate fractional-order Pseudo-Jacobi–Fourier moments (APJFMs) and do not introduce any geometric errors or numerical integration errors during calculation.

3 Accurate fractional-order pseudo-Jacobi–Fourier moments

To analyze APJFMs' characteristics in the fractional order, we give a general definition of AFPJFMs and discuss orthogonality, recursion, and invariance. Also, we investigate the influence of the fractional-order parameter on the zeros' distribution and the ROI feature extraction ability.

Table 1 η_k and z_k for ten-point Gaussian integral node

k	η_k	z_k
①	0.066671344	− 0.973906529
②	0.149451349	− 0.865063367
③	0.219086363	− 0.679409568
④	0.269266719	− 0.433395394
⑤	0.295524225	− 0.148874339
⑥	0.295524225	0.148874339
⑦	0.269266719	0.433395394
⑧	0.219086363	0.679409568
⑨	0.149451349	0.865063367
⑩	0.066671344	0.973906529

3.1 Definition

Let the fractional-order parameter α be a positive rational number $\alpha \in R^+$. Let the variable $r = r^\alpha$ ($r \in [0,\ 1]$), accurate fractional-order recursive Pseudo-Jacobi radial basis functions can be obtained (refer to "Appendix A"):

$$\begin{aligned} J_n(\alpha, r) &= J_n(r^\alpha) \\ &= (-1)^n\left[\frac{(2n+4)}{(n+3)(n+1)}(1-r^\alpha)\alpha r^{3\alpha-2}\right]^{\frac{1}{2}} \\ &\quad \times P_n(\alpha, r), \end{aligned} \tag{24}$$

here

$$P_n(\alpha, r) = (L_1 t + L_2)P_{n-1}(\alpha, r) + L_3 P_{n-2}(\alpha, r), \quad n \geq 2, \tag{25}$$

L_1, L_2, L_3 are the same as Eq. (9).

The recursive initial values of $P_n(\alpha, r)$ are:

$$P_0(\alpha, r) = \frac{\Gamma(4)}{\Gamma(3)}, \quad P_1(\alpha, r) = \frac{\Gamma(5)}{\Gamma(3)}\left(1 - \frac{5}{3}r^\alpha\right). \tag{26}$$

The radial basis functions satisfies the orthogonal condition:

$$\int_0^1 J_n(\alpha, r)J_{n'}(\alpha, r)r\mathrm{d}r = \delta_{nn'}. \tag{27}$$

Based on the above, we define AFPJFMs with repetition of $m \in Z$ and order of $n \in N$ as:

$$\begin{aligned} \tilde{C}_{nm} &= \frac{1}{2\pi}\int_0^1\int_0^{2\pi} V_{nm}^*(\alpha, r, \theta)f(r,\theta)r\mathrm{d}r\mathrm{d}\theta \\ &= \frac{1}{2\pi}\int_0^1\int_0^{2\pi} J_n(\alpha, r)\exp(-jm\theta)f(r,\theta)r\mathrm{d}r\mathrm{d}\theta. \end{aligned} \tag{28}$$

Similarly, the image can be reconstructed by AFPJFMs and basis functions as:

$$\begin{aligned} f(r,\theta) &= \sum_{n=0}^{+\infty}\sum_{m=-\infty}^{+\infty}\tilde{C}_{nm}\cdot V_{nm}(\alpha, r, \theta) \\ &\approx \sum_{n=0}^{K}\sum_{m=-K}^{K}\tilde{C}_{nm}\cdot V_{nm}(\alpha, r, \theta). \end{aligned} \tag{29}$$

3.2 Zeros of radial kernels

The position and number of the radial basis functions' zeros are related to the moments' ability to capture high-frequency information [5–7]. The zeros distribution is also a fundamental property because it is related to information

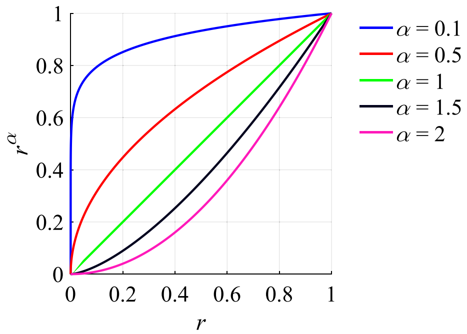


Fig. 4 Schematic diagram of variable substitution $r = r^\alpha$

suppression. It means that calculated moments emphasize certain parts while ignoring other factors of the image. Unlike classical PJFMs, the fractional-order parameter can control the AFPJFMs' zeros distribution, which shows that the focus described by AFPJFMs can be transferred to a more resolving image area. We call it local/ROI feature extraction. This ability comes from the variable replacement $r = r^\alpha$ in Eq. (24). Figure 4 shows the relationship between r^α and r. Figures 5, 6, 7 show some visual examples of radial kernels $J_n(\alpha, r)$. In summary, we have the following conclusions on the zeros' distribution and ROI feature extraction of AFPJFMs:

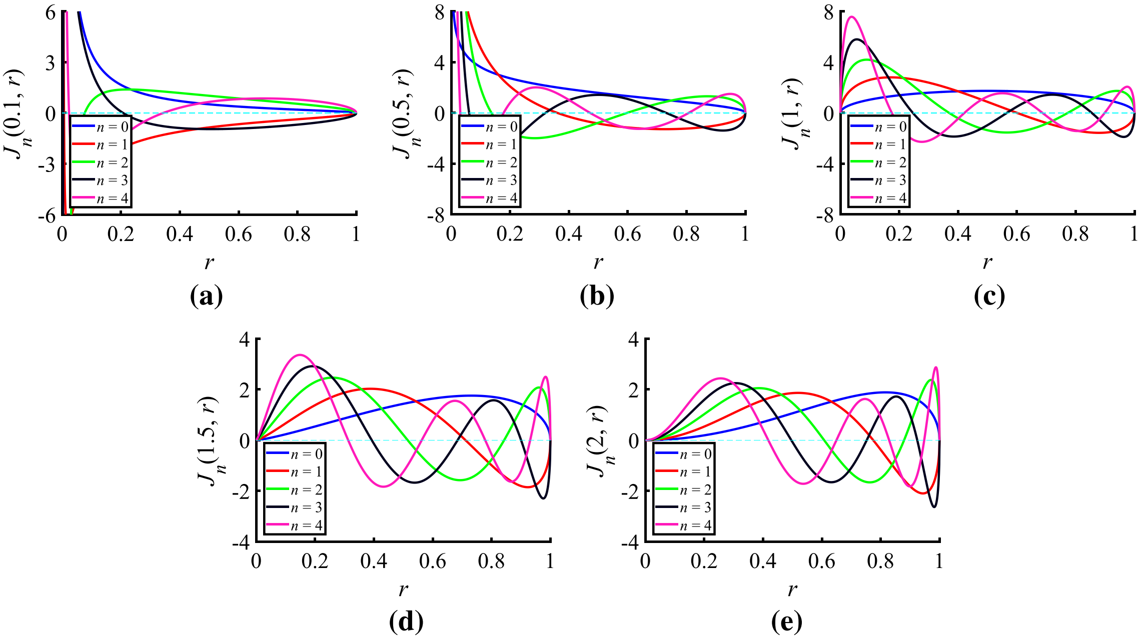


Fig. 5 AFPJFMs' radial kernels with $\alpha = 0.1,\ 0.5,\ 1,\ 1.5,\ 2$ from **a**–**e** when $\alpha = 1$,the zeros are uniform. When $\alpha < 1$ or $\alpha > 1$,the zeros biased toward 0 or 1

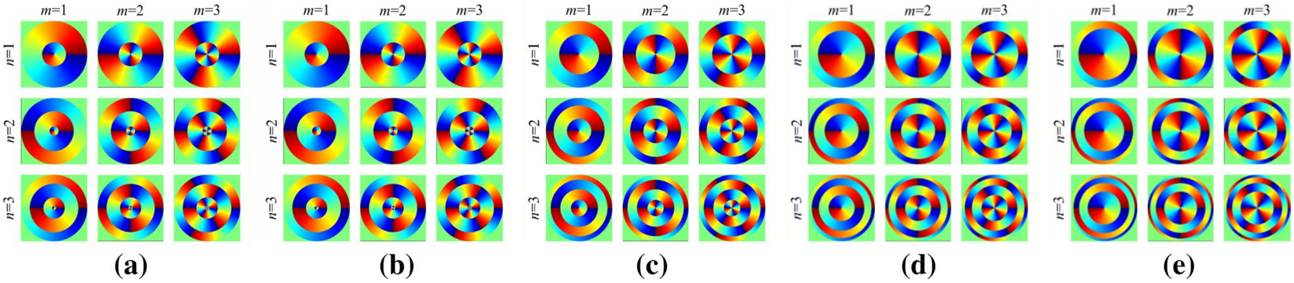
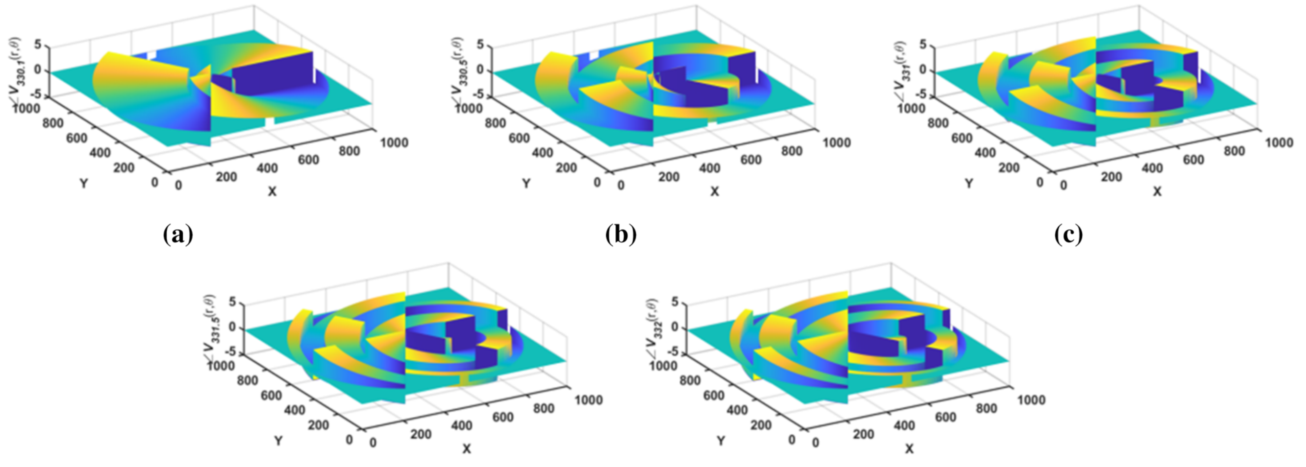


Fig. 6 The 2D view of AFPJFMs' phase angle $\angle V_{nm\alpha}(r, \theta) = \arctan(\mathrm{Im}(V_{nm\alpha}(r, \theta))/\mathrm{Re}(V_{nm\alpha}(r, \theta)))$ with $\alpha = 0.1,\ 0.5,\ 1,\ 1.5,\ 2$ from **a**–**e** (The color from blue to yellow indicates that the phase angle is from small to large)



(a) (b) (c)

(d) (e)

Fig. 7 The 3D view of AFPJFMs' phase angle $\angle V_{nm\alpha}(r,\theta) = \arctan(\mathrm{Im}(V_{nm\alpha}(r,\theta))/\mathrm{Re}(V_{nm\alpha}(r,\theta)))$ with $\alpha = 0.1,\ 0.5,\ 1,\ 1.5,\ 2$ from **a**–**e** ($n = 3, m = 3$, X and Y are the horizontal and vertical coordinates of the image)

- When $\alpha = 1$, the distribution of radial kernels' zeros is basically uniform. At this time, the moment value describes the entire image area equally.
- When $\alpha < 1$, there will be $r = r^\alpha > r$ in the interval $(0,\ 1)$ and move the radial kernels's zeros to the $r = 0$ direction. The moments value emphasizes the information in the central area of the image.
- When $\alpha > 1$, there will be $r = r^\alpha < r$ in the interval $(0,\ 1)$ and move the radial kernels' zeros to the $r = 1$ direction. The moments value emphasizes the information in the outer area of the image.

4 Accurate quaternion fractional-order pseudo-Jacobi–Fourier moments

In [10, 35, 36, 40], PJFMs are only applied to image processing of the grayscale images and cannot be directly applied to color images. Therefore, in this section, we will extend AFPJFMs to the color image domain and propose accurate quaternion fractional-order Pseudo-Jacobi–Fourier moments (AQFPJFMs) to suit color image processing. AQFPJFMs reflect the color information and the correlation between the color channels. Therefore, AQFPJFMs have better image representation and distinguishing capabilities.

4.1 Definition

Mathematically, quaternion [41] is an extended number system of the complex number system, which is usually expressed in the following form:

$$\boldsymbol{q} = a + b\boldsymbol{i} + c\boldsymbol{j} + d\boldsymbol{k}. \tag{30}$$

$a,\ b,\ c,\ d$ are quaternion real parts, and $\boldsymbol{i},\ \boldsymbol{j},\ \boldsymbol{k}$ are quaternion imaginary parts. We list some basic operation rules of the quaternion imaginary unit below:

$$\begin{cases} \boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = -1 \\ \boldsymbol{ij} = -\boldsymbol{ji} = \boldsymbol{k} \\ \boldsymbol{jk} = -\boldsymbol{kj} = \boldsymbol{i} \\ \boldsymbol{ki} = -\boldsymbol{ik} = \boldsymbol{j} \end{cases}. \tag{31}$$

The quaternion conjugate and modulus are:

$$\boldsymbol{q}^* = a - b\boldsymbol{i} - c\boldsymbol{j} - d\boldsymbol{k}, \quad |\boldsymbol{q}| = \sqrt{a^2 + b^2 + c^2 + d^2}. \tag{32}$$

Given any two quaternions $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$, they satisfy the following properties:

$$(\boldsymbol{q}_1 \cdot \boldsymbol{q}_2)^* = \boldsymbol{q}_2^* \cdot \boldsymbol{q}_1^*. \tag{33}$$

In recent years, the quaternion theory is widely used in color images. Assuming a continuous color image as:

$$f(r,\theta,z)=\begin{bmatrix} f(r,\theta,\ 1)\\ f(r,\theta,\ 2)\\ f(r,\theta,\ 3)\end{bmatrix} =\begin{bmatrix} f_R(r,\theta)\\ f_G(r,\theta)\\ f_B(r,\theta)\end{bmatrix},\ r\in[0,1],\ \theta\in[0,2\pi],\ z\in\{1,\ 2,\ 3\}. \tag{34}$$

$f_R(r,\theta)$, $f_G(r,\theta)$, and $f_B(r,\theta)$ represent three color channels of the color image $f(r,\theta,z)$, respectively. Here we can define the color image $f(r,\theta,z)$ in pure quaternion form as:

$$f(r,\theta,z)=f_R(r,\theta)\boldsymbol{i}+f_G(r,\theta)\boldsymbol{j}+f_B(r,\theta)\boldsymbol{k}. \tag{35}$$

Since quaternion multiplication is not commutative, from [42], we can get the following two definitions of AQFPJFMs:

$$\begin{aligned} H^L_{nm\alpha} &= \int_0^1\int_0^{2\pi} V^*_{nm\alpha}(\mu,r,\theta)\cdot f(r,\theta,z)r\mathrm{d}r\mathrm{d}\theta\\ &= \int_0^1\int_0^{2\pi}\frac{1}{2\pi}J_n(r,\alpha)\exp(-\mu m\theta)f(r,\theta,z)r\mathrm{d}r\mathrm{d}\theta.\\ H^R_{nm\alpha} &= \int_0^1\int_0^{2\pi} f(r,\theta,z)\cdot V^*_{nm\alpha}(\mu,r,\theta)r\mathrm{d}r\mathrm{d}\theta\\ &= \int_0^1\int_0^{2\pi} f(r,\theta,z)\frac{1}{2\pi}J_n(r,\alpha)\exp(-\mu m\theta)r\mathrm{d}r\mathrm{d}\theta. \end{aligned} \tag{36}$$

$\boldsymbol{\mu}$ is a pure quaternion in any unit, $\boldsymbol{\mu}=(\boldsymbol{i}+\boldsymbol{j}+\boldsymbol{k})/\sqrt{3}$ is used in this article. $H^L_{nm\alpha}$ and $H^R_{nm\alpha}$ are called left AQFPJFMs coefficients and right AQFPJFMs coefficients, respectively. According to Eq. (33), we prove that $H^L_{nm\alpha}$ and $H^R_{nm\alpha}$ have the following relationship (refer to "Appendix B"):

$$H^L_{nm\alpha}=-\left(H^R_{(-n)(-m)\alpha}\right)^*. \tag{37}$$

In this paper, AQFPJFMs represent to the left AQFPJFMs.

AQFPJFMs coefficients $H^L_{nm\alpha}$ (or $H^R_{nm\alpha}$) and the complete orthogonal basis functions $V_{nm\alpha}(r,\theta)$ can be used to reconstruct the color image $f(r,\theta,z)$:

$$\begin{aligned} f(r,\theta,z) &= \sum_{n=0}^{+\infty}\sum_{m=-\infty}^{+\infty} V_{nm\alpha}(r,\theta)\cdot H^L_{nm\alpha},\\ f(r,\theta,z) &= \sum_{n=0}^{+\infty}\sum_{m=-\infty}^{+\infty} H^R_{nm\alpha}\cdot V_{nm\alpha}(r,\theta). \end{aligned} \tag{38}$$

If the reconstruction restriction condition $|n|,\ |m|\le K$ is added, the approximate reconstruction color image $\tilde{f}(r,\theta,z)$ can be obtained by Eq. (39):

$$\begin{aligned} f(r,\theta,z)\approx\tilde{f}(r,\theta,z) &= \sum_{n=0}^{K}\sum_{m=-K}^{K} V_{nm\alpha}(r,\theta)\cdot H^L_{nm\alpha},\\ f(r,\theta,z)\approx\tilde{f}(r,\theta,z) &= \sum_{n=0}^{K}\sum_{m=-K}^{K} H^R_{nm\alpha}\cdot V_{nm\alpha}(r,\theta). \end{aligned} \tag{39}$$

By Eqs. (20)–(23), we can obtain (refer to "Appendix C"):

$$H^L_{nm\alpha}=A^L_{nm\alpha}+B^L_{nm\alpha}\boldsymbol{i}+C^L_{nm\alpha}\boldsymbol{j}+D^L_{nm\alpha}\boldsymbol{k}, \tag{40}$$

where

$$\begin{cases} A^L_{nm\alpha} = -\frac{1}{\sqrt{3}}[\mathrm{Im}(H_{nm\alpha}(f_R))+\mathrm{Im}(H_{nm\alpha}(f_G))+\mathrm{Im}(H_{nm\alpha}(f_B))]\\ B^L_{nm\alpha} = \mathrm{Re}(H_{nm\alpha}(f_R))+\frac{1}{\sqrt{3}}[\mathrm{Im}(H_{nm\alpha}(f_B))-\mathrm{Im}(H_{nm\alpha}(f_G))]\\ C^L_{nm\alpha} = \mathrm{Re}(H_{nm\alpha}(f_G))+\frac{1}{\sqrt{3}}[\mathrm{Im}(H_{nm\alpha}(f_R))-\mathrm{Im}(H_{nm\alpha}(f_B))]\\ D^L_{nm\alpha} = \mathrm{Re}(H_{nm\alpha}(f_B))+\frac{1}{\sqrt{3}}[\mathrm{Im}(H_{nm\alpha}(f_G))-\mathrm{Im}(H_{nm\alpha}(f_R))] \end{cases}. \tag{41}$$

Here $H_{nm\alpha}(\cdot)$ represents the AFPJFMs coefficients of $\cdot$, Im $(\cdot)$ represents the imaginary part of the imaginary number $\cdot$, and Re $(\cdot)$ represents the real part of the imaginary number $\cdot$.

4.2 Invariance

4.2.1 Rotation invariance

$f(r,\theta,z)$ is the color image, ϑ is the rotation angle. Then the rotated image $f^{\mathrm{rot}}(r,\theta,z)$ can be defined as:

$$f^{\mathrm{rot}}(r,\theta,z)=f(r,\theta+\vartheta,z). \tag{42}$$

Next we will prove that the AQFPJFMs' magnitude coefficients of the rotated image $f^{\mathrm{rot}}(r,\theta,z)$ have rotation invariance:

$$\begin{aligned} \tilde{H}^L_{nm\alpha} &= \frac{1}{2\pi}\int_0^1\int_0^{2\pi} J^*_n(\alpha,r)\exp(-jm\theta)f^{\mathrm{rot}}(r,\theta)r\mathrm{d}r\mathrm{d}\theta\\ &= \frac{1}{2\pi}\int_0^1\int_0^{2\pi} J^*_n(\alpha,r)\exp(-jm\theta)f(r,\theta+\vartheta,z)r\mathrm{d}r\mathrm{d}\theta\\ &= \frac{1}{2\pi}\int_0^1\int_0^{2\pi} J^*_n(\alpha,r)\exp(-jm(\theta-\vartheta))f(r,\theta,z)r\mathrm{d}r\mathrm{d}\theta\\ &= \exp(jm\vartheta)\frac{1}{2\pi}\int_0^1\int_0^{2\pi} J^*_n(\alpha,r)(-jm\theta)f(r,\theta,z)r\mathrm{d}r\mathrm{d}\theta\\ &= \exp(jm\vartheta)H^L_{nm\alpha}. \end{aligned} \tag{43}$$

It can be seen that rotated image only affects the phase of AQFPJFMs, and does not affect the magnitude of AFPJFMs. So there is:

$$\left|\tilde{H}^{L}_{nm\alpha}\right| = \left|\exp(jm\vartheta)H^{L}_{nm\alpha}\right| = \left|H^{L}_{nm\alpha}\right|,$$
$$\left|\tilde{H}^{R}_{nm\alpha}\right| = \left|H^{R}_{nm\alpha}\exp(jm\vartheta)\right| = \left|H^{R}_{nm\alpha}\right|. \quad (44)$$

4.2.2 Scaling invariance

Theoretically, AQFPJFMs are not scaling invariance. However, in the calculation of AQFPJFMs' process, the image functions are all normalized and mapped to the unit circle. All the scaling images are the same image after being mapped to the unit circle. Therefore, we can consider the AQFPJFMs coefficients to be scaling invariance.

5 Image representation via mixed low-order moments

Low-order moments reflect more robust low-frequency information, global feature descriptions usually use low-order moments to improve robustness. However, features only containing a few low-order moments usually cannot distinguish multiple objects in an image. Therefore, in actual situations, high-order moments are needed to improve the discriminability, but this comes at the cost of robustness. Based on this, we propose a strategy called mixed low-order moments feature (MLMF). It uses fractional-order parameters to improve discriminability and robustness.

5.1 Definition

Mathematically, we define MLMF based on AQFPJFMs as:

$$\boldsymbol{v} = \{\mathcal{M}(H_{nm\alpha})\cdot w(\alpha,n,m):(n,m)\in\Phi,\alpha\in\Omega\}. \quad (45)$$

$\mathcal{M}(\bullet)$ generates the rotation invariant of AQFPJFMs $H_{nm\alpha}$. In this paper, $\mathcal{M}(\bullet)$ is the magnitude of AQFPJFMs. $w(\alpha,n,m)$ is the weight functions, we can set $w(\alpha,n,m)$ to strengthen specific features by changing the order (n,m) and fractional-order parameter α. Φ is the set of order (n,m), priority is given to low order. In this article, we use the largest order K to define Φ:

$$\Phi(K) = \{(n,m) : n\in N,\ m\in Z,\ |n|,|m|\le K\}. \quad (46)$$

5.2 Robustness

In this section, we will design an experiment to verify the robustness of AQFPJFMs with mixed low-order moments features. First, we define the AQFPJFMs' MLMF feature $\boldsymbol{v}$ of the image f as:

$$\boldsymbol{v}(f) = \left\{\left|H_{nm\alpha}(f)\right| : (n,m)\in O,\ \alpha\in S\right\}. \quad (47)$$

where

$$O = \{(0,0),\ (0,1),\ (1,0)\},\quad S = \{0.1, 0.5, 1, 2, 4\}. \quad (48)$$

Table 2 Image attack parameters

Attack type	Level
Rotation	Rotation angle: 5°, 45°, 90°
Scaling	Scaling factor: 0.5, 1.5, 2
Average filter	Kernel size: 5×5, 7×7
Gaussian filter	Kernel size: 5×5; Standard derivation: 1, 2
Salt and pepper noise	Noise density: 0.05, 0.1
Gaussian noise	Kernel size: 5×5; Mean: 0; Variance:0.005, 0.01
JPEG compression	Quality factor: 30, 10
Laplace sharpening	Kernel: [0, – 1, 0; – 1, 5, – 1; 0, – 1, 0]

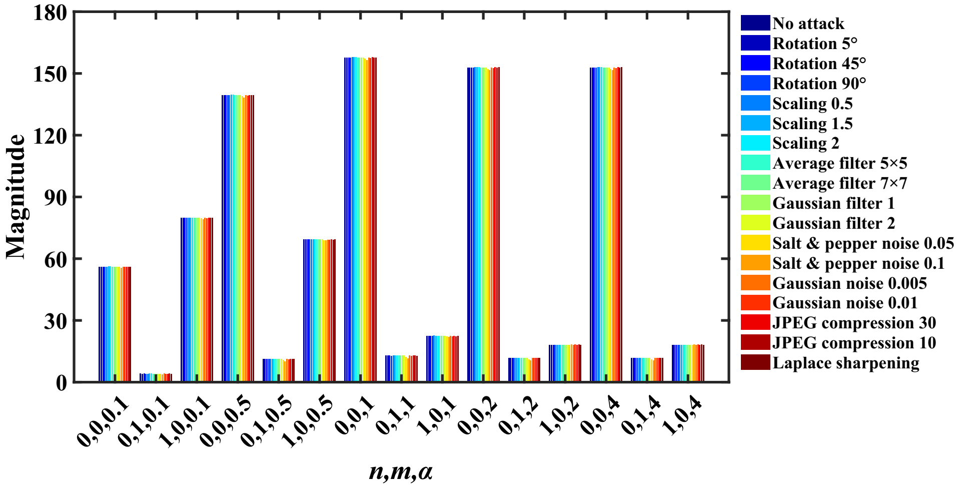


Fig. 8 Robustness of AQFPJFMs mixed low-order moment features under various attacks

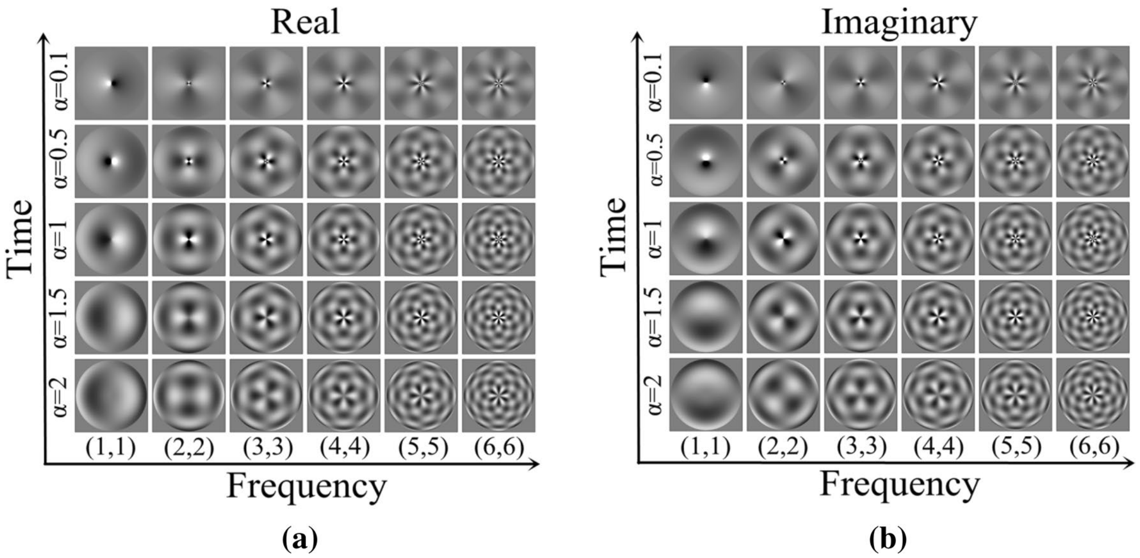


Fig. 9 **a** The real parts of AQFPJFMs' basis functions, **b** the imaginary parts of AQFPJFMs' basis functions

To intuitively understand the robustness of AQFPJFMs hybrid low-order moment features, Fig. 8 shows the feature $\boldsymbol{v}$ of the original image and each attack image. Here we set $w\,(\alpha, n, m) = 1$, attack parameters are listed in Table 2. We can see that the variation range of the characteristic $\boldsymbol{v}$ under each attack and intensity is small. Therefore, experiments have proved that AQFPJFMs hybrid low-order moment features have good robustness.

5.3 Time–frequency analysis and discriminability

The MLFM strategy is based on the AQFPJFMs' time–frequency analysis capabilities. Figure 9 shows the different parts of AQFPJFMs' basis functions. The following conclusions can be drawn:

- When we fixed the fractional-order parameter α, the order (n, m) increases and zeros' number of the basis functions $V_{nm\alpha}(r, \theta)$ will increase accordingly.
- When we fixed the order (n, m), zeros' number of the basis functions $V_{nm\alpha}(r, \theta)$ does not change. The position of the zeros will be affected by the fractional-order parameter α.

This phenomenon proves that AQFPJFMs have time–frequency analysis capabilities: The fractional-order parameter α affects the time-domain characteristics, the order (n, m) affects the frequency-domain characteristics. Different fractional-order parameters α reflect different time-domain characteristics. Combining these moments will make the features more distinguishable. At the same time, since the features only contain low-frequency components, better robustness is ensured.

6 Experiments and application

In this section, we will prove the effectiveness of AQFPJFMs and MLMF through several experiments. All the tests are implemented with MATLAB R2017a and run under the Microsoft Windows 7 environment equipped with a 3.60 GHz CPU and 16 GB memory computer.

6.1 Computation complexity and numerical stability

To prove the correctness and availability of the PJFMs' recursive calculation and polar pixel tiling scheme, the image "Lena" (128 × 128) will be reconstructed in the following three ways:

- Direct calculation method of PJFMs.
- Recursive calculation method of PJFMs.
- Recursive calculation method of APJFMs.

We use the mean square reconstruction error (MSRE) here to evaluate the reconstruction quality of the image:

$$\text{MSRE} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} [f(x,y) - \tilde{f}(x,y)]^2 \mathrm{d}x\mathrm{d}y \Bigg/ \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} [f(x,y)]^2 \mathrm{d}x\mathrm{d}y, \tag{49}$$

where $\tilde{f}(x,y)$ represents the reconstructed image, $f(x,y)$ represents the original image. Figure 10 shows the reconstructed image, the comparison of the MSRE values and calculation time are shown in Fig. 11. These results indicate that:

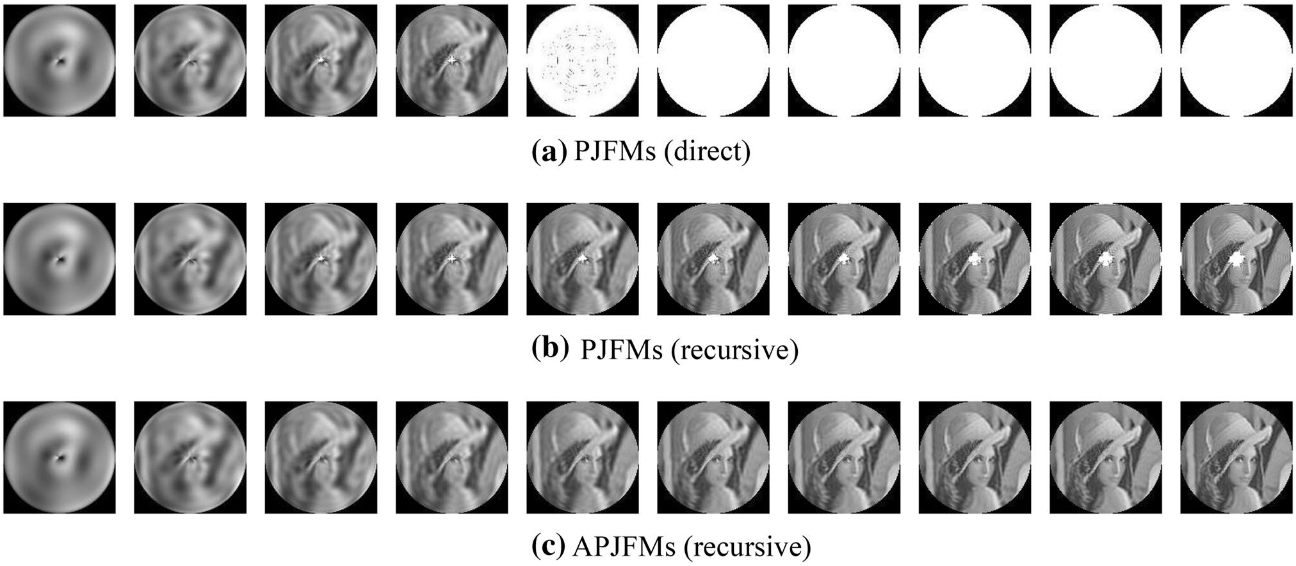


Fig. 10 Reconstructed image Lena(128 × 128) with order $K = 5,\ 10 \ldots,\ 50$ from left to right: **a** direct calculation of PJFMs. **b** Recursive calculation of PJFMs. **c** Recursive calculation of APJFMs

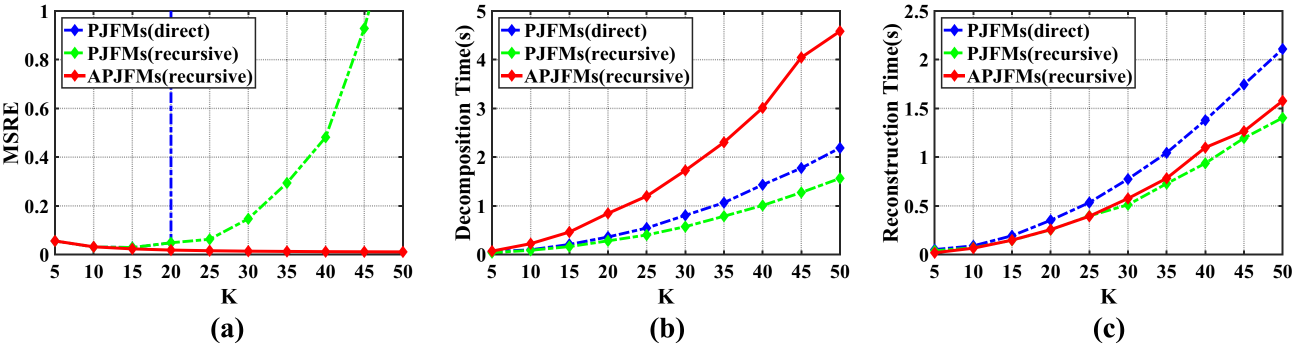


Fig. 11 Data comparison of grayscale image "Lena" (128 × 128) with order $K = 5,\ 10 \ldots,\ 50$ from left to right. **a** MSRE values, **b** decomposition times, **c** reconstruction times

- When $K < 22$, PJFMs' direct and recursive calculation method will obtain consistent subjective reconstruction results and objective reconstruction accuracy. Therefore, our recursive calculation method is correct.
- When $K \geq 22$, the direct calculation method of PJFMs has serious numerical instability problems, and the computational complexity is relatively high. Since the recursive calculation method of PJFMs does not involve the factorial (or gamma function) of large numbers, it exhibits better stability during $K \geq 22$, and the computational complexity is relatively low. Therefore, our recursive calculation method is effective and meets the theoretical expectations in Sect. 2.2.
- Compared with the recursive calculation of PJFMs, the image reconstruction ability of APJFMs has been greatly improved. Therefore, our polar pixel tiling scheme is effective.

The recursive calculation method of APJFMs is used in the experiments after this section.

6.2 Image reconstruction

This section will analyze the significance of the fractional-order parameter on the image reconstruction ability of AFPJFMs and AQFPJFMs. As we mentioned in Sect. 3.2, when $\alpha = 1$, AFPJFMs and AQFPJFMs equally extract global image features, and when the value of α deviates from 1, AFPJFMs and AQFPJFMs more reflect the local characteristics of the image. Therefore, this experiment sets the

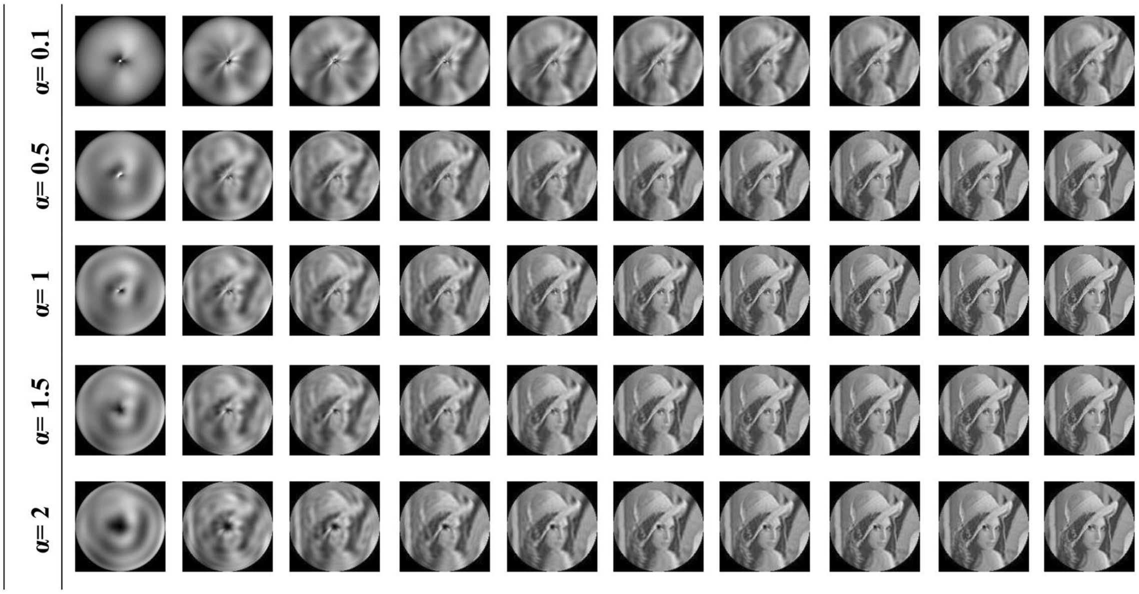


Fig. 12 Recursive computation of AFPJFMs' reconstructed grayscale image Lena (128 × 128) with order $K = 5,\ 10\ldots,\ 50$ from left to right and $\alpha = 0.1,\ 0.5,\ 1,\ 1.5,\ 2$ from top to bottom

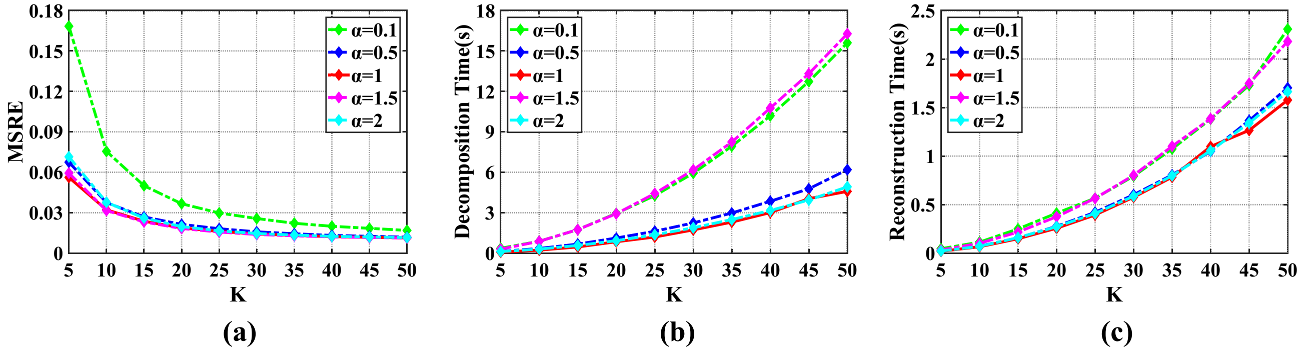


Fig. 13 Recursive computation of AQFPJFMs' reconstructed color image Lena (128 × 128) with order $K = 5,\ 10\ldots,\ 50$ from left to right and $\alpha = 0.1,\ 0.5,\ 1,\ 1.5,\ 2$ from top to bottom

parameter as: $\alpha = \{0.1,\ 0.5,\ 1,\ 1.5,\ 2\}$. Figure 12 shows the gray image "Lena" (128 × 128) with different fractional-order parameter under AFPJFMs' recursive method, Fig. 13 shows the color image "Lena" (128 × 128) with different fractional-order parameter under AQFPJFMs' recursive method. The corresponding reconstruction error, decomposition time, and reconstruction time are given in Figs. 14 and 15. We can see:

- When $\alpha = 1$, AFPJFMs' recursive calculation method will obtain consistent subjective reconstruction results and objective reconstruction accuracy with APJFMs' recursive calculation method. Therefore, our fractional expansion is correct.
- AQFPJFMs' recursive calculation method will obtain similar subjective reconstruction results and objective reconstruction accuracy with AFPJFMs' recursive calculation method. Therefore, quaternion expansion is effective.
- From the perspective of global features, the reconstructed image effect is better than PJFMs because of the adjust zeros distribution and polar pixel tiling scheme. However, when $\alpha < 0.5$, the image center will produce a vital integration error. On the contrary, the integration accu-

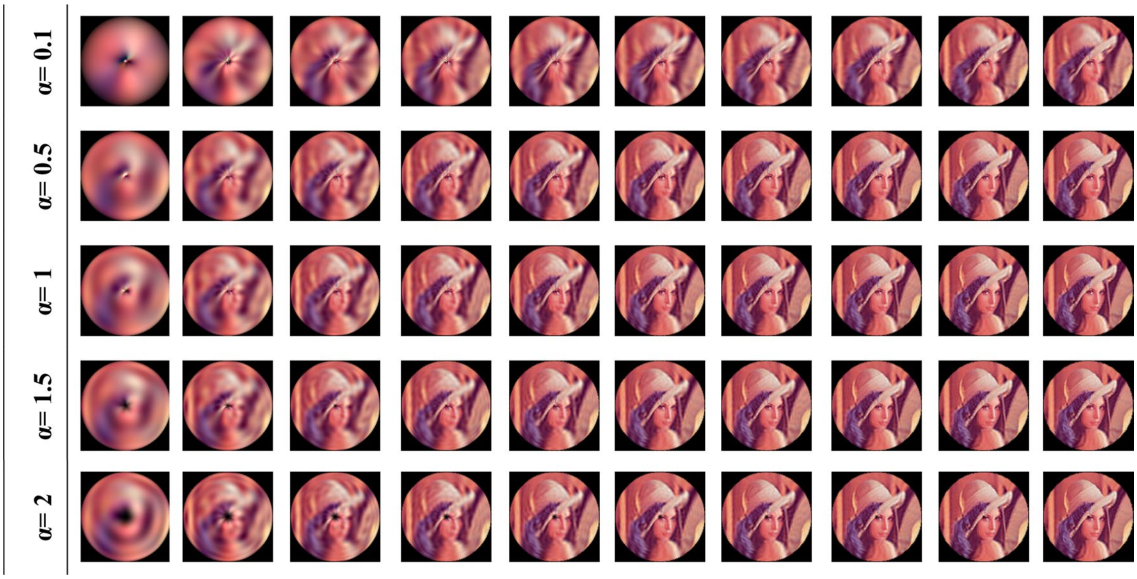
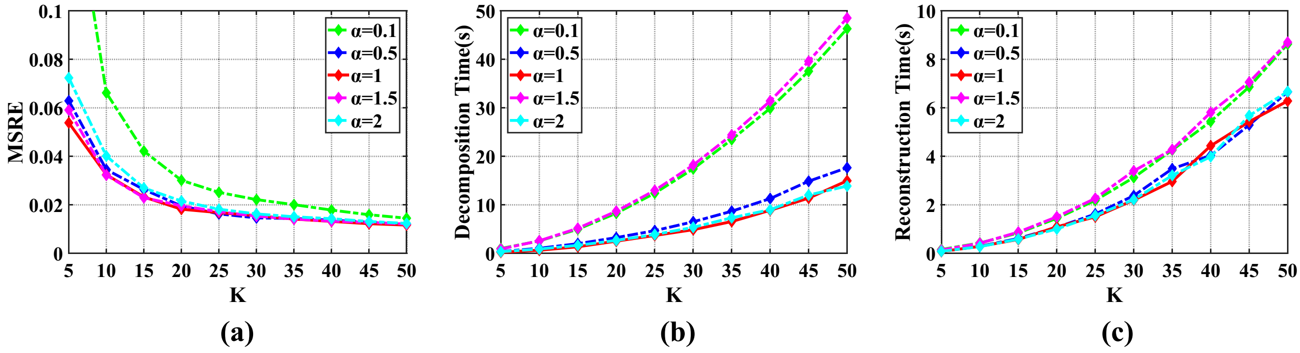


Fig. 14 AFPJFMs' data comparison of grayscale image "Lena" (128 × 128) with order $K = 5,\ 10\ldots,\ 50$ from left to right. **a** MSRE values, **b** decomposition times, **c** reconstruction times



(a)

(b)

(c)

Fig. 15 AQFPJFMs' data comparison of color image "Lena" (128 × 128) with order $K = 5,\ 10\ldots,\ 50$ from left to right. **a** MSRE values, **b** decomposition times, **c** reconstruction times

racy and numerical stability of the image center when $\alpha \geq 0.5$ are better.

- From the perspective of local features, when $\alpha < 1$, the image reconstruction effect of the central area is better than the outer area of the image. When $\alpha > 1$, the reconstruction effect of the outer area of the image is better than that of the central area of the image. This is in line with the theory we gave in Sect. 3.2. By observing Figs. 12 and 14, we can see that AFPJFMs and AQFPJFMs show more obvious spatial characteristics when $\alpha > 1$.

6.3 Application to image zero-watermarking

The watermarked image obtained by traditional digital watermarking technology will be distorted. Image distortion is unacceptable in some applications, such as military imaging systems, artwork scanning, and medical diagnosis. Moreover, this method is difficult to balance robustness, imperceptibility, and capacity due to its inherent contradictions [43]. Based on this, Wen [44] offered a new zero-watermarking algorithm. The zero-watermarking algorithm does not change the host image but constructs a watermark from the host image's characteristics. In this process, the digital image is absolutely fidelity. Therefore, zero-watermarking has perfect imperceptibility and effectively solves the balance between robustness, imperceptibility, and watermark capacity.

Moments and moments invariants have constantly been researching hotspots in the area of zero-watermarking due to their excellent image description ability and invariance. However, moments based on the zero-watermarking algorithm still has problems to be solved, including:

- Traditional zero-watermarking algorithm is generally applicable to the protection of grayscale images, in reality, color images to be protected are more common;
- Traditional zero-watermarking algorithm paid too much attention to robustness and ignoring the distinction. This phenomenon leads to a high false detection rate;
- Most zero-watermarking algorithms based on frequency domain features, their features are not robust to geometric attacks. Therefore, they can only improve robustness through machine learning (such as SVM). Also, their time complexity and applicability are inferior;
- In the algorithm based on the moment feature, the moment calculation methods almost have the problems of enormous calculation complexity, numerical instability, and low calculation accuracy, which directly affect the performance of the algorithm.

Therefore, we applied AQFPJFMs and MLMF in zero-watermarking. Compared with other advanced algorithms, our proposed algorithm has the following advantages:

- We proposed a new moment AQFPJFMs to effectively reflect the correlation between color information and color channels.
- We proposed an image description method based on AQFPJFMs called mixed low-order moments features (MLMF). The low-order AQFPJFMs have good robustness. The moments' value of the fractional-order parameter is mixed to enhance the features' discriminability, thereby MLMF achieving a good balance of robustness and discriminability.
- We proposed a new moment calculation scheme AQFPJFMs, which has significant advantages over existing calculation methods by calculation speed, numerical stability, and calculation accuracy.

The process of our zero-watermarking algorithm is as follows:

6.3.1 Initial setup

Host image $I = \{f(i,j,z) : f \in [0,\ 255], (i,j) \in [1,\ N_I]^2, z \in \{1,\ 2,\ 3\}\}$.
Fractional-order parameter $\alpha = \{0.1,\ 0.5,\ 1,\ 1.5,\ 2\}$.
Weight functions $w(\alpha, n, m) = 1$.
Logo $W = \{w(i,j) : w \in \{0,1\}, (i,j) \in [1,\ N_W]^2\}$.
Key $\mathrm{key}1 = \{x_0 = 0.5,\ \mu = 0.45,\ n = N_W\}$, $\mathrm{key}2 = \{a = 2,\ b = 3,\ It = 10\}$, $\mathrm{key}3 = \{a = 1,\ b = 2,\ It = 5\}$.

6.3.2 Use AQFPJFMs and MLMF to generation zero-watermarking

1. Get the R, G, B color channel components of the host image I.
2. Calculate the AQFPJFMs coefficients according to Eq. (41).
3. Generate mixed low-order moments features:

$$\begin{gathered}\mathbf{v} = \{\mathcal{M}(H_{nm\alpha}) \cdot w(\alpha, n, m)\} \\ \left((n,m) \in \{|n|, |m| \le \left\lceil \frac{\sqrt{1+8N_W/|\alpha|}-3}{4} \right\rceil\}, \alpha \in \{0.1, 0.5, 1, 2, 4\}\right)\end{gathered}. \tag{50}$$

4. Quantify and MLMF $\boldsymbol{v} = \{v_i\}_{i=1}^{N_W}$ to obtain quantized features $\boldsymbol{bv} = \{bv_i\}_{i=1}^{N_W}$:

$$bv_i = \begin{cases} 0 & v_i < T \\ 1 & v_i \ge T \end{cases}, \quad T = \text{median}\ (\text{mean}(\boldsymbol{v}_1), \text{mean}(\boldsymbol{v}_2),\ \ldots\ , \text{mean}(\boldsymbol{v}_{N_W})). \tag{51}$$

5. Use key1 and asymmetric Tent map to generate a chaotic sequence $\{x_i\}$:

$$x_i = \begin{cases} \frac{x_{i-1}}{\mu} & 0 \le x_{i-1} \le \mu \\ \frac{1-x_{i-1}}{1-\mu} & \mu < x_{i-1} \le 1 \end{cases}. \tag{52}$$

6. Use chaotic sequences $\{x_i\}$ and quantized features bv to perform left, right, and cyclic shift operations [45] to generate feature maps W.
7. Use key2 and key3 to scramble the watermarking W and the feature map B through the generalized Arnold transformation to obtain the scrambled watermarking WS and the scrambled feature map BS;

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & b \\ a & ab+1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \bmod N_W. \tag{53}$$

(x', y') represents the position of the pixel after transformation, and (x, y) represents the position of the pixel before transformation.

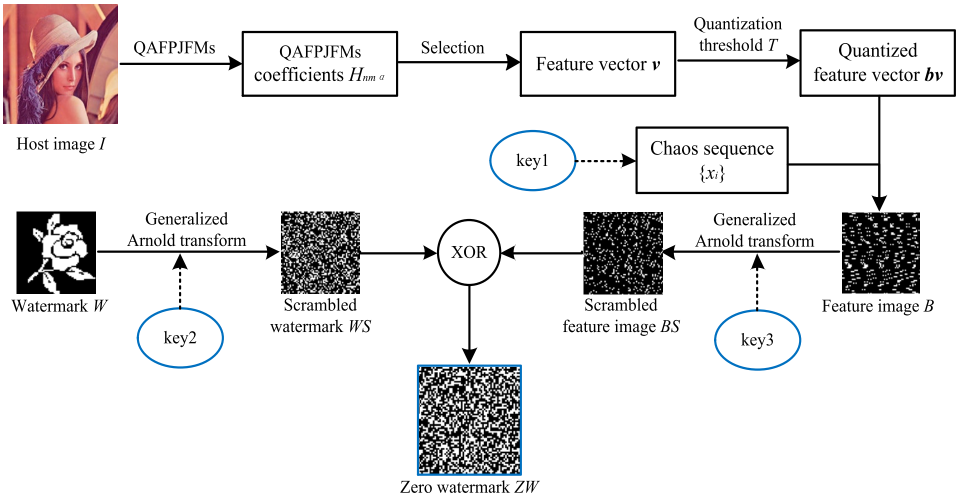
8. Take the XOR of the scrambled watermarking WS and the scrambled feature map BS to get the zero-watermarking ZW:

$$\mathrm{ZW} = \mathrm{XOR}\ (\mathrm{WS},\ \mathrm{BS}). \tag{54}$$

9. Send the zero-watermark ZW, key1, key2, key3, and the identity information $\mathrm{ID}_{\mathrm{Owner}}$ of the copyright owner to a trusted organization through a confidential channel, requesting it to be authenticated. After accepting the request, the trusted organization will verify the related message received and stamp it with a digital time stamp:

$$C_{TA} = h_{TA}(\mathrm{ZW}||\mathrm{key1}||\mathrm{key2}||\mathrm{key3}||\mathrm{ID}_{\mathrm{Owner}}||t). \tag{55}$$

Fig. 16 The generation process of our proposed zero-watermarking algorithm



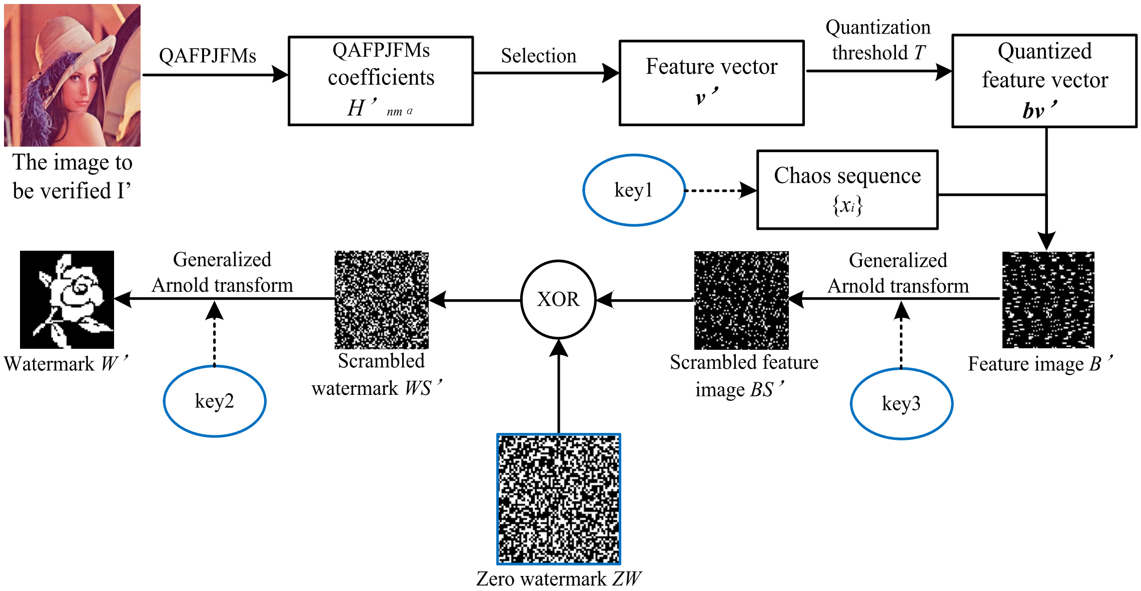

The image to be verified I'
QAFPJFMs
QAFPJFMs coefficients $H'_{nm\alpha}$
Selection
Feature vector v'
Quantization threshold T
Quantized feature vector bv'
key1
Chaos sequence {xi}
Watermark W'
Generalized Arnold transform
key2
Scrambled watermark WS'
XOR
Scrambled feature image BS'
Generalized Arnold transform
key3
Feature image B'
Zero watermark ZW

Fig. 17 The verification process of our proposed zero-watermarking algorithm

t is a digital timestamp means the authentication time; $h_{TA}(\cdot)$ is a single-channel hash function; | | marks the concatenation of data.

6.3.3 Use AQFPJFMs and MLMF to verification zero-watermarking

1. The verifier downloads information from the trusted organization, including: zero-watermarking ZW, key1, key2, key3 and identity information $\mathrm{ID}_{\mathrm{Owner}}$, uses the trusted organization to calculate whether the timestamp is complete and authentic:

$$C'_{TA} = h_{TA}(\mathrm{ZW}||\mathrm{key1}||\mathrm{key2}||\mathrm{key3}||\mathrm{ID}_{\mathrm{Owner}}||t). \tag{56}$$

2. Perform steps B.1 to B.6 on I' to obtain a feature map B' .
3. Use key3 to scramble the feature map B' through the generalized Arnold transformation, then we obtain the scrambled feature map BS′:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} ab+1 & -b \\ -a & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \bmod N_W. \tag{57}$$

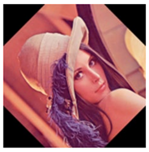
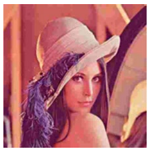
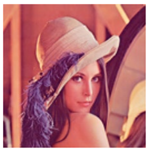
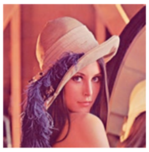
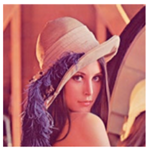
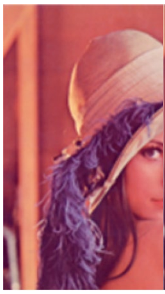
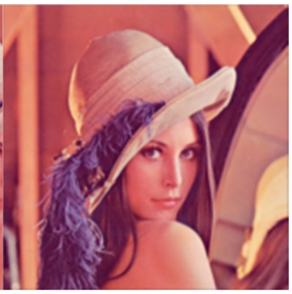
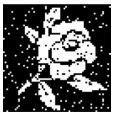
4. Take the XOR of the zero-watermarking ZW and the scrambled feature map BS′ to get the scramble watermark WS′:

$$\mathrm{WS}' = \mathrm{XOR}(\mathrm{BS}', \ \mathrm{ZW}). \tag{58}$$

5. Use key2 to reverse the scrambled watermarking WS′, then we obtain the detected watermarking W':

$$W' = \{w'(i,j) : w' \in \{0,1\}, (i,j) \in [1, \ N_W]^2\}. \tag{59}$$

Table 3 "Lena" image under different attacks with corresponding PSNR, BER values, and retrieved watermarks

Attack	Rotation with cropping 15°	Rotation with cropping 45°	Rotation with no cropping 2°	JPEG compression 10	JPEG compression 30	Scaling 0.25	Scaling 2.0
PSNR	11.6	10.3	14.6	27.5	30.9	28.6	42.3
Attacked image							
BER	0	0	0	0	0	0	0
Retrieved watermark							
Attack	Gaussian noise 0.01	Salt & Pepper noise 0.01	Surround cropping 1/4	Sharpening	UnZign (6,6)	Median filtering 3×3	Blurring
PSNR	20.3	25.1	8.5	21.3	26.4	33.8	28.1
Attacked image							
BER	0	0	0.0454	0	0	0	0
Retrieved watermark							

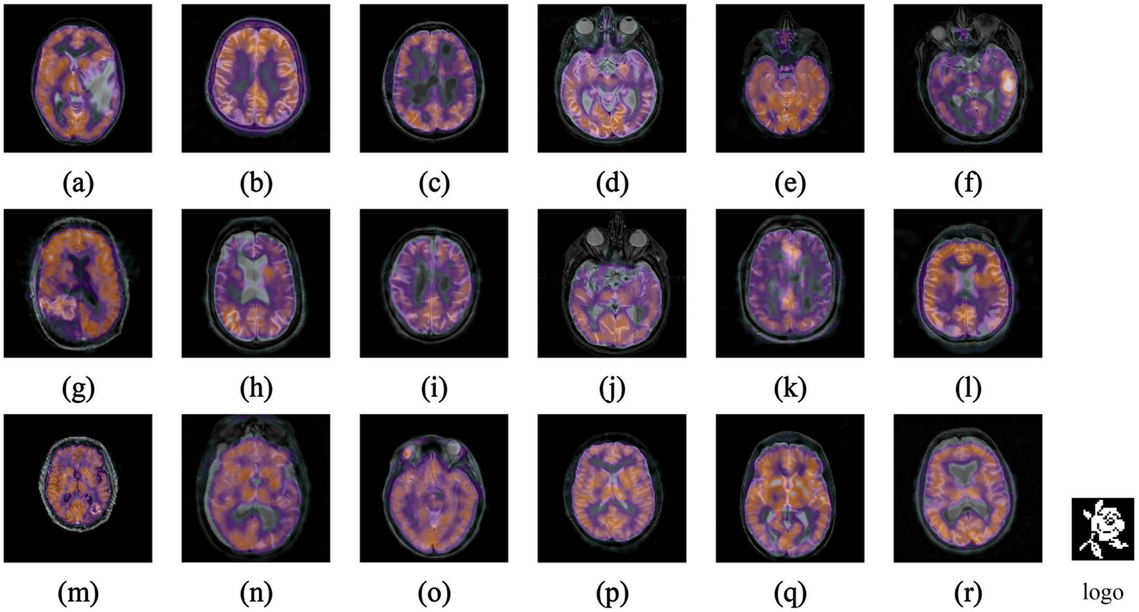


Fig. 18 Original MRI medical images and logo

Table 4 BER comparison results with other zero-watermarking algorithms (Whole Brain Atlas)

Attacks	PSNR	Proposed	Yang2020 [47]	Xia2019 [48]	Wang2017 [49]	Wang2016 [50]	Chang2008 [51]
Rotation							
5°	18.2236	**0.0000**	**0.0000**	0.0059	0.0206	0.3320	0.0037
45°	15.2647	**0.0000**	**0.0000**	0.0052	0.0235	0.3155	0.0042
Scaling							
0.25	20.7022	**0.0000**	0.0002	0.0274	0.0527	0.0438	0.0117
4	25.1038	**0.0000**	0.0001	0.0020	0.0049	0.0168	0.0088
JPEG compression							
30	22.9507	**0.0000**	0.0007	0.0059	0.0275	0.0084	0.0054
70	24.7500	**0.0011**	0.0009	0.0026	0.0234	0.0048	0.0037
Upper left corner cropping							
1/16	49.6328	**0.0000**	0.0002	**0.0000**	**0.0000**	0.1743	**0.0000**
1/8	29.5355	**0.0000**	0.005	**0.0000**	**0.0000**	0.2256	**0.0000**
Salt and pepper noise (0.01)	23.7615	0.0011	**0.0009**	0.0260	0.0400	0.0446	0.0698
Gaussian noise (0.01)	21.3016	0.0106	**0.0077**	0.0200	0.0430	0.0344	0.0285
Gaussian filtering (3×3)	21.6548	**0.0000**	0.0001	0.0024	0.0052	0.0241	0.0092
Median filtering (3×3)	24.9545	**0.0038**	0.0039	0.0098	0.0147	0.0249	0.0178

The above is all the processes of the zero-watermark algorithm. Figure 16 shows the generation process of our proposed zero-watermarking algorithm, Fig. 17 shows the verification process of our proposed zero-watermarking algorithm.

Next, the peak signal-to-noise ratio (PSNR) of the carrier image is estimated to the degradation after being attacked:

$$\mathrm{PSNR} = 10\lg\left\{\left(3\cdot 255^2\cdot N_I^2\right)/\sum_{z=1}^{3}\sum_{i=1}^{N_I}\sum_{j=1}^{N_I}\left(f(i,j,z)-f'(i,j,z)\right)^2\right\}. \tag{60}$$

We use the detected watermark's bit error rate (BER) to estimate the robustness of the algorithm:

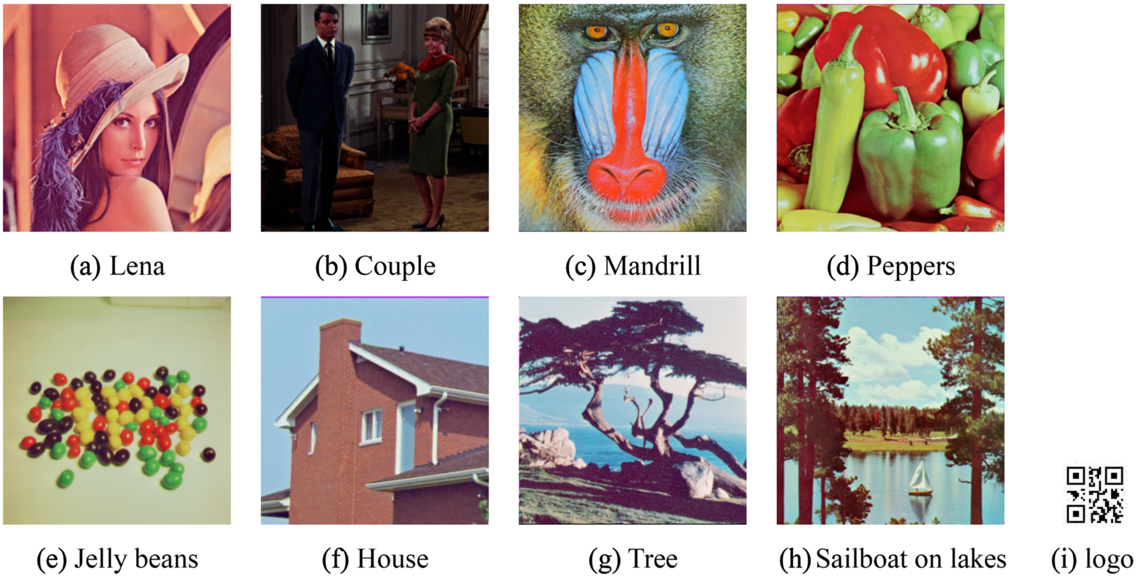

(a) Lena (b) Couple (c) Mandrill (d) Peppers (e) Jelly beans (f) House (g) Tree (h) Sailboat on lakes (i) logo

Fig. 19 Original color image and logo

Table 5 AR comparison results with other watermarking algorithms (USC-SIPI)

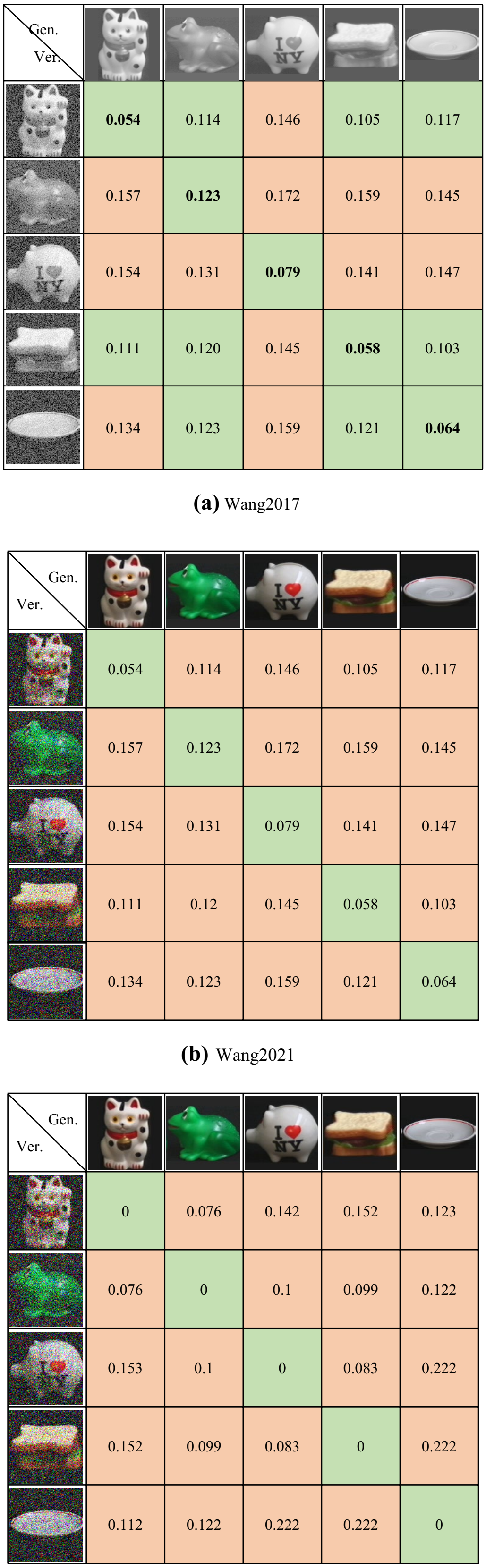
Attacks	PSNR	Proposed	Xia2021 [53]	Kang2020 [54]	Liu2019 [55]
JPEG					
30	29.3325	**1.0000**	0.9923	0.9762	0.9091
70	31.5819	**1.0000**	0.9968	0.9914	0.9932
Rotation					
5°	15.8961	**1.0000**	0.9967	0.9892	0.9150
45°	10.8333	**1.0000**	0.9946	0.9860	0.8555
Scaling and resizing to original size					
0.25	27.1446	**1.0000**	0.9895	0.9585	0.5762
4	41.6412	**1.0000**	0.9978	0.9971	0.9893
Gaussian noise (0.01)	20.1584	**1.0000**	0.9813	0.9470	0.9609
Gaussian filtering (3 × 3)	29.2265	**1.0000**	0.9978	0.9946	0.9629
Median filtering (3 × 3)	31.7905	**1.0000**	0.9906	0.9818	0.9727
Salt and peppers noise (0.03)	20.5037	**1.0000**	0.9735	0.9544	0.9424
Upper left corner cropping					
1/16	17.3245	0.9965	**1.0000**	**1.0000**	0.9053
1/8	14.448	0.9921	**1.0000**	**1.0000**	**1.0000**

$$\mathrm{BER} = \sum_{i=1}^{N_W} \sum_{j=1}^{N_W} \mathrm{XOR}(w(i,j),\ w'(i,j))/N_W^2. \tag{61}$$

Here, we take the color image "Lena" (512 × 512) as an example. Table 3 gives the images of "Lena" under various attacks, and their corresponding PSNR values, the detected watermarking and BER values. These results show that our proposed algorithm has excellent robustness. In the face of various attacks, our algorithm is still identifiable.

To test the effectiveness of our algorithm, we use the mean value of the detected watermark BER to compare with other similar zero-watermarking algorithms. In this experiment, eighteen colorful MRI medical images from the "Whole Brain Atlas" image library [46], Fig. 18 shows the test images whose size was 256 × 256, and a binary image with the size of 32 × 32. The experimental results will be compared with the five most advanced zero-watermarking algorithms [47–51]. Among them, Yang2020 [47], Xia2019 [48], Wang2017 [49], and Wang2016 [50] are some zero-watermarking algorithms based on moments, while Chang2008 [51] is the zero-watermarking algorithm based on spatial domain features. It can be seen from Table 4 that

Fig. 20 Discriminability results. **a** Wang2017. **b** Wang2021, **c**. Proposed algorithm ▶



Gen. Ver.					
	0.054	0.114	0.146	0.105	0.117
	0.157	**0.123**	0.172	0.159	0.145
	0.154	0.131	**0.079**	0.141	0.147
	0.111	0.120	0.145	**0.058**	0.103
	0.134	0.123	0.159	0.121	**0.064**

(a) Wang2017

Gen. Ver.					
	0.054	0.114	0.146	0.105	0.117
	0.157	0.123	0.172	0.159	0.145
	0.154	0.131	0.079	0.141	0.147
	0.111	0.12	0.145	0.058	0.103
	0.134	0.123	0.159	0.121	0.064

(b) Wang2021

Gen. Ver.					
	0	0.076	0.142	0.152	0.123
	0.076	0	0.1	0.099	0.122
	0.153	0.1	0	0.083	0.222
	0.152	0.099	0.083	0	0.222
	0.112	0.122	0.222	0.222	0

(c) Proposed

for most attacks, our proposed algorithm has the lowest bit error rate and higher robustness. This is because the AQFP-JFMs' low-order moments and MLMF algorithms are more robust to conventional image attacks (sharpening, blurring, compression, noise, etc.). For [47–50], the algorithms have to use high-order moments to increase the watermark capacity, so their robustness is poor.

To ensure the universality of proposed algorithm, we select 30 color images with the size of 256×256 as carrier images from the USC-SIPI image database [52]. A binary image with the size of 32×32 as a watermark. Figure 19 shows some of these images.

The watermark of [53, 54] is composed of two different kinds of moments. The watermark of algorithm [55] is composed of DTCWT-DCT. Table 5 shows the comparison of the proposed algorithm and the above algorithm. We use the detected watermark accuracy rate (AR) to estimate the robustness of the algorithm:

$$\mathrm{AR} = 1 - \mathrm{BER}. \tag{62}$$

Obviously, under most attacks, the AR values of our proposed algorithm is the highest. These experiments further confirmed our previous analysis; MLMF and AQFP-JFMs improve the robustness of the proposed algorithm. Since most zero-watermarking algorithms are publicly verifiable, in addition to their robustness, discriminability is also extremely important. However, the existing zero-watermarking algorithms often ignore the discriminability that will lead zero-watermarking algorithm may mistakenly confirm some carrier to be verified. Therefore, we designed the following discriminability experimental program: First, we choose five carrier images in the Coil-100 image database [56] and generate the corresponding zero-watermarking. Then, we use the attack version (zero-mean Gaussian noise) of these five images to verify and record the BER of the detected watermark. Finally, set a verification threshold. Only when the BER is less than this threshold, the algorithm will confirm the verification. For the algorithm's discriminability, here we use the false positive ratio (FPR) to measure it:

$$\mathrm{FPR} = \mathrm{FP}/(\mathrm{TP} + \mathrm{FP}). \tag{63}$$

TP and FP, respectively, represent the number of the algorithm correctly times confirmed and incorrectly confirmed. The test results of our algorithm will be compared with Wang2017 [49] and Wang2021 [57]. Figure 20 shows the discriminability test results comparison between Wang2017, Wang2021 and our algorithm. The red background shows the failed verification. The green

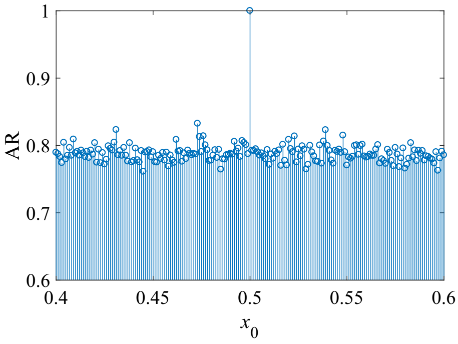


Fig. 21 Security experimental result

background shows the passed verification. The noise variance of Wang2017, Wang2021, and our proposed algorithm is 0.01; the verification threshold of Wang2017, Wang2021, and our proposed algorithm is 0.13, 0.22, and 0.04, respectively; the FPR of Wang2017, Wang2021, and our proposed algorithm is 0.615, 0, and 0. Here, robustness requires confirmation and verification when the carrier content is consistent, even if the verification carrier is attacked. This required that the BER on the diagonal of Fig. 20 should be as low as possible. The discriminability requires that when the carrier content is consistent, even if the verification carrier is attacked, robustness can be used for confirmation and verification. It means that the off-diagonal BER of Fig. 20 should be as high as possible. If the algorithm is good enough, there should be a clear boundary between the BER of the above two cases. For Wang2017, the length of the generated feature is equal to the length of the watermark, so the feature contains high-order PCET coefficients ($K = 32$), which has better discriminability. Simultaneously, the robustness of high-order coefficients is poor (Fig. 20a BER on the diagonal is too high), which makes Wang2017's FPR too large. For Wang2021, the algorithm has obvious boundaries, so it has better discriminability than Wang2017. But in Fig. 20b,

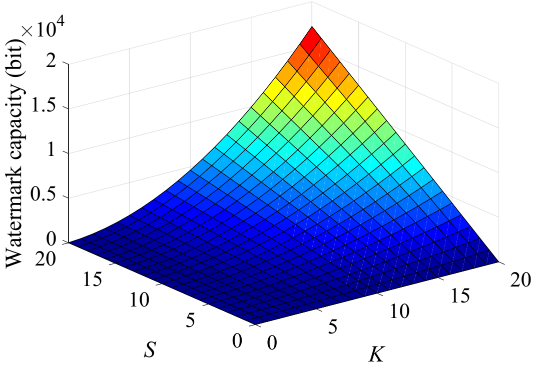


Fig. 22 Watermark capacity experimental result

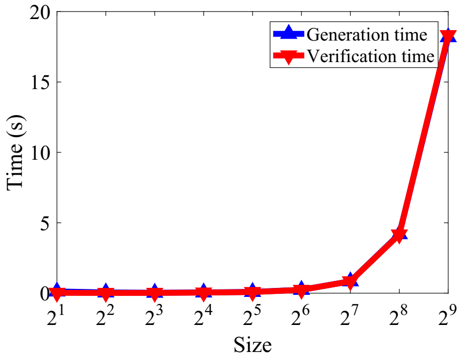


Fig. 23 Speed experimental result

we can see that the discriminative values of Wang2021 is generally too large. Compared with Fig. 20c, the values on the diagonal of Wang2021 are not 0, but the values on the diagonal of our proposed algorithm are all 0, the boundary is clearer. Therefore, the discriminability of our proposed algorithm is better than Wang2021. This is because the proposed algorithm is based on AQFPJFMs and MLMF. On the one hand, MLMF is composed of the low-order moments of $K = 2$, which ensures that the algorithm has good robustness. On the other hand, based on the AQFPJFMs' different fractional-order parameters, the rich time-domain information contained in MLMF enhances the discriminability of the algorithm. It can be seen that compared with other watermarking algorithm methods, our proposed zero-watermarking algorithm performs better in both robustness and discriminability.

To prove the security of our proposed algorithm, we test the impact of changing a single $x_0 \in \text{key1}$ on the algorithm security. Here, AR is used to check the security of the zero-watermarking algorithm.

We increase x_0 from 0.4 to 0.6 with a step size of 0.001. Figure 21 shows that the carrier image to be inspected can pass the verification only when the x_0 is correct ($x_0 = 0.5$). When the key x_0 is incorrect, the value of AR fluctuates around 0.8. It proves that the key is the crux to the security of the algorithm. When the key is not disclosed, even if all the algorithm details are totally exposed, our proposed zero-watermarking algorithm still has sufficient security.

Our proposed zero-watermarking algorithm's watermark capacity is:

$$C = S \cdot (K + 1)\,(2K + 1). \tag{64}$$

K is the maximum order, C is the watermark capacity (bit), and S is the number of selected parameters s. Figure 22 shows the relationship between K, S, and watermark capacity. Obviously, adding K and S can increase the watermark capacity of the zero-watermarking algorithm, but we recommend that users increase S instead of increasing K, because the low-order moments values are more robust.

The calculation time of our proposed algorithm is mainly spent on calculating the AQFPJFMs of the carrier image. Because we use a recursive calculation method, the time complexity of the moment is reduced from $O(N^2K^2)$ to $O(N^2K)$. (N is the size of carrier image, K is the maximum order.) Although using polar pixel tiling scheme and the increase of fractional-order parameter will increase the time complexity, the calculation time of our proposed zero-watermarking algorithm will not be significantly affected because the low-order moments and recursive calculation method we used. Therefore, the time complexity of our zero-watermarking algorithm is $O\ (3 \cdot 10 \cdot N \cdot N^2 \cdot S \cdot K) = O\ (SN^3K)$ (S is the number of fractional-order parameter α). Figure 23 shows the relationship between the zero-watermarking algorithm spent time and the carrier image's size.

7 Conclusion

In this paper, first, we proposed the recursive method of PJFMs, because the recursive calculation method does not involve the factorial (or gamma function) of large numbers, so it has better numerical stability; second, we proposed the accurate algorithm of PJFMs, called APJFMs. The polar pixel tiling scheme can cancel geometric and numerical integration errors, so we can reconstruct high-order images better; third, we proposed a new set of generalized fractional orthogonal moments, called AFPJFMs. Due to the introduction of fractional-order parameter, users can adjust the radial basis functions' zeros distribution by changing the fractional-order parameter according to different application backgrounds; fourthly, we extended AFPJFMs to the color images by quaternion. AQFPJFMs are realized. It has been improved in terms of accuracy, numerical stability and applicability; finally, we make full use of the time–frequency analysis capabilities of AQFPJFMs to propose MLMF and apply it to the digital image zero-watermarking algorithm. Compared with other advanced zero-watermarking algorithms, our proposed algorithm proves that MLMF has significant advantages in both global feature robustness and discriminability. Also, after testing, our proposed zero-watermarking algorithm has high security and has certain advantages in zero-watermarking capacity and calculation speed.

For future work, there are still many aspects of this paper that need to be improved. First, the nearest neighbor interpolation method in the AQFPJFMs algorithm affects the accuracy of the algorithm. We should use a higher-precision interpolation method to achieve upsampling; second, because of the powerful image description ability of MLMF to improve system performance, we consider applying AQFPJFMs and MLMF in other digital image processing fields.

Appendices

Appendix A

By Eq. (4), the weighting function is:$w(r) = (1 - r) \cdot r^2$.

Here replace the independent variable r of the radial basis function with the new variable $r = r^\alpha, r \in [0, 1]$, and the new weight function can be obtained as:

$$w'(r^\alpha) = (1 - r^\alpha) \cdot \alpha r^{\alpha-1} \cdot r^{2\alpha}.$$

We define a new radial basis functions:

$$\begin{aligned} J'_n(r^\alpha) &= \left[\frac{w'(r)}{rb_n}\right]^{\frac{1}{2}} P_n(r^\alpha) = \left[\frac{(1 - r^\alpha) \cdot \alpha r^{\alpha-1} \cdot r^{2\alpha}}{rb_n}\right]^{\frac{1}{2}} P_n(r^\alpha) \\ &= \left[\frac{(1 - r^\alpha) \cdot \alpha r^{3\alpha-2}}{b_n}\right]^{\frac{1}{2}} P_n(r^\alpha) \\ &= (-1)^n \left[\frac{(2n+4)}{(n+3)(n+1)}(1 - r^\alpha)\alpha r^{3\alpha-2}\right]^{\frac{1}{2}} \times P_n(\alpha, r). \end{aligned}$$

The proof is complete.

Appendix B

Taking the conjugate of $H^L_{nm\alpha}$, there are:

$$\begin{aligned} \left(H^L_{nm\alpha}\right)^* &= \left(\int_0^1 \int_0^{2\pi} \frac{1}{2\pi} J_n(r,\alpha) \exp(-\mu m\theta) f(r,\theta,z) r \mathrm{d}r \mathrm{d}\theta\right)^* \\ &= \int_0^1 \int_0^{2\pi} \frac{1}{2\pi} (J_n(r,\alpha) \exp(-\mu m\theta) f(r,\theta,z))^* r \mathrm{d}r \mathrm{d}\theta \\ &= \int_0^1 \int_0^{2\pi} \frac{1}{2\pi} (J_n(r,\alpha))^* f(r,\theta,z) (\exp(-\mu m\theta))^* r \mathrm{d}r \mathrm{d}\theta \\ &= -\int_0^1 \int_0^{2\pi} J_n(r,\alpha) f(r,\theta,z) \frac{1}{2\pi} \exp(-\mu m\theta) r \mathrm{d}r \mathrm{d}\theta \\ &= -H^R_{(-n)(-m)\alpha}. \end{aligned}$$

Therefore, $H^L_{nm\alpha} = -(H^R_{(-n)(-m)\alpha})^*$.

Appendix C

$$
\begin{aligned}
H^L_{nm\alpha} &= \int_0^1 \int_0^{2\pi} \frac{1}{2\pi} J_n(r,\alpha)\exp(-\boldsymbol{\mu} m\theta) f(r,\theta,z) r\mathrm{d}r\mathrm{d}\theta \\
&= \int_0^1 \int_0^{2\pi} \frac{1}{2\pi} J_n(r,\alpha)(f_R(r,\theta)\boldsymbol{i} + f_G(r,\theta)\boldsymbol{j} + f_B(r,\theta)\boldsymbol{k}) r\mathrm{d}r\mathrm{d}\theta \\
&= \int_0^1 \int_0^{2\pi} \frac{1}{2\pi} J_n(r,\alpha)\exp(-\boldsymbol{\mu} m\theta) f_R(r,\theta) r\mathrm{d}r\mathrm{d}\theta \cdot \boldsymbol{i} \\
&\quad + \int_0^1 \int_0^{2\pi} \frac{1}{2\pi} J_n(r,\alpha)\exp(-\boldsymbol{\mu} m\theta) f_G(r,\theta) r\mathrm{d}r\mathrm{d}\theta \cdot \boldsymbol{j} \\
&\quad + \int_0^1 \int_0^{2\pi} \frac{1}{2\pi} J_n(r,\alpha)\exp(-\boldsymbol{\mu} m\theta) f_B(r,\theta) r\mathrm{d}r\mathrm{d}\theta \cdot \boldsymbol{k} \\
&= [\mathrm{Re}(H_{nm\alpha}(f_R)) + \boldsymbol{\mu}\mathrm{Im}(H_{nm\alpha}(f_R))] \cdot \boldsymbol{i} \\
&\quad + [\mathrm{Re}(H_{nm\alpha}(f_G)) + \boldsymbol{\mu}\mathrm{Im}(H_{nm\alpha}(f_G))] \cdot \boldsymbol{j} \\
&\quad + [\mathrm{Re}(H_{nm\alpha}(f_B)) + \boldsymbol{\mu}\mathrm{Im}(H_{nm\alpha}(f_B))] \cdot \boldsymbol{k} \\
&= A^L_{nm\alpha} + B^L_{nm\alpha}\boldsymbol{i} + C^L_{nm\alpha}\boldsymbol{j} + D^L_{nm\alpha}\boldsymbol{k}.
\end{aligned}
$$

Acknowledgements This work was supported partially by the National Natural Science Foundation of China (Nos. 61472171 & 61701212), Natural Science Foundation of Liaoning Province (No. 2019-ZD-0468), and Scientific Research Project of Liaoning Provincial Education Department (No. LJKZ0985).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical standards All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki Declaration and its later amendments or comparable ethical standards.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. Hu MK (1962) Visual pattern recognition by moment invariants. IRE Trans Inf Theory 2(8):179–187
2. Teague MR (1980) Image-analysis via the general-theory of moments. J Opt Soc Am 69(8):920–930
3. Pawlak M (2014) Over 50 years of image moments and moment invariants. Gate Comput Sci Res 73(2):91–110
4. Teh CH, Chin RT (1988) On image analysis by the method of moments. IEEE Trans Pattern Anal Mach Intell 10(4):556–561
5. Sheng YL, Shen LX (1994) Orthogonal Fourier–Mellin moments for invariant pattern recognition. J Opt Soc Am 11(6):1748–1757
6. Ping ZL, Wu RG, Sheng YL (2002) Image description with Chebyshev–Fourier moments. J Opt Soc Am 19(9):1748–1754
7. Khotanzad A, Hong YH (1990) Invariant image recognition by Zernike moments. IEEE Trans Pattern Anal Mach Intell 12(5):489–497
8. Ping Z, Ren H, Jian Z et al (2007) Generic orthogonal moments: Jacobi–Fourier moments for invariant image description. Pattern Recogn 40(4):1245–1254
9. Bailey RR, Srinath MD (1996) Orthogonal moment features for use with parametric and non-parametric classifiers. IEEE Trans Pattern Anal Mach Intell 18(4):389–399
10. Amu G, Hasi S, Yang X et al (2004) Image analysis by pseudo-Jacobi (p = 4, q = 3)-Fourier moments. Appl Opt 43(10):2093–2101
11. Camacho C, Báez-Rojas JJ, Toxqui-Quitl C, Padilla-Vivanco A (2014) Color image reconstruction using quaternion Legendre–Fourier moments in polar pixels. In: 2014 IEEE international conference on mechatronics, electronics and automotive engineering (ICMEAE), Cuernavaca, pp 3–8
12. Amu G, Hasi S, Ai AZ (2015) Research progress of moment invariant image analysis. J Inner Mongolina Agric Univ 26(4):146–150
13. Qi S, Zhang Y, Wang C et al (2023) A survey of orthogonal moments for image representation: theory, implementation, and evaluation. ACM Comput Surv 55(1):1–35. https://doi.org/10.1145/3479428
14. Liao SX, Pawlak M (1998) On the accuracy of Zernike moments for image analysis. IEEE Trans Pattern Anal Mach Intell 20:1358–1364
15. Wee CY, Paramesran R (2007) On the computational aspects of Zernike moments. Image Vis Comput 25(6):967–980
16. Biswas R, Biswas S (2012) Polar Zernike moments and rotational invariance. Opt Eng 51(8):1–9
17. Mukundan R, Ramakrishnan KR (1995) Fast computation of Legendre and Zernike moments. Pattern Recogn 28(9):1433–1442
18. Papakostas GA, Boutalis YS, Karras DA, Mertzios BG (2007) Fast numerically stable computation of orthogonal Fourier–Mellin moments. IET Comput Vis 1(1):11–16
19. Hosny KM, Shouman MA, Abdel Salam HM (2011) Fast computation of orthogonal Fourier–Mellin moments in polar coordinates. J Real-Time Image Process 6(2):73–80
20. Walia E, Singh C, Goyal A (2012) On the fast computation of orthogonal Fourier–Mellin moments with improved numerical stability. J Real-Time Image Process 7(4):247–256
21. Xin Y, Pawlak M, Liao S (2012) Accurate computation of Zernike moments in polar coordinates. IEEE Trans Image Process 6(7):996–1004
22. Camacho-Bello C, Padilla-Vivanco A, Toxqui-Quitl C et al (2016) 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
23. Bhrawy A, Zaky M (2016) A fractional-order Jacobi Tau method for a class of time-fractional PDEs with variable coefficients. Math Methods Appl Sci 39(7):1765–1779
24. Vargas-Vargas H, Camacho-Bello C, Rivera-López JS et al (2021) Some aspects of fractional-order circular moments for image analysis. Pattern Recogn Lett 149:99–108
25. Su X, Tao R, Kang X (2019) Analysis and comparison of discrete fractional Fourier transforms. Signal Process 160(7):284–298
26. Kazem S, Abbasbandy S, Kumar S (2013) Fractional-order Legendre functions for solving fractional-order differential equations. Appl Math Model 37(7):5498–5510
27. Zhang H, Li Z, Liu Y (2016) Fractional orthogonal Fourier–Mellin moments for pattern recognition. In: 2016 Chinese conference on pattern recognition (CCPR). Springer, Singapore, pp 766–778
28. Benouini R, Batioua I, Zenkouar K, Zahi A, Najah S, Qjidaa H (2019) Fractional-order orthogonal Chebyshev moments and moment invariants for image representation and pattern recognition. Pattern Recogn 86:332–343
29. Yang H, Qi S, Tian J et al (2021) Robust and discriminative image representation: fractional-order Jacobi–Fourier moments. Pattern Recogn 115:107898

30. Hosny KM, Darwish MM, Aboelenen T (2020) Novel fractional-order polar harmonic transforms for gray-scale and color image analysis. J Frankl Inst 357(4):2533–2560
31. Wang C, Gao H, Yang M et al (2021) Invariant image representation using novel fractional-order polar harmonic Fourier moments. Sensors 21(4):1544
32. Chen B, Yu M, Su Q, Shim HJ, Shi YQ (2018) Fractional quaternion Zernike moments for robust color image copy-move forgery detection. IEEE Access 6:56637–56646
33. Wang C, Hao Q, Ma B et al (2021) Fractional-order quaternion exponential moments for color images. Appl Math Comput 400:126061
34. Yamni M, Karmouni H, Sayyouri M et al (2021) Robust zero-watermarking scheme based on novel quaternion radial fractional Charlier moments. Multimed Tools Appl 80(14):21679–21708
35. Upneja R, Singh C (2015) Fast computation of Jacobi–Fourier moments for invariant image recognition. Pattern Recogn 48(5):1836–1843
36. Sáez JL (2017) Comments on "fast computation of Jacobi–Fourier moments for invariant image recognition." Pattern Recogn 67:16–22
37. Singh C, Upneja R (2012) Accurate computation of orthogonal Fourier Mellin moments. J Math Imaging Vis 44(3):411–431
38. Singh C, Walia E, Upneja R (2013) Accurate calculation of Zernike moments. Inf Sci 233:255–275
39. Karakasis EG, Papakostas GA, Koulouriotis DE, Tourassis VD (2013) A unified methodology for computing accurate quaternion color moments and moment invariants. IEEE Trans Image Process 23(2):596–611
40. Camacho-Bello C (2014) High-precision and fast computation of Jacobi–Fourier moments for image description. J Opt Soc Am 31(1):124–134
41. Hamilton WR (1866) Elements of quaternions. Longmans, Green, & Company, London
42. Camacho-Bello C, Padilla-Vivanco A, Toxqui-Quitl C et al (2016) Reconstruction of color biomedical images by means of quaternion generic Jacobi–Fourier moments in the framework of polar pixels. J Med Imaging 3(1):57–66
43. Petitcolas APF (2000) Watermarking schemes evaluation. IEEE Signal Process Mag 17(5):58–64
44. Wen Q, Sun TF, Wang SX (2003) Concept and application of zero-watermark. Acta Electron Sin 31:214–216
45. Shao Z, Shang Y, Zeng R, Shu H, Coatrieux G, Wu J (2016) Robust watermarking scheme for color image based on quaternion-type moment invariants and visual cryptography. Signal Process Image Commun 48:12–21
46. The Whole Brain Atlas. http://www.med.harvard.edu/AANLIB/home.html
47. Yang H, Qi S, Niu P, Wang X (2020) Color image zero-watermarking based on fast quaternion generic polar complex exponential transform. Signal Process Image Commun 82:115747
48. Xia ZQ, Wang XY, Zhou W, Li R, Wang C, Zhang C (2019) Color medical image lossless watermarking using chaotic system and accurate quaternion Polar Harmonic transforms. Signal Process 157:108–118
49. Wang CP, Wang XY, Chen XJ, Zhang C (2017) Robust zero-watermarking algorithm based on polar complex exponential transform and logistic mapping. Multimed Tools Appl 76(24):26355–26376
50. Wang CP, Wang XY, Xia ZQ, Zhang C, Chen XJ (2016) Geometrically resilient color image zero-watermarking algorithm based on quaternion exponent moments. J Vis Commun Image Represent 41:247–259
51. Chang CC, Lin PY (2008) Adaptive watermark mechanism for rightful ownership protection. J Syst Softw 81(7):1118–1129
52. The USC-SIPI image database. http://sipi.usc.edu/database/
53. Xia Z, Wang X, Han B et al (2021) Color image triple zero-watermarking using decimal-order polar harmonic transforms and chaotic system. Signal Process 180:0165–1684
54. Kang X, Zhao F, Chen Y et al (2020) Combining polar harmonic transforms and 2D compound chaotic map for distinguishable and robust color image zero-watermarking algorithm. J Vis Commun Image Represent 70:1047–3203
55. Liu J, Li J, Ma J et al (2019) A robust multi-watermarking algorithm for medical images based on DTCWT-DCT and Henon map. Appl Sci 9(4):700–722
56. Coil-100. http://www.cs.columbia.edu/CAVE/software/softlib/coil-100.php
57. Wang XY, Wang L, Tian JL et al (2021) Color image zero-watermarking using accurate quaternion generalized orthogonal Fourier–Mellin moments. J Math Imaging Vis 63:708–734

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.