



ELSEVIER

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Fast computation of Krawtchouk moments

Barmak Honarvar Shakibaei Asli^{a,*}, Jan Flusser^b^a Integrated Lightwave Research Group, Department of Electrical, Faculty of Engineering, University of Malaya, Lembah Pantai, 50603 Kuala Lumpur, Malaysia^b Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Pod vodárenskou věží 4, 182 08 Praha 8, Czech Republic

ARTICLE INFO

Article history:

Received 24 May 2013

Received in revised form 18 July 2014

Accepted 29 July 2014

Available online 7 August 2014

Keywords:

Krawtchouk polynomial

Krawtchouk moment

Geometric moment

Impulse response

Fast computation

Digital filter

ABSTRACT

The paper describes the calculation of the Krawtchouk Moments (KMs) from an image, which is a computationally demanding task. We present two original methods that use the outputs of cascaded digital filters in deriving KMs. The first approach uses the digital filter outputs to form geometric moments (GMs) and the KMs are obtained via GMs. The second method uses a direct relationship to obtain KMs from the digital filter outputs. This is possible thanks to the formulation of Krawtchouk polynomials in terms of binomial functions, which are equivalent to the digital filter outputs. In this study, the performance of the proposed techniques is compared with other existing methods of KMs calculation. The experimental study shows that the first and the second proposed techniques perform 57% and 87% faster than the recurrence method for a real image of a size 128×128 pixels, which performs a significant improvement.

© 2014 Elsevier Inc. All rights reserved.

1. Introduction

Thanks to a rapid development of imaging technologies, sensors, and a growing number of application areas, computer image analysis has become an established discipline. One of the major areas is designing of efficient image descriptors. Among them, various kinds of moment functions have played an important role and have been widely used to extract image features for object recognition [12,5,10,6,53,16,24], template matching [15], edge detection [22], robot vision [31], robust watermarking [14,50,28] and data compression [1]. Moments are projections of the image onto a polynomial basis and often perform an advantageous alternative to harmonic [30] and locally-supported [48] bases. Since moments have been applied on images of complex scenes, we can observe an ongoing concern of the researchers for reducing the computation time without any loss of accuracy.

The computational time is determined basically by three factors. The first one is what polynomial basis we used for moment construction, the second one is the particular numerical algorithm, and the last one is the class of images we work with (for instance, moments of binary images can be calculated much faster than that of general graylevel images).

At the beginning, the authors mostly worked with geometric moments (i.e. moments w.r.t. the basis $x^p y^q$) and several frequently-cited papers used GMs to construct various invariants [23,10,11]. Hence, numerous methods for fast computation of GMs have appeared (see [12], Chapter 7, for a general survey and [39] for a survey of binary image algorithms).

In order to reach better numerical stability of the moments, several authors proposed to use various orthogonal polynomial bases and consequently *continuous orthogonal moments*. This idea comes from Teague [40] who proposed to employ Zernike moments (ZMs) which are orthogonal inside a unit circle. Orthogonality on a circle is an advantageous property

* Corresponding author. Tel.: +60 12 2554105.

E-mail addresses: barmak.honarvar@gmail.com (B. Honarvar Shakibaei Asli), flusser@utia.cas.cz (J. Flusser).

when one constructs rotation invariants. If this is not the main concern, then moments orthogonal on a square, such as Legendre moments (LMs) [41], Tchebichef moments (TMs) [34], Gauss–Hermite moments (GHMs) [49], Gegenbauer moments and Jacobi moments [12] are used more frequently because they do not require polar resampling of the image. Orthogonal moments have found a wide range of applications in character recognition, face recognition, direction-of-arrival estimation, and trademark segmentation and retrieval [38,8,9,25,13,37,21,17]. However, using these continuous moments for discrete images brings certain problems in computation: they require coordinate transformation of the image into the area of orthogonality and suitable discrete approximation of the integrals. Such an approximation usually violates the exact orthogonality and results in the occurrence of errors when calculating the moments numerically.

To eliminate this kind of errors, *discrete orthogonal moments* appeared in the literature. Typically they are constructed in 1D domain and their extensions to higher dimensions is done by multiplication of respective 1D polynomials which results in a discrete orthogonality on a hypercube. They are particularly suitable for digital images since they do not require coordinate transformation and the discretization step in the moment computation. Discrete Tchebichef, Krawtchouk, dual Hahn and Racah moments (TMs, KMs, HMs, and RMs) are typical representatives of this moment family [34,33,36,51,35,43,52,57,56].

Yap et al. [51] proposed to use the Krawtchouk moments as image descriptors. Thanks to discrete orthogonality of the KMs, their experimental results show that the KMs are better in terms of reconstruction error when compared with Zernike, Legendre and Tchebichef moments. They derived KM invariants by using indirect method (IDM). In IDM, moment invariants are obtained by using invariants of GMs. In [51], Krawtchouk polynomials are expressed in terms of monomials and KM invariants are derived using a linear combination of GM invariants. Hence by using these moment invariants, perfect invariance cannot be achieved for digital images since the derivation of these invariants are not based on Krawtchouk polynomials. Venkataramana and Raj [44] derived scale and translation invariants of KMs using direct method (DM). They derived the scale invariants by algebraically eliminating the scale factor contained in the scaled KMs. They proposed the modified KMs which preserve the translation invariant property by assuming that discrete weighted Krawtchouk polynomials are periodic with each period equal to the number of data points and these polynomials are shifted to the centroid. KMs have been also used in invariant image watermarking by Zhang et al. [54].

When considering numerical properties and implementation issues, the motivation for introducing OG moments becomes particularly apparent. Orthogonal polynomials can be evaluated by recurrent relations which yields in fast and numerically stable calculation. From this point of view it is not generally recommended to evaluate OG polynomials by expanding them into standard powers. Such approach would inevitably lead to a loss of precision due to possible overflow and/or underflow. Even if these expansions are commonly presented in literature, they should serve for illustration and/or theoretical considerations rather than for numerical calculations. Many algorithms have been developed for fast computation of moment functions such as GMs, ZMs, LMs, TMs and exact Zernike moments (EZMs). Hatamian [18] used cascaded all-pole filters to compute GMs consecutively. Honarvar and Paramesran [19] used a set of lower output values of digital filters to formulate the GMs. They derived the new formulation of GMs based on the Stirling numbers. Honarvar et al. [4] proposed another algorithm based on a reduced 2D digital filter structure for fast implementation of GMs. Kotoulas and Andreadis [26] applied an accumulator grid to compute ZMs in a real-time domain. Zhou et al. [55] proposed two new algorithms for efficient computation of LMs. Wang and Wang [46] presented parallel recursive computation of the inverse LMs for signal and image reconstruction. The same authors proposed a recursive method for computation of Tchebichef moments and their inverse transform using Clenshaw's formula [45]. Lim et al. [29] used cascaded digital filters to compute EZMs. Chang et al. [7] showed a hardware structure to implement TMs via GMs. Honarvar et al. [20] proposed a fast recursive computation of TM and its inverse transform based on Z-transform. Raj and Venkataramana [35] proposed a fast computation algorithm of inverse KM transform using Clenshaw's recurrence formula.

In this paper, we concentrate on numerical calculation of Krawtchouk moments since they already approved advantageous properties for image analysis. We propose two novel algorithms to derive the KMs by means of cascade digital filters with a suitable impulse response. In the first algorithm, the KMs are obtained from the GMs via digital filter outputs. In the second method, the Krawtchouk polynomials are modified to be used directly with the same digital filter outputs. The desired image is considered as an input of the proposed digital filter structure and the filter outputs are converted to KMs using linear combinations. Experimental results on real images demonstrates that this structure is faster than the traditional algorithms which have used the direct and recursive computation of KMs.

This paper is organized as follows. Section 2 presents the definition of Krawtchouk polynomials and moments. A brief review of GMs implementation using digital filter outputs is given in Section 3. Section 4 shows the proposed methods to compute KMs via cascaded digital filters. Section 5 validates the theoretical frameworks presented in the previous sections through experimental study. Concluding remarks are given in Section 6.

2. Krawtchouk polynomials and moments

In this section, we recall Krawtchouk polynomials and their expansion in terms of monomials. These polynomials were introduced by Soviet Ukrainian mathematician Mikhail P. Krawtchouk (sometimes spelled Kravchuk) in 1929 [27] and later generalized into a form of Meixner polynomials [32]. Krawtchouk polynomials were introduced into image processing area by Yap et al. [51].

2.1. Krawtchouk polynomials

The n th degree classical Krawtchouk polynomial is defined as

$$K_n(x; p, N) = \sum_{k=0}^n a_{k,n,p} x^k = {}_2F_1 \left(-n, -x; -N; \frac{1}{p} \right), \tag{1}$$

where $x, n = 0, 1, 2, \dots, N, N > 0, p \in (0, 1)$ is a localization parameter and ${}_2F_1(\cdot)$ is the generalized hypergeometric function

$${}_2F_1(a, b; c; z) = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k}{(c)_k} \frac{z^k}{k!}, \tag{2}$$

and $(x)_k$ is the Pochhammer symbol given by

$$(x)_k = x(x+1)(x+2) \cdots (x+k-1), \quad k \geq 1 \quad \text{and} \quad (x)_0 = 1. \tag{3}$$

The normalized and weighted Krawtchouk polynomials, $\bar{K}_n(x; p, N)$ are defined as [51]

$$\bar{K}_n(x; p, N) = K_n(x; p, N) \sqrt{\frac{w(x; p, N)}{\rho(n; p, N)}}, \tag{4}$$

where the weight function, $w(\cdot)$ and the square norm, $\rho(\cdot)$ are given as

$$w(x; p, N) = \binom{N}{x} p^x (1-p)^{N-x}, \tag{5}$$

and

$$\rho(n; p, N) = \left(\frac{p-1}{p} \right)^n \frac{n!}{(-N)_n}. \tag{6}$$

The normalized and weighted Krawtchouk polynomials have the following three-term recurrence relation:

$$p(n-N)\bar{K}_{n+1}(x; p, N) = A[p(N-2n) + n-x]\bar{K}_n(x; p, N) - Bn(1-p)\bar{K}_{n-1}(x; p, N), \tag{7}$$

where

$$A = \sqrt{\frac{(1-p)(n+1)}{p(N-n)}},$$

$$B = \sqrt{\frac{(1-p)^2(n+1)n}{p^2(N-n)(N-n+1)}},$$

with

$$\bar{K}_0(x; p, N) = \sqrt{w(x; p, N)},$$

$$\bar{K}_1(x; p, N) = \left(1 - \frac{1}{Np}x \right) \sqrt{w(x; p, N)}.$$

The weighted Krawtchouk polynomials up to the degree three for $p = 0.5$ and $p = 0.2$ are shown in Fig. 1(a) and (b), respectively. The graphs illustrate the role of the localization parameter p which enables to shift the polynomials to the region of interest.

2.2. Krawtchouk moments

The 2D Krawtchouk moment of order $(n+m)$ of a discrete image intensity function $f(x, y)$ with a support of $N \times M$ pixels is defined as [51]

$$Q_{nm} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \bar{K}_n(x; p_1, N-1) \bar{K}_m(y; p_2, M-1) f(x, y). \tag{8}$$

The orthogonality property leads to the following inverse moment transform

$$f(x, y) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} Q_{nm} \bar{K}_n(x; p_1, N-1) \bar{K}_m(y; p_2, M-1). \tag{9}$$

If only the moments of order up to (N_{max}, M_{max}) are computed, then the reconstructed image in (9) can be approximated by

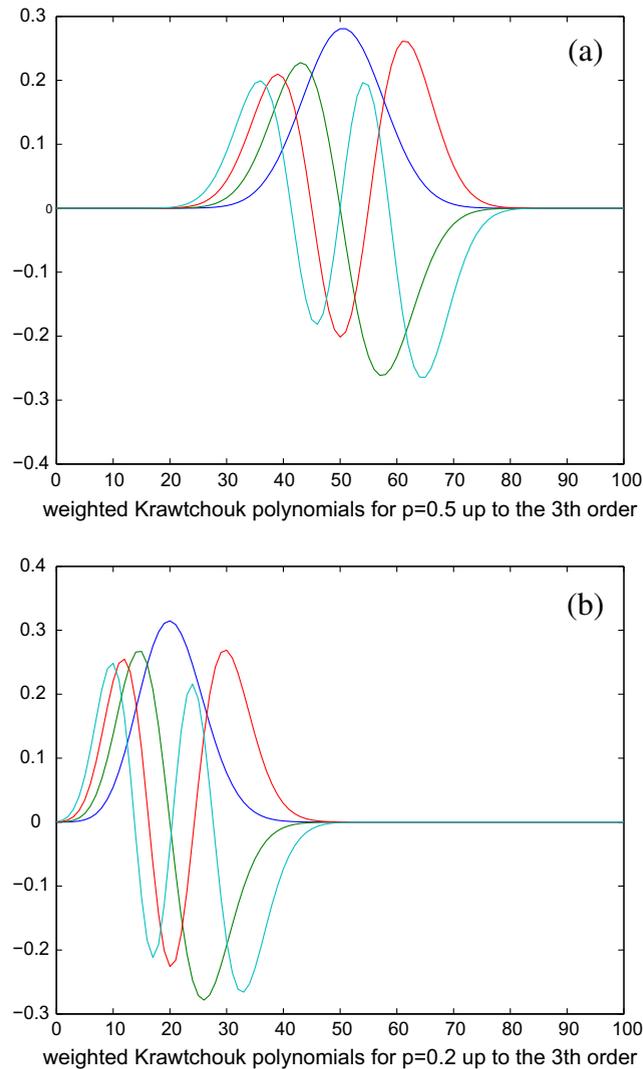


Fig. 1. Weighted Krawtchouk polynomials up to the degree three for $N = 100$: (a) $p = 0.5$ and (b) $p = 0.2$.

$$\tilde{f}(x, y) = \sum_{n=0}^{N_{\max}} \sum_{m=0}^{M_{\max}} Q_{nm} \bar{K}_n(x; p_1, N-1) \bar{K}_m(y; p_2, M-1). \quad (10)$$

The Krawtchouk moments in (8) can be expressed in terms of GMs as [51]

$$Q_{nm} = [\rho(n, p_1, N) \rho(m, p_2, M)]^{-\frac{1}{2}} \sum_{i=0}^n \sum_{j=0}^m a_{i,n,p_1} a_{j,m,p_2} m_{ij}, \quad (11)$$

where m_{ij} are the geometric moments that can be obtained from

$$m_{ij} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} x^i y^j f(x, y), \quad (12)$$

and $\{a_{k,n,p}\}$ are coefficient determined by (1).

3. Computation of the GMs via cascaded digital filters

Hatamian [18] proposed an all-pole filter based structure to compute GMs up to the 16th order. Fig. 2 shows a single pole digital filter with transfer function $1/(z-1)$, which is equivalent to an accumulator with unity feedback. This accumulator

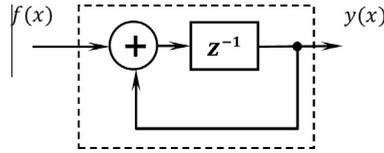


Fig. 2. Single-pole filter structure as an accumulator in feed-forward path.

has a delay in the feed-forward path. The cascaded digital filter outputs correspond to the transfer function $1/(z - 1)^{p+1}$, as can be shown in the following form

$$y_p(N + p + 1) = \sum_{x=0}^{N-1} f(x) \binom{x + p + 1}{p}, \tag{13}$$

where p is the GMs order and N is the length of 1D signal, $f(x)$. Notice that, the input image should be reversed. As shown in Fig. 4(a), the digital filter outputs are sampled at point $N + p + 1$.

The relationship between GMs and the all-pole digital filter outputs for a 2D image is expressed as

$$m_{p,q} = \sum_{r=0}^p C_{p,r} \sum_{s=0}^q C_{q,s} y_{r,s}, \tag{14}$$

where y_r is the r th digital filter output and $C_{p,r}$ is a matrix of coefficients directly obtained from the impulse response of the all-pole digital filters as given in [3]:

$$C_{pr} = \begin{cases} 0, & p < r, \\ (-1)^p, & r = 0, \\ rC_{p-1,r-1} - (r+1)C_{p-1,r}, & \text{otherwise.} \end{cases} \tag{15}$$

To overcome the problem caused by the delay elements in the feed-forward path, Wong and Siu [47] moved the delay element to the feedback path of the filter, as shown in Fig. 3. The transfer function is given as

$$H_p(z) = \left(\frac{1}{1 - z^{-1}} \right)^{p+1}. \tag{16}$$

In this case, the relationship between the digital filter outputs and the input signal is

$$y_p(N - 1) = \sum_{x=0}^{N-1} f(x) \binom{x + p}{p}. \tag{17}$$

As shown in Fig. 4(b), the outputs are sampled at the same point $N - 1$. However, for a 2D image, the GMs are obtained from the digital filter outputs and a different matrix coefficient is used, as shown below:

$$m_{p,q} = \sum_{r=0}^p D_{p,r} \sum_{s=0}^q D_{q,s} y_{r,s}, \tag{18}$$

where the coefficients of $D_{p,r}$ are obtained from the following recurrence formula shown in [2].

$$D_{pr} = \begin{cases} 0, & p > 0, r = 0, \\ 1, & p = 0, r = 0, \\ r(D_{p-1,r-1} - D_{p-1,r}), & \text{otherwise.} \end{cases} \tag{19}$$

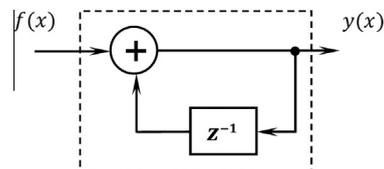
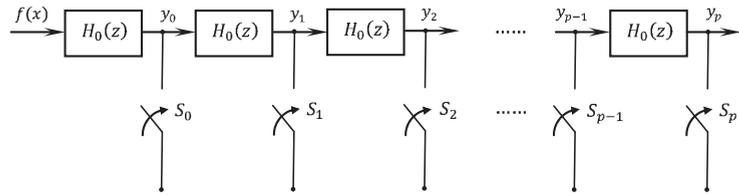


Fig. 3. Single-pole single-zero filter structure as an accumulator in feed-back path.



- (a) Sampling times for feed-forward accumulator → $y_0(N + 1)$ $y_1(N + 2)$ $y_3(N + 3)$ $y_{p-1}(N + p)$ $y_p(N + p + 1)$
- (b) Sampling times for feed-back accumulator → $y_0(N - 1)$ $y_1(N - 1)$ $y_3(N - 1)$ $y_{p-1}(N - 1)$ $y_p(N - 1)$
- (c) Sampling times for proposed method → $y_0(N - 1)$ $y_1(N - 2)$ $y_3(N - 3)$ $y_{p-1}(N - p)$ $y_p(N - p - 1)$

Fig. 4. 1D cascaded digital filter structure for generating desired outputs up to p th order based on three sampled outputs using different time intervals. (a) Hatamian’s approach using the sampled outputs at the increasing time intervals, $N + p + 1$. Each block is similar to the filter shown in Fig. 2. (b) Wong and Siu’s approach using the sampled outputs at the same time interval, $N - 1$. Each block is similar to the filter shown in Fig. 3 and (c) proposed approach using the sampled outputs at the decreasing time intervals, $N - p - 1$. Each block is similar to the filter shown in Fig. 3.

4. Computation of the KMs via cascaded digital filters

In this section, we first show that a cascaded feed-back digital filter outputs can be sampled at the earlier time intervals. In the second subsection, we define a weighted geometric moments based on the weight function of Krawtchouk polynomials to represent the KMs with respect to them. Third subsection shows how the KMs can be computed efficiently in terms of the cascaded digital filter outputs. Finally, a matrix representation of 2D KMs expresses the direct calculation of these moments based on the filter outputs.

4.1. Feed-back accumulator outputs with earlier sampled time intervals

The transfer function of a cascaded filter, $H_p(z) = 1/(1 - z^{-1})^{p+1}$ can be created by a series of p cascaded accumulators. Such a filter as shown in Fig. 4(c), given an input $f(x)$, produces a sampled output at earlier points of $N - p - 1$ as

$$y_p(N - p - 1) = \sum_{x=0}^{N-1} f(x) \binom{x}{p}. \tag{20}$$

As shown in Fig. 4(c), this formulation makes it possible for the digital filter outputs to be sampled at earlier points, $N - 1, N - 2, N - 3, \dots, N - p - 1$. Meanwhile, this set of output values starts to decrease after $p/2$ moment orders. In this case, the GMs are obtained from the digital filter outputs, y_r , as follows:

$$m_p = \sum_{r=0}^p E_{p,r} y_r, \tag{21}$$

where the coefficients of $E_{p,r}$ are calculated from the following recurrence formula:

$$E_{pr} = \begin{cases} 0, & p < r, \\ 1, & r = 0, \\ rE_{p-1,r-1} + (r + 1)E_{p-1,r}, & \text{otherwise.} \end{cases} \tag{22}$$

A 2D accumulator grid is shown in Fig. 5 and consists of one row and $(p + 1)$ columns based on single-pole single-zero filters. The filter output for an input image, $f(x, y)$ with size $N \times M$ is given by the following expression:

$$y_{p,q}(N - p - 1, M - q - 1) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} \binom{i}{p} \binom{j}{q} f(i, j). \tag{23}$$

4.2. Computation of KMs using GMs via digital filters

Since a weighted function, $w(x; p, N)$ is used to obtain the KMs, hence the GMs need to be weighted by the same function. We can define the GMs of a 2D weighted image $\tilde{f}(x, y)$ using the discrete approximation as

$$m_{pq} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} x^p y^q \tilde{f}(x, y), \tag{24}$$

where $\tilde{f}(x, y) = f(x, y)W(x, y)$ and $W(x, y)$ is defined as

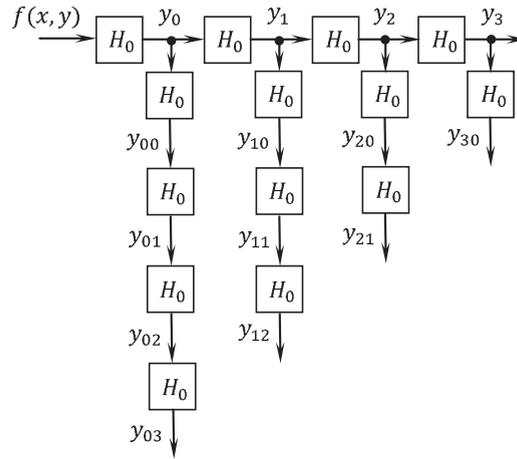


Fig. 5. A 2D digital filter structure for generating desired outputs up to third order. Each block in the array is a single-pole single-zero filter similar to the filter in Fig. 3.

$$W(x, y) = \sqrt{w(x; p_1, N - 1)w(y; p_2, M - 1)}. \tag{25}$$

By using (21), the GMs can be expressed as a set of linear combination of digital filter outputs:

$$m_{pq} = \sum_{r=0}^p \sum_{s=0}^q E_{pr} E_{qs} y_{r,s}, \tag{26}$$

where

$$y_{r,s} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \tilde{f}(x, y) \binom{x}{p} \binom{y}{q}. \tag{27}$$

Note that (27) is the weighted version of (20) in 2D form. We now show a relationship between the Krawtchouk polynomials and monomials. First, by expanding the Pochhammer symbol in terms of binomial coefficients, we obtain:

$$(-x)_k = (-1)^k k! \binom{x}{k}. \tag{28}$$

With the above identity and the definition of hypergeometric function in (2), the orthogonal Krawtchouk polynomial in (1) can also be written as

$$K_n(x; p; N) = \sum_{k=0}^n \frac{(-n)_k}{(-N)_k} \left(-\frac{1}{p}\right)^k \binom{x}{k}. \tag{29}$$

Eq. (29) can be expressed in terms of monomial, x as

$$K_n(x; p; N) = \sum_{k=0}^n C_k(n, p, N) \sum_{i=0}^k s(k, i) x^i, \tag{30}$$

where

$$C_k(n, p, N) = \frac{(-n)_k}{k!(-N)_k} \left(-\frac{1}{p}\right)^k,$$

and $s(k, i)$ are the Stirling numbers of the first kind [42], which satisfies

$$\binom{x}{k} = \frac{1}{k!} \sum_{i=0}^k s(k, i) x^i. \tag{31}$$

The polynomial expansion in (30) is useful in expressing the KMs in terms of GMs. Using (8) and (30), the KMs of an image can be expressed in terms of weighted GMs as follows:

$$Q_{nm} = A_n A_m \sum_{k=0}^n C_k(n, p_1, N - 1) \sum_{i=0}^m C_l(m, p_2, M - 1) \sum_{j=0}^k \sum_{j=0}^m s(k, i) s(l, j) m_{ij}, \tag{32}$$

where

$$A_n = [\rho(n; p, N - 1)]^{-\frac{1}{2}}$$

Using (26), the KMs of order (n, m) , can be obtained

$$Q_{nm} = A_n A_m \sum_{k=0}^n C_k(n, p_1, N - 1) \sum_{l=0}^m C_l(m, p_2, M - 1) \times \sum_{i=0}^k \sum_{j=0}^m s(k, i) s(l, j) \sum_{r=0}^i \sum_{s=0}^j E_{i,r} E_{j,s} y_{r,s}(N - r - 1, M - s - 1). \tag{33}$$

Fig. 6 shows the generating process of the KMs computation using filter outputs via GMs in block diagram form.

4.3. Direct computation of KMs via digital filters

In this subsection, a direct relationship between KMs and digital filter outputs is derived. Then, by using the 2D KMs definition in (8), the weighted and normalized functions in (4) and the binomial form of the Krawtchouk polynomials in (29), we obtain the following expression:

$$Q_{nm} = A_n A_m \sum_{k=0}^n k! C_k(n, p_1, N - 1) \binom{x}{k} \sum_{l=0}^m l! C_l(m, p_2, M - 1) \binom{y}{l} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \tilde{f}(x, y), \tag{34}$$

or

$$Q_{nm} = A_n A_m \sum_{k=0}^n k! C_k(n, p_1, N - 1) \sum_{l=0}^m l! C_l(m, p_2, M - 1) \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \tilde{f}(x, y) \binom{x}{k} \binom{y}{l}. \tag{35}$$

The last double summations of Eq. (35) are equivalent to the digital filter outputs which are described in (27). Therefore, the 2D KMs can be acquired from cascaded digital filter outputs using the following formula:

$$Q_{nm} = A_n A_m \sum_{k=0}^n \sum_{l=0}^m k! l! C_k(n, p_1, N - 1) C_l(m, p_2, M - 1) y_{k,l}(N - k - 1, M - 1 - l). \tag{36}$$

All the aforementioned stages are shown in block diagram form in Fig. 7.

4.4. Matrix representation of the proposed direct method

In matrix form, the generated KMs in (36) can be defined as a matrix \mathbf{Q}

$$\mathbf{Q} = (A_n A_m) \mathbf{\Psi}_1 \mathbf{Y} \mathbf{\Psi}_2^T, \tag{37}$$

where $(\cdot)^T$ denotes the transpose of the matrix and

$$\mathbf{Q} = \{Q_{nm}\}_{n,m=0}^{n,m=N-1}, \tag{38}$$

$$\mathbf{\Psi}_i = \left\{ \frac{\binom{n-m+1}{m}}{p_i^m (1-N)_m} \right\}_{n,m=0}^{n,m=N-1}, \quad i = 1, 2.$$

$$\mathbf{Y} = \{y_{nm}\}_{n,m=0}^{n,m=N-1}.$$

\mathbf{Y} is a 2D matrix with the elements of filter outputs. For a 4×4 image, the matrix representation of KMs in terms of filter outputs can be expressed as

$$\begin{pmatrix} Q_{00} & Q_{01} & Q_{02} & Q_{03} \\ Q_{10} & Q_{11} & Q_{12} & Q_{13} \\ Q_{20} & Q_{21} & Q_{22} & Q_{23} \\ Q_{30} & Q_{31} & Q_{32} & Q_{33} \end{pmatrix} = (A_n A_m) \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & \frac{1}{p_1(1-N)} & 0 & 0 \\ 1 & \frac{2}{p_1(1-N)} & \frac{1}{p_1^2(1-N)(2-N)} & 0 \\ 1 & \frac{3}{p_1(1-N)} & \frac{6}{p_1^2(1-N)(2-N)} & \frac{6}{p_1^3(1-N)(2-N)(3-N)} \end{pmatrix} \begin{pmatrix} y_{00} & y_{01} & y_{02} & y_{03} \\ y_{10} & y_{11} & y_{12} & y_{13} \\ y_{20} & y_{21} & y_{22} & y_{23} \\ y_{30} & y_{31} & y_{32} & y_{33} \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & \frac{1}{p_2(1-N)} & \frac{2}{p_2(1-N)} & \frac{3}{p_2(1-N)} \\ 0 & 0 & \frac{1}{p_2^2(1-N)(2-N)} & \frac{6}{p_2^2(1-N)(2-N)} \\ 0 & 0 & 0 & \frac{6}{p_2^3(1-N)(2-N)(3-N)} \end{pmatrix}. \tag{39}$$

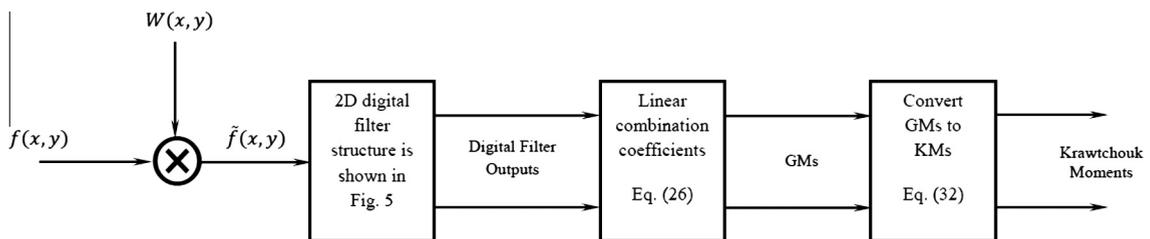


Fig. 6. Block Diagram for generating KMs via GMs.

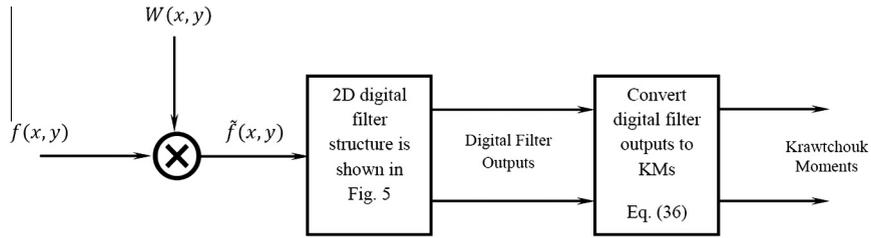


Fig. 7. Block diagram for direct generating of the KMs via filter outputs.

If $p_1 = p_2 = \frac{1}{2}$, the KMs of the second order can be reduced to the following equations:

$$\begin{aligned}
 Q_{02} &= \sqrt{3} \left(y_{00} - \frac{4}{3}y_{01} + \frac{4}{3}y_{02} \right), \\
 Q_{20} &= \sqrt{3} \left(y_{00} - \frac{4}{3}y_{10} + \frac{4}{3}y_{20} \right), \\
 Q_{11} &= 3 \left(y_{00} - \frac{2}{3}y_{01} - \frac{2}{3}y_{10} + \frac{4}{9}y_{11} \right).
 \end{aligned}
 \tag{40}$$

5. Experimental results

We have conducted various numerical experiments in order to prove the validity, accuracy and the efficiency of the proposed methods. The detailed description of these numerical experiments is presented in this section. The performance of the proposed methods is evaluated and compared with the existing methods for computing KMs. This section is divided into two subsections.

In the first subsection, the accuracy of the proposed methods is proved by computing of the KMs using recursive algorithm. Its efficiency is discussed with respect to the computational time of the KMs in the second subsection. CPU elapsed times are used to show the efficiency of the proposed methods where these elapsed times are computed and compared using both the proposed methods (filter method via GMs and filter direct method) and the existing conventional methods (KMs via GMs and recursive methods) in the same computing environment.

5.1. Accuracy of the proposed methods

In order to analyze the effect of Eqs. (8), (33) and (36) in computing of KMs, ten random orders of KMs were computed for two gray-scale real images of a resolutions of 128×128 ('Bridge') and 256×256 ('Man'). These images are benchmark images which are commonly used in the image processing field and are shown in Fig. 8.

The first experiment verifies the accuracy of the derived KMs. Table 1 displays the exact values of KMs derived by recursive method and those calculated by the proposed digital filter via GMs and the proposed digital filter direct method for 128×128 image. In this experiment, we compute the KM values for $p = 0.1, 0.5, 0.9$, respectively. It is clear from the



Fig. 8. Gray-scale images. (a) Bridge (128×128) and (b) Man (256×256).

Table that the KMs derived from the both methods are very close to the exact values but still the direct method is slightly more accurate than the filter via GMs. It is because Eq. (33) requires the computation of coefficients $C_k, E_{i,r}$ and the Stirling numbers, while Eq. (36) only requires the computation of coefficients C_k .

In the second experiment, we repeated the previous setup but used the image of ‘Man’ of a size of 256×256 pixels. Table 2 displays the KM values derived from (8) and compares them with the KM values derived from (33) and (36) for $p = 0.1, 0.5, 0.9$. We can see that the conclusion is basically the same as in the first experiment. Hence, we can summarize that both our proposed algorithms do not noticeably affect the values of KMs even if the parameter p changes.

5.2. Computational time

In moment calculation, the time is a critical issue because in general the calculation of moments is time expensive and fast algorithms may help a lot. Their importance is even more apparent if we are aware that a typical application of moments is in object recognition where a close-to-realtime performance is desirable. We tested the time complexity of the proposed methods (Eqs. (33) and (36)) and compared it to two reference algorithms: the indirect method (Eqs. (11) and (12)) and the recursive algorithm (Eqs. (7) and (8)). The experiments were performed on a PC equipped with 3.0 GHz CPU and 2 GB RAM.

In this experiment, four gray-scale standard benchmark images of different sizes were used as test images. The images are ‘Boat’, ‘Cameraman’, ‘Peppers’ and ‘Lena’ with resolutions of 64×64 , 128×128 , 256×256 and 512×512 pixels, respectively. They are shown in Fig. 9.

In all images we set $p_1 = p_2 = 0.5$ and computed KMs using the aforementioned algorithms. Table 3 contains the computation time up to the indices (60, 60) for the ‘Boat’ image. The results show that both our proposed methods perform by far better than the two traditional algorithms. Moreover, the method, which uses the filter output directly, significantly outperforms all others including the filter method via GMs. As an example, for zero order KM, the direct digital filter method is about 109 times faster than the recursive method and for 60th order of KMs for the same image, it is 28 times faster than the recursive method.

We repeated this experiment for the other three images. The maximum index of the evaluated KMs was always equal to the size of the image. The results are summarized in Table 4 for the ‘Cameraman’, in Fig. 10 for the ‘Peppers’, and in Fig. 11

Table 1

Comparison of the computed values of the KMs using recursive, digital filter via GMs and digital filter direct methods for image shown in Fig. 8(a).

Orders	$p_1 = p_2 = 0.1$			$p_1 = p_2 = 0.5$			$p_1 = p_2 = 0.9$		
	Recursive method Eqs. (7) and (8)	Proposed method Eq. (33)	Proposed method Eq. (36)	Recursive method Eqs. (7) and (8)	Proposed method Eq. (33)	Proposed method Eq. (36)	Recursive method Eqs. (7) and (8)	Proposed method Eq. (33)	Proposed method Eq. (36)
$Q_{0,0}$	11.3148	11.3150	11.3141	9.8762	9.8766	9.8759	5.1329	5.1331	5.1325
$Q_{9,50}$	-0.6951	-0.6947	-0.6953	0.0120	0.0122	0.1217	0.6478	0.6479	0.6473
$Q_{31,95}$	0.1204	0.1209	0.1201	-0.0971	-0.0968	-0.0969	0.3807	0.3803	0.3808
$Q_{4,54}$	0.6712	0.6720	0.6717	1.1741	1.1745	1.1743	1.2374	1.2375	1.2371
$Q_{63,110}$	-0.0548	-0.0535	-0.0541	0.0350	0.0353	0.0348	0.1752	0.1750	0.1755
$Q_{20,86}$	0.4140	0.4141	0.4144	0.0723	0.0725	0.0721	0.4279	0.4276	0.4281
$Q_{106,35}$	-0.2232	-0.2230	-0.2236	0.0202	0.0206	0.0202	0.1853	0.1858	0.1856
$Q_{97,44}$	-0.1389	-0.1382	-0.1392	-0.1022	-0.1025	-0.1020	0.3666	0.3669	0.3662
$Q_{127,14}$	0.0260	0.0263	0.0258	0.0127	0.0125	0.0129	-0.0148	-0.0150	-0.0144
$Q_{127,127}$	0.0027	0.0025	0.0029	-0.0190	-0.0192	-0.0188	-0.0211	-0.0209	-0.0215

Table 2

Comparison of the computed values of the KMs using recursive, digital filter via GMs and digital filter direct methods for image shown in Fig. 8(b).

Orders	$p_1 = p_2 = 0.1$			$p_1 = p_2 = 0.5$			$p_1 = p_2 = 0.9$		
	Recursive method Eqs. (7) and (8)	Proposed method Eq. (33)	Proposed method Eq. (36)	Recursive method Eqs. (7) and (8)	Proposed method Eq. (33)	Proposed method Eq. (36)	Recursive method Eqs. (7) and (8)	Proposed method Eq. (33)	Proposed method Eq. (36)
$Q_{0,0}$	9.5407	9.5402	9.5409	5.9698	5.9693	5.9694	13.3342	13.3343	13.3347
$Q_{15,8}$	-0.7600	-0.7603	-0.7606	-1.3402	-1.3406	-1.3401	1.1947	1.1942	1.1948
$Q_{34,158}$	0.2211	0.2208	0.2212	0.0538	0.0541	0.0532	0.1983	0.1980	0.1983
$Q_{127,127}$	0.4715	0.4716	0.4718	-0.0198	-0.0199	-0.0194	0.1725	0.1724	0.1727
$Q_{96,25}$	-0.6975	-0.6972	-0.6977	-0.0621	-0.0626	-0.0619	0.8048	0.8042	0.8049
$Q_{142,230}$	-0.0260	-0.0258	-0.0262	0.0027	0.0021	0.0029	0.0684	0.0682	0.0685
$Q_{185,31}$	-0.0890	-0.0888	-0.0889	0.0235	0.0231	0.0238	0.4821	0.4819	0.4823
$Q_{225,255}$	-0.0063	-0.0061	-0.0064	0.0105	0.0108	0.0102	-0.0109	-0.0110	-0.0107
$Q_{240,100}$	0.0067	0.0062	0.0064	-0.0589	-0.0582	-0.0589	-0.0111	-0.0116	-0.0113
$Q_{255,255}$	-0.00037	-0.00034	-0.00038	0.0106	0.0108	0.0105	0.00071	0.00071	0.00070

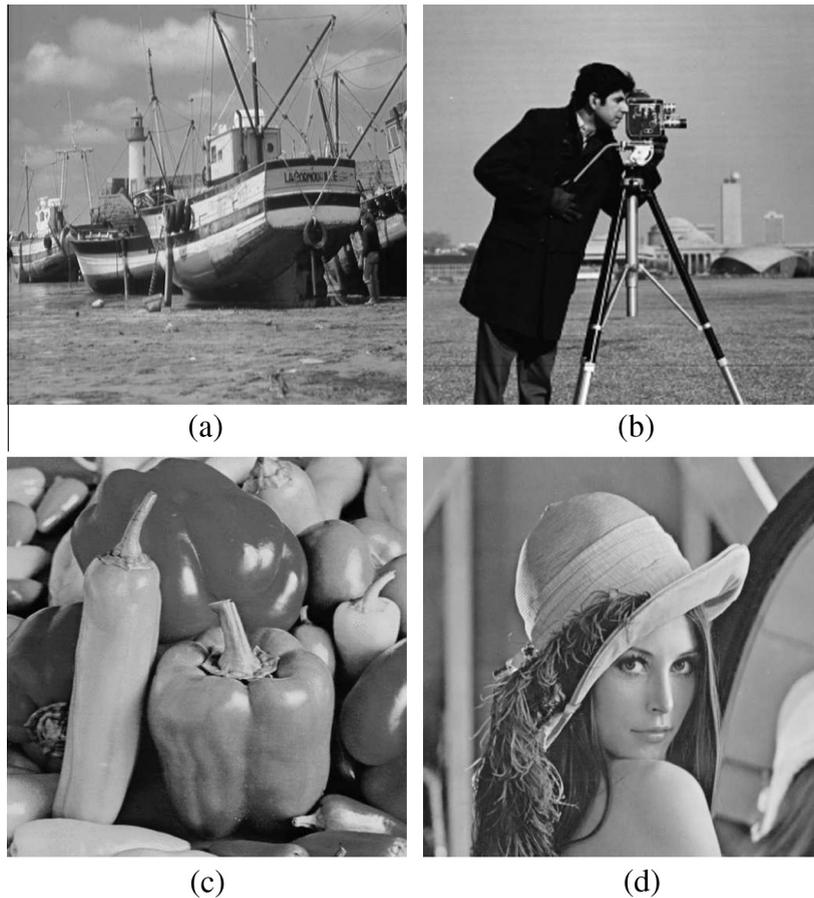


Fig. 9. Set of gray-scale images to compute the KMs using four different algorithms: (a) Boat (64×64), (b) Cameraman (128×128), (c) Peppers (256×256) and (d) Lena (512×512).

Table 3

Elapsed CPU times in milliseconds: computation of KMs for gray-scale 'Boat' image of size 64×64 shown in Fig. 9(a) using different methods.

Order	Expansion in terms of GMs using Eq. (32)	Recurrence method using Eq. (7)	Digital filter method via GMs using Eq. (33)	Digital filter method directly using Eq. (36)
0	150.8	78.0	3.4	0.71
5	187.2	176.4	7.9	1.08
10	270.4	250.4	15.6	1.96
15	473.2	374.4	37.5	2.76
20	577.2	452.7	51.9	3.28
25	728.0	535.6	62.8	3.95
30	987.1	660.3	81.7	4.74
35	1210.0	717.6	102.3	5.83
40	1354.5	790.9	194.5	8.65
45	1839.5	1686.3	287.5	10.37
50	2356.1	1976.8	401.4	24.93
55	2604.6	2133.5	596.3	59.24
60	2826.7	2356.7	719.6	82.75

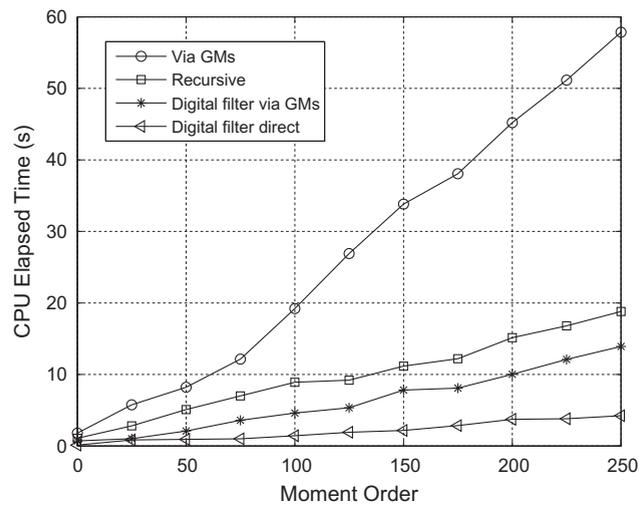
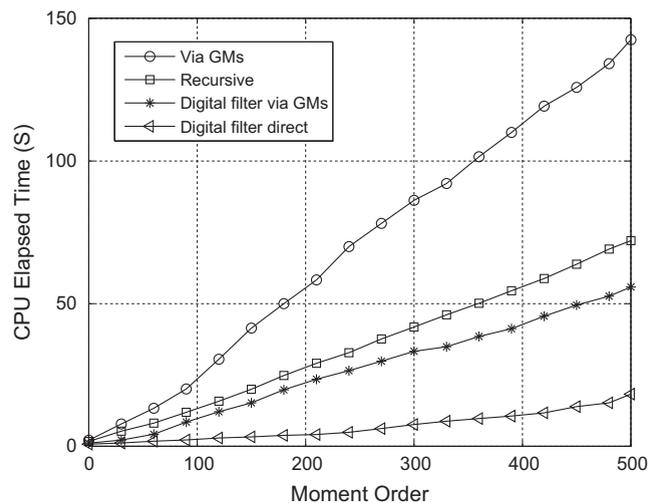
for the "Lena" image. In all these cases the results are consistent with the first experiment, regardless of the image size. Computation of the KMs using direct digital filter is significantly faster than the other methods. Computation by means of digital filtering via GMs is the second fastest one, still much faster than two reference methods.

6. Conclusion

The contribution of this paper lies in the efficient filter implementation of calculation of discrete Krawtchouk moments, which are important for image analysis. We have formulated the Krawtchouk polynomials in terms of only one combination

Table 4Elapsed CPU times in seconds: computation of KMs for gray-scale 'Cameraman' image of size 128×128 shown in Fig. 9(b) using different methods.

Order	Expansion in terms of GMs using Eq. (32)	Recurrence method using Eq. (7)	Digital filter method via GMs using Eq. (33)	Digital filter method directly using Eq. (36)
0	0.3767	0.1820	0.0079	0.00185
10	0.9874	0.7956	0.0198	0.00487
20	1.5666	1.4300	0.0736	0.00601
30	2.9812	2.1008	0.1051	0.00988
40	3.3498	2.8340	0.2765	0.01963
50	4.0987	3.5152	0.6129	0.05395
60	5.0827	4.1340	0.9972	0.20038
70	5.8766	4.7996	1.4725	0.36920
80	7.3478	5.3248	1.8973	0.49971
90	8.0865	6.1360	2.2549	0.66129
100	9.1591	6.9620	2.8723	0.78236
110	11.4554	7.4568	3.0145	0.97635
120	14.7125	7.9821	3.3984	1.00255

**Fig. 10.** Elapsed CPU times in seconds: full set of KMs for gray-scale 'Pepper' image of size 256×256 shown in Fig. 9(c) using different methods.**Fig. 11.** Elapsed CPU times in seconds: full set of KMs for gray-scale 'Lena' image of size 512×512 shown in Fig. 9(d) using different methods.

form to be used with filter outputs to generate KMs. We have proposed two fast algorithms, based on the cascaded filters, for computation of KMs of real gray-scale images. We have proved experimentally that both proposed methods yield almost accurate KM values with negligible errors. In terms of speed, which was our main concern, we proved that the both proposed methods perform significantly faster than the reference standard methods and that the direct digital filtering is faster than the digital filtering via GMs. At the same time we have proven that both new methods are suitable for large grayscale images. This is the main conclusion of the paper.

Acknowledgement

This work has been partially supported by the Grant No. P103/11/1552 of the Czech Science Foundation, Czech Republic.

References

- [1] N.A. Abu, S.L. Wong, H. Rahmalan, S. Sahib, Fast and efficient 4×4 Tchebichef moment image compression, *Majlesi J. Electr. Eng.* 4 (3) (2010).
- [2] M. Al-Rawi, Fast Zernike moments, *J. Real-Time Image Process.* 3 (1–2) (2008) 89–96.
- [3] M. Al-Rawi, Y. Jie, Practical fast computation of Zernike moments, *J. Comput. Sci. Technol.* 17 (2) (2002) 181–188.
- [4] B. Honarvar, R. Paramesran, K.-H. Chang, A reduced 2-D digital filter structure for fast implementation of geometric moments, 2011.
- [5] S. Belkasim, M. Shridhar, M. Ahmadi, Pattern recognition with moment invariants: a comparative study and new results, *Pattern Recognit.* 24 (12) (1991) 1117–1138.
- [6] J. Boyce, W. Hossack, Moment invariants for pattern recognition, *Pattern Recognit. Lett.* 1 (56) (1983) 451–456.
- [7] K.H. Chang, R. Paramesran, B. Honarvar, C.L. Lim, Efficient hardware accelerators for the computation of Tchebichef moments, *IEEE Trans. Circ. Syst. Video Technol.* 22 (3) (2012) 414–425.
- [8] S. Farokhi, S.M. Shamsuddin, J. Flusser, U.U. Sheikh, M. Khansari, K. Jafari-Khouzani, Rotation and noise invariant near-infrared face recognition by means of Zernike moments and spectral regression discriminant analysis, *J. Electron. Imaging* 22 (1) (2013) 1–11. Article No. 013030.
- [9] S. Farokhi, S.M. Shamsuddin, U.U. Sheikh, J. Flusser, M. Khansari, K. Jafari-Khouzani, Near infrared face recognition by combining Zernike moments and undecimated discrete wavelet transform, *Digital Signal Process.* 31 (0) (2014) 13–27.
- [10] J. Flusser, T. Suk, Pattern recognition by affine moment invariants, *Pattern Recognit.* 26 (1) (1993) 167–174.
- [11] J. Flusser, T. Suk, Degraded image analysis: an invariant approach, *IEEE Trans. Pattern Anal. Mach. Intell.* 20 (6) (1998) 590–603.
- [12] J. Flusser, T. Suk, B. Zitová, Moments and Moment Invariants in Pattern Recognition, Wiley, Chichester, 2009.
- [13] S. Ghosal, R. Mehrotra, Segmentation of range images: an orthogonal moment-based integrated approach, *IEEE Trans. Robot. Autom.* 9 (4) (1993) 385–399.
- [14] S. Golabi, M.S. Helfroush, H. Danyali, M. Owjimehr, Robust watermarking against geometric attacks using partial calculation of radial moments and interval phase modulation, *Inform. Sci.* 269 (0) (2014) 94–105.
- [15] A. Goshtasby, Template matching in rotated images, *Pattern Anal. Mach. Intell., IEEE Trans. PAMI-7* (3) (1985) 338–344.
- [16] L. Guo, M. Dai, M. Zhu, Quaternion moment and its invariants for color object classification, *Inform. Sci.* 273 (0) (2014) 132–143.
- [17] J.A. Martin, H.M. Santos, J. Lope, Orthogonal variant moments features in image analysis, *Inform. Sci.* 180 (6) (2010) 846–860.
- [18] M. Hatamian, A real-time two-dimensional moment generating algorithm and its single chip implementation, *IEEE Trans. Acoust., Speech Signal Process.* 34 (3) (1986) 546–553.
- [19] B. Honarvar, R. Paramesran, A new formulation of geometric moments from lower output values of digital filters, *J. Circ., Syst. Comput.* (2014) 1450055.
- [20] B. Honarvar, R. Paramesran, C.-L. Lim, The fast recursive computation of Tchebichef moment and its inverse transform based on Z-transform, *Digital Signal Process.* 23 (5) (2013) 1738–1746.
- [21] K.M. Hosny, A systematic method for efficient computation of full and subsets Zernike moments, *Inform. Sci.* 180 (11) (2010) 2299–2313.
- [22] H.S. Hsu, Moment preserving edge detection and its application to image data compression, *Opt. Eng.* 32 (1993) 1596–1608.
- [23] M.K. Hu, Visual pattern recognition by moment invariants, *IRE Trans. Inform. Theory* 8 (2) (1962) 179–187.
- [24] L. Ke, W. Qian, X. Jian, P. Weigu, 3D model retrieval and classification by semi-supervised learning with content-based similarity, *Inform. Sci.* 281 (0) (2014) 703–713.
- [25] Y. Kim, W. Kim, Content-based trademark retrieval system using a visually salient feature, *Image Vis. Comput.* 16 (1213) (1998) 931–939.
- [26] L. Kotoulas, I. Andreadis, Real-time computation of Zernike moments, *IEEE Trans. Circ. Syst. Video Technol.* 15 (6) (2005) 801–809.
- [27] M. Krawtchouk, Sur une généralisation des polynômes d'Hermite, *C.R. Acad. Sci., Paris* 189 (1929) 620–622.
- [28] L. Leida, L. Shushang, A. Ajith, P. Jeng-Shyang, Geometrically invariant image watermarking using polar harmonic transforms, *Inform. Sci.* 199 (0) (2012) 1–19.
- [29] C.-L. Lim, B. Honarvar, K.-H. Thung, P. Raveendran, Fast computation of exact Zernike moments using cascaded digital filters, *Inform. Sci.* 181 (17) (2011) 3638–3651.
- [30] L. Manhua, L. Shuxin, Z. Qijun, Fingerprint orientation field reconstruction by weighted discrete cosine transform, *Inform. Sci.* 268 (0) (2014) 65–77.
- [31] V. Markandey, R. deFigueiredo, Robot sensing techniques based on high-dimensional moment invariants and tensors, *IEEE Trans. Robot. Autom.* 8 (2) (1992) 186–195.
- [32] J. Meixner, Orthogonale polynomsysteme mit einer besonderen gestalt der erzeugenden funktion, *J. London Math. Soc.* s1-9 (1) (1934) 6–13.
- [33] R. Mukundan, Some computational aspects of discrete orthonormal moments, *IEEE Trans. Image Process.* 13 (8) (2004) 1055–1059.
- [34] R. Mukundan, S. Ong, P. Lee, Image analysis by Tchebichef moments, *IEEE Trans. Image Process.* 10 (9) (2001) 1357–1364.
- [35] P. Raj, A. Venkataramana, Fast computation of inverse Krawtchouk moment transform using Clenshaw's recurrence formula, in: *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, vol. 4, 16 2007–October 19 2007, pp. IV-37–IV-40.
- [36] H. Shu, H. Zhang, B. Chen, P. Haigr, L. Luo, Fast computation of Tchebichef moments for binary and grayscale images, *IEEE Trans. Image Process.* 19 (12) (2010) 3171–3180.
- [37] C. Singh, E. Walia, N. Mittal, Rotation invariant complex Zernike moments features and their applications to human face and character recognition, *IET Comput. Vis.* 5 (5) (2011) 255–265.
- [38] C. Singh, E. Walia, R. Upneja, Accurate calculation of Zernike moments, *Inform. Sci.* 233 (0) (2013) 255–275.
- [39] T. Suk, C. HöSchl IV, J. Flusser, Decomposition of binary images – a survey and comparison, *Pattern Recognit.* 45 (12) (2012) 4279–4291.
- [40] M.R. Teague, Image analysis via the general theory of moments, *J. Opt. Soc. Am.* 70 (8) (1980) 920–930.
- [41] C.-H. Teh, R. Chin, On image analysis by the methods of moments, *IEEE Trans. Pattern Anal. Mach. Intell.* 10 (4) (1988) 496–513.
- [42] N.M. Temme, *Special Functions: An Introduction to the Classical Functions of Mathematical Physics*, Wiley-Interscience, New York, 1996.
- [43] A. Venkataramana, P. Ananth Raj, Image watermarking using Krawtchouk moments, in: *Computing: Theory and Applications, 2007. ICCTA '07. International Conference on*, March 2007, pp. 676–680.
- [44] A. Venkataramana, P.A. Raj, Combined scale and translation invariants of Krawtchouk moments, in: *National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics, 2008*.
- [45] G. Wang, S. Wang, Recursive computation of Tchebichef moment and its inverse transform, *Pattern Recognit.* 39 (1) (2006) 47–56.

- [46] G.-B. Wang, S.-G. Wang, Parallel recursive computation of the inverse Legendre moment transforms for signal and image reconstruction, *IEEE Signal Process. Lett.* 11 (12) (2004) 929–932.
- [47] W.-H. Wong, W.-C. Siu, Improved digital filter structure for fast moments computation, *IEE Proc. – Vis., Image Signal Process.* 146 (2) (1999) 73–79.
- [48] W. Xiang-Yang, Y. Hong-Ying, Z. Yu, F. Zhong-Kai, Image denoising using SVM classification in nonsubsampling contourlet transform domain, *Inform. Sci.* 246 (0) (2013) 155–176.
- [49] B. Yang, M. Dai, Image analysis by Gaussian-Hermite moments, *Signal Process.* 91 (10) (2011) 2290–2303.
- [50] X. yang Wang, P. pan Niu, H. ying Yang, C. peng Wang, A. long Wang, A new robust color image watermarking using local quaternion exponent moments, *Inform. Sci.* 277 (0) (2014) 731–754.
- [51] P.T. Yap, R. Paramesran, S.H. Ong, Image analysis by Krawtchouk moments, *IEEE Trans. Image Process.* 12 (11) (2003) 1367–1377.
- [52] P.T. Yap, R. Paramesran, S.H. Ong, Image analysis using Hahn moments, *IEEE Trans. Pattern Anal. Mach. Intell.* 29 (11) (2007) 2057–2062.
- [53] H. Zhang, H. Shu, G. Han, G. Coatrieux, L. Luo, J. Coatrieux, Blurred image recognition by Legendre moment invariants, *IEEE Trans. Image Process.* 19 (3) (2010) 596–611.
- [54] L. Zhang, W. Xiao, Z. Ji, Local affine transform invariant image watermarking by Krawtchouk moment invariants, *IET Inform. Secur.* 1 (3) (2007) 97–105.
- [55] J. Zhou, H. Shu, L. Luo, W. Yu, Two new algorithms for efficient computation of Legendre moments, *Pattern Recognit.* 35 (5) (2002) 1143–1152.
- [56] H. Zhu, H. Shu, J. Liang, L. Luo, J.-L. Coatrieux, Image analysis by discrete orthogonal Racah moments, *Signal Process.* 87 (4) (2007) 687–708.
- [57] H. Zhu, H. Shu, J. Zhou, L. Luo, J.L. Coatrieux, Image analysis by discrete orthogonal dual Hahn moments, *Pattern Recognit. Lett.* 28 (13) (2007) 1688–1704.