

## CHAPTER 2

---

### Accuracy Analysis of Moment Functions

---

Simon Liao

Moment methods have been the subject of intensive research since the concept of image moments was introduced by Hu in 1962 [6]. Different types of conventional continuous orthogonal moments, defined in the rectangular region and circular domain, have been investigated as the unique image features for applications in fields of pattern recognition and image analysis. For a general study of continuous orthogonal moments, please refer to [11, 12, 4].

In this chapter, we will conduct the accuracy analysis of continuous moment functions defined in both the rectangular region and circular domain, analyze the computational errors of those moment functions, and propose solutions to improve the computing accuracy of moments, especially for the higher order moment functions.

---

Simon Liao  
The University of Winnipeg  
Winnipeg, Manitoba, Canada, R3B 2E9  
e-mail: s.liao@uwinnipeg.ca

## 2.1 Moment Functions Defined in the Rectangular Region

The feature representation capability of different types of image moment functions defined in the rectangular regions, such as the Geometric moments, Central moments, Legendre moments, and Gegenbauer moments has been widely studied and applied in several areas of computer vision, including image analysis and recognition. The moment-based image features can capture global properties of an image and represent it in the moment space uniquely.

In this section, we will focus on some accuracy issues of computing moment functions defined in the rectangular region, and propose some general solutions in order to achieve more precisely calculated moments.

### 2.1.1 Accuracy Analysis

The general two-dimensional  $(p + q)$ -th continuous moment of an image function  $f(x, y)$  defined in the rectangular region, using a moment weighting kernel  $\psi_{pq}(x, y)$ , is given by

$$\Psi_{pq} = \int_x \int_y \psi_{pq}(x, y) f(x, y) dx dy, \quad (2.1)$$

where  $p, q = 0, 1, 2, \dots$

However, for a digital image, the analog image function  $f(x, y)$  needs to be digitized into its discrete version  $f(x_i, y_j)$ . Therefore, the double integration in Eq.(2.1) would have to be approximated by some summation formulas. A commonly used formula is

$$\hat{\Psi}_{pq} = \sum_x \sum_y \psi_{pq}(x_i, y_j) f(x_i, y_j) \Delta x \Delta y, \quad (2.2)$$

where  $\Delta x$  and  $\Delta y$  are the sampling intervals in the  $x$  and  $y$  directions.

If the moment kernel function  $\psi_{pq}(x_i, y_j)$  is fairly distributed within each pixel  $(i, j)$ , Eq.(2.2) would provide a relatively accurate approximation of Eq.(2.1). However, when the orders of moment kernel functions increase, in most cases, the distributions within pixel  $(i, j)$  vary significantly.

Figure 2.1 shows the distribution of Legendre polynomials  $P_{120}(x)P_{80}(y)$  within a corner pixel  $(256, 1)$ . It is obvious that the integration of  $P_{120}(x)P_{80}(y)$  in this pixel is very different from  $\Delta x \Delta y$ .

To improve the computational accuracy of moment functions defined in the rectangular region, we can rewrite Eq.(2.2) to

$$\hat{\Psi}_{pq} = \sum_x \sum_y f(x_i, y_j) h_{pq}(x_i, y_j), \quad (2.3)$$

where

$$h_{pq}(x_i, y_j) = \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} \psi(x, y) dx dy. \quad (2.4)$$

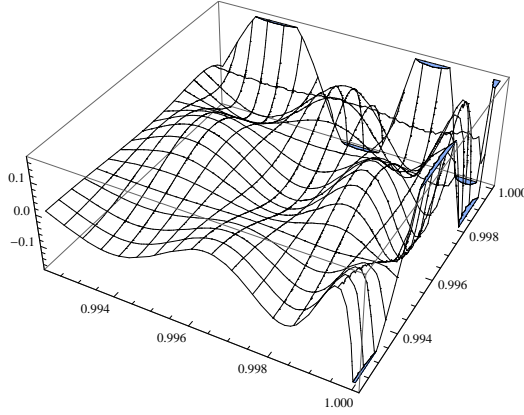


Figure 2.1: The distribution of  $P_{120}(x)P_{80}(y)$  in a corner pixel,  $(256, 1)$ , of a  $256 \times 256$  image.

By using some well-known numerical integration techniques, the double integration in Eq.(2.4) can be approximated with various accuracies. For example, the alternative extended Simpson's rule was applied to deal with a similar situation [3, 9]. We can also apply a straightforward numerical scheme, which divides a pixel into  $k \times k$  sub regions that weight the same, to calculate the double integration in Eq.(2.4) with the substantially improved computational accuracy [19].

In the following subsection, based on our accuracy analysis and the suggested numerical scheme, we will examine the Legendre moment functions defined in the rectangular region.

### 2.1.2 Legendre Moments

Legendre moment is one of the important continuous orthogonal moments defined in the rectangular region and has been well investigated since the earlier years of moment-based descriptors studies [16, 17, 9].

The  $m$ -th order Legendre polynomial is defined by [14]

$$P_m(x) = \frac{1}{2^m m!} \frac{d^m}{dx^m} (x^2 - 1)^m, \quad (2.5)$$

with the recurrent formula

$$P_{m+1}(x) = \frac{2m+1}{m+1} x P_m(x) - \frac{m}{m+1} P_{m-1}(x). \quad (2.6)$$

The Legendre polynomials  $\{P_m(x)\}$  are a complete orthogonal basis set on the interval  $[-1, 1]$  [2]:

$$\int_{-1}^{+1} P_m(x) P_n(x) dx = \frac{2}{2n+1} \delta_{mn}, \quad (2.7)$$

where  $\delta_{mn}$  is the Kronecker symbol.

The  $(m, n)$ -th order of Legendre moment of an image function  $f(x, y)$  is defined on the square  $[-1, 1] \times [-1, 1]$

$$\lambda_{mn} = \frac{(2m+1)(2n+1)}{4} \int_{-1}^{+1} \int_{-1}^{+1} f(x, y) P_m(x) P_n(y) dx dy, \quad (2.8)$$

where  $m, n = 0, 1, 2, \dots$

According to the orthogonality property of the Legendre moments, an original image function  $f(x, y)$  can be reconstructed from an infinite series of its Legendre moments

$$f(x, y) = \sum_{m=0}^{\infty} \sum_{n=0}^m \lambda_{m-n, n} P_{m-n}(x) P_n(y). \quad (2.9)$$

In practice, however, if only Legendre moments of order  $\leq T$  are given, the original image function  $f(x, y)$  can only be approximated by a truncated series

$$f(x, y) \simeq f_T(x, y) = \sum_{m=0}^T \sum_{n=0}^m \lambda_{m-n, n} P_{m-n}(x) P_n(y). \quad (2.10)$$

For a digitalized image, we can only observe an image function  $f(x, y)$  at discrete pixels, so the discrete version of  $f(x, y)$  becomes  $f(x_i, y_j)$ . Assuming that a digital image function  $f(x_i, y_j)$  is sized by  $M \times N$ , referring to Eq.(2.3) and Eq.(2.4), we can compute the Legendre moments  $\lambda_{mn}$  with our proposed numerical scheme

$$\hat{\lambda}_{mn} = \sum_{i=1}^M \sum_{j=1}^N f(x_i, y_j) h_{mn}(x_i, y_j), \quad (2.11)$$

where

$$h_{mn}(x_i, y_j) = \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} P_m(x) P_n(y) dx dy. \quad (2.12)$$

To verify the more accurately computed Legendre moments, we would examine the image reconstructions determined by

$$\hat{f}_T(x, y) = \sum_{m=0}^T \sum_{n=0}^m \hat{\lambda}_{m-n, n} P_{m-n}(x) P_n(y), \quad (2.13)$$

which is a newer version of Eq.(2.10) with the Legendre moments  $\lambda_{m-n, n}$  replaced by their approximations given by Eq.(2.11).

Figure 2.2 shows two testing images utilized in this research. Both of the testing images are sized by  $256 \times 256$  with 256 different gray levels.

To compare the reconstructed images with the original testing image, we have adopted the Peak Signal to Noise Ratio (PSNR) as the measurement, which is image independent and can be used to evaluate the reconstruction performance generally. PSNR is the ratio between the maximum power of the signal and the affecting noise, and is defined as



Figure 2.2: Two testing images sized by  $256 \times 256$  with 256 different gray levels.

$$PSNR = 10 \log_{10} \left( \frac{G_{Max}^2}{MSE} \right), \quad (2.14)$$

where  $G_{Max}$  is the maximum gray level of the image, which is 255 in our case, and  $MSE$  is the Mean Square Error defined by

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N [f(x_i, y_j) - \hat{f}(x_i, y_j)]^2. \quad (2.15)$$

First, we conducted the image reconstructions on Fig.(2.2a) by using different maximum Legendre moment orders with various  $k \times k$  numerical schemes. Table 2.1 displays some PSNR values to measure the reconstruction performances on Fig.(2.2a).

Table 2.1: PSNRs of different  $k \times k$  numerical schemes for reconstruction performances on Figure 2.2 (a)

	$T = 40$	$T = 80$	$T = 120$	$T = 160$	$T = 200$	$T = 240$
$1 \times 1$	21.4604	20.6813	13.6755	12.6527	8.5273	6.3311
$3 \times 3$	21.5906	24.0981	25.7981	25.9551	21.8930	21.0419
$5 \times 5$	21.5927	24.1444	25.9015	27.2255	27.3102	28.0546
$7 \times 7$	21.5929	24.1536	25.9223	27.3241	28.2903	28.8333
$9 \times 9$	21.5930	24.1560	25.9419	27.2999	28.5504	29.2834
$11 \times 11$	21.5930	24.1569	25.9515	27.3113	28.5163	29.6366

Figure 2.3 demonstrates some images reconstructed from Fig.(2.2a) with various  $k \times k$  numerical schemes and different maximum Legendre moment orders. It can be observed that the images reconstructed from higher orders of Legendre moments with  $7 \times 7$ ,  $9 \times 9$ , and  $11 \times 11$  numerical schemes are very close to the original Fig.(2.2a) visually.

Figure 2.4 shows some images reconstructed from Fig.(2.2b) with the  $11 \times 11$  numerical scheme from different orders of Legendre moments, while Fig.(2.5) plots

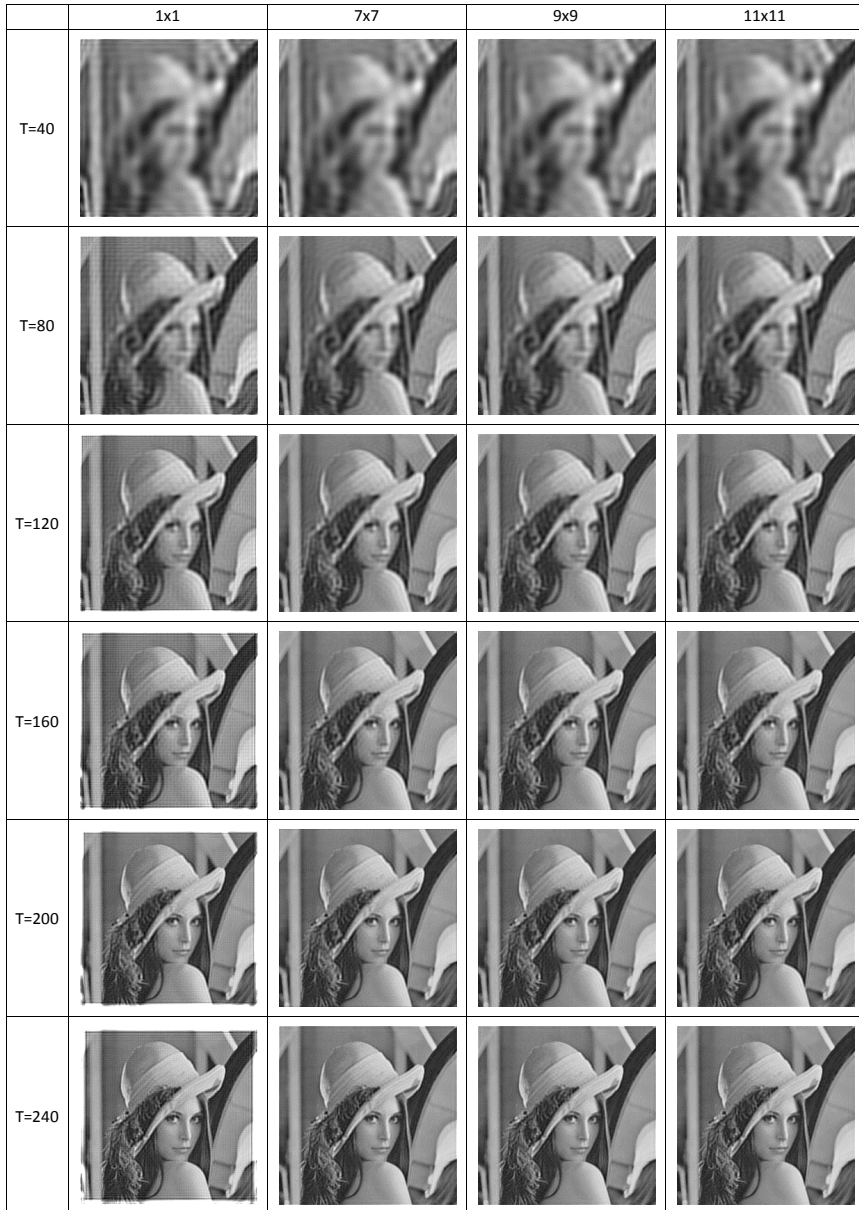


Figure 2.3: Some reconstructed images from different Legendre moments orders with various  $k \times k$  numerical schemes on Fig.(2.2a).

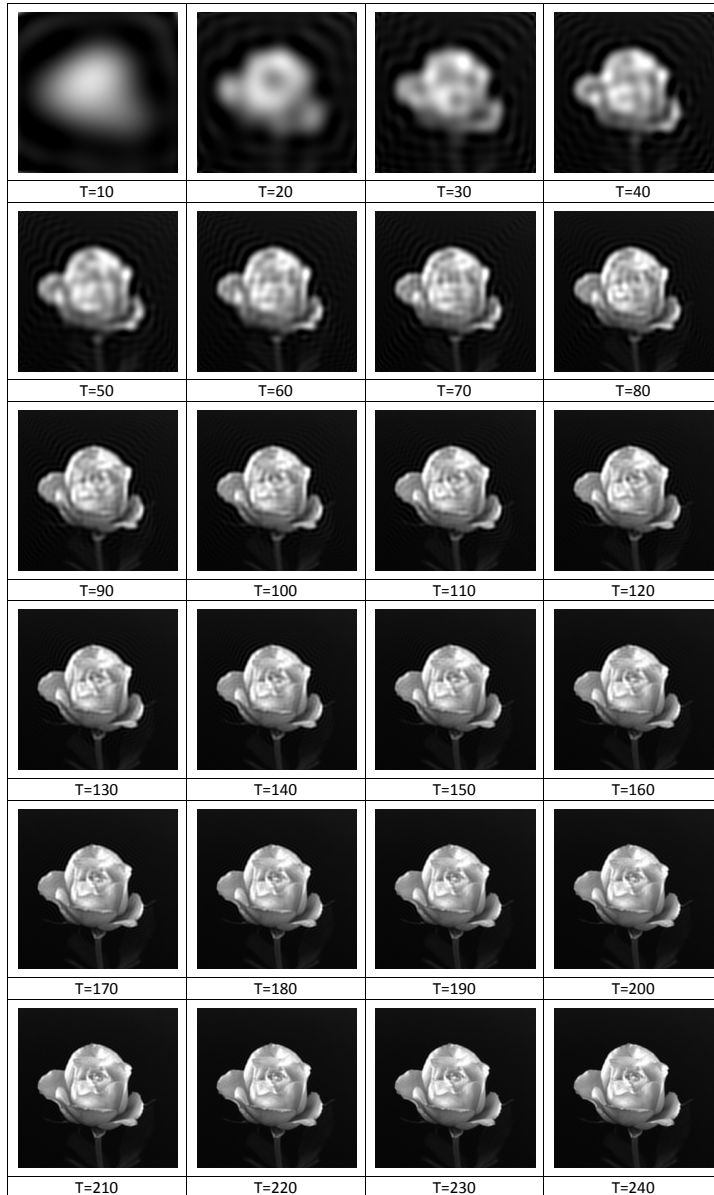


Figure 2.4: Some reconstructed images from different Legendre moments orders with  $11 \times 11$  numerical scheme on Fig.(2.2b).

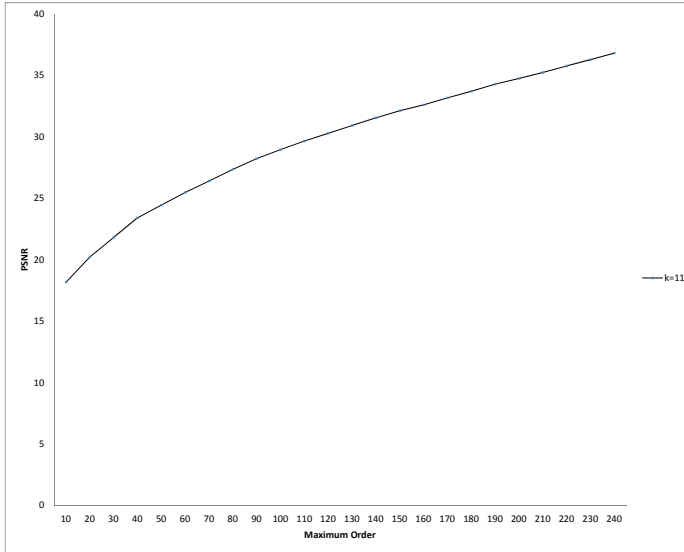


Figure 2.5: PSNRs of the reconstruction performances on Fig.(2.2b) with  $11 \times 11$  numerical scheme and different maximum orders of Legendre moments.

the PSNRs of the image reconstruction performances.

While the proposed solutions based on our analysis can improve the accuracy of computing Legendre moment functions substantially, it should be noted that the computing time of these processes is high. We conducted our experiment on a desktop computer with 12.0 GB RAM and an i7 - 2600 CPU at 3.40 GHz. For an image sized by  $256 \times 256$ , Table 2.2 shows the program running time for computing the Legendre moments of order 240, with different  $k \times k$  numerical schemes.

Table 2.2: Computing time of the Legendre moments of order 240 with different  $k \times k$  numerical schemes

	$k = 1$	$k = 3$	$k = 5$	$k = 7$	$k = 9$	$k = 11$
Hours	5.1502	16.597	38.829	71.597	115.54	170.91

Although only the Legendre moment functions are studied here, the proposed solution based on our accuracy analysis is expected to increase the computational accuracy of other conventional continuous moments, defined in the rectangular region as well.



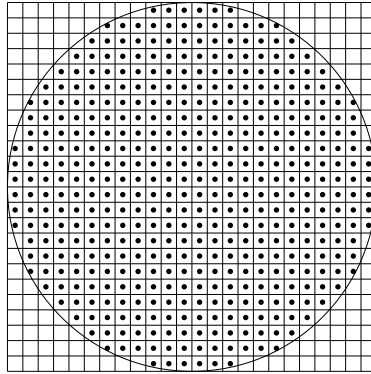


Figure 2.6: Implementing a circularly defined function in a Cartesian plane.

## 2.2 Moments Functions Defined in a Circular Domain

Due to several advanced fundamental properties, particularly the distinctive property of being invariant to rotations and reflections, the orthogonal moments defined on a circular domain have been the subject of extensive theoretical studies since they were introduced by Teague in 1980 [16]. In this class of moments, Zernike, pseudo-Zernike, and Fourier-Mellin moments have been used in numerous important applications in image analysis and recognition, ophthalmology, optical engineering, watermarking, and face recognition. However, some technical difficulties related to the computation of these moments, especially the issues of accuracy and efficiency, have bottlenecked the broader usage of this class of moments in applications [13].

### 2.2.1 Accuracy Analysis

In this subsection, we will examine a couple of important accuracy issues related to moment functions defined in a circular domain.

#### Geometric Errors

In image processing and analysis, most digital images are obtained by using the Cartesian image model, though other models such as the method based on the polar coordinate system [21] have been investigated as well. However, in general, implementing a set of circularly defined moment functions in a Cartesian plane is error prone; especially, when the computations are related to the pixels along the boundary of unit circle.

Usually, when computing a circularly defined moment function, if the centre of a pixel falls inside the border of unit circle  $x^2 + y^2 \leq 1$ , this pixel will be used in the computation; otherwise, it will be discarded. However, as shown in Fig.(2.6), some pixels are not entirely inside the unit circle; on the other hand, some of the interior

circle is not covered by the pixels. Therefore, the area covered by the moment function computation is not exactly the area of the unit circle.

In the case of most moment functions defined in a circular domain, the unit circle is located in a  $2 \text{ units} \times 2 \text{ units}$  square which is composed of  $N \times N$  pixels. Therefore, the area of the unit circle is  $\pi$ . If we use  $T(N)$  to represent the total number of pixels whose centres fall inside the unit circle, the summation of the areas of all these pixels is

$$A_{pixels} = T(N) \frac{4}{N^2}. \quad (2.16)$$

Then, the geometric error between the unit circle and the summation of all the pixels used in the circularly defined moment function computation is

$$R(N) = T(N) \frac{4}{N^2} - \pi. \quad (2.17)$$

It is crucial to know how fast the geometric error  $R(N)$  converges to zero when  $N$  tends to infinity. In fact, this issue is closely related to a famous problem in analytic number theory, attributed originally to Gauss and referred to as "The Lattice Points of a Circle Problem" [8]. Gauss' problem on the number of points inside a circle is to determine the correct order of magnitude of  $R(N)$  as  $N \rightarrow \infty$ .

According to the results from [5, 7, 8], the smallest possible order of magnitude of  $R(N)$  is

$$R(N) = O(N^{-\frac{3}{2}}). \quad (2.18)$$

This remains an open problem in number theory.

We would refer to the detailed discussion on this issue to [13] and take the conclusion that the geometric error is inherent from implementing circular defined moment functions in a Cartesian plane.

### Approximation Error

The general two-dimensional continuous moment function of an image  $f(x, y)$  defined in a circular domain, with a moment weighting kernel  $\psi_{pq}(x, y)$ , is given by

$$\Psi_{pq} = \iint_{x^2+y^2 \leq 1} \psi_{pq}(x, y) f(x, y) dx dy. \quad (2.19)$$

For digital image processing, when the analog image function  $f(x, y)$  is digitized into its discrete version  $f(x_i, y_j)$ , the double integration in Eq.(2.19) must be approximated by some summation formulas. One of the commonly used formulas is

$$\hat{\Psi}_{pq} = \sum_{x_i^2+y_j^2 \leq 1} \psi_{pq}(x_i, y_j) f(x_i, y_j) \Delta x \Delta y, \quad (2.20)$$

where  $\Delta x$  and  $\Delta y$  are the sampling intervals in the  $x$  and  $y$  directions.

If the distribution of the moment kernel function  $\psi_{pq}(x_i, y_j)$  is fairly smooth within each pixel  $(i, j)$ , Eq.(2.20) would provide a relatively justified approximation of Eq.(2.19).

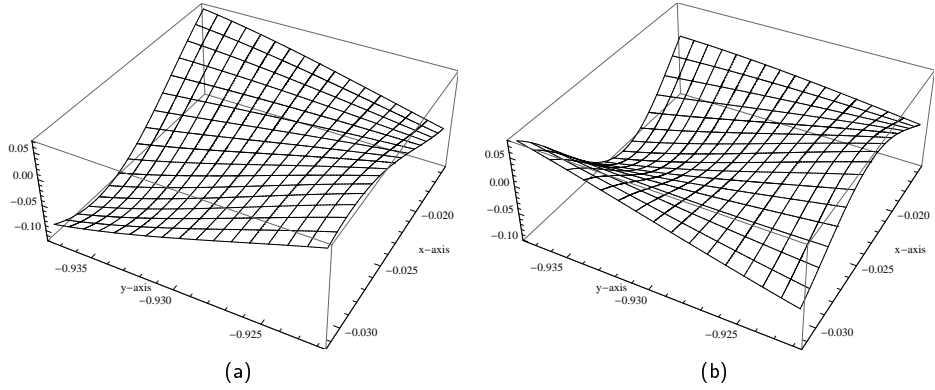


Figure 2.7: (a) The distribution of Zernike polynomial  $V_{100,70}(x, y)$  within the pixel located at  $(5, 63)$ ; (b) The distribution of pseudo-Zernike polynomial  $V_{100,70}(x, y)$  within the pixel located at  $(5, 63)$ .

However, for most cases, the distribution within pixel  $(i, j)$  vary significantly when the orders of moment kernel functions increase.

Figure 2.7 shows two examples of the distributions of two moment weighting kernel functions, Zernike polynomial  $V_{100,70}(x, y)$  and pseudo-Zernike polynomial  $V_{100,70}(x, y)$ , within a pixel located at  $(5, 63)$  of an image sized at  $128 \times 128$ . It is obvious that when the order of a kernel function rises, the accuracy of using  $\Delta x \Delta y$  to approximate the double integrations in Eq.(2.19) will decrease.

To improve the computational accuracy of moment functions defined in a circular domain, we rewrite Eq.(2.20) to

$$\hat{\Psi}_{pq} = \sum_{x_i^2 + y_j^2 \leq 1} \sum f(x_i, y_j) h_{pq}(x_i, y_j), \quad (2.21)$$

where

$$h_{pq}(x_i, y_j) = \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} \psi(x, y) dx dy. \quad (2.22)$$

Then the issue turns into how to compute the double integration in Eq.(2.22) more accurately. Some two-dimensional numerical integration techniques can be applied to improve the accuracy of the double integrations, such as N-dimensional cubature formulas [10].

An efficient and straightforward numerical scheme is to divide a pixel into  $k \times k$  sub regions with the same weights [19]. By averaging the values of all sub regions, we can considerably increase the computational accuracy.

It should be noted that if the moment kernel function  $\psi_{pq}(x, y)$  is defined in the unit circle  $x^2 + y^2 \leq 1$ , we need to discard all pixels that have any sub region falling outside the unit circle.

In the following subsections, based on our accuracy analysis, we will examine some moment functions defined in a circular domain.

### 2.2.2 Zernike Moments

The utilization of Zernike moments in the fields of image analysis and recognition was pioneered by Teague in 1980 [16].

A set of complex orthogonal functions defined over the unit disk was introduced by Zernike in 1934 [18]. The  $(p, q)$ -th order of Zernike function is defined as

$$V_{pq}(x, y) = R_{pq}(\rho) \exp(jq\theta), \quad x^2 + y^2 \leq 1, \quad (2.23)$$

where  $\rho = \sqrt{x^2 + y^2}$  is the length of the vector from the origin to the pixel  $(x, y)$ , and  $\theta = \arctan(y/x)$  is the angle between the vector and the  $x$  axis. The real-valued radial polynomial  $R_{pq}(\rho)$  is defined as [17]

$$R_{pq}(\rho) = \sum_{s=0}^{(p-|q|)/2} (-1)^s \frac{(p-s)!}{s! \left(\frac{p+|q|}{2} - s\right)! \left(\frac{p-|q|}{2} - s\right)!} \rho^{p-2s}, \quad (2.24)$$

where  $p - |q|$  is an even number, and the integer  $q$  takes positive, negative, or zero values, and satisfies

$$|q| \leq p. \quad (2.25)$$

The radial polynomial  $R_{pq}(\rho)$  is an orthogonal polynomial satisfying

$$\int_0^1 R_{pl}(\rho) R_{ql}(\rho) \rho d\rho = \frac{1}{2(p+1)} \delta_{pq}, \quad (2.26)$$

which leads to the orthogonality relation for  $\{V_{pq}(x, y)\}$  in the two-dimensional circular domain [10]

$$\iint_{x^2+y^2 \leq 1} V_{pq}^*(x, y) V_{p'q'}(x, y) dx dy = \frac{\pi}{p+1} \delta_{pp'} \delta_{qq'}, \quad (2.27)$$

where  $\delta_{pp'} = 1$  if  $p = p'$ , and 0 otherwise.

The Zernike moment of order  $p$  with repetition  $q$  is defined as

$$A_{pq} = \iint_{x^2+y^2 \leq 1} f(x, y) V_{pq}^*(x, y) dx dy, \quad (2.28)$$

where  $*$  denotes complex conjugate.

The fundamental feature of the Zernike moments is their rotational invariance. If an image  $f(x, y)$  is rotated by an angle  $\alpha$  counterclockwise, then the Zernike moment  $A_{pq}^{(\alpha)}$  of the rotated image is given by

$$A_{pq}^{(\alpha)} = A_{pq} e^{-jq\alpha}, \quad (2.29)$$



Figure 2.8: The original testing image is sized by  $128 \times 128$  with 256 gray levels.

which leads to

$$|A_{pq}^{(\alpha)}| = |A_{pq}|. \quad (2.30)$$

Thus, the magnitudes of the Zernike moments can be used as rotationally invariant image features. It should be noted, however, for digital images, the invariance properties of some Zernike moments are not perfectly valid [20].

To compute the Zernike moments  $A_{pq}$  in a Cartesian plane, referring to Eq.(2.21) and Eq.(2.22), we have the formula

$$\hat{A}_{pq} = \sum_{x_i^2 + y_j^2 \leq 1} f(x_i, y_j) h_{pq}(x_i, y_j), \quad (2.31)$$

where

$$h_{pq}(x_i, y_j) = \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} V_{pq}^*(x, y) dx dy. \quad (2.32)$$

The completeness and orthogonality of the Zernike function set  $\{V_{pq}(x, y)\}$  allow us to represent an image function  $f(x, y)$  by its Zernike moments

$$f(x, y) = \sum_{p=0}^{\infty} \sum_{q=-p}^p \frac{p+1}{\pi} A_{pq} V_{pq}(x, y), \quad (2.33)$$

where  $(p+1)/\pi$  is the normalizing constant. In practice, however, Eq.(2.33) needs to be approximated with a finite set of Zernike moments in a Cartesian plane

$$\hat{f}(x_i, y_j) = \sum_{p=0}^T \sum_{q=-p}^p \frac{p+1}{\pi} \hat{A}_{pq} V_{pq}(x_i, y_j), \quad (2.34)$$

where  $T$  is the truncation parameter indicating how many moments are taken into account [10].

To verify our proposed scheme for more accurate circularly defined moment functions, we would examine the image reconstruction from Zernike moments determined by Eq.(2.34). We have used a  $128 \times 128$  image with 256 gray levels as our testing image, which is shown in Fig.(2.8).

Figure 2.9 shows some reconstructed images by using different truncation parameters and  $k$  schemes, and Table 2.3 displays the PSNR values to measure the qualities of those reconstructed images.

Table 2.3: PSNRs of different  $k \times k$  numerical schemes for reconstruction performances on Fig.(2.8) from Zernike moments

	$k = 1$	$k = 3$	$k = 5$	$k = 7$
$T = 20$	21.330	21.169	21.172	21.172
$T = 40$	23.177	23.688	23.678	23.674
$T = 60$	23.772	25.516	25.542	25.544
$T = 80$	22.282	27.291	27.374	27.388
$T = 100$	21.580	28.902	29.043	29.055
$T = 120$	19.681	30.104	30.747	30.716

It can be observed that the images reconstructed from higher orders of Zernike moments with  $5 \times 5$  and  $7 \times 7$  numerical schemes visually are very close to the original testing image shown in Fig.(2.8).

### 2.2.3 pseudo-Zernike Moments

A modified version of Zernike functions, which is the so-called pseudo-Zernike functions, was derived by Bhatia and Wolf in [1]. This set of polynomials is different from that of Zernike and the real-valued radial polynomials  $R_{nm}(\rho)$  are defined as

$$R_{nm}(\rho) = \sum_{s=0}^{n-|m|} \frac{(-1)^s (2n+1-s)! \rho^{n-s}}{s!(n+|m|+1-s)!(n-|m|-s)!}, \quad (2.35)$$

where  $n = 0, 1, 2, \dots, \infty$ , and  $m$  is restricted to  $|m| \leq n$  only.

Compared to the Zernike polynomials, the set of pseudo-Zernike polynomials contains  $(n+1)^2$  linearly independent polynomials of degree  $\leq n$ , while the set of Zernike polynomials contains only  $\frac{1}{2}(n+1)(n+2)$  linearly independent polynomials of degree  $\leq n$  [17].

The pseudo-Zernike moment of order  $n$  with repetition  $m$  is defined as

$$A_{nm} = \iint_{x^2+y^2 \leq 1} f(x,y) V_{nm}^*(x,y) dx dy. \quad (2.36)$$

The pseudo-Zernike moments are rotational invariant as well. If an image  $f(x,y)$  is rotated  $\alpha$  degrees counterclockwise, the pseudo-Zernike moments of the resulting image are

$$A_{nm}^{(\alpha)} = A_{nm} e^{-jq\alpha}. \quad (2.37)$$

This leads to

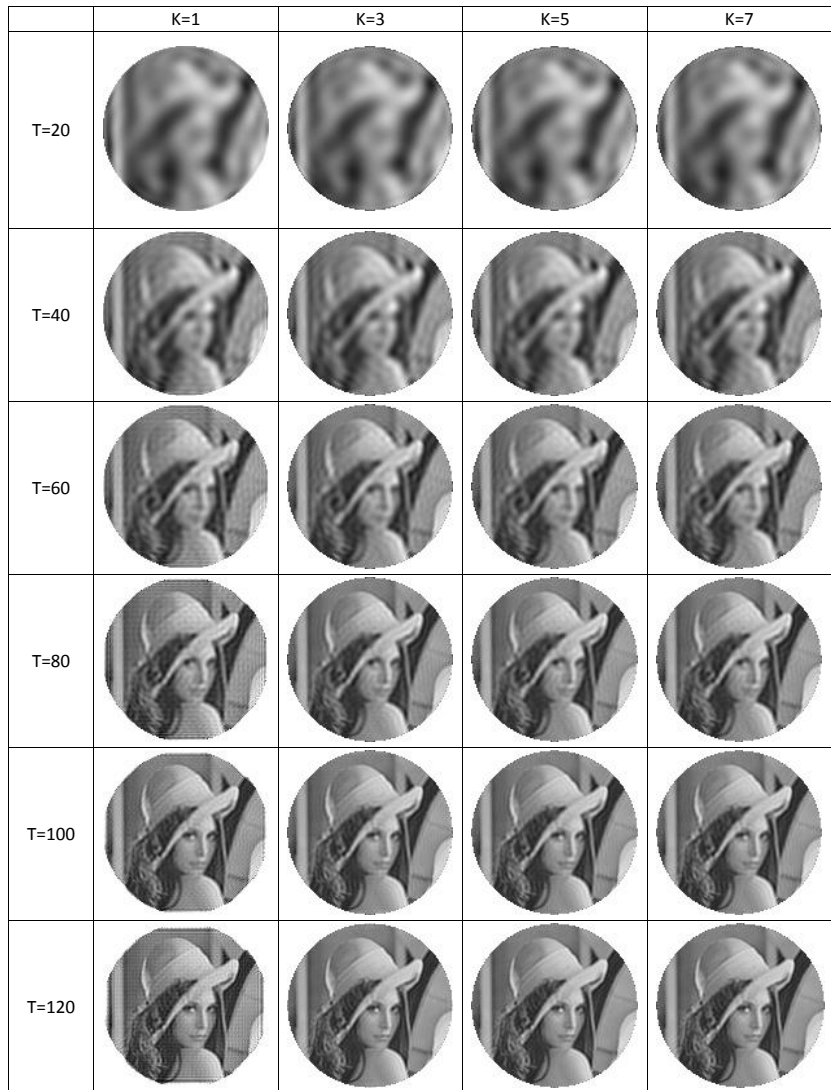


Figure 2.9: Reconstructed images from Zernike moments with different  $T$  values and  $k$  schemes.

$$|A_{nm}^{(\alpha)}| = |A_{nm}|, \quad (2.38)$$

which indicates that the magnitudes of the pseudo-Zernike moments are rotationally invariant.

To compute the pseudo-Zernike moments  $A_{nm}$  in a Cartesian plane, referring to Eq.(2.21) and Eq.(2.22), we apply the formula

$$\hat{A}_{nm} = \sum_{x_i^2+y_j^2 \leq 1} \sum f(x_i, y_j) h_{nm}(x_i, y_j), \quad (2.39)$$

where

$$h_{nm}(x_i, y_j) = \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} V_{nm}^*(x, y) dx dy. \quad (2.40)$$

Due to the orthogonality of the pseudo-Zernike functions, we can represent an image function  $f(x, y)$  by its pseudo-Zernike moments

$$f(x, y) = \sum_{p=0}^{\infty} \sum_{q=-p}^p \frac{p+1}{\pi} A_{nm} V_{nm}(x, y). \quad (2.41)$$

To reconstruct the image function  $f(x, y)$  with a finite set of its pseudo-Zernike moments in a Cartesian plane, Eq.(2.41) can be approximated by

$$\hat{f}(x_i, y_j) = \sum_{p=0}^T \sum_{q=-p}^p \frac{p+1}{\pi} \hat{A}_{nm} V_{nm}(x_i, y_j), \quad (2.42)$$

where  $\hat{A}_{nm}$  is defined in Eq.(2.39), and  $T$  is the truncation parameter deciding the number of pseudo-Zernike moments which are taken into account for image reconstruction.

To validate the proposed scheme for computing the pseudo-Zernike moment functions, we have conducted the image reconstructions from pseudo-Zernike moments by applying Eq.(2.42). The same testing image shown in Fig.(2.8) is utilized in this investigation.

Figure 2.10 represents some reconstructed images from the pseudo-Zernike moments with different truncation parameters and  $k$  schemes. Table 2.4 displays the PSNR values to measure the quality of those reconstructed images.

Comparing the results shown in Fig.(2.10) and Table 2.4 with those of Fig.(2.9) and Table 2.3, it is noticeable that the images reconstruction performances of the higher orders of pseudo-Zernike moments are superior to Zernike moments. This can be explained by the fact that the set of pseudo-Zernike polynomials contains about twice as many linearly independent polynomials of the same degree as the set of Zernike polynomials does.



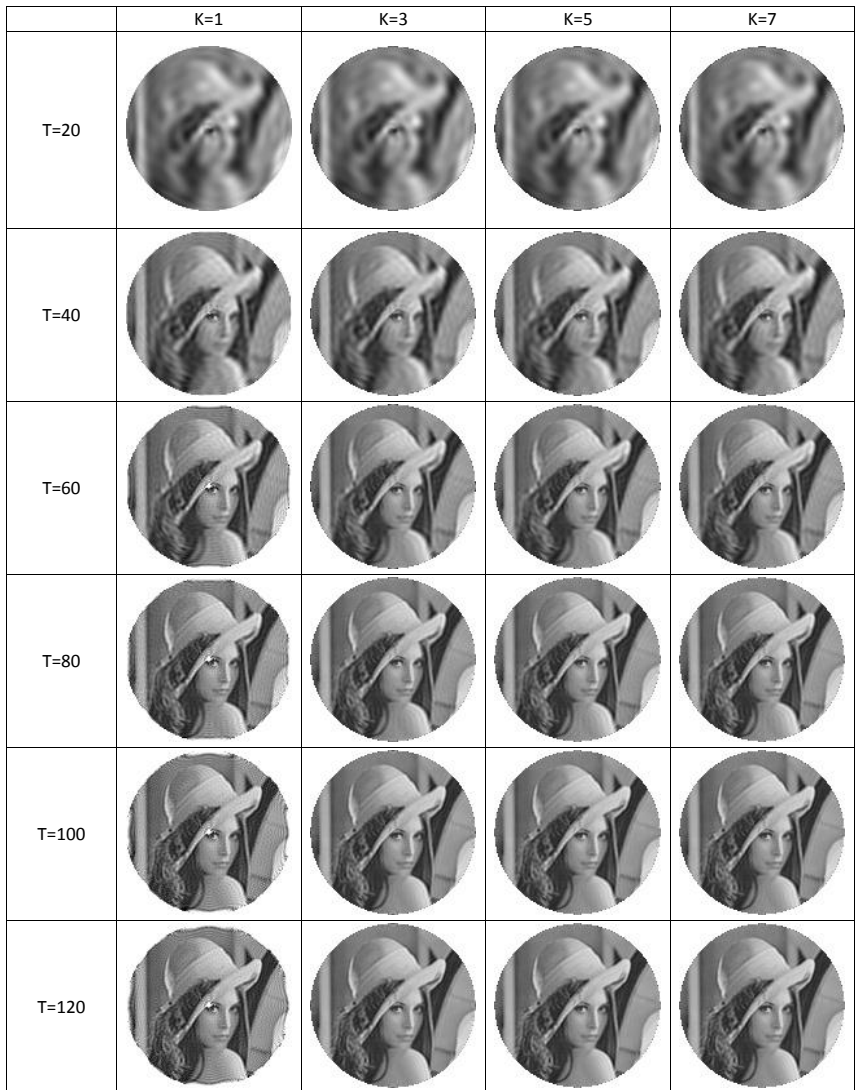


Figure 2.10: Reconstructed images from pseudo-Zernike moments with different  $T$  values and  $k$  schemes.

Table 2.4: PSNRs of different  $k \times k$  numerical schemes for reconstruction performances on Fig.(2.8) from pseudo-Zernike moments

	$k = 1$	$k = 3$	$k = 5$	$k = 7$
$T = 20$	22.693	22.599	22.588	22.585
$T = 40$	24.273	25.476	25.468	25.468
$T = 60$	22.980	27.796	27.882	27.893
$T = 80$	22.238	29.780	29.888	29.902
$T = 100$	21.092	31.025	31.213	31.231
$T = 120$	20.275	31.893	31.997	32.029

## 2.2.4 Orthogonal Fourier-Mellin moments

Sheng and Shen introduced orthogonal Fourier-Mellin moments in 1994 [15]. The orthogonal Fourier-Mellin moments are defined in a polar coordinate system over the interior of the unit circle

$$\Phi_{nm} = \frac{1}{2\pi\alpha_n} \int_0^{2\pi} \int_0^1 f(r, \theta) Q_n(r) \exp(-jm\theta) r dr d\theta, \quad (2.43)$$

where  $f(r, \theta)$  is an image function, the circular harmonic order  $m = 0, \pm 1, \pm 2, \dots$ , and

$$\alpha_n = \frac{1}{2(n+1)} \quad (2.44)$$

is a normalization constant. The polynomials  $Q_n(r)$  are

$$Q_n(r) = \sum_{s=0}^n \alpha_{ns} r^s, \quad (2.45)$$

where

$$\alpha_{ns} = (-1)^{n+s} \frac{(n+s+1)!}{(n-s)!s!(s+1)!} \quad (2.46)$$

are called coefficients of the  $n$ th polynomial with  $n$  starting from zero.

Since the set of  $Q_n(r)$  is orthogonal over the range  $0 \leq r \leq 1$

$$\int_0^1 Q_n(r) Q_k(r) r dr = \alpha_n \delta_{nk}, \quad (2.47)$$

where  $\delta_{nk}$  is the Kronecker symbol, the basis functions  $Q_n(r) \exp(-jm\theta)$  in Eq.(2.43) are orthogonal over the unit circle.

To compute the orthogonal Fourier-Mellin moments  $\Phi_{nm}$  in a Cartesian plane, referring to Eq.(2.21) and Eq.(2.22), we have

$$\hat{\Phi}_{nm} = \frac{n+1}{\pi} \sum_{x_i^2 + y_j^2 \leq 1} f(x_i, y_j) h_{nm}(x_i, y_j), \quad (2.48)$$

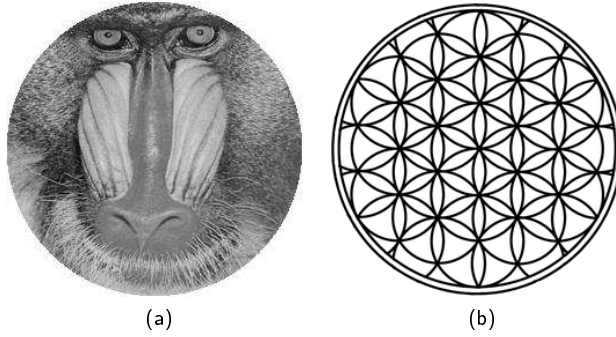


Figure 2.11: The testing images are sized by  $256 \times 256$  with 256 gray levels.

where

$$h_{nm}(x_i, y_j) = \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} Q_n(r) \exp(-jm\theta) dx dy. \quad (2.49)$$

The orthogonality of the set  $Q_n(r) \exp(-jm\theta)$  allows us to reconstruct an image function defined in the unit circle by the inverse orthogonal Fourier-Mellin transform

$$f(r, \theta) = \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} \Phi_{nm} Q_n(r) \exp(jm\theta). \quad (2.50)$$

In practice, with a finite set of orthogonal Fourier-Mellin moments  $\Phi_{nm}$  computed in a Cartesian plane, an approximate version of  $f(r, \theta)$ ,  $\hat{f}(r, \theta)$ , can be obtained by

$$\hat{f}(r, \theta) = \sum_{n=0}^{N_{max}} \sum_{m=-M_{max}}^{M_{max}} \hat{\Phi}_{nm} Q_n(r) \exp(jm\theta), \quad (2.51)$$

where  $-M_{max} \leq m \leq M_{max}$  and  $0 \leq n \leq N_{max}$ .

To attest our scheme to compute the orthogonal Fourier-Mellin moments, we have conducted the image reconstructions on two  $256 \times 256$  images with 256 gray levels, which are shown in Fig.(2.11a) and (2.11b).

Figure 2.12 represents some reconstructed Fig.(2.11a) images from the orthogonal Fourier-Mellin moments with the same order  $n = m = 220$  and different  $k \times k$  schemes. Visually, as shown in Fig.(2.12), the improvements on the reconstructed images are substantial as  $k$  increases.

Table 2.5 displays the PSNR values of some reconstructed images with different orders and numerical schemes.

A collection of reconstructed Fig.(2.11a) images, with the same numerical integration scheme  $k = 15$  and different maximum radial and harmonic orders, is displayed in Fig.(2.13).

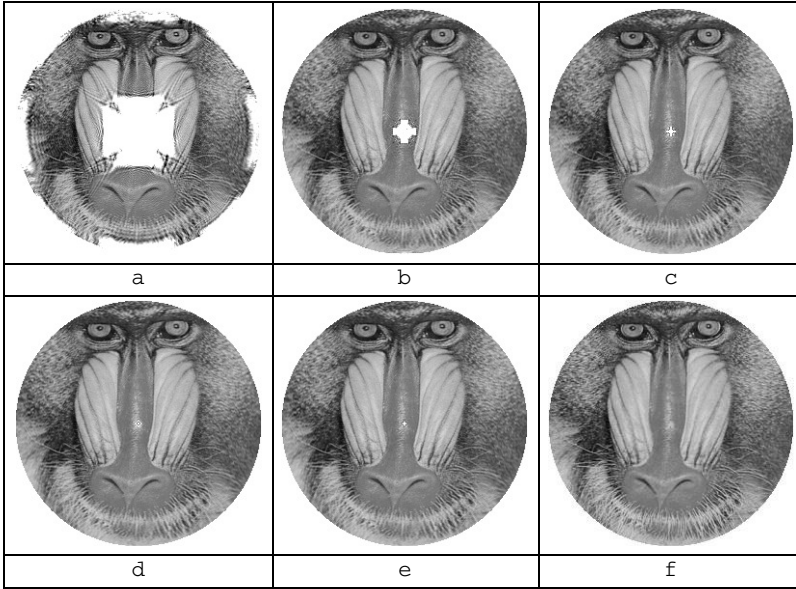


Figure 2.12: Sub-figures (a) to (e) are the reconstructed images from orders  $n = m = 220$  with the numerical integration scheme of  $k = 1, 3, 7, 11,$  and  $15,$  respectively. The sub-figure (f) shows the original image for comparison.

Table 2.5: Some PSNR values of reconstructed images with different numerical  $k \times k$  schemes

Order( $n = m$ )	10	45	80	115	150	185	220
$1 \times 1$	21.291	23.357	23.093	21.399	19.166	17.044	15.052
$3 \times 3$	21.295	23.733	25.186	26.380	27.463	27.847	27.535
$7 \times 7$	21.295	23.763	25.308	26.779	28.352	30.053	31.883
$11 \times 11$	21.295	23.767	25.317	26.801	28.461	30.327	32.489
$15 \times 15$	21.295	23.766	25.323	26.803	28.473	30.352	32.710

By using the testing image Fig.(2.11b), we have conducted some image reconstructions with different maximum radial and harmonic orders. The results are shown in Fig.(2.14).

The top three images in the right column of Fig.(2.14) illustrate that the radial patterns are better reconstructed as the value of harmonic order  $m$  increases. From the bottom three images in the left column, however, we can observe that the harmonic patterns are better reconstructed when the radial order  $n$  increases.

Based on the above observations, we conclude that the radial order  $n$  and harmonic order  $m$  of the orthogonal Fourier-Mellin moments preserve image information in different orientations. More specifically, the harmonic order  $m$  of the orthogonal Fourier-Mellin moments contain more information on radial patterns, while the radial

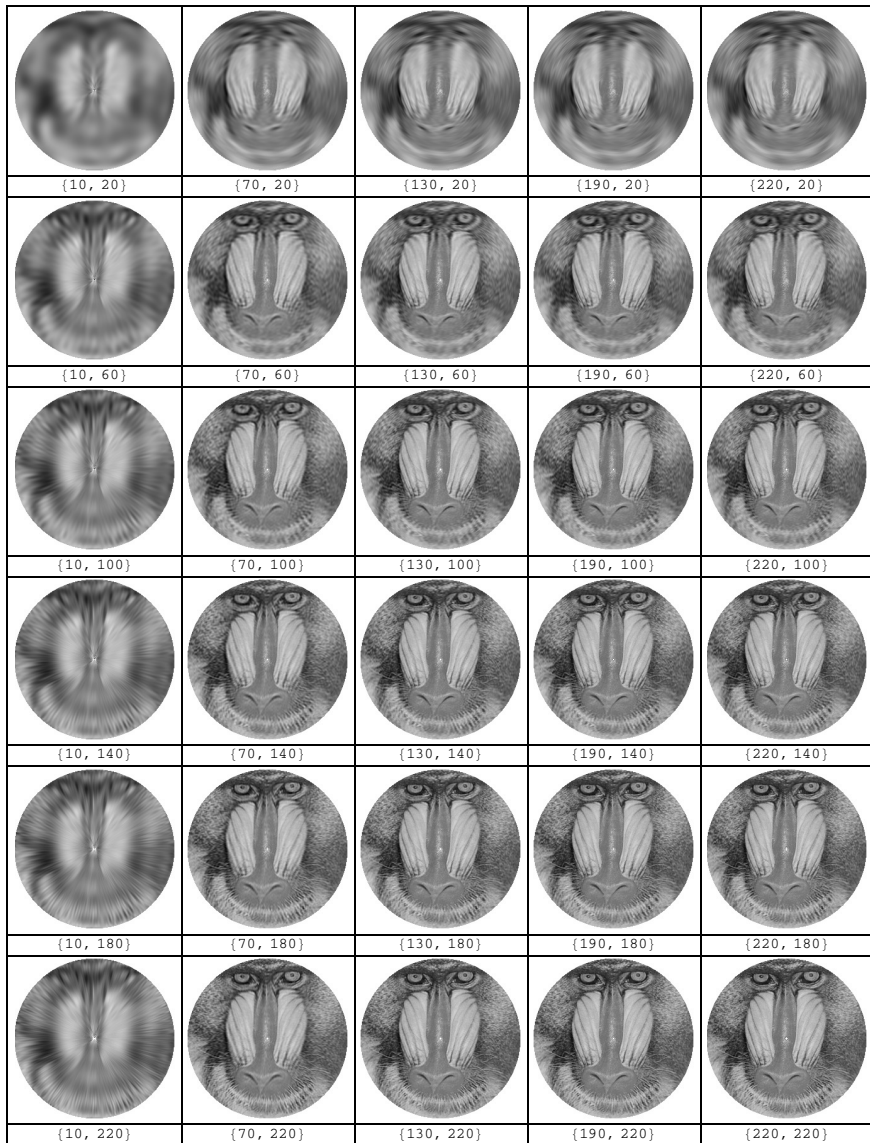


Figure 2.13: Reconstructed images from applying the numerical integration scheme  $k = 15$  with different maximum radial and harmonic orders.

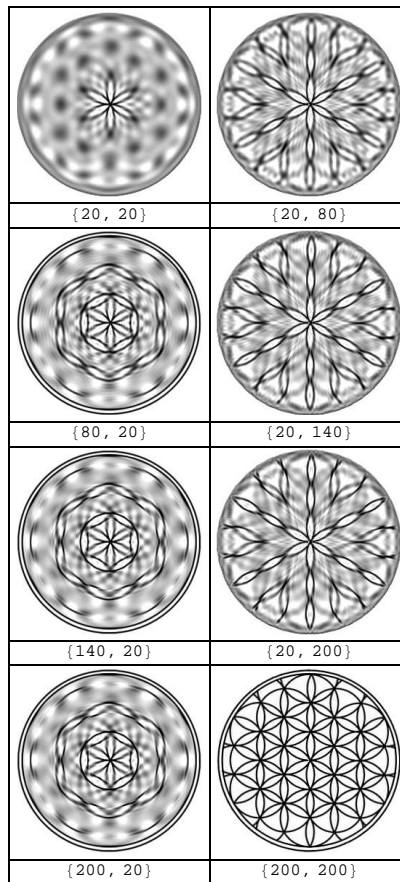


Figure 2.14: Some reconstructed images with different  $\{n, m\}$ .

order  $n$  of the orthogonal Fourier-Mellin moments hold more information on harmonic patterns.

## 2.3 Concluding Remarks

In this chapter, we have analyzed the general computational errors of moment functions defined in both the rectangular region and circular domain. We have also proposed some algorithms and techniques to improve the accuracy of moment computing, especially for the higher order of moment functions. To demonstrate the more precisely computed different types of conventional continuous orthogonal moments defined in both the rectangular region and circular domain, the image reconstructions from different types of conventional continuous orthogonal moment functions are examined with satisfied performances.

Based on our accuracy analysis of moment functions and the proposed solutions, we expect that further progress will be made in the research of various types of conventional continuous moments, defined in both the rectangular region and circular domain.

## Acknowledgment

The author wishes to thank Amy Chiang and Hong Dai for their help in preparing this chapter.

## References

- [1] A.B. Bhatia and E. Wolf. On the circle polynomials of zernike and related orthogonal sets. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 50, pages 40–48, 1954.
- [2] R. Courant and D. Hilbert. *Methods of mathematical physics*, volume 1. Wiley, 2008.
- [3] B.P. Flannery, W.H. Press, S.A. Teukolsky, and W. Vetterling. Numerical recipes in c. *Press Syndicate of the University of Cambridge, New York*, 1992.
- [4] J. Flusser, T. Suk, and B. Zitova. *Moments and Moment Invariants in Pattern Recognition*. Wiley, Chichester, 2009.
- [5] G.H. Hardy. On the expression of a number as the sum of two squares. *Quart. J. Math.*, 46:263–283, 1916.
- [6] M.K. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179–187, 1962.
- [7] A.E. Ingham. On two classical lattice point problems. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 36, pages 131–138, 1940.
- [8] E. Landau. *Vorlesungen über Zahlentheorie*. Berlin, 1927.
- [9] S.X. Liao and M. Pawlak. On image analysis by moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(3):254–266, 1996.

- [10] S.X. Liao and M. Pawlak. On the accuracy of Zernike moments for image analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1358–1364, 1998.
- [11] R. Mukundan and K.R. Ramakrishnan. *Moment Functions in Image Analysis: Theory and Applications*. World Scientific, Singapore, 1998.
- [12] M. Pawlak. *Image Analysis by Moments: Reconstruction and Computational Aspects*. Wroclaw University of Technology Press, Wroclaw, 2006. <http://www.dbc.wroc.pl/dlibra/doccontent?id=1432&from=&dirids=1>.
- [13] M. Pawlak and S.X. Liao. On the recovery of a function on a circular domain. *IEEE Transactions on Information Theory*, 48(10):2736–2753, 2002.
- [14] G. Sansone. *Orthogonal functions*, volume 9. Dover Publications, 1991.
- [15] Y. Sheng and L. Shen. Orthogonal Fourier-Mellin moments for invariant pattern recognition. *J. Opt. Soc. Am.*, 11(6):1748–1757, 1994.
- [16] M.R. Teague. Image analysis via the general theory of moments. *J. Opt. Soc. Am.*, 70(8):920–930, 1980.
- [17] C.H. Teh and R.T. Chin. On image analysis by the methods of moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):496–513, 1988.
- [18] von F. Zernike. Beugungstheorie des schneidenverfahrens und seiner verbesserten form, der phasenkontrastmethode. *Physica*, 1(7):689–704, 1934.
- [19] X. Wang and S. Liao. Image reconstruction from orthogonal Fourier-Mellin moments. In *Image Analysis and Recognition*, volume 7950 of *Lecture Notes in Computer Science*, pages 687–694. Springer Berlin Heidelberg, 2013.
- [20] Y. Xin, S. Liao, and M. Pawlak. Circularly orthogonal moments for geometrically robust image watermarking. *Pattern Recognition*, 40(12):3740–3752, 2007.
- [21] Y. Xin, M. Pawlak, and S. Liao. Accurate computation of Zernike moments in polar coordinates. *IEEE Transactions on Image Processing*, 16(2):581–587, 2007.