

Review of Computing Algorithms for Discrete Fractional Fourier Transform

Muhammad Irfan, Liying Zheng and Haroon Shahzad

Department of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China

Abstract: Discrete Fractional Fourier Transform (DFRFT) has received lots of attention in last two decades because of its superior benefits and wide applications in various fields. In this study we present a comparative analysis of the most famous algorithms for the computation of DFRFT. Analysis is done on the parameters like time complexity, accuracy, consistency, basic mathematical properties and generalization of Discrete Fourier Transform (DFT) and approximation of continuous Fractional Fourier Transform (FRFT). Main objective of the research is to portray the major advantages and disadvantages of the previously proposed algorithms so that appropriate algorithm may be selected as per requirements. On the basis of study it has been observed that there exist several definitions and algorithm for computing the DFRFT. These algorithms are based on different techniques, such as eigenvectors, chirp convolution, spectral decomposition, non-Fresnel integral, or orthogonal projection. Each of these algorithms has its own advantages and disadvantages. Despite of these developments, we still feel that there has been dire need of a standard definition and computing method for accurate and efficient computation of DFRFT.

Keywords: Computing algorithm, discrete fractional Fourier transform, Fourier transform, time-frequency representation, time varying signals

INTRODUCTION

The Discrete Fourier Transform (DFT) has proved itself a powerful tool in various fields especially for signal processing and analysis. The Fractional Fourier Transform (FRFT) is a generalization of the Fourier transforms (Soo-Chang and Min-Hung, 2001). The FRFT was introduced in 1980. Since then, the FrFT has been a hot research topic because of its benefits and advantages. It is being used in various fields such as quantum mechanics, quantum optics, optical systems, signal and image processing, communication and solution of differential equations (Namias, 1980; Xingpeng *et al.*, 2004). Fourier transform provides the signal's spectral content, but it fails when we need the time location of the spectral components. Time-frequency patterns are important in analysis of signals like non-stationary or time-varying signals. In order to analyze such signals, time-frequency representations are used. The FRFT has proved itself a powerful tool for the analysis of time-varying signals by representing rotation of a signal in time-frequency plane (David, 1999; Mendlovic and Ozaktas, 1993; Liying and Daming, 2010; Alieva and Bastiaans, 2001; Soo-Chang and Man Hung, 1996). In various fields where Fourier Transform (FT) and frequency domain concepts are used, performance can be enhanced through the use of FRFT (Ozaktas *et al.*, 1999). It has close relationship with some other transforms like Fourier, chirp, wavelet and linear

canonical transforms (Soontorn, 2002; Rajiv and Kulbir, 2005). Discrete counter part of FRFT, DFRFT has attracted lots of attention in last two decades. The DFRFT has been digitally computed using various approaches that can be classified as eigenvectors (decomposition) based (Çag˘atay *et al.*, 2000; Ozaktas *et al.*, 2001; Bultheel and Martinez-Sulbaran, 2004; Soo-Chang *et al.*, 2005; Bai, 2010), chirp convolution based (Ozaktas *et al.*, 1996; Arikan *et al.*, 1996), weighted summation based (Min-Hung, 2003), orthogonal projection based (Soo-Chang *et al.*, 1999), computation algorithm with zooming-in ability (Zhao *et al.*, 2008), spectral decomposition of kernel and linear combination-type DFRFT methods, group theory and impulse train type derivations and so on. These definitions and computing methods have their own advantages and disadvantages, with respect to time complexity, accuracy, internal consistency, analytical elegance as that of DFT, generalization of DFT in the same sense as continuous FRFT generalizes to continuous ordinary Fourier transform, satisfaction of properties of continuous FRFT and recovery of signal from transformation (Atakishiyev *et al.*, 1999; Cariolaro *et al.*, 2000; Soo-Chang and Min Hung, 1997) etc.

In this study we present a comparative analysis of some famous definitions and computing methods of the DFRFT. So far, there does not exist any standard definition of DFRFT, as in the case of DFT. Thus, many authors claimed their proposed definition and computing

method to be better than others. In such situation we have done a comparative analysis of some famous methods of the DFRFT including eigenvectors (decomposition) based computing method, chirp convolution based computing method, weighted summation based method, orthogonal projection based method, non-Fresnel integral based computing method etc. And the parameters like time complexity, accuracy, consistency, acceptability, generic properties, generalization of DFT and approximation of FRFT and recovery of signal from transformation have been evaluated and discussed.

The study is organized as follows: “Preliminaries” introduces the fractional Fourier transform and its discrete counterpart DFRFT. In “Computing Methods for DFRFT” we discuss various computing methods of DFRFT. In “Analysis and Results” we have done analysis of some famous DFRFT computing methods. “Conclusion” provides the concluding remarks.

PRELIMINARIES

Fractional Fourier transforms: The FRFT was introduced in mathematical literature with the same notation in 1980. By studying the brief history and development of the FRFT, it seems that it was independently reinvented by a number of researchers. FRFT is the generalization of the classical Fourier transform. It depends on a parameter α , which is $\alpha = a \pi/2$, as shown in Fig. 1 and can be interpreted as a rotation by an angle α in the time-frequency plane or it can also be defined as decomposition of the signal in terms of chirps.

It is classified as a member of more general class of integral transformations known as quadratic complex exponential, so, it's intrinsically complex to evaluate through analytical methods as well as numerical integrations. The main reason is that such operations require very high sampling rates which increased time complexity (Ozaktas *et al.*, 1996).

Mathematically, the FRFT can be defined by various ways; for example, through linear integral transform by specifying its linear transform kernel, by fractional power of ordinary Fourier transform through Hermite-Gaussian functions which are eigenfunctions of Fourier transform, as a special one parameter subclass of the class of linear canonical transform as it represents rotation in the time-frequency plane, through coordinate multiplication and differentiation operators.

However any definition of FRFT should obey the following three postulates to qualify (Bai, 2010):

- It should be continuous for all real values of α .
- Can be reduced to an ordinary order when α is an integer.
- It should obey the additive property i.e., $F^{\alpha+\beta} [f(x)] = F^{\alpha} [F^{\beta} \{f(x)\}] = F^{\beta} [F^{\alpha} \{f(x)\}]$

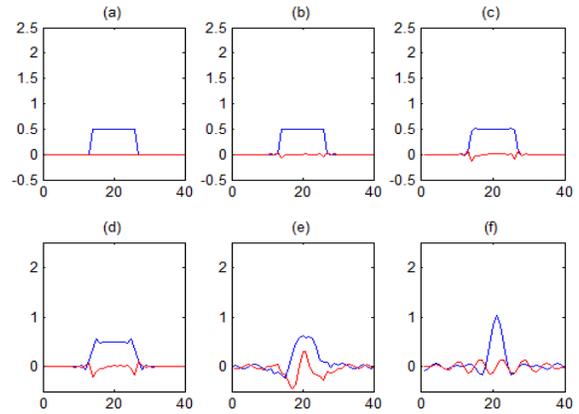


Fig. 1: Real (blue) and imaginary (red) parts of FRFT of rectangle function computed for different value of rotation angle α

Below given is the most famous definition of FRFT. Hermite-Gaussian functions are considered as eigenfunctions of ordinary Fourier transform (Ozaktas *et al.*, 1999, 1996). Here $\Psi(n)$ denotes the Hermite-Gaussian functions and its value is:

$$\Psi_n(x) = \frac{2^{1/4}}{\sqrt{2^n n!}} H_n(\sqrt{2\pi} x) \exp(-\pi x^2) \quad (1)$$

The α th order continuous FRT can be defined (Çag˘atay *et al.*, 2000) for interval $0 < |\alpha| < 2$ through its integral kernel:

$$\{F^\alpha f\}(t_\omega) = \int_{-\infty}^{\infty} K_\alpha(t_\omega, t) f(t) dt \quad (2)$$

$$K_\alpha(t_\omega, t) = K_\Phi e^{j\pi(t2_\alpha \cot\Phi - 2t_\omega t \csc\Phi + t^2 \cot\Phi)} \quad (3)$$

where

$$K_\Phi = e^{-j(\frac{\pi \operatorname{sgn}(\Phi)}{4} \frac{\Phi}{2})} / |\sin\Phi|^{0.5}$$

and

$$\Phi = \alpha\pi / 2$$

The kernel in (3) has the following spectral expansion:

$$K_\alpha(t_\omega, t) = \sum_{k=-\infty}^{\infty} \Psi_k(t_\omega) e^{-j\pi/2ka} \Psi_k(t) \quad (4)$$

where, $\Psi_k(t)$ denotes the k th Hermite-Gaussian function and t_α denotes the variable in the α th-order fractional Fourier domain. This kernel maps the function into its fractional Fourier transform. As fractional Fourier transform is generalization of Fourier transform, when $\alpha = 1$ in (2) it reduces to the ordinary Fourier transform (Çag˘atay *et al.*, 2000).

Discrete fractional Fourier transform: Santhanam and McClellan first reported the study on DFRFT in 1995 (Rajiv and Kulbir, 2005). In 2000, Pei and Ding classified definitions of DFRFT according to the methodologies used for calculations (Pei and Ding, 2000; Soo-Chang and Jian-Jiun, 2000). DFRFT definitions can also be classified into various types like:

- Simplest way to compute DFRFT is through the sampling of the continuous fractional Fourier transform, it is called direct form of DFRFT. But this definition has drawbacks like it does not obey some basic properties like unitary, additively, reversibility and closed-form properties
- In improved sampling-type continuous FRFT is properly sampled to get the better results
- DFRFT can also be computed using the linear combination of identity operator, DFT, time inverse operation and IDFT, that's why this typed is called linear combination-type. Computed transform matrix is orthogonal, additive and reversible but it has drawbacks like the results do not match with the continuous FRFT
- Eigenvector decomposition-type DFRFT can be computed through eigenvectors of the DFT matrix and fractional power of the DFT matrix. Results of DFRFT computed in this way are similar to the continuous FRFT; it also satisfies the basic properties like orthogonality, additively and reversibility
- Group theory-type DFRFT derived through the multiplication of DFT and periodic chirps. This definition satisfies the properties like rotational property of Wigner distribution, additively and the reversibility. This definition has its own limitations
- Impulse train-type DFRFT is considered as a special case of continuous FRFT as the input function is taken as periodic and equally spaced impulse train. Biggest drawback of this approach is that it is could not be defined for all fractions.

Here we take a definition which is in spirit of the definition in Eq. (4) (Ozaktas *et al.*, 2001). Discrete fractional Fourier transform matrix F^α can be defined as the discrete analogue of Eq. (4):

$$F_{ll'}^\alpha = \sum_{n=0}^N h_{gnl} e^{-\frac{i\alpha n n'}{2}} h_{gnl'} \quad (5)$$

where, h_{gnl} is the l th element of h_{g_n} and $F_{ll'}^\alpha$ is the $l'l'$ th element of F^α , with $0 \leq l \leq N-1, 0 \leq l' \leq N-1$. From (5) it can be deduced that fractional Fourier transform of a vector f is simply $f_a = F^\alpha f$. The above given definition of discrete fractional Fourier transform fulfills the basic properties of FRT like unitarity, index additivity and reduction to ordinary transform when $\alpha = 1$ and approximation of continuous FRFT. It is clear from

the above equation that discrete fractional Fourier transform can also be defined in relation with discrete Hermite-Gaussian and function can be expanded through discrete Hermite-Gaussian functions (Ozaktas *et al.*, 2001). Here h_{g_n} denotes the discrete Hermite-Gaussian function.

COMPUTING METHODS FOR DFRFT

DFRFT is difficult to compute by direct sampling of continuous FRFT because the kernel function of the continuous FRFT exhibits drastic oscillation and the oscillation amplitude varies a lot for different order of FRFT (Qi-Wen *et al.*, 2009). Such intrinsic complexities attracted numerous researchers to devise DFRFT computing algorithms. Such algorithms can be divided into two broad categories, fast FRFT algorithms and discrete FRFT algorithms. Most of the fast algorithms are FFT based. These fast algorithms can be further divided into two types, single FFT based (Cariolaro *et al.*, 1998; Marinho and Bernardo, 1998) and (fast) convolution based (Xingpeng *et al.*, 2004).

In this section we will have deep analysis of some famous approaches devised by different authors, as it's impossible to analyze all approaches here in this study.

Eigenvectors based method: Various authors adopted eigenvectors based algorithm for definition and computation of DFRFT (Soo-Chang and Man Hung, 1996; Çağatay *et al.*, 2000; Ozaktas *et al.*, 2001; Bultheel and Martinez-Sulbaran, 2004; Soo-Chang *et al.*, 2005; Soo-Chang and Min Hung, 1997). Candan, Kutay and Ozaktas (Çağatay *et al.*, 2000) proposed the definition which is based on a particular set of eigenvectors of DFT matrix which are said to be discrete counterpart of set of Hermite-Gaussian functions. The Hermite-Gaussian functions are considered to be eigenfunctions of fractional Fourier transform, it can be inferred from this fact that the Hermite-Gaussian functions are eigenfunctions of the ordinary Fourier transform when $\alpha = 1$ (Bultheel and Martinez-Sulbaran, 2004) as shown in Fig. 2. This definition of the DFRFT generalizes the discrete Fourier transform in the same way that continuous fractional Fourier transform generalizes the ordinary continuous Fourier transform; furthermore the definition is unitary, index additive and reduces to DFT for unit order (Çağatay *et al.*, 2000).

Mathematically it can be computed as:

$$F^\alpha [m, n] = u_k [m] \sum_{n=0}^{N-1} (\lambda_k)^\alpha u_k [n] \quad (6)$$

where, $u_m [n]$ is an arbitrary orthonormal eigenvector set of DFT matrix and λ_k be the associated eigenvalues. We have to generate F^α matrix to compute discrete FRT, in order to generate F^α matrix following steps are required. First of all generate matrices V and H . V is a matrix that decomposes an arbitrary vector $f[n]$ into its

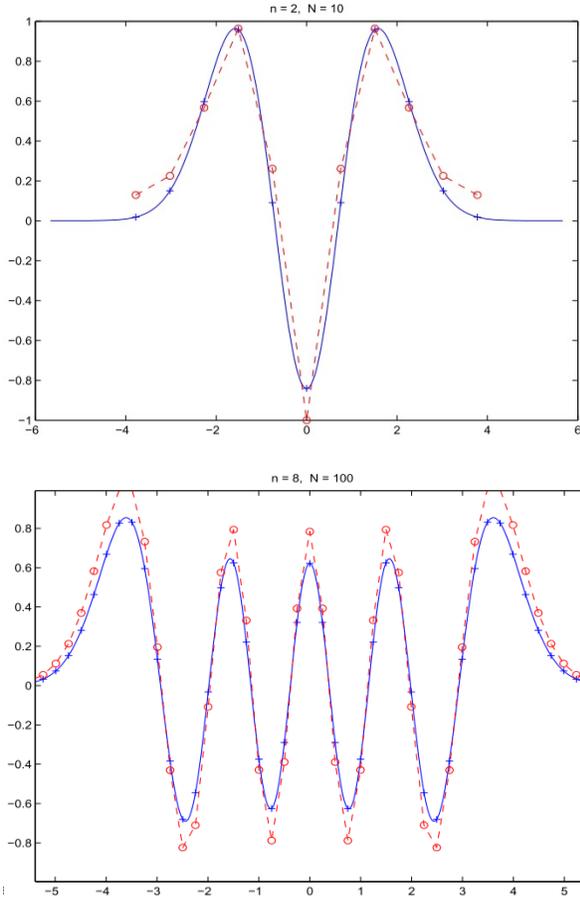


Fig. 2: The continuous Gauss-hermite functions (solid line) and the eigenvectors (o) of the discrete FRFT matrix

even and odd components. It maps the even part of the N -dimensional vector $f[n]$ to the first $(\frac{N}{2} + 1)$ components and odd part to the remaining components. Furthermore, V matrix is both unitary and symmetric. H is a real symmetric matrix, it has tridiagonal structure and commutes with DFT matrix (Bradley and Kenneth, 1982).

Eigenvector set of H as well as DFT matrix is unique and orthogonal. H is a discrete counter part of continuous Hermite-Gaussian matrix. After finding V and H matrices, next step is to find E_v and O_d matrices and find their eigenvectors e_k, o_k and eigenvalues.

$$VHV^{-1} = VHV = \begin{bmatrix} E_v & 0 \\ 0 & O_d \end{bmatrix} \quad (7)$$

Since eigenvectors of the DFT matrix are either even or odd vectors (McClellan and Parks, 1972), so the eigenvectors of VHV^{-1} are also either even or odd vectors. Then sort these eigenvectors and values after it compute $u_{2k} = V[e_k^t | 0 \dots 0]^T$ and $u_{2k+1} = V[0 \dots 0 | o_k^t]^T$ through e_k, o_k . Then F^α is computed as defined in (6). In this method computation of eigenvectors takes the larger part of computation time.

Total computation time complexity of this method is $O(N^2)$ which is very high but the main advantages of this algorithm are accuracy, generalization and consistency.

Pei and Hei (2000) and Soo-Chang and Man Hung (1996) proposed similar definition that is based upon the Eigen-decomposition of the DFT kernel matrix. After determining the eigenvalues of DFT kernel the transform kernel for DFRFT can easily be defined by taking fractional power of the eigenvalues. It gives similar transforms as those of continuous FRFT and it also holds properties like rotation etc. Computational complexity of this approach is $O(N^2)$ (Soo-Chang and Min Hung, 1997), which is very high.

Chirp convolution based method: The FRFT is closely related to the chirp function. Chirp based computation of DFRFT was proposed and elaborated by various authors (Xingpeng *et al.*, 2004; Bultheel and Martinez-Sulbaran, 2004; Ozaktas *et al.*, 1996; Arikan *et al.*, 1996). The FRFT is a member of more general class, i.e., linear canonical transform whose member can be broken down into succession of simpler operations. So Chirp based computation of DFRFT is based on the breakdown of FRFT into simple operations like chirp multiplication chirp convolution and again chirp multiplication. Mathematically, it can be written as Ozaktas *et al.* (1996):

$$g(x) = \exp\left[-i\pi x^2 \tan\left(\frac{\phi}{2}\right)\right] f(x) \quad (8)$$

$$g'(x) = A_\phi \int_{-\infty}^{\infty} \exp[i\pi\beta(x-x')^2] g(x') dx' \quad (9)$$

where $\beta = \csc\phi$

$$f_\alpha(x) = \exp\left[-i\pi x^2 \tan\left(\frac{\phi}{2}\right)\right] g'(x) \quad (10)$$

In Eq. (8-10), a signal $f(x)$ is firstly multiplied by a chirp function. $f(x)$ should be interpolated before this step. Then $g(x)$ is convolved with chirp function, after that the result $g'(x)$ is multiplied by a chirp function again. This algorithm works for $-1 < \alpha \leq 1$. If value of α is outside this interval different properties are used to compute the desired result. Before the computation of DFRFT the signal was interpolated, so at the end the resultant $f_\alpha(x)$ will be subsampled as the result should have the same length as that of original signal. The reason for expanding the signal and padding it with zeros was mainly because the convolution of signal should not be contaminated by boundary effects in the middle of the convolution signal. There exist various algorithms for interpolation and sub sampling of the signals.

Equation (8-10) can also be written as Bultheel and Martinez-Sulbaran (2004).

$$x^2 \cot \alpha - 2x \operatorname{Ecsca} + \operatorname{E}^2 \cot \alpha = x^2 (\cot \alpha - \operatorname{csca}) + (x - \operatorname{E})^2 \operatorname{csca} + \operatorname{E}^2 (\cot \alpha - \operatorname{csca}) \quad (11)$$

Eq. (11) is equivalent to Eq. (8-10), applying chirp convolution with sweep rate csca between two chirp multiplications with sweep rate $\cot \alpha$.

Definition proposed here satisfies the basic requirement of mapping from the samples of original function to the samples of the fractional Fourier transform. However in contrary to the ordinary DFT, this definition does not uniquely determine the discrete fractional Fourier transform, furthermore it is not exactly the fractional analog of Fast Fourier transform (FFT). It uses FFT techniques with complexity $N \log N$, it is a fast algorithm. The computational complexity of proposed method is $O(N \log N)$, which is reasonable.

Orthogonal projection based method: In Soo-Chang *et al.* (1999), orthogonal projection based definition and method to compute DFRFT is proposed. It uses Hermite eigenvectors of DFT and retains the eigenvalue to eigenfunction relation as that of continuous FRFT:

$$A = \hat{U}^T X \quad (12)$$

$$F^{2\alpha/\pi} = \hat{U} D^{2\alpha/\pi} \hat{U}^T X = \hat{U} D^{2\alpha/\pi} A \quad (13)$$

$$= \sum_{n=0}^{N-1} a_k e^{-jk\alpha} \hat{u}_k \quad (14)$$

Here $A = [a_0, a_1, \dots, \dots, a_{N-1}]^T$, coefficients a'_k are the inner products of signal and eigenvectors, $\hat{U} = [\hat{u}_0 | \hat{u}_1 | \dots | \hat{u}_{N-1}]$ for odd value of N and for even value of N its value is $\hat{U} = [\hat{u}_0 | \hat{u}_1 | \dots | \hat{u}_{N-2} | \hat{u}_N]$ and D is a diagonal matrix having different values for even and odd length signal. It provides similar transform and rotational properties as those of continuous FRFT. Furthermore, it has the same relation to FRFT as that of DFT to continuous Fourier transform. It provides similar transforms and results as those of the continuous case. Moreover, it obeys unitary and rotation properties. The time complexity for implementing this algorithm is $O(N^2)$, which is very high.

Weighted summation method: The In this approach discrete fractional Fourier transform can be computed by weighted summation of DFRFT's with special angles. Pei and Hei proposed separate methods for even and odd length signals (Soo-Chang and Min-Hung, 2001; Min-Hung and Soo-Chang, 2003), whereas a generic form of this method valid for signals of any length, either even or odd, proposed in Min-Hung and Soo-Chang (2003).

For a discrete signal x having odd length N with rotation angle α is computed as:

$$X_\alpha = \sum_{n=0}^{N-1} B_n X_{n\beta} \quad (15)$$

where $\beta = 2\pi/N$ and weighted coefficients B_n can be computed as:

$$B_n = \operatorname{IDFT}\{e^{-jka}\} = \frac{1}{N} \sum_{n=0}^{N-1} e^{-jka} e^{\frac{j2\pi nk}{N}} \quad (16)$$

The weighting coefficients are computed from IDFT operation.

For a signal x having even length:

$$X_\alpha = \sum_{n=0}^N B_n X_{n\beta} \quad (17)$$

where, $\beta = 2\pi/(N+1)$ and weighted coefficients B_n can be computed as

$$B_n = \operatorname{IDFT}\{e^{-jka}\} = \frac{1}{N+1} \sum_{n=0}^N e^{-jka} e^{\frac{j2\pi nk}{N+1}} \quad (18)$$

These The special angles are multiples of $2\pi/N$ for the odd case and are multiples of $2\pi/(N+1)$ for even length. The N -point and $(N+1)$ point IDFT is required for odd length and for even length signals respectively. In both cases an odd-point IDFT is computed to get the weighting coefficients. The popular FFT algorithm can only be applied while the length is power of 2; therefore, it cannot help us get the weighted coefficients. So weighting coefficients are obtained from an IDFT operation both for even and odd length signals. Computational complexity of this method is $O(N^2)$, which is very high. Furthermore it requires (N^2) storage space.

Non-Fresnel integral based method: Now we discuss an alternative definition and method to compute discrete fractional Fourier transform. This approach does not require Fresnel integral. Eq. for definition of FRFT through this method can be written as Ozaktas *et al.* (1996):

$$\{F^\alpha f\}(x) = A_\Phi e^{-i\pi\alpha x^2} \int_{-\infty}^{\infty} e^{-i2\pi\beta x x'} \left[e^{i\pi\alpha x'^2} f(x') \right] dx' \quad (19)$$

where, $\alpha = \cot\Phi$ and $\beta = \operatorname{csc}\Phi$.

Derived equation for computation of discrete fractional Fourier transform is as

$$\{F^\alpha f\}\left(\frac{m}{2\Delta x}\right) = (A_\Phi/2\Delta x) \sum_{n=-N}^N e^{i\pi\left(\alpha\left(\frac{m}{2\Delta x}\right)^2 - 2\beta\left[\frac{mn}{(2\Delta x)^2}\right] + \alpha\left(\frac{n}{2\Delta x}\right)^2\right)} f\left(\frac{n}{2\Delta x}\right) \quad (20)$$

Eq. (20) gives us the samples of the fractional Fourier transform in terms of the original function.

Computation of discrete fractional Fourier transform using this equation has $O(N^2)$ time complexity. It can be modified to reduce time complexity, as:

$$\{F^\alpha f\}(m/2\Delta x) = A_\phi/2\Delta x e^{i\pi(a-\beta)\left(\frac{m}{2\Delta x}\right)^2} \sum_{n=-N}^N e^{i\pi\beta\left(\frac{m-n}{2\Delta x}\right)^2} e^{i\pi(a-\beta)^2 f\left(\frac{n}{2\Delta x}\right)} \quad (21)$$

In Eq. (21) we can see that it is convolution of $e^{i\pi\beta\left(\frac{m-n}{2\Delta x}\right)^2}$ and chirp-modulated function. As we know that convolution can be in $O(N \log N)$ time using FFT. So, the overall computational time complexity of this method is reduced to $O(N \log N)$. This method maps the N samples of the original function to the N samples of the transform very accurately. But it does not fulfill properties like unitary and index additivity.

Algorithm with zooming-in ability: One of the major disadvantages of chirp based algorithm and some of other famous algorithms discussed above is that, they can only calculate the entire fractional spectra with a fixed low resolution. In various applications such as the detection and estimation, this algorithm with fixed resolution cannot be used as it does not meet these requirements. Biggest advantage of this algorithm is that user can freely choose computational resolution and zoom in any interested portion of the fractional spectra. This ultimately improves estimation accuracy. Mathematically it can be defined as Zhao *et al.* (2008).

$$f_{\alpha,\lambda,p}[m] = \left(\frac{A_\phi}{2\sqrt{N}}\right) e^{j\pi(aN)(\lambda)^2} e^{\frac{j\alpha\lambda m\pi}{P}} e^{\frac{j\pi(a-p\beta)m^2}{4NP^2}} \times \sum_{n=-N}^N e^{\frac{j\pi\beta(m-n)^2}{4PN}} \left[e^{\frac{j\pi(p\alpha-\beta)n^2}{4PN}} e^{-j\pi\beta\lambda N} \overline{f}[m] \right] \quad (22)$$

In (22), λ is the shift factor having value $\lambda = \frac{x_i}{\Delta x}$, $-0.5 \leq \lambda \leq 0.5$, it is used for relative location of x_i in entire spectra interval $[-\frac{\Delta x}{2}, \frac{\Delta x}{2}]$ and P is a zoom factor, its value is $P = \frac{1}{(\Delta l 2\Delta x)}$, actually it's the measurement of the ratio of the standard sampling $1/((2\Delta x))$ to the calculated space Δl . Eq. (22) works for range $0.5 \leq |\alpha| \leq 1.5$. By using index additivity property of FRFT the properties like $F^\alpha = F^{-1}F^{1+p}$ and $F^\alpha = F^1F^{p-1}$, it can be extended to all values of α . Time complexity of this algorithm is same as that of chirp convolution based algorithm is $O(N \log N)$. Main advantage of this approach is that it offers the fine evaluation of structure of the partial spectra which ultimately improves estimation accuracy. Furthermore this algorithm provides flexible and efficient analysis of the fractional spectra.

Discrete Fourier transform based method: This definition of FRFT is the enhanced version of Shih's definition (Shih, 1995). Proposed algorithm for computing DFRFT is derived from the FFT and is based on the periodicity of the eigenvalues of the FRFT (Bai, 2010). This definition may also be generalized to compute two-dimension DFRFT, by which the fractional order spectrum of the digital images may be obtained.

Shih proposed the definition of the FRFT which is based on the fact that the FRFT is a weighted composition of the j th order Fourier transforms of the original function, where ($j = 0, 1, 2, 3$). This definition is based on the inference that the DFRFT is a weighted composition of the first four orders of the DFT. Mathematically it can be expressed as:

$$F^\alpha[f(n)] = \sum_{m=0}^3 \exp\left(-\frac{i3\pi(a-m)}{4}\right) \cos\left(\frac{\pi(a-m)}{2}\right) \times \cos\left(\frac{\pi}{4}\right) (\pi\alpha - m4) f_m(k) \quad (23)$$

where $f_m(k)$ is the m th order of the DFT of the original sequence $f(n)$, for $m = 0, 1, 2, 3$, here DFT can be computed through use of FFT in an efficient way. Computational complexity of this algorithm is $O(N \log N)$, which is very reasonable.

ANALYSIS AND RESULTS

General behavior and properties: It is believed that FRFT is richer in theory and more flexible in application, but it is expensive in implementation. Here in this section DFRFT of a rectangle signal is computed using the above mentioned approaches with different values of transform order, α .

In Fig. 3, the FRFT is computed with eigenvectors based approach. General properties of this approach are accuracy, generalization of DFT and consistency. It is clear from Fig. 3 that the FRFT value is equal to ordinary DFT for $\alpha = 1$ and original signal is recovered for $\alpha = 2$. Although eigenvectors based method has similar results as those of continuous FRFT, but it requires complex matrix calculations and its physical explanation is not explicit enough. In this method signals may be recovered from their transforms with errors (Bai, 2010).

In Fig. 4, the FRFT is computed using the chirp convolution based approach. The FRFT is equal to the ordinary DFT for $\alpha=1$ and original signal is recovered for $\alpha = 2$. Other graphs show the value of the FRFT for different values of α . It satisfies the basic requirement of mapping from the samples of original function to the samples of the fractional Fourier transform's his approach has a major drawback that this algorithm can only calculate the entire fractional spectra with a fixed

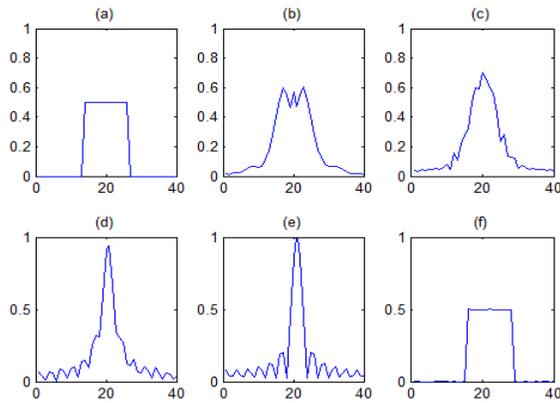


Fig. 3a: Rectangle function data. Magnitudes of its fractional Fourier transform computed through eigenvectors based approach for (b) $\alpha = 0.25$, (c) $\alpha = 0.50$, (d) $\alpha = 0.75$, (e) $\alpha = 1.0$ and (f) $\alpha = 2.0$ same as original function

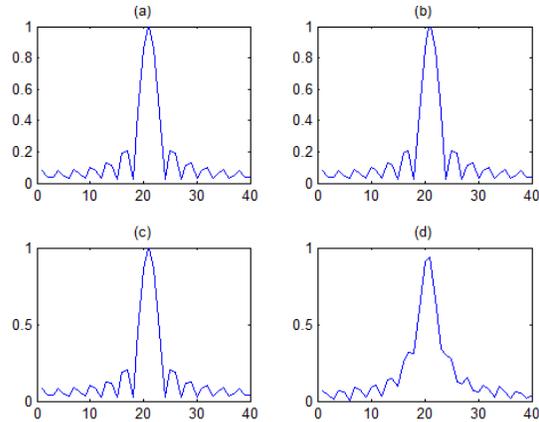


Fig. 6a: Ordinary DFT of rectangle function computed through *fftw*. Magnitudes of fractional Fourier transforms computed through (b) Eigenvectors based approach, (c) Chirp convolution based approach and (d) Orthogonal projection based approach

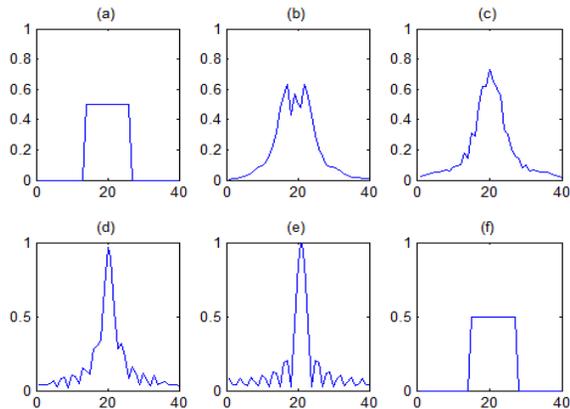


Fig. 4a: Rectangle function data. Magnitudes of its fractional Fourier transform computed through chirp convolution based approach for (b) $\alpha = 0.25$, (c) $\alpha = 0.50$, (d) $\alpha = 0.75$, (e) $\alpha = 1.0$ and (f) $\alpha = 2.0$ same as original function

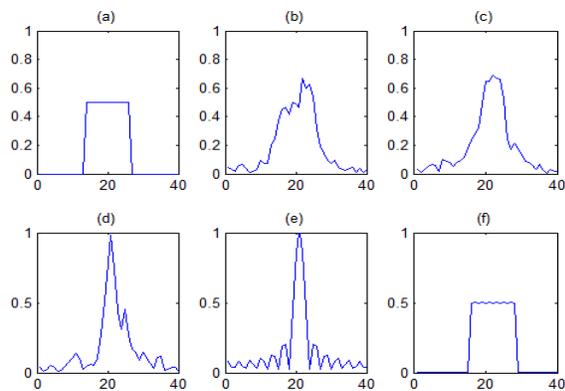


Fig. 5a: Rectangle function data. Magnitudes of its fractional Fourier transform computed through orthogonal projection based approach for (b) $\alpha = 0.25$, (c) $\alpha = 0.50$, (d) $\alpha = 0.75$, (e) $\alpha = 1.0$ and (f) $\alpha = 2.0$ same as original function

low resolution. It means that for an N -point input sample, the outputs of the above algorithms are N -point uniformly spaced samples over the entire fractional spectra, resulting in the fixed resolution computation.

Because of this limitation these algorithms cannot be used in many applications such as detection and estimation, zooming etc (Zhao *et al.*, 2008). Furthermore, this algorithm works for $-1 < \alpha \leq 1$. If value of α is outside this interval different properties like $\{F^2 f\}(x) = f(-x)$ and $\{F^4 f\}(x) = f(x)$ to compute the desired result.

In Fig. 5, the FRFT is computed using orthogonal projection based approach. It provides similar transforms and results as those of the continuous case. Original signal is recovered for $\alpha = 2$. One drawback of such numerical integration based FRFT methods is that they do not obey rotation rules and the signal cannot be recovered from its inverse transform (Soo-Chang *et al.*, 1999).

Most of the algorithms discussed above couldn't provide the zoom function, it's only the zooming-in based algorithm which provides this functionality furthermore it offers the fine evaluation of structure of the partial spectra which ultimately improves estimation accuracy.

Reduction to ordinary Fourier transforms: One of the important properties of fractional Fourier transform is that it should reduce to ordinary DFT for $\alpha = 1$. Figure 6 shows the behavior of different approaches in this regard. It is clear from Fig. 6 that almost all the approaches mentioned above produce the same result as that of ordinary DFT for $\alpha = 1$, however eigenvectors based approach and chirp convolution based approach produces better results as compared to orthogonal projection based approach.

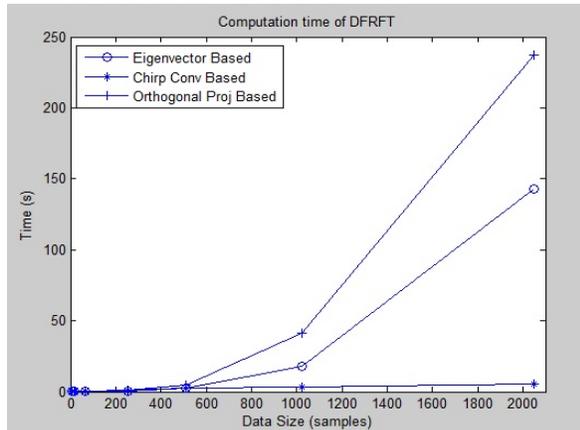


Fig. 7: Computation time of DFRFT

Computation time: Computation time complexity is an important factor in comparative analysis of various approaches used for computation of FRFT.

Different approaches have different time complexity. Generally, eigenvectors based approach, weighted summation based approach, and orthogonal projection based approach have $O(N^2)$ whereas computational time complexity of chirp convolution based is $O(N \log N)$. However, in some scenarios, chirp based methods are not preferred, as they need coordinate scaling which requires double interpolation, which in result requires extra computation power (Ozaktas *et al.*, 1996; Yongun *et al.*, 2004). Time taken for computation of DFRFT of a chirp signal of different length through different approaches is represented through graph in Fig. 7. It is clear from the figure that time taken by chirp convolution based approach is very less as compared to time taken by eigenvectors based as well as orthogonal projection based approach. In this experiment, DFRFT of a chirp signal is computed. In weighted summation based generic method, although the computational complexity is $O(N^2)$, however, when length of signal is in power of 2, then weighted coefficients for angular decomposition can be computed in a fast way through FFT.

CONCLUSION

In this study, we have presented a critical review and comparative analysis of some important existing approaches available for computation of discrete fractional Fourier transform. Comparison is done on the basis of parameters such as computation complexity, accuracy, and consistency, generalization of DFT, and approximation of continuous FRFT. We have taken different types of simulated data and processed it through implementation of above-mentioned methods. Processing results indicate that all above-mentioned definitions can be reduced to ordinary DFT for $\alpha = 1$. It is also observed that none of the methods fulfill all the basic properties.

Eigenvectors based definition and orthogonal projection based method correctly approximate the continuous FRFT, whereas these two methods take huge time for computation of DFRFT as compared to chirp convolution based method.

ACKNOWLEDGMENT

This study was supported in part by the National Natural Science Foundation of China (Grant No. 61003128).

REFERENCES

- Alieva, T. and M.J. Bastiaans, 2001. Wigner distribution and fractional fourier transform. Proceeding of 6th International Symposium on Signal Processing and its Applications (ISSPA), Aug. Kuala Lumpur, Malaysia, pp: 13-16.
- Arikan, O., M.A. Kutay, H.M. Ozaktas and O.K. Akdemir, 1996. The discrete fractional fourier transform. Proceeding of IEEE SP Symp. Time-Freq. Anal., 4: 205-207.
- Atakishiyev, N.M., L.E. Vicent and K.B. Wolf, 1999. Continuous vs. discrete fractional fourier transform. J. Comput. Appl. Math., 107: 73.
- Bai, T., 2010. A fast two-dimension discrete fractional fourier transform algorithm and its application on digital watermarking. Proceeding of 12th IEEE International Conference on Communication Technology (ICCT '10).
- Bradley, W.D. and S. Kenneth, 1982. Eigenvectors and functions of the discrete fourier transform. IEEE T. Acoust. Speech, Assp-30(1).
- Bultheel, A. and H.E. Martinez-Sulbaran, 2004. Computation of the Fractional Fourier Transform. Kath. Univ., Leuven.
- Çağatay, C., M.A. Kutay and H.M. Ozaktas, 2000. The discrete fractional fourier transform. IEEE T. Signal Process., 48(5).
- Cariolaro, G., T. Erseghe, P. Kraniuskas and N. Laurenti, 1998. A unified framework for the fractional fourier transform. IEEE T. Signal Process., 46(12).
- Cariolaro, G., T. Erseghe, P. Kraniuskas and N. Laurenti, 2000. Multiplicity of fractional fourier transforms and their relationships. IEEE T. Signal Process., 48(1): 227-241.
- David, M., 1999. The fractional fourier transform: A tutorial. Proceedings of the IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing (NSIP'99), Antalya, Turkey.
- Liyang, Z. and S. Daming, 2010. Maximum amplitude method for estimating compact fractional fourier domain. IEEE Signal Proc. Let., 17(3).
- Marinho, F.J. and L.M. Bernardo, 1998. Numerical calculation of fractional fourier transforms with a single fast-fourier transform algorithm. J. Opt. Soc. Am. A, 15: 2111-2116.

- McClellan, J.H. and T.W. Parks, 1972. Eigenvalue and eigenvector decomposition of the discrete fourier transform. *IEEE T. Acoust. Speech*, AU-20: 66-74.
- Mendlovic, D. and H.M. Ozaktas, 1993. Fractional fourier transforms and their optical implementation-I. *J. Opt. Soc. Am. A*, 10: 1875-1881.
- Min-Hung, Y., 2003. Angular decompositions for the discrete fractional signal transforms. *Proceedings of International Symposium on Circuits and Systems (ISCAS '03)*.
- Min-Hung, Y. and P. Soo-Chang, 2003. A method for the discrete fractional fourier transform computation. *IEEE T. Signal Process.*, 51(3).
- Namias, V., 1980. The fractional fourier transform and its application in quantum mechanics. *J. Inst. Math. Appl.*, 25: 241-265.
- Ozaktas, H.M., O. Arikan, M.A. Kutay and G. Bozdaki, 1996. Digital computation of the fractional fourier transform. *IEEE T. Signal Process.*, 44(9).
- Ozaktas, H.M., M.A. Kutay and D. Mendlovic, 1999. *Introduction to the Fractional Fourier Transform and its Applications: In Advances in Imaging Electronics and Physics*. Ch. 4. Academic, New York.
- Ozaktas, H.M., Z. Zalevsky and M.A. Kutay, 2001. *The Fractional Fourier Transform with Applications in Optics and Signal Processing*. John Wiley and Sons Inc., Wiley.
- Pei, S.C. and J.J. Ding, 2000. Closed form discrete fractional and affine fractional transforms. *IEEE T. Signal Process.*, 48: 1338-1353.
- Qi-Wen, R., Z. Hai-Ying, Z. Zhong-Zhao and S. Xue-Jun, 2009. The analysis of the discrete fractional fourier transform algorithms. *Proceeding of Canadian Conference on Electrical and Computer Engineering (CCECE '09)*.
- Rajiv, S. and S. Kulbir, 2005. Fractional fourier transform: A novel tool for signal processing. *J. Indian Inst. Sci.*, 85: 11-26.
- Shih, C.C., 1995. Fractional of fourier transform. *Opt. Comm.*, 118: 495-498.
- Soo-Chang, P. and D. Jian-Jiun, 2000. Closed-form discrete fractional and affine fourier transforms. *IEEE T. Signal Process.*, 48(5).
- Soo-Chang, P. and Y. Man-Hung, 1996. Discrete fractional fourier transform. *Proceeding of IEEE International Symposium on Circuits and Systems (ISCAS)*.
- Soo-Chang, P. and Y. Min-Hung, 1997. Improved discrete fractional fourier transform. *Opt. Lett.*, 22(14).
- Soo-Chang, P. and Y. Min-Hung, 2001. A novel method for discrete fractional fourier transform computation. *Proceeding of the 2001 IEEE International Symposium on Circuits and Systems, ISCAS, Sydney, NSW*.
- Soo-Chang, P., Y. Min-Hung and T. Chien-Cheng, 1999. Discrete fractional fourier transform based on orthogonal projection. *IEEE T. Signal Process.*, 47(5).
- Soo-Chang, P., H. Wen-Liang and D. Jian-Jiun, 2005. Discrete fractional fourier transform based on new nearly tridiagonal commuting matrices. *Proceeding of IEEE International Conference on Acoustics, Speech and Signal Processing, Proceedings (ICASSP '05)*.
- Soontorn, O., 2002. The generalized discrete fractional fourier transforms. *Proceeding of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.
- Xingpeng, Y., T.X.W. Qiaofeng, X. Yong, Y. Yingbai and J. Guofan, 2004. Improved fast fractional-fourier-transform algorithm. *J. Opt. Soc. Am. A*, 21(9).
- Yongun, S., Y. Shaopu, L. Xiandong and P. Cunzhi, 2004. Effect of interpolation methods on fast computation of fractional fourier transform. *Proceeding of 7th International Conference on Signal Processing (ICSP '04)*.
- Zhao, X., R. Tao and Y. Wang, 2008. A fast algorithm for fractional fourier transform with zooming-in ability. *Proceeding of 4th International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM '08*.