Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Recursive Computation of Forward Krawtchouk Moment Transform Using Clenshaw's Recurrence Formula

A. Venkataramana, Lecturer
Dept of ECE, Government Polytechnic College
Mahabubnagar-509001, Andhrapradesh, India.
Email : avramana@yahoo.com

Dr. P.Ananth Raj, Professor
Dept of ECE, College of Engineering, Osmania University
Hyderabad- 500 007, Andhrapradesh, India.
Email: peruguananth@yahoo.com,

Abstract-Forward Krawtchouk moment transform (FKMT) needs more computations when a straightforward method of computation is chosen. Hence, this paper proposes a simple recursive method to reduce the number of computations using a Clenshaw's recurrence formula. Further, it is also shown that the number of computations can still be reduced by using the symmetry property of weighted Krawtchouk polynomials that are used to define FKMT. The proposed approach is verified with theoretical calculations as well as CPU time required for computation of FKMT of 1D and 2D points. The obtained recursive structure is simple and modular, Hence, it is suitable for parallel VLSI implementation.

KeywordsImageReconstruction,Verylargescaleintegration,Krawtchouk moment transform, Clenshaw's recurrence formula

1. INTRODUCTION

During the last few years, many moment transforms such as Geometric, Legendre, Zernike, Tchebichef, Krawtchouk etc. are proposed and applied for solving different problems in image processing and computer vision. In order to use them for real time applications, recursive algorithms are proposed. Some of the recursive algorithms proposed using Clenshaw's recurrence formula are reported in references [5-10]. Simulation results given in [4] show that Krawtchouk moments are better in terms of reconstruction error when compared with other moments. Hence, this paper proposes a recursive method for fast computation of FKMT for signal and image data points using Clenshaw's recurrence formula. The recursive structure proposed is simple and modular hence, it is suitable for parallel Very Large Scale Integration (VLSI) implementation.

This paper is organized into six sections. Background on Krawtchouk moments is given in section 2I. Section 3 presents details about Clenshaw's recurrence formula. Proposed recursive method of computation of FKMT for both signal and image data points is presented in section 4. Proposed method with symmetry property consideration is given section 5. Finally, the last section presents conclusions about the work.

2. BACKGROUND ON KRAWTCHOUK MOMENTS

The p[th] order forward Krawtchouk moments transform Q_p for a N point 1D signal $f(x)$ is defined as

$$Q_p = \sum_{x=0}^{N-1} \bar{K}_p(x;\lambda,N-1) f(x) \quad (1)$$

where $\bar{K}_p(x;\lambda,N-1)$ is the p[th] order weighted Krawtchouk polynomial [4] is defined as

$$\bar{K}_p(x;\lambda,N-1) = K_p(x;\lambda,N-1)\sqrt{\frac{w(x;\lambda,N-1)}{\rho(p;\lambda,N-1)}} \quad (2)$$

In the above expression, $K_p(x;\lambda,N-1)$ is the p[th] order discrete Krawtchouk polynomial [4] defined as

$$K_p(x;\lambda,N-1) = {}_2F_1(-p,-x;-(N-1);\frac{1}{\lambda}) \quad (3)$$

for x, p=0,1,2,….N-1, The parameter $\lambda \in (0,1)$, ${}_2F_1$ is the hypergeometric function, defined as

$${}_2F_1(a,b;c;z) = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k z^k}{(c)_k k!} \quad (4)$$

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

$$(a)_k = a(a+1)....(a+k-1) \quad (5)$$

The weight function $w(x;\lambda,N-1)$ is given by

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

The weight function $w(x;\lambda,N-1)$ can be recursively computed using

$$w(x+1;\lambda,N-1) = (\frac{N-1-x}{x+1})\frac{\lambda}{1-\lambda} w(x;\lambda,N-1) \quad (6)$$

$$with \quad w(0;\lambda,N-1) = (1-\lambda)^{N-1}$$

and $\rho(p;\lambda,N-1)$ is the squared norm, which is given by

$$\rho(p;\lambda,N-1) = (-1)^p (\frac{1-\lambda}{\lambda})^p \frac{p!}{(-N+1)_p}$$

The squared norm can be recursively calculated using the formula

$$\rho(p+1;\lambda,N-1) = \frac{(\lambda-1)(p+1)}{\lambda(p+1-N)} \rho(p;\lambda,N-1) \quad (7)$$

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

with $\rho(0;\lambda,N-1)=1$

Recursive formula to compute Krawtchouk polynomials with respect to 'x' is given by

$$\lambda(N-1-x)K_p(x+1;\lambda,N-1)=$$
$$(x+\lambda N-\lambda-p-2\lambda x)K_p(x;\lambda,N-1)$$
$$-x(1-\lambda)K_p(x-1;\lambda,N-1)$$

for $x=1\ to\ N-2$

with $K_p(0;\lambda,N-1)=1\ and\ K_p(1;\lambda,N-1)=1-\dfrac{p}{(N-1)\lambda}$

The recursive formula for weighted Krawtchouk polynomials is derived as

$$\bar{K}_p(x+1;\lambda,N-1)=\frac{(x+\lambda N-\lambda-p-2\lambda x)}{\sqrt{\lambda(N-1-x)(x+1)(1-\lambda)}}\bar{K}_p(x;\lambda,N-1)$$
$$-\sqrt{\frac{x(N-x)}{(x+1)(N-1-x)}}\bar{K}_p(x-1;\lambda,N-1) \tag{8}$$

The initial values for the above recursive formula can be obtained as

$$\bar{K}_p(0;\lambda,N-1)=K_p(0;\lambda,N-1)\sqrt{\frac{w(0;\lambda,N-1)}{\rho(p;\lambda,N-1)}}=\sqrt{\frac{w(0;\lambda,N-1)}{\rho(p;\lambda,N-1)}}$$

and

$$\bar{K}_p(1;\lambda,N-1)=K_p(1;\lambda,N-1)\sqrt{\frac{w(1;\lambda,N-1)}{\rho(p;\lambda,N-1)}}=(1-\frac{p}{(N-1)\lambda})\sqrt{\frac{w(1;\lambda,N-1)w(0;\lambda,N-1)}{\rho(p;\lambda,N-1)w(0;\lambda,N-1)}}$$
$$=(1-\frac{p}{(N-1)\lambda})\sqrt{\frac{w(1;\lambda,N-1)}{w(0;\lambda,N-1)}}\sqrt{\frac{w(0;\lambda,N-1)}{\rho(p;\lambda,N-1)}}=(1-\frac{p}{(N-1)\lambda})\sqrt{\frac{(N-1)\lambda}{(1-\lambda)}}\bar{K}_p(0;\lambda,N-1)$$

Given a set of Krawtchouk moments Q_p up to order $N_{\max}$ for a digital signal $f(x)$, one can compute $f(x)$ from Krawtchouk moments using the formula

$$f(x)=\sum_{p=0}^{N_{\max}}Q_p\bar{K}_p(x;\lambda,N-1) \tag{9}$$

Forward Krawtchouk moments transform Q_{pq} of order (p,q) for a digital image $f(x,y)$ of size N x M are defined[4] as

$$Q_{pq}=\sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\bar{K}_p(x;\lambda_1,N-1)\bar{K}_q(y;\lambda_2,M-1)f(x,y) \tag{10}$$

Given a set of Krawtchouk moments up to order ($N_{\max}$, $M_{\max}$) , then the inverse moment transform can be computed using

$$f(x,y)=\sum_{p=0}^{N_{\max}}\sum_{q=0}^{M_{\max}}Q_{pq}\bar{K}_p(x;\lambda_1,N-1)\bar{K}_q(y;\lambda_2,M-1) \tag{11}$$

3. CLENSHAW'S RECURRENCE FORMULA

Clenshaw's recurrence formula [9] is an efficient way to evaluate a sum of products of indexed coefficients by functions that satisfy a recurrence relation. Suppose that the desired sum is

$$I(x)=\sum_{n=0}^{J}c_nF_n(x) \tag{12}$$

in which $F_n(x)$ satisfies the recurrence relation as follows

$$F_{n+1}(x)=\alpha(n,x)F_n(x)+\beta(n,x)F_{n-1}(x) \tag{13}$$

for some functions $\alpha(n,x)$ and $\beta(n,x)$

Then Clenshaw's recurrence formula states that the sum $I(x)$ can be evaluated by

$$I(x)=\beta(1,x)F_0(x)\psi_2+F_1(x)\psi_1+c_0F_0(x) \tag{14}$$

where the quantities ψ_n can be obtained from the following recurrence:

$$\psi_{J+2}=\psi_{J+1}=0$$
$$\psi_n=\alpha(n,x)\psi_{n+1}+\beta(n+1,x)\psi_{n+2}+c_n \tag{15}$$

for $n=J,J-1,......1$ and solve backward to obtain ψ_2 and ψ_1

4. PROPOSED RECURSIVE METHOD

In this section, we propose a recursive method for fast computation of FKMT for signal and image data points using a Clenshaw's recurrence formula. To compute FKMT for a 1D signal as given in eq.(1) using the Clenshaw's recurrence formula, let us consider

$$F_x(p)=\bar{K}_p(x;\lambda,N-1) \tag{16}$$

Comparing eq.(13) with eq.(8), we get

$$\alpha(x,p)=\frac{(x+\lambda N-\lambda-p-2\lambda x)}{\sqrt{\lambda(N-1-x)(x+1)(1-\lambda)}} \tag{17}$$

and

$$\beta(x,p)=\beta(x)=-\sqrt{\frac{x(N-x)}{(x+1)(N-1-x)}} \tag{18}$$

According to eq.(15), we define

$$\psi_{N+1}=\psi_N=0$$
$$\psi_x=\alpha(x,p)\psi_{x+1}+\beta(x+1)\psi_{x+2}+f(x) \tag{19}$$
$$for\ x=N-1,........,\ 0$$

with $\alpha(N-1,p)=0,\ \beta(N-1)=0,\ \beta(N)=0$

Using eq.(14), we get

$$Q_p=[f(0)F_0(p)+F_1(p)\psi_1+\beta(1)F_0(p)\psi_2]$$
$$=[f(0)\bar{K}_p(0;\lambda,N-1)+\bar{K}_p(1;\lambda,N-1)\psi_1+\beta(1)\bar{K}_p(0;\lambda,N-1)\psi_2] \tag{20}$$

From eq.(19), we get

$\psi_0=\alpha(0,p)\psi_1+\beta(1)\psi_2+f(0)$ from which we obtain

$$f(0)=\psi_0-\alpha(0,p)\psi_1-\beta(1)\psi_2$$

Substituting the above $f(0)$ in eq.(20),

$$Q_p=\psi_0\bar{K}_p(0;\lambda,N-1)-\alpha(0,p)\psi_1\bar{K}_p(0;\lambda,N-1)$$
$$-\beta(1)\psi_2\bar{K}_p(0;\lambda,N-1)+\bar{K}_p(1;\lambda,N-1)\psi_1+\beta(1)\psi_2\bar{K}_p(0;\lambda,N-1)$$
$$=\psi_0\bar{K}_p(0;\lambda,N-1)-\alpha(0,p)\psi_1\bar{K}_p(0;\lambda,N-1)+\bar{K}_p(1;\lambda,N-1)\psi_1$$
$$=\psi_0\bar{K}_p(0;\lambda,N-1)+\psi_1[\bar{K}_p(1;\lambda,N-1)-\alpha(0,p)\bar{K}_p(0;\lambda,N-1)]$$

By substituting the values of $\bar{K}_p(1;\lambda,N-1)\ and\ \alpha(0,p)$ in the above equation, we obtain

$$Q_p=\psi_0\bar{K}_p(0;\lambda,N-1)+\psi_1[(1-\frac{p}{(N-1)\lambda})\sqrt{\frac{(N-1)\lambda}{(1-\lambda)}}\bar{K}_p(0;\lambda,N-1)$$
$$-\frac{(\lambda N-\lambda-p)}{\sqrt{\lambda(N-1)(1-\lambda)}}\bar{K}_p(0;\lambda,N-1)]$$

which on simplification gives

$$Q_p=\psi_0\bar{K}_p(0;\lambda,N-1) \tag{21}$$

The proposed recursive approach for the 1D FKMT (Q_p) computation can be summarized as follows

$$\begin{aligned} &Initialize\ \ \psi_{N+1} = \psi_N = 0 \\ &Do\ \ for\ \ x = N-1:-1:0 \\ &\psi_x = \alpha(x,p)\psi_{x+1} + \beta(x+1)\psi_{x+2} + f(x) \\ &end \end{aligned} \quad (22)$$

$$Q_p = \psi_0 \bar{K}_p(0;\lambda,N-1)$$

In the above presented approach, we recursively generate ψ_x from a input sequence $f(x)$ $(x = N-1,....,0)$. At N[th] step, we obtain ψ_0 which is used to evaluate Q_p as given in (22). A recursive structure for implementation of 1D FKMT (Q_p) according to eq.(22) is shown in Fig.1. The box z^{-1} shown in figure represents a delay element.

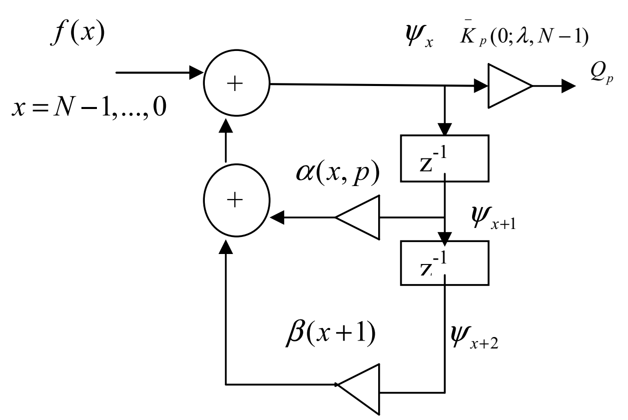

Figure1. Recursive structure for the 1 D FKMT computation

In the proposed recursive method for computing each FKMT, we need (2N+1) multiplications and 2N additions provided that the recursive coefficients $\alpha(x,p)$ *and* $\beta(x+1)$ are pre-computed. In order to compute the FKMT as given in (1) by using the straightforward method, the polynomial values are first computed using eq.(8). For each 'x', eq.(8) requires 2 multiplications and 1 addition provided that the recursive coefficients and the values of $\bar{K}_p(0;\lambda,N-1)$ and $\bar{K}_p(1;\lambda,N-1)$ are also pre-computed and saved. A total of 2(N-2) multiplications and (N-2) additions are required for computing all (N-2) values of 'x'. Further, N multiplications and (N-1) additions are required in the summing process of eq.(1). As a result, a total of 2(N-2)+N=(3N-4) multiplications and (N-2)+(N-1)=(2N-3) additions are required for computation of single p[th] order moment by using straightforward method. The same is given in table1. It is observed that there is a slight increase in the number of additions when compared with straightforward method, but the total number of computations (multiplications + additions) are reduced.

The above presented approach can be extended to 2D image data points. The FKMT for 2D case as given in eq.(10) for parameters $\lambda_1 = \lambda_2 = \lambda$ can be expressed as

$$Q_{pq} = \sum_{x=0}^{N-1} \bar{K}_p(x;\lambda,N-1)\, w_x(q) \quad (23)$$

where

$$w_x(q) = \sum_{y=0}^{M-1} \bar{K}_q(y;\lambda,M-1) f(x,y) \quad (24)$$

The coefficients $w_x(q)$ defined in eq.(24) are evaluated first for each $x = N-1,.....,1,0$ according to the above presented approach.

$$\begin{aligned} &Initialize\ \ \psi_{M+1} = \psi_M = 0 \\ &Do\ \ for\ \ y = M-1:-1:0 \\ &\psi_y = \alpha(y,q)\psi_{y+1} + \beta(y+1)\psi_{y+2} + f(x,y) \\ &end \end{aligned} \quad (25)$$

$$w_x(q) = \psi_0 \bar{K}_q(0;\lambda,M-1)$$

Then $w_x(q)$ are used to evaluate Q_{pq} defined in eq.(23) which is given below

$$\begin{aligned} &Initialize\ \ \varphi_{N+1} = \varphi_N = 0 \\ &Do\ \ for\ \ x = N-1:-1:0 \\ &\varphi_x = \alpha(x,p)\varphi_{x+1} + \beta(x+1)\varphi_{x+2} + w_x(q) \\ &end \end{aligned} \quad (26)$$

$$Q_{pq} = \varphi_0 \bar{K}_p(0;\lambda,N-1)$$

The computational complexity is as follows. In eq.(25), the evaluation of $w_x(q)$ requires (2M+1) multiplications and 2M additions for each value of 'x'. Since there are N values of 'x', a total of (2M+1)N = 2MN+N multiplications and 2MN additions are required. Eq.(26) requires (2N+1) multiplications and 2N additions. Hence, computation of one Q_{pq} using the proposed implementation of eqs.(25) and (26) requires a total of (2MN+N)+(2N+1) = (2MN+3N+1) multiplications and (2MN+2N) additions. The straightforward method of computation of one 2D FKMT is just two stage computation as given in eqs.(23) and(24). The first stage computation of eq.(23) can be considered as 1D FKMT computation for 'N' values of 'x'. For each 'x', a single 1D moment calculation requires (3M-4) multiplications and (2M-3) additions. Hence, the total number of computations for 'N' values of 'x' is (3M-4)N multiplications and (2M-3)N additions. The second stage computation of eq.(24) can also be seen as a single 1D computation which requires another (3N-4) multiplications and (2N-3) additions. Hence the straightforward method of computation of single 2D Krawtchouk moment transform as given in eqs.(23) and (24) requires a total of (3M-4)N+(3N-4) = (3MN-N-4) multiplications and (2M-3)N+(2N-3) = (2MN-N-3) additions. This is shown in table1.

5. SYMMETRY PROPERTY CONSIDERATION

The symmetry property [4] of weighted Krawtchouk polynomials is given by

$$\bar{K}_p(x;\lambda,N-1) = (-1)^p \bar{K}_p(N-1-x;\lambda,N-1) \quad (27)$$

In the case of 'N' even, Krawtchouk moments as given in eq.(1) can be expressed as

$$Q_p = \sum_{x=0}^{\frac{N}{2}-1} \bar{K}_p(x;\lambda,N-1) g(x) \quad (28)$$

where $g(x) = f(x) + (-1)^p f(N-1-x)$ (29)

for $x = 0, 1, ... \frac{N}{2} - 1$

In case of 'N' odd, the signal can be zero-padded to make the 'N' even. Similar to the approach for 1D FKMT computation presented in section IV, eq.(28) can be computed as follows

$$Initialize\ \psi_{\frac{N}{2}+1} = \psi_{\frac{N}{2}} = 0$$

$$Do\ for\ x = \frac{N}{2} - 1 : -1 : 0$$

$$\psi_x = \alpha(x, p)\psi_{x+1} + \beta(x+1)\psi_{x+2} + g(x) \quad (30)$$

$$end$$

$$where\ g(x) = f(x) + (-1)^p f(N-1-x)$$

$$Q_p = \psi_0 \bar{K}_p(0; \lambda, N-1)$$

This approach first pre-operates the input sequence $f(x)$ $(x = N-1, ..., 0)$ to obtain the new sequence $g(x)$ $(x = \frac{N}{2} - 1, ..., 0)$ then ψ_x is recursively generated from the new input sequence. At the $\frac{N}{2}^{th}$ step, the transform Q_p is computed. The recursive structure for this case is same as shown in figure1 with $f(x)$ replaced by $g(x)\ for\ (x = \frac{N}{2} - 1, ..., 0)$ which can be computed by using pre-operation given by eq.(29). In the above method, pre-operation requires N/2 additions. Hence, a total of (N+1) multiplications and (N+N/2) = 3N/2 additions are required for computation of each transform element Q_p. Number of multiplications are reduced by 'N' and the number of additions are reduced by N/2 when compared with the proposed approach with out symmetry considerations. Hence, the proposed method with symmetry property consideration can provide faster results than the approach with out considering the symmetry property of weighted Krawtchouk polynomials. We extend the above presented approach to 2D case for the computation of FKMT of image $f(x, y)$. First we compute the coefficients $w_x(q)$ for each $x = \frac{N}{2} - 1, ..., 1, 0$ using the above presented method for 1D case. Then $w_x(q)$ are used to evaluate Q_{pq}. The number of multiplications and additions required for computation of 1D and 2D FKMT using symmetry property is also given in table1.

To know the cpu time used for the calculation of FKMT, 'cputime' function available in MATLAB is used. The programs are run on 3GHz, Pentium 4 processor. The computed cpu time for calculation of various orders (40, 80, 120) of FKMT for Lena image using the straightforward method without and with symmetry and proposed recursive method without and with symmetry property consideration are (9.31, 26.78,60.40), (6.78,26.67, 60.03), (11.09,24.50,40.85) and (3.71,8.67,14.93) seconds respectively. It is found that proposed recursive computation of FKMT with symmetry property consideration takes lesser computation time.

Table.1. Computational complexity comparison for a single 1D FKMT (Q_p) and 2D FKMT (Q_{pq})

Method	Multiplications 1D : 2D	Additions 1D : 2D
Straightforward method	(3N-4) : (3MN-N-4)	(2N-3) :(2MN- N-3)
Proposed recursive method	(2N+1):(2MN+3N+1)	(2N):(2MN+2N)
Straightforward method with symmetryconsideration	(3N/2-4):(3MN/4-N/2-4)	(3N/2-3): (5MN/4-N/2-3)
Proposed recursive method with symmetry property consideration	(N+1):(MN/2+3N/2+1)	(3N/2):(5MN/4+N)

6. CONCLUSIONS

This paper proposed a simple recursive method for fast computation of FKMT using Clenshaw's recurrence formula. It is shown that the number of computations can still be reduced by considering the symmetry property of weighted Krawtchouk polynomials. The proposed approach is simple and modular in structure and hence, suitable for low cost parallel VLSI implementation.

REFERENCES

[1] Ming-Kuin Hu , " Visual Pattern Recognition by Moment Invariants," *IRE Transaction on Information Theory*, vol.8, pp.179-187, Feb.1962.

[2] M.R.Teague, " Image Analysis Via the General Theory of Moments" , *J. optical soc Am.,* vol.70, pp.920-930 , Aug.1980

[3] R.Mukundan, S.H.Ong and P.A.Lee, " Image Analysis by Tchebichef Moments", *IEEE Transaction on Image Processing*, vol.10, no.9, pp.1357-1363, Sept.2001.

[4] Pew-Thian Yap, Raveendran Parameshran and S.H. Ong "Image Analysis by Krawtchouk Moments", *IEEE Transaction on Image Processing*,vol.12, no.11,pp.1367-1377, Nov.2003

[5] H.C.Chiang and J.C.Liu, "Recursive implementation for the forward and inverse MDCT in MPEG audio coding*," IEEE Signal Processing Lett*., vol.3,pp.116-118,April1996.

[6] Vladimir Nikolajevic and Gerhard Fettweis, " Computation of forward and inverse MDCT using Clenshaw's recurrence formula," *IEEE Transaction on Signal Processing*,vol.51,no.5pp.1439- 1444, May2003.

[7] Guo-Bao Wang and Shi-gang Wang, "Parallel recursive computation of the inverse Legendre moment transforms for signal and image reconstruction" *IEEE Signal Processing Lett*., vol.11,no.12,pp.929-932,Dec.2004.

[8] Guobao Wang and Shigang Wang, "Recursive Computation of Tchebichef Moment and its inverse transform," *Pattern Recognition*, vol.39,pp.47-56,2006

[9] W.Press, S.A.Teukolsky, W.T.Vetterling and B.P.Flannery, Numerical Recipes in C, Cambridge University Press, U.K., 1992.

[10] P.Ananth Raj and A.Venkataramana, " Fast Computation of Inverse Krawtchouk Moment Transform using Clenshaw's Recurrence Formula, " accepted for publication in *IEEE Int. Conf. in Image Processing*, Sept.9-13,2007, SanAntonio,Texas,USA

[11] Botros, N.M.Jian Yang Feinsilver, P.Schott, " Hardware realization of Krawtchouk transform using VHDL and FPGAs", IEEE Transaction on Industrial Electronics, vol.49, pp.1306-1312, Dec.2002.