SIViP (2014) 8:1543–1553
DOI 10.1007/s11760-012-0391-4

ORIGINAL PAPER

Image and video processing using discrete fractional transforms

Neeru Jindal · Kulbir Singh

Received: 1 December 2011 / Revised: 9 October 2012 / Accepted: 9 October 2012 / Published online: 23 October 2012

Abstract The mathematical transforms such as Fourier transform, wavelet transform and fractional Fourier transform have long been influential mathematical tools in information processing. These transforms process signal from time to frequency domain or in joint time–frequency domain. In this paper, with the aim to review a concise and self-reliant course, the discrete fractional transforms have been comprehensively and systematically treated from the signal processing point of view. Beginning from the definitions of fractional transforms, discrete fractional Fourier transforms, discrete fractional Cosine transforms and discrete fractional Hartley transforms, the paper discusses their applications in image and video compression and encryption. The significant features of discrete fractional transforms benefit from their extra degree of freedom that is provided by fractional orders. Comparison of performance states that discrete fractional Fourier transform is superior in compression, while discrete fractional cosine transform is better in encryption of image and video. Mean square error and peak signal-to-noise ratio with optimum fractional order are considered quality check parameters in image and video.

Keywords Discrete fractional Fourier transforms · Discrete fractional Cosine transforms · Discrete fractional Hartley transforms · Image · Video

N. Jindal (✉) · K. Singh
Department of ECE, Thapar University, Patiala 147004, Punjab, India
e-mail: neerusingla99@gmail.com

K. Singh
e-mail: ksingh@thapar.edu

1 Introduction

The French scientist Jean Baptiste Joseph Fourier suggested the use of Fourier transform (FT) while solving a heat conduction dilemma in 1807. However, with the extension of research theme and area, the fractional power of FT operator appeared in the mathematical literature in 1929 [1–3]. The fractional Fourier transform (FrFT) has later on established many applications in the area of quantum mechanics [4], signal processing [5–7], pattern recognition [8], and optical, image and video processing [9–11]. The FrFTs are well known as rotational Fourier transform or angular Fourier transform in few papers [12,13]. The continuous FrFT is often implemented in optics [14,15]. It is a recognized fact that difference equations have a richer solution set than their continuous limit differential equations [16]. The window functions in the fractional domain using different discrete fractional Fourier transform (DFrFT) classes have been analyzed in [17]. Fractional Fourier domain decomposition for continuous and discrete signals and systems has been introduced in [18]. Nowadays, digital implementation of fractional Fourier transform is applied in many applications areas [19–24]. Several discrete fractional transforms have been successfully used for one-dimensional (1-D), two-dimensional (2-D) and three-dimensional (3-D) signals [19–25].

The goal of this paper is to review the definitions of discrete fractional Fourier transform (DFrFT), discrete fractional cosine transform (DFrCT) and discrete fractional Hartley transform (DFrHT) and their applications in the area of image and video processing. The paper is organized as follows: Section 2 presents definitions of Fractional transforms, 1-D/2-D DFrFT, DFrCT, DFrHT. Section 3 presents the simulation results for image and video compression and encryption using discrete fractional transforms. The conclusion of this paper is given in the last section.

2 Fractional transforms

In the last decade, researchers have delved in the fractional transforms and their applications [27,28]. Fractional transforms are one of the key tools in many disciplines such as communication, signal processing, and control systems etc. The Namias in 1980 [4] introduced FrFT, a generalization of the classical FT. The FrFT is a cohesive time-frequency transform which can reveal the characteristics of signal gradually changing from time domain to frequency domain with its order increasing from 0 to 1. Performance of FrFT becomes better with extra degree of freedom. The results of the Namias were refined by the McBride and Kerr in 1987 [26] by developing an operational calculus to define the FrFT. The one-dimensional FrFT is defined as FrFT of order α of $x(u)$ denoted by X_α(u) [27]:

$$X_\alpha(u) = \int_{-\infty}^{\infty} x(t)\, K_\alpha(t,u)\, \mathrm{d}t \tag{1}$$

$$K_\alpha(t,u) = \begin{cases} \sqrt{\frac{1-j\cot\alpha}{2\pi}}\, e^{\,j((u^2+t^2)/2)\cot\alpha - jut\,\mathrm{cosec}(\alpha)} & \text{if } \alpha \text{ is not multiple of } \pi \\ \delta(t-u) & \text{if } \alpha \text{ is multiple of } 2\pi \\ \delta(t+u) & \text{if } \alpha+\pi \text{ is multiple of } 2\pi \end{cases} \tag{2}$$

where $\alpha = a\pi/2$ indicates the rotation angle of the transformed signal for FrFT, a is the fractional order of the FrFT, and the FrFT operator is designated by X_α. From definition, it is apparent that FrFT is periodic with period 4 and additive in index. The simplest generalization of the FrFT for two dimensions is given by Eqs. (3) and (4):

$$X_{\alpha,\beta}(u,s) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} K_{\alpha,\beta}(u,s;t,r)x(t,r)\mathrm{d}t\mathrm{d}r \tag{3}$$

$$K_{\alpha,\beta}(u,s;t,r) = K_\alpha(u,t)K_\beta(s,r) \tag{4}$$

In case of two-dimensional FrFT, we have to consider two angles of rotation, $\alpha = a\pi/2$ and $\beta = b\pi/2$. If one of these angles of rotation is zero, then the two-dimensional kernel reduces to one-dimensional kernel. The signal can be recovered back by 2-D FrFT operation with inverted angles $(-\alpha, -\beta)$.

Although the fractional Fourier transforms have been the mainstay in transform, a more recent approach from the last decade known as DFrFT simplifies both the mathematics and physical analysis. The advantages of discrete fractional transforms are that it is an evolutionary computational method, high compression ratio and that its fractional part provides extra degree of freedom in computations. Huffman decoding [29] is difficult due to different code lengths at decoder and some prior knowledge of probabilities of symbols is required from encoder, whereas in case of discrete fractional transforms, only the inverse keys of encoder are required at the decoder.

The usefulness of discrete transforms in image and video processing applications is presented in paper.

2.1 Discrete fractional Fourier transforms

After the continuous FrFT has been derived, many researchers have attempted their best to develop a discrete counterpart of it. It was the Santhanam and McClellan [30] who have first reported the work in 1995. The properties such as unitarity, index additivity, reduction to DFT when the order is equal to unity and approximation of the continuous FrFT are to be fulfilled by a valid discrete input/discrete output FrFT.

Discrete fractional Fourier transforms must obey the rotational properties quite like the continuous FrFT. These rotational properties can be easily realized by the power law of the kernel matrix in discrete case. So the fractional power of kernel is required for computing the DFrFT. In [31,32], a method for computing the DFrFT has been proposed. DFrFT can be derived as linear combination of identity operation, discrete Fourier transforms (DFT), time inverse operation and inverse DFT (IDFT) [33]. Unfortunately, this DFrFT cannot have similar outputs as those with the continuous FrFT. The development of current improved DFrFT introduced a commuting matrix **M** to compute the real eigenvectors of the DFT kernel matrix **H**:

$$M = \begin{bmatrix} 2 & 1 & 0 & 0 & \cdots & 0 \\ 1 & 2\cos\omega & 1 & 0 & \cdots & 0 \\ 0 & 1 & 2\cos 2\omega & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & & \vdots \\ 1 & 0 & 0 & 0 & \cdots & 2\cos(N-1)\omega \end{bmatrix} \tag{5}$$

where $\omega = 2\pi/N$, and N is the size of the DFT kernel matrix. Matrix **M** commutes with the matrix **H** and satisfies commutative property: MH = HM. The eigenvectors of matrix **M** and **H** are the same, but their eigenvalues are different. The eigenvectors of **M** are orthonormal to each other, and eigenvalues are real because **M** is a real symmetric matrix. In [32,34], Pei and Yeh used the DFT eigenvectors obtained from matrix **M** to construct the DFrFT kernel. The eigenvectors of matrix **M** are treated as discrete Hermite functions [34,35]. The transformation kernel of DFrFT can be easily defined by determining the fractional powers of the eigenvalues. The transform kernel of the DFrFT can be calculated as

$$D_\alpha = F^{2\alpha/\pi} = VD^{2\alpha/\pi}V^T \tag{6}$$

Table 1 Eigenvalues assignment rule of DFrFT kernel matrix

N	The eigenvalues
$4m$	$e^{-jk\alpha}, k = 0, 1, 2, \ldots, (4m-2), 4m$
$4m+1$	$e^{-jk\alpha}, k = 0, 1, 2, \ldots, (4m-1), 4m$
$4m+2$	$e^{-jk\alpha}, k = 0, 1, 2 \ldots, 4m, (4m+2)$
$4m+3$	$e^{-jk\alpha}, k = 0, 1, 2, \ldots, (4m+1), (4m+2)$

where α indicates the rotation angle of the DFrFT. $V = [v_0|v_1|\cdots|v_{N-2}|v_{N-1}]$ for N odd, $V = [v_0|v_1|\cdots|v_{N-2}|v_N]$ for N even, and V_k is the kth-order DFT Hermite eigenvector. $D^{2\alpha/\pi}$ is the diagonal matrix with eigenvalues of DFrFT in the diagonal entries. A method for finding the DFT Hermite eigenvectors v_k is presented in [22,32]. In Table 1, there exists a jump in the last eigenvalues for the two even-length cases and (6) can be written as:

$$D_\alpha = \sum_{k=0}^{N-1} e^{-jk\alpha} v_k v_k^T, \quad \text{for } N = 4m+1, 4m+3 \tag{7}$$

The DFrFT of a signal can be computed with a transformation kernel with Eq. (8).

$$D_\alpha x = F^{2\alpha/\pi} x = V D^{2\alpha/\pi} V^T x \tag{8}$$

To compute inverse DFrFT (IDFrFT) [33], DFrFT is calculated with order –α.

$$x = F^{-2\alpha/\pi} D_\alpha = V D^{-2\alpha/\pi} V^T D_\alpha \tag{9}$$

For 2-D DFrFT, two individual angles of rotation α and β in two dimensions are taken and can be implemented by row–column computation in case of 2-D separable kernel. Then, the forward and inverse 2-D DFrFT for (m, n) and (p, q) points are defined with separable form as:

$$D_{(\alpha,\beta)}(m,n) = \sum_{p=0}^{M-1}\sum_{q=0}^{N-1} x(p,q) D_{(\alpha,\beta)}(p,q,m,n) \tag{10}$$

$$x(p,q) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} D_{(\alpha,\beta)}(m,n) D_{(-\alpha,-\beta)}(p,q,m,n) \tag{11}$$

The 2-D DFrFT also keep the preferred properties of continuous 2-D FrFT. In the present paper, applications of DFrFT in two dimensions, that is, in image and video compression and encryption, are discussed.

2.2 Discrete fractional Cosine transform

The discrete fractional cosine transform (DFrCT) is a general form of discrete Cosine transform (DCT), which has an additional free parameter that can be used in all applications where DCT is found to be useful. The DCT of a sequence $\{x[n], 0 \le n \le N-1\}$ is defined [37–41] with Eqs. (12) and (13):

$$X(k) = \alpha(k) \sum_{n=0}^{N-1} x[n] \cos\left[\frac{(2n+1)\pi k}{2N}\right], \quad \text{for } 0 \le k \le N-1 \tag{12}$$

where

$$\alpha(k) = \begin{cases} \frac{1}{\sqrt{N}} & \text{for } k = 0 \\ \sqrt{\frac{2}{N}} & \text{for } 1 \le k \le N-1 \end{cases} \tag{13}$$

The elements of one-dimensional DCT kernel matrix are given by equation (14):

$$E_{DCT}(k,n) = \begin{cases} \frac{1}{\sqrt{N}}, k=0; & 0 \le n \le N-1 \\ \sqrt{\frac{2}{N}} \cos\left[\frac{(2n+1)\pi k}{2N}\right]; & 1 \le k \le N-1; 0 \le n \le N-1 \end{cases} \tag{14}$$

Because the sequence is orthogonal, the inverse DCT (IDCT) can be recovered as:

$$x[n] = \sum_{k=0}^{N-1} \alpha(k) X(k) \cos\left[\frac{(2n+1)\pi k}{2N}\right], \quad 0 \le n \le N-1 \tag{15}$$

The DFrCT uses the eigen decomposition of the DCT kernel. And the exclusive eigenvectors are obtained from the even Hermite–Gauss eigenvectors of the Fourier matrix in the cosine case. The kernel matrix of N point DFrCT is defined as [36,42]:

$$C_{N,\alpha} = V_N D_N^{2\alpha/\pi} V_N^T = V_N \begin{bmatrix} 1 & & 0 & \\ & e^{-2j\alpha} & & \\ & & \ddots & \\ & & & e^{-j2(N-1)\alpha} \end{bmatrix} V_N^T \tag{16}$$

where $V_N = \left[v_0 | v_1 | \cdots | v_{2N-2}\right]$, v_k is the eigenvector derived from kth-order DFT Hermite eigenvector.

DFrCT has the mathematical property of unitarity, additivity of rotations, periodicity and reality [35]. In image compression and encryption, two-dimensional DFrCT is used. Two-dimensional DFrCT is used as two times one-dimensional DFrCT (row-wise and column- wise). Two angles of rotation α and β in two dimensions are taken separately. Applications of DFrCT for 2-D are reviewed with simulation results.

2.3 Discrete fractional Hartley transform

The need of fractional order and its computation has been investigated in many of the engineering and science applications [43,44]. Thus, the transform kernel $K_H^{\alpha}\left(u, u'\right)$of fractional Hartley transform is given by:

$$K_H^{\alpha}(u, u') = \sum_{n=0}^{\infty} \lambda_n^{\alpha} e_n(u) e_n(u') \tag{17}$$

The relation between fractional Hartley transforms (FrHT) and fractional Fourier transform (FrFT) is given as:

$$g_H^{\alpha}(u') = \frac{1 + e^{\frac{j\alpha\pi}{2}}}{2} g_F^{\alpha}(u') + \frac{1 - e^{\frac{j\alpha\pi}{2}}}{2} g_F^{\alpha}(-u') \tag{18}$$

The two-dimensional FrHT can be computed by applying one-dimensional transform row-wise and column-wise separately. The relation between DFrFT and DFrHT is given by:

$$g_F^{\alpha}(u') = \frac{1 + e^{\frac{j\alpha\pi}{2}}}{2} g_H^{\alpha}(u') + e^{j(\pi/2)\alpha} \frac{1 - e^{\frac{j\alpha\pi}{2}}}{2} g_H^{\alpha}(-u') \tag{19}$$

Moreover, if the real and imaginary parts of the DFrHT are both even symmetric, then $g_F^{\alpha}\left(u'\right) = g_H^{\alpha}$ means that DFrFT is equal to DFrHT.

3 Discrete fractional transforms applications

The applications of discrete fractional transforms in image and video are discussed in this section.

3.1 Image compression

To expedite transmission and reduce storage requirements, image compression means image with smallest possible number of bits without loss of information [45,46] at receiving end. Two fundamental components of compression are redundancy and irrelevancy reduction. Certain image processing operations such as compression and encryptions are more perceptive and proficient if we process the image in a different domain like frequency domain and spatial domain [47]. A motivating aspect of using a frequency domain representation of an image is that it is much more efficient to decorrelate an image in the frequency domain than in the spatial domain. For image compression, different frequency domains techniques are discussed in the literature. In our paper, we are reviewing image compression with 2-D discrete fractional transforms. The following steps are used for image compression:

Step 1: An image is first separated into non-overlapped subimages. The most popular subimage sizes are 8×8 AND 16×16. For simulation results, implementation scheme subimage size chosen is 8×8.

Step 2: A 2-D discrete fractional transform (DFrFT, DFrCT and DFrHT) is applied to each block at optimum value of fractional order with selected compression percentage. The degree of data reduction as a result of the compression process is known as compression percentage. For simplicity, the 'α' order along (rows) and y (columns) directions is taken to be same. This is done to convert the gray-scale levels of pixels in spatial domain into coefficients in transform domain.

Step 3: The quantization of these coefficients is done to selectively eliminate or more coarsely quantize the coefficients that carry the least information. A compromise can be made between image quality and compression percentage by adjusting the coarseness of the quantizer called cutoff value. The optimized quantized coefficients are arranged from lower-frequency to higher-frequency components and further compressed by efficient run-length coding approach.

Step 4: At decoding end, simply contrary process of encoding using inverse 2-D discrete fractional transform is performed. Inverse discrete fractional transform is obtained by inverted value of 'α' that was used in forward discrete fractional transform with the same value.

Various parameters are used to study the properties of the compressed image. The most commonly used parameters to estimate image-compression algorithms are compression percentage, peak signal- to-noise ratio (PSNR) and mean square error (MSE). In Eq. (21), the PSNR value in decibels (dB) is used to measure the difference between the decoded image 'r' and the original image 'o'. In general, the larger the PSNR value, the better the image quality.

$$\text{MSE} = \left[\frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [r(i, j) - o(i, j)]^2 \right] \tag{20}$$

$$\text{PSNR} = 10 \log_{10} \left[\frac{M \times N}{MSE} \right] \tag{21}$$

where $M \times N$ is the size of the image.

Numerical simulations have been executed to estimate the strength of the image compression technique. The original images taken are Baboon, Boats, House, Flower and Pepper with 256×256 pixels. These images are compressed to the compression percentages of 10 %, 20, 30, 40, 50 and 75 % by varying values of fractional order 'a'. The minimum value of compression percentage taken in the simulation is 10 %, and the maximum is 75 %. We observed that the optimum compression performance is achieved for values of fractional order 'a' varying between 0 and 1. The optimum value of 'a'

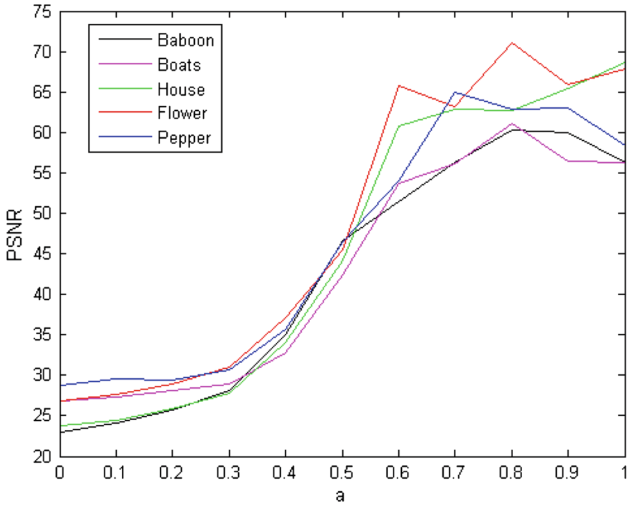

Fig. 1 Fractional-order a vs. PSNR of different images using DFrFT with 10 %

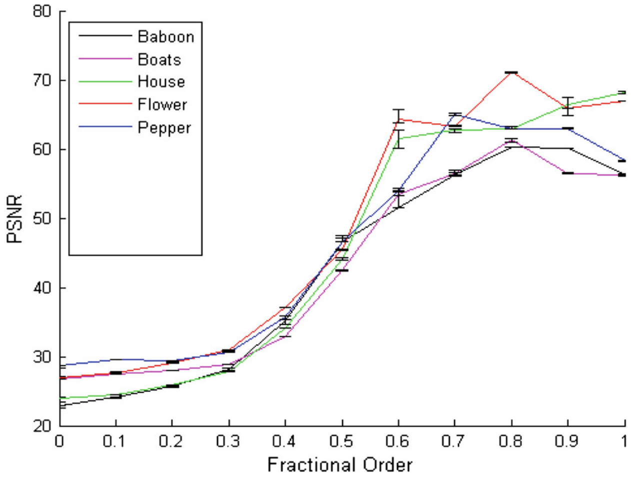

Fig. 2 Error bar graph of different images using DFrFT with 10 %

(a_{opt}) is dependent both on the image and on the compression percentage. Figure 1 shows the plots for PSNR vs. 'a' for different images at selected compression percentage of 10 %. Larger PSNR and low MSE are considered for good-quality compressed image. MSE and PSNR are metric parameters that illustrate the performance of discrete fractional transforms.

The optimum value of 'a' (a_{opt}) is selected for maximum PSNR and minimum MSE at particular compression percentage for an image in Fig. 1. When simulation results are repeated for number of times, the error bar graph is shown in Fig. 2.

In Table 2, for Baboon image, performance of DFrFT at 20 % compression percentage, when a_{opt} is 0.91, gives high PSNR of 50.663, and low MSE of 0.412. Table 3 gives the results of different images for 'a_{opt}' at selected compression percentage (10 %), compressed using DFrFT and DFrCT.

Table 2 MSE and PSNR at optimum a and selected compression for Baboon using DFrFT

Image	Compression percentage	a_{opt}	MSE	PSNR
Baboon	10	0.92	0.046	60.08
	20	0.91	0.412	50.663
	30	0.91	1.513	45.19
	40	0.92	4.23	40.77
	50	0.91	9.84	37.111
	75	0.97	66.14	28.83

Figure 3 is the original Baboon image of 256 × 256 and shows the compressed images for compression percentage = 10 % using DFrFT, DFrCT and DFrHT.

Figure 4 shows effect on PSNR for compression percentages of 10, 20, 30, 40, 50 and 75 % at optimum value 'a' for Baboon and Barbara images using all three transforms.

Figure 4 shows that at the highest compression percentage, PSNR is lowest, and at the lowest compression percentage, PSNR is highest. We observed that the DFrFT is better than DFrCT and DFrHT for both images. However, the performance of DFrCT is better than DFrHT in Baboon image but wavering in the Barbara image. So, it is evident that DFrFT provides better PSNR in comparison with DFrCT and DFrHT. The reason is that in time-frequency representations, generally a plane has two orthogonal axes corresponding to time and frequency, respectively. If a signal is represented along time axis, its Fourier transform is represented along the frequency axis. The Fourier transform operator is viewed as a change in the representation of the signal corresponding to a counterclockwise axis rotation of $\pi/2$ rad. In context of DFrFT, for any vector at 45°, it performs same for time and frequency plane equally. Image compression with DFrFT performs better in frequency domain and tries to save bandwidth by varying fractional order 0–1.

The comparison of discrete fractional transforms and JPEG [48] is given in Table 4 for Lena and Pepper images.

It was observed that for (8 × 8) blocks, discrete fractional transform shows better performance in image quality except in encoding and decoding time of CPU. The improvement in PSNR for Lena and Pepper images is 9.89 and 8.21 dB, respectively.

3.2 Image encryption

An encryption technique transforms the image such that only authorized users can understand the meaning of the image using encryption keys. A number of review papers are available on image and video encryption providing a more or less inclusive overview of the techniques proposed so far [49–54].

Table 3 MSE & PSNR at optimum a and selected compression (10 %) using DFrFT and DFrCT

Image	Compression percentage	Using DFrFT			Using DFrCT		
		a_{opt}	MSE	PSNR	a_{opt}	MSE	PSNR
Boat	10	0.93	0.1118	56.555	0.92	0.3281	51.08
House	10	0.93	0.010	66.99	0.98	0.1718	54.69
Pepper	10	0.91	0.0226	63.49	0.95	0.3271	51.10
Aerial	10	0.96	0.14	55.48	0.97	0.2993	50.69
Jet plane	10	0.92	0.021	63.68	0.94	0.2968	50.68
Baboon	10	0.92	0.046	60.08	0.93	0.1093	54.23
Flower	10	0.94	0.0102	66.94	0.94	868.16	18.74

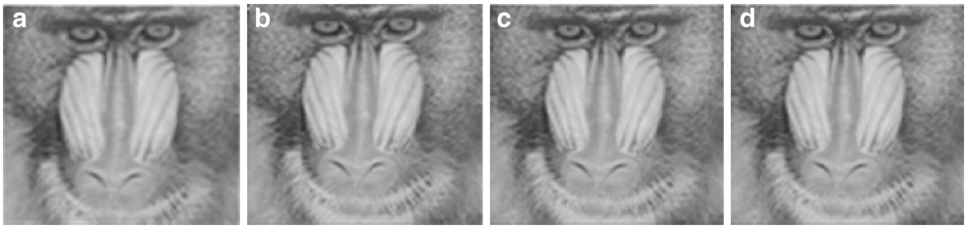

Fig. 3 Simulation results of Baboon image with three transforms at compression percentage = 10 %. **a** The original Baboon image, **b** compressed using DFrFT, **c** compressed using DFrCT, **d** compressed using DFrHT

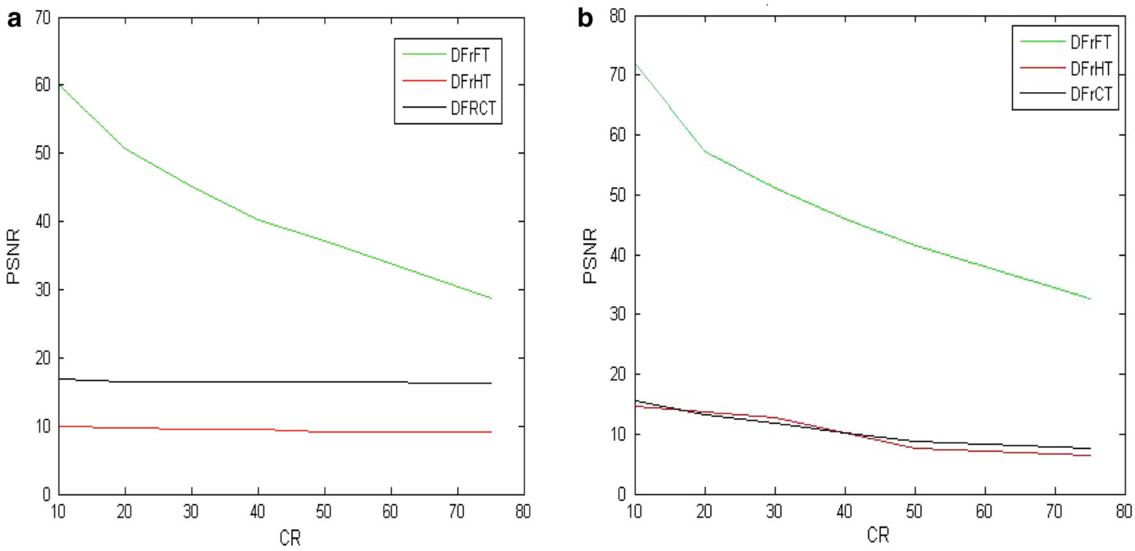

Fig. 4 Compression percentage vs. PSNR of **a** Baboon and **b** Barbara image for DFrFT, DFrCT and DFrHT

Table 4 Comparison of discrete fractional transforms and JPEG

Algorithm	Image size	PSNR in dB	CPU time	
			Encoding time (s)	Decoding time (s)
Discrete fractional transforms	Lena (256 × 256)	44.55	5.33	5.33
	Pepper (256 × 256)	42.28	3.48	3.48
JPEG [48]	Lena (256 × 256)	34.66	0.12	0.12
	Pepper (256 × 256)	34.27	0.2	0.2

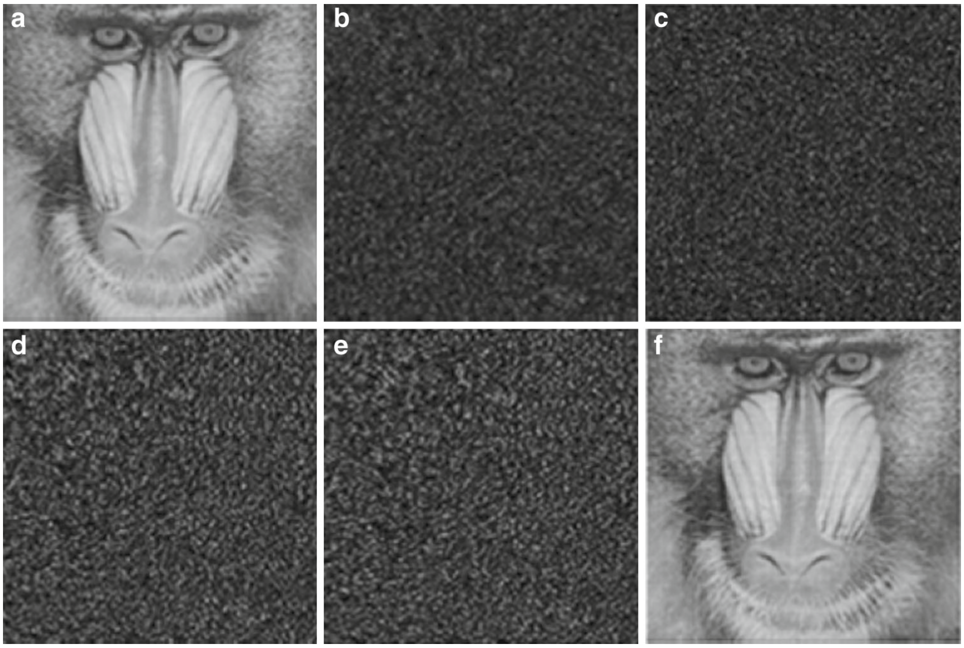

Fig. 5 Simulation results with DFrCT. **a** original Baboon image, **b** encrypted Baboon image (1, 0.33, 0.8) with PSNR = 79.072, **c** incorrectly decrypted image (−0.6, −0.33, −0.8) with PSNR = 54.31, **d** incorrectly decrypted image (1, −0.40, −0.8) with PSNR = 78.92, **e** incorrectly decrypted image (−1, −0.33, −0.5) with PSNR = 78.41, **f** correctly decrypted image (−1, −0.33, −0.8) with PSNR = 80.04

It has been recently noticed that cascaded discrete fractional transforms with random phase filtering can be used in encryption of images [55–59]. Tao et al. in [60] proposed a scheme utilizing random phase encoding in the fractional Fourier domain; two images can be encrypted into one encrypted image with stationary white distribution. Image encryption benefits from the extra degree of freedom that is provided by discrete fractional orders. The n-stage of discrete fractional transforms can provide n-dimensional extra keys indicated by the fractional orders. Two individual fractional orders along x-axis and y-axis are used in two-dimensional discrete fractional-order transforms. Such a system can have $n-1$ random phase filters, so that the total encryption keys can be increased to as many as $3n-1$.

The order 'α' along x and y directions is taken to be the same, that is, $\alpha_x = \alpha_y = \alpha$, and three stages of DFrCT are cascaded together. In the intermediate planes, two randomly encoded phase masks are used. The algorithm consists of two parts, encryption to encrypt the image and decryption to retrieve the image back.

Consider real-valued two-dimensional image data, $f(x_0, y_0)$, which is to be encrypted. The image is discrete-fractional-cosine-transformed three times using fractional-orders α_1, α_2 and α_3, respectively. In the intermediate stages, we put two random phase masks (RPM),

$$p_1(x_1, y_1) = \exp\left[-\iota 2\pi\varphi_1(x_1, y_1)\right] \tag{22}$$

and

$$p_2(x_2, y_2) = \exp\left[-\iota 2\pi\varphi_2(x_2, y_2)\right] \tag{23}$$

serving as phase filters, respectively. Here, functions $\varphi_1(x_1, y_1)$ and $\varphi_2(x_2, y_2)$ are randomly generated homogeneously distributed functions with values (0, 1). Thus, the resultant transformed function $\Omega(x, y)$ can be written as in Eqs. (24–26):

$$\Omega(x, y) = F_c^{\alpha 3}\left\{\Omega_2(x_2, y_2)p(x_2, y_2)\right\} \tag{24}$$

with

$$\Omega_2(x, y) = F_c^{\alpha 2}\left\{\Omega_1(x_1, y_1)p(x_1, y_1)\right\} \tag{25}$$

and

$$\Omega_1(x_1, y_1) = F_c^{\alpha 1}\left\{f(x_0, y_0)\right\} \tag{26}$$

The final resultant function $\Omega(x, y)$ is the encrypted image. The decryption process is the inverse operation with respect to encryption. First, take a DFrCT of order-α_3 on the encrypted image $\Omega(x, y)$, multiply the random phase mask $p_2^*(x_1, y_1)$ and then get the midterm function $\Omega_2(x_2, y_2)$. Next, perform a fractional Fourier transform of order-α_2 on the function $\Omega_2(x_2, y_2)$ and then multiply it by the phase mask $p_1^*(x_1, y_1)$; thus, function $\Omega_1(x_1, y_1)$ will be recovered. After another fractional Fourier transform of order-α_1 on the function $\Omega_1(x_1, y_1)$, finally get the original image $f(x_0, y_0)$. Here, the mask $p_2^*(x_1, y_1)$ and $p_1^*(x_1, y_1)$ are the complex conjugate of $p_2(x_1, y_1)$ and $p_1(x_1, y_1)$, respectively.

Encryption and decryption results for Baboon, Boats, Flower, House and Pepper images of 256 × 256 pixels using discrete fractional transforms (DFrFT, DFrCT) are obtained. Numerical simulations have been performed to examine the performance of discrete fractional transforms

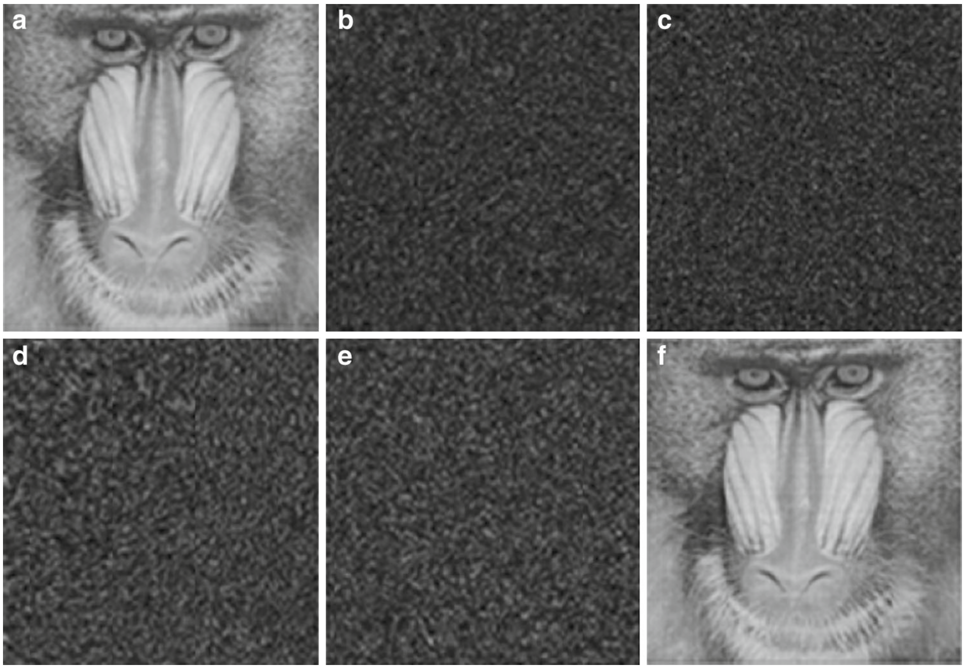

Fig. 6 Simulation results with DFrFT **a** original Baboon image, **b** encrypted Baboon image (0.5, 0.5, 0.5) with PSNR = 60.46, **c** incorrectly decrypted image (−0.6, −0.5, −0.5) with PSNR = 54.46, **d** incorrectly decrypted image (−0.5, −0.4, −0.5) with PSNR = 53.20, **e** incorrectly decrypted image (−.5, −0.5, −0.7) with PSNR = 53.20, **f** correctly decrypted image (−0.5, −0.5, −0.5) with PSNR = 53.20

(DFrFT, DFrCT). The original Baboon image is encrypted with fractional keys {1, 0.33, 0.8} using DFrCT and {0.5, 0.5, 0.5} using DFrFT as shown in Figs. 5 and 6, respectively. The effect on PSNR is calculated, and it is observed that in DFrCT, the value of PSNR decreases. Figure 5 shows that PSNR affects much more in DFrCT as compared to DFrFT even if single right key is not used for decryption.

There are many aspects of security analysis for evaluating how a system behaves with common noise attacks. The main noise of consideration is salt and pepper noise. The angle of rotation $\alpha = a\pi/2$ depends on fractional-order 'a'. The length of fractional part of a can be increased, and it will be difficult for attacker to find the correct key. Figure 7 shows salt & pepper noise and decrypted image for DFrFT.

3.3 Video encryption

From the past decade, many people around the world acquire, utilize and share videos via Internet [61–65]. Security and protection of video contents are the main demand of users. Raw video data consist of a series of still images. We have discussed image encryption techniques as priority.

An encryption and decryption algorithm is implemented for video encryption based on discrete fractional transforms. A number of examples of video sequences at 25 frames per second have been encrypted using discrete fractional cosine transforms. Four successive frames from a video clip, and their encrypted form based on DFrCT are depicted in Figs. 8 and 9, respectively.

The advantages of the proposed algorithm are as follows:

a. Multi-keys used to encrypt video frame will enhance the security strength of video.
b. Three-dimensional discrete fractional transform is separable transform, and it can be implemented as a series of one-dimensional transform (or as a two-dimensional transform followed by a one-dimensional transform). Each frame of video clip is encrypted with encryption key column-wise and row-wise and so creates difficulty for adversary to decrypt each frame of same video scene.
c. Because of individual frame encryption, if a frame is corrupted or lost during transmission, it does not affect the decryption of other frames. Therefore, time is saved by avoiding iteration transmission of remaining frames due to single lost frame.

At receiving end, the authorized user will use inverted decryption keys and get original frames. Figures 10 and 11 shows the results with wrong keys, and Fig. 12 shows the results with right encryption keys. A variety of video clips with different encryption keys using DFrFTs are simulated. Videos of different frame sizes are taken to calculate the mean square error (MSE) and peak signal-to-noise ratio (PSNR) using DFrFT and DFrCT as shown in Table 5.

It is observed that DFrCT gives superior PSNR and less MSE for video encryption in comparison with DFrFT.

Strong encryption algorithm must be capable of resisting attack of salt–pepper noise in channel. Figure 13 shows original building video frame and its decrypted frame at receiver after salt–pepper noise attack.

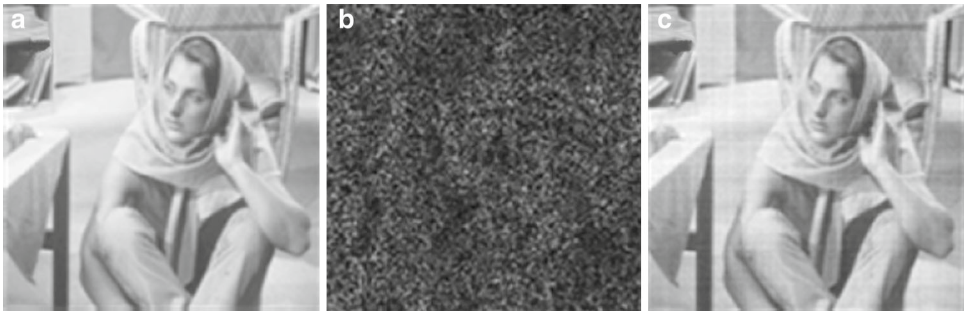

Fig. 7 **a** Original Barbara image, **b** encrypted using DFrFT and attacked with salt–pepper noise, and **c** decrypted using DFrFT

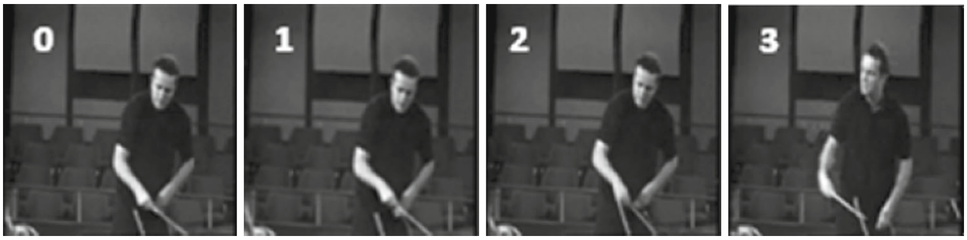

Fig. 8 A sequence of four frames (0–3): note the motion of arms

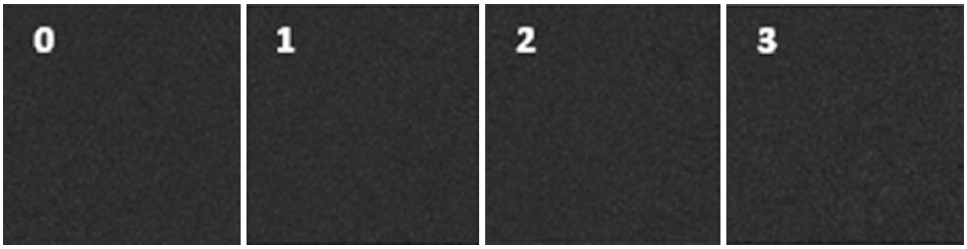

Fig. 9 Encrypted video frames of their respective frames

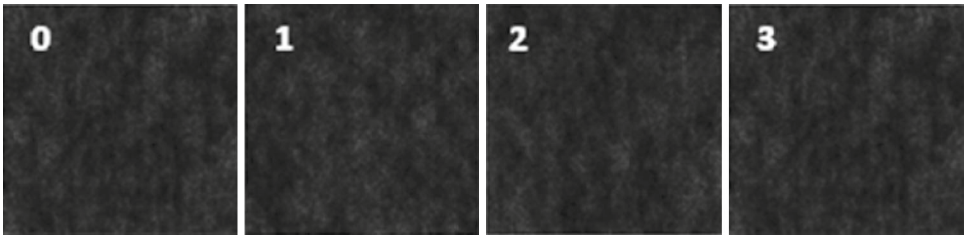

Fig. 10 Decrypted video frames of their respective frames (with wrong keys)

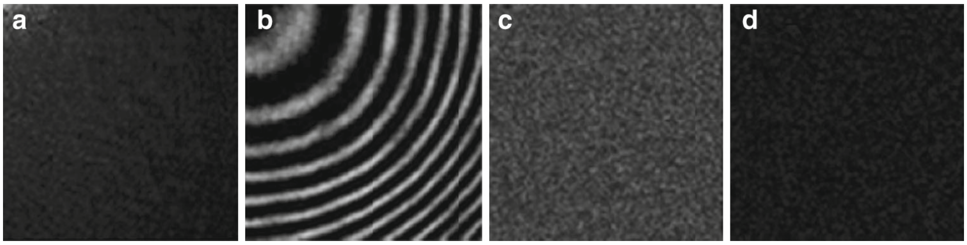

Fig. 11 **a** Decrypted frame with first key wrong, **b** with second key wrong, **c** with third key wrong and **d** with all keys wrong

Table 5 MSE and PSNR for different test videos with DFrFT and DFrCT

Test video	Frame size ($M \times N$)	With DFrFT		With DFrCT	
		MSE	PSNR	MSE	PSNR
Mars	600 × 480	0.013574	66.803573	0.021606	64.785145
Person	480 × 360	0.029564	63.422864	0.000056	90.636911
Building	480 × 640	0.322862	53.040639	0.000375	82.395493
Corn	720 × 1,152	0.005990	70.356811	0.000001	109.524013
Sea shore	480 × 854	0.177648	55.635208	0.000188	85.388985

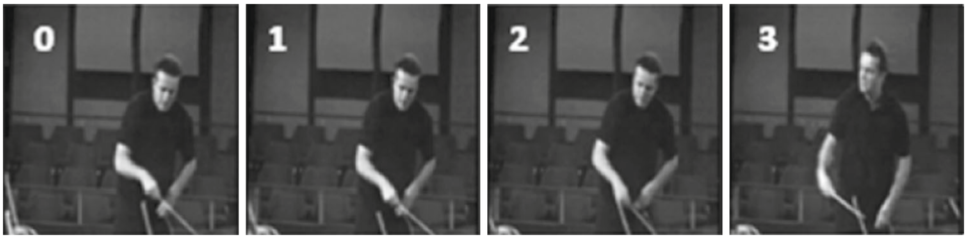

Fig. 12 Decrypted video frames of their respective frames (with right keys)

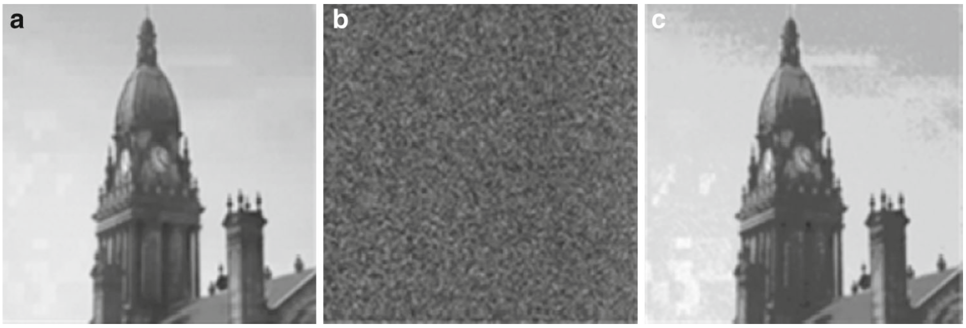

Fig. 13 Building video frame. **a** Original, **b** encrypted using DFrCT and attacked with salt–pepper noise, **c** decrypted using DFrCT

4 Conclusions

This paper has reviewed that discrete fractional transforms are used extensively and researched in various applications to improve efficiency. MSE and PSNR values for different test images at selected compression percentage and optimum fractional-order 'a' are computed using discrete fractional transforms for image compression. It has been concluded that DFrFTs perform better with high PSNR and low MSE in image compression. Discrete fractional transforms with multi-keys enhance security in image encryption. DFrCT results give abundant variations in PSNR with change in single key as compared to DFrFT. Hence, DFrCT establishes better sensitivity of encryption keys for authorized users. Encryption of different frame size videos with discrete fractional cosine transforms gives better PSNR as compared to DFrFTs. The high PSNR of the order of 79 dB makes the algorithm suitable for real-time applications. The future scope will be the implementation of other discrete fractional transforms in image and video applications.

Acknowledgments The authors acknowledge the support provided by the Department of Electronics & Communication Engineering, Thapar University Patiala, Punjab (India), for carrying out the Research work and anonymous reviewers for their valuable suggestions.

References

1. Wiener, N.: Hermitian polynomials and Fourier analysis. J. Math. Phys. **8**, 70–73 (1929)
2. Condon, E.U.: Immersion of the Fourier transform in a continuous group of functional transformations. Proc. Nat. Acad. Sci. USA **23**(3), 158–164 (1937)
3. Kober, H.: Wurzeln aus der Hankel-, Fourier- und aus an-deren stetigen transformationen. Q. J. Math. Oxford Ser. **10**, 45–49 (1939)
4. Namias, V.: The fractional order Fourier transform and its application to quantum mechanics. J. Inst. Maths. Appl. **25**, 241–265 (1980)
5. Almeida, L.B.: The fractional Fourier transform and time-frequency representations. In: IEEE Trans. Signal Process. **42**, 3084–3091 (1994)
6. Vijaya, C., Bhat, J.S.: Signal compression using discrete fractional Fourier transform and set partitioning in hierarchical tree. Signal Process. **86**(8), 1976–1983 (2006)
7. Tao, R., Deng, B., Wang, Y.: Research progress of the fractional Fourier transform in signal processing. Sci. China Ser. F Inf. Sci. **49**(1), 1–25 (2006)
8. Bracewell, R. N.: The Fourier Transform and Its Applications, 2nd Revised. McGraw Hill, NewYork (1986).
9. Jindal, N., Singh, K.: Image encryption using discrete fractional transforms. International Conference on Advances in Recent Technologies in Communication and Computing. Published by IEEE Computer Society, 165–167 (2010).
10. Abomhara, M., Zakaria, O., Khalifa, O.O.: An overview of video encryption techniques. Int. J. Comput. Theory Eng. **2**(1), 103–110 (2010)
11. Gonzalez, R. C., Woods, R. E.: Digital Image Processing, 3rd edition (2008).
12. Santhanam, B., McClellan, J.H.: The discrete rotational Fourier transform. IEEE Trans. Signal Process. **42**, 994–998 (1996)
13. Lohmann, A.W.: Image rotation, Wigner rotation and the fractional Fourier transform. J. Opt. Soc. Am. A **10**, 2181–2186 (1993)
14. Mendlovic, D., Ozaktas, H.M.: Fractional Fourier transforms and their optical implementation. I. J. Opt. Soc. Am. A **10**, 1875–1881 (1993)
15. Ozaktas, H.M., Mendlovic, D.: Fractional Fourier transforms and their optical implementation. II. J. Opt. Soc. Am. A **10**, 2522–2531 (1993)
16. Atakishiyev, N.M., Vicent, L.E., Wolf, K.B.: Continuous vs. discrete fractional Fourier transforms. J. Comput. Appl. Math. **107**, 73–95 (1999)

17. Singh, K.: Performance of discrete fractional Fourier transform classes in signal processing applications, Ph.D. Thesis, Thapar University (2006).
18. Yetik I.S., Kutay M.A., Ozaktas, H., Ozaktas, H.M.: Continuous and discrete fractional Fourier domain decomposition. In: IEEE, 93–96 (2000).
19. Ozaktas, H.M., Zalevsky, Z., Kutay, M.A.: The Fractional Fourier Transform with Applications in Optics and Signal Processing. Wiley, New York (2000)
20. Pei, S.C., Yeh, M.H.: A novel Method for discrete fractional Fourier transform computation. ISCAS **2001**(2), 585–588 (2001)
21. Yeh, M.H., Pei, S.C.: A method for the discrete-time fractional Fourier transform computation. Signal Process. **51**(6), 1663–1669 (2003)
22. Pei, S.C., Yeh, M.H., Tseng, C.C.: Discrete fractional Fourier transform based on orthogonal projections. In: IEEE Trans. Signal Process. **47**(5), 1335–1348 (1999)
23. Djurovic, I., Stancovic, S., Pitas, I.: Digital watermarking in the fractional Fourier transformation domain. J. Netw. Comput. Appl. **24**, 167–173 (2001)
24. Rubio, J.G.V., Santhanam, B.: On the multiangle centered discrete fractional Fourier transform. In: IEEE Signal Process. Lett., **12**(4), (2005).
25. Jain, A.K.: Fundamentals of Digital Image Processing. Prentice-Hall, Englewood Cliffs (1989)
26. McBride, A.C., Keer, F.H.: On Namia's fractional Fourier transform. IMA J. Appl. Math **239**, 159–175 (1987)
27. Bultheel, A., Mart ínez, H.: Computation of the fractional Fourier transform. Appl. Comput. Harmon. Anal. **16**(3), 182–202 (2004)
28. Narayanan, V.A., Prabhu, K.M.M.: The fractional Fourier transform: theory, implementation and error analysis. Microprocess. Microsyst. **27**, 511–521 (2003)
29. Bultheel, A., Martinez, H.: A shattered survey of the fractional Fourier transform. TW Reports, TW337, 42 pages, Department of Computer Science, K.U. Leuveven, Leuven, Belgium (April 2002).
30. Santhanam, B., McClellan, J.H.: The DRFT-a rotation in time-frequency space. Proc. IEEE Int. Conf. Acoust. Speech Signal Process (ICASSP) **2**, 921–925 (1995)
31. Candan, C.: Discrete fractional Fourier transform. M.S. thesis, Bikent Univ., Ankara, Turkey (1998).
32. Pei, S.C., Yeh, M.H.: Improved discrete fractional Fourier transform. Opt. Lett. **22**, 1047–1049 (1997)
33. Santhanam, B., McClellan, J.H.: The discrete rotational Fourier transform. Signal Process. **44**(4), 994–998 (1996)
34. Pei, S.C., Yeh, M.H.: Discrete fractional Fourier transform. Proc. IEEE Int. Symp. Circuits Syst. 536–539 (1996).
35. Yeh, M.H., Pei, S.C.: A method for the discrete fractional Fourier transform computation. In: IEEE Trans. Signal Process. **51**(3), 889–891 (2003)
36. Pei, S.C., Yeh, M.H.: The discrete fractional cosine and sine transforms. IEEE Trans. Signal Process. **49**, 1198–1207 (2001)
37. Gerek, O.N., Erden, M.F.: The discrete fractional cosine transform. Proceedings of the IEEE Balkan Conference on Signal Processing, Communications, Circuits and Systems, Istanbul, Turkey (2000).
38. Lohmann, A.W., Mendlovic, D., Zalevsky, Z., Dorch, R.G.: Some important fractional transformations for signal processing. Opt. Commun. **125**, 18–20 (1996)
39. Yip, P.: Sine and Cosine transforms. In: Poularikas, A.D. (ed.) Transforms, The Handbook Applications. CRC Press, Alabama (1996)
40. Prudnikov, A.P., Brychkov, Y.A., Marichev, O.I.: Integrals and Series VI: Elementary Functions. Gordon and Breech Science Publishers, New York (1986)
41. Ahmed, N., Natarajan, T., Rao, K.R.: Discrete cosine transform. In: IEEE Trans. Comp. C **23**, 90–93 (1974).
42. Narasimha, M.J., Peterson, A.M.: On the computation of the discrete cosine transform. In: IEEE Trans. Commun. **26**(6), 934–936 (1978)
43. Shu, H., Wang, Y., Senhadji, L., Luo, L.: Direct computation of type-II discrete Hartley transform. Proc. IEEE Signal Process. Lett. **14**, 329–332 (2007)
44. Jiang, L., Shu, H., Wu, J., Wang, L., Senhadji, L.: A novel split-radix fast algorithm for 2-D discrete Hartley transform. Proc. IEEE Trans. Circuits Syst. **57**, 911–924 (2010)
45. Pratt, W.K.: Digital Image Processing, 2nd edn. Wiley, New York (1991)
46. Sid-Ahmed, M.A.: Image Processing–Theory, Algorithms, and Architectures. McGraw-Hill, New York (1995)
47. Thyagarajan, K.S.: Still Image and video compression with MATLAB. Wiley, New York (2011)
48. Chen, C.C.: On the selection of image compression algorithms. In: IEEE 14th International Conference on Pattern Recognition (ICPR'98) (1998).
49. Netravali, A.N., Haskell, B.G.: Digital Pictures—Representation and Compression. Plenum Press, New York (1988)
50. Ozturk, I., Sogukpinar, I.: Analysis and comparison of image encryption algorithms. World Acad. Sci. Eng. Technol. **3**, 26–30 (2005)
51. Singh, N., Sinha, A.: Optical image encryption using fractional Fourier transform and chaos. Opt. Lasers Eng. **46**, 117–123 (2008)
52. Hennely, B., Sheridan, J.T.: Image encryption and fractional Fourier transform. Optik **114**, 251–265 (2003)
53. Liu, S., Yu, L., Zhu, B.: Optical image encryption by cascaded fractional Fourier transform with random phase filtering. Opt. Commun. **187**, 57–63 (2001)
54. Jayaraman, S., Esakkirajan, S., Veerakumar, T.: Digital Image Processing. Tata McGraw Hill Pte. Ltd, New Delhi (2009)
55. Puri, A., Chen, T.: Multimedia Systems, Standards, and Networks. Marcel Dekker Inc., New York (2000)
56. Trappe, W., Washington, L.C.: Introduction to Cryptography with Coding Theory. Prentice Hall, Upper Saddle River, NJ (2001)
57. Unnikrishnan, G., Joseph, J., Singh, K.: Optical encryption by double-random phase encoding in the fractional fourier domain. Opt. Lett. **25**, 887–889 (2000)
58. Liu, Z.J., Liu, S.T.: Double image encryption based on iterative fractional Fourier transform. Opt. Commun. **275**, 324–329 (2007)
59. Joshi, M., Shakher, C., Singh, K.: Image encryption and decryption using fractional Fourier transform and radial Hilbert transform. Opt. Lasers Eng. **46**, 522–526 (2008)
60. Tao, R., Xin, Y., Wang, Y. : Double image encryption based on random phase encoding in the fractional Fourier domain. Opt. Express **15**, (2007).
61. Puri, A., Chen, T.: Multimedia Systems, Standards, and Networks. Marcel Dekker Inc., New York (2000)
62. Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A concrete security treatment of symmetric encryption. Proceedings of the 38th Symposium on Foundations of Computer Science, IEEE (1997).
63. Bovik, A.C.: The Essential Guide to Image Processing. Elsevier, Burlington, MA (2008)
64. Qiao, L., Nahrstedt, K.: Comparison of MPEG encryption algorithms. Int. J. Comput. Graphics **22**(3), (1998).
65. Maniccam, S.S., Bourbakis, N.G.: Image and video encryption using SCAN patterns. Pattern Recognit. **37**(4), 725–737 (2004)