Accurate and Fast Discrete Polar Fourier Transform

A. Averbuch — CS Department, Tel-Aviv University, Tel-Aviv 69978 Israel

R.R. Coifman — Math. Department, Yale University, New-Haven 06520 CT

D.L. Donoho — Stat. Department, Stanford University, Stanford 94305 CA

M. Elad — CS Department, The Technion, Haifa 32000 Israel

M. Israeli — CS Department, The Technion, Haifa 32000 Israel

Abstract

In this article we develop a fast high accuracy Polar FFT. For a given two-dimensional signal of size $N \times N$, the proposed algorithm's complexity is $O(N^2 \log N)$, just like in a Cartesian 2D-FFT. A special feature of our approach is that it involves only 1-D equispaced FFT's and 1D interpolations. A central tool in our approach is the pseudo-polar FFT, an FFT where the evaluation frequencies lie in an oversampled set of non-angularly equispaced points. The pseudo-polar FFT plays the role of a halfway point – a nearly-polar system from which conversion to Polar Coordinates uses processes relying purely on interpolation operations. We describe the conversion process, and compare accuracy results obtained by unequally-sampled FFT methods to ours and show marked advantage to our approach.

1 Introduction

Fourier analysis is a fundamental tool in mathematics and mathematical physics, and also in theoretical treatments of signal and image processing. The discovery, popularization, and digital realization of fast algorithms for Fourier analysis – so called FFT – has had far reaching implications in science and technology in recent decades. The scientific computing community regards the FFT as one of the leading algorithmic achievements of the 20th century [1]. In fact, even ordinary consumer-level applications now involve FFT's – think of web browser decoding a JPEG images – so that development of new tools for Fourier analysis of digital data may be of potentially major significance.

In this paper we develop tools associated with Fourier analysis in which the set of frequencies is equispaced when viewed in polar coordinates. When dealing with images defined over the continuum, $f(x) = f(x_1, x_2),\ x = (x_1, x_2) \in \mathbf{R}^2$, let

$$\hat{f}(\phi) = \int f(x) \exp(-i\phi' x) dx$$

be the usual continuum Fourier transform of f. Writing the frequency $\phi = \{rcos(\theta), rsin(\theta)\}$ in polar coordinates, we let $\tilde{f}(r,\theta) = \hat{f}(\phi(r,\theta))$. In this paper, the term *Polar Fourier Transform* refers to the operation $\tilde{f}(r,\theta) = \mathcal{PF}\{f(x)\}$, namely, getting $f(x)$ in Cartesian variables and computing $\tilde{f}(r,\theta)$ defined with polar variables in the frequency domain. While changes of variables are, of course, banal *per se*, their significance lies in the change of viewpoint they provide. The polar FT can be a powerful tool for organizing our understanding of operators and functions on the two dimensional continuum.

Turning to digital images sampled over a Cartesian grid, we similarly desire a transform that produces polar coordinates in the frequency domain, and in a way that have many of the properties of the continuum polar FT, including relations to rotation, registration, Radon transform, and so on. Naturally, we desire such transform to be a fast one, similar to the digital transform that produces the Cartesian grid in the frequency domain. Unfortunately, the prevailing belief seems to be that there cannot exist such an algorithm [2]. This is intimately related to the strong reliance of the FFT on separability of the axes, and the equispaced samples on both grids - properties lost when dealing with polar coordinates.

In this paper we propose a notion of polar FT which is well suited for digital data – a procedure which is faithful to the continuum polar FT concept, highly accurate, fast, and generally applicable. We define the polar grid of frequencies $\xi_{p,q} = \{\xi_x[p,q], \xi_y[p,q]\}$ in the circle inscribed in the fundamental region $\xi \in [-\pi, \pi)^2$, and, given digital Cartesian data $f[i_1, i_2]$ we define the polar FT to be the collection of samples $\{F(\xi_{p,q})\}$, where $F(\xi)$ is the trigonometric polynomial

$$F(\xi_{p,q}) = \sum_{i_1} \sum_{i_2} f[i_1, i_2] \exp\left(-i_1 \xi_x[p,q] - i_2 \xi_y[p,q]\right).$$

Thinking of the polar Discrete Fourier Transform (PDFT) mapping $\mathcal{PDFT} : f[i_1, i_2] \rightarrow F(\xi_{p,q})$ as a

linear operator, we can also consider a generalized inverse procedure of it, going back from discrete polar Fourier data to cartesian spatial data.

This paper briefly defines the polar FT concept for digital data, the associated fast algorithms, and discuss its features such as accuracy and computational complexity. In this paper we concentrate on the forward transform. We refer the reader to a more detailed report of our work in [3].

2 Current State of the Art

For evaluating the Fourier transform of a equally spaced Cartesian grid image over non-equispaced non Cartesian set of frequencies, a body of literature we refer to as USFFT (short for Unequally Spaced Frequency Fourier Transform) proposes the use of the regular FFT, followed by interpolation [4, 5, 6, 7, 8, 9, 10, 11, 12, 13].

The existing state-of-the-art expressed by this literature can be adapted to produce an approximate Polar FFT (PFFT) and its inverse. While we are unaware of any publication which does so, we describe here the basic principles of such an adaptation. A suggested PFFT, based on our interpretation of ideas in the existing literature on USFFT, is described in Algorithm A.

Task : Given $f[i_1, i_2]$, $0 \le i_1, i_2 \le N-1$, compute the polar Fourier data $F(\xi_{p,q})$.

Step 1 : Define an oversampled $NS \times NS$ Cartesian grid frequencies $\bar{\xi}_{k_1,k_2} \in [-\pi, \pi]^2$.

Step 2 : Apply regular FFT at $\bar{\xi}_{k_1,k_2}$.

Step 3 : For each polar destination frequency point identify nearby points from $\bar{\xi}_{k_1,k_2}$.

Step 4 : Perform approximate interpolation from the neighbors in $\bar{\xi}_{k_1,k_2}$ to the desired polar location at $\xi_{p,q}$.

Algorithm A - Polar Fast Fourier Transform (PFFT) based on USFFT methodology.

There are various uncertainties, complexities, and shortcomings associated with an implementation using the current state-of-the-art ideas from the USFFT: (i) the need to choose parameters; (ii) the need to adopt high oversampling S in order to obtain high interpolation accuracy; (iii) the burden of identifying, for each irregular grid point, the corresponding Cartesian points and loading those from memory efficiently; and (iv) the need to have the "forward" and the "inverse" polar FT's defined this way to relate well to each other.

3 The New Approach

The approach we propose for PFFT factors the problem into two steps: first, a *Pseudo-Polar* FFT is applied, in which a pseudo-polar sampling set is used, and second, a conversion from pseudo-polar to polar FT is performed.

3.1 Pseudo-Polar FFT

At the heart of the method proposed here for the PFFT we use the pseudo-polar FFT – an FFT where the evaluation frequencies lie in an oversampled set of non-angularly equispaced points (see Figure 1). The pseudo-polar grid is separated into two groups – the Basically Vertical (BV) and the Basically Horizontal (BH) subsets. The BV group (filled dots in Figure 1) is defined by

$$BV = \left\{ \begin{array}{c} \xi_y = \frac{\pi \ell}{N} \quad \text{for} \quad -N \le \ell < N \\ \xi_x = \frac{2\pi m \ell}{N^2} \quad \text{for} \quad -\frac{N}{2} \le m < \frac{N}{2} \end{array} \right\} \qquad (1)$$

and a similar definition describes the BH group. Whereas the polar grid is built as the points on the intersection between linearly growing concentric circles and angularly equispaced rays, the pseudo-polar uses a set of linearly growing concentric squares and a linearly growing sloped rays.

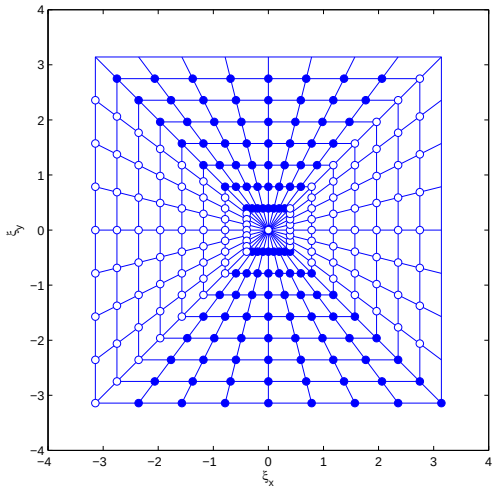

Figure 1: The pseudo-polar grid and its separation into BV and BH coordinates ($N = 8$).

This polar-like 2D grid enables fast Fourier computation. This grid has been explored by many since the 1970-s and until recently. The pioneers in this field are Mersereau and Oppenheim [14] who proposed the Linogram grid as an alternative grid to the polar one. Later work by Munson and others [15, 16] have shown

how these ideas can be extended and used for tomography. Recently, The pseudo-polar grid (essentially the Linogram with a proper choice of oversampling) was proposed as the base for a stable forward and inverse Radon transform called *Fast Slant-Stack* [17].

For this grid we have the following fundamental result [17, 3]:

Theorem 1 *Given the signal* $f[i_1, i_2]$, $0 \le i_1, i_2 < N$, *the EXACT evaluation of the FT on the oversampled pseudo-polar grid with* NS *concentric squares and* $2NP$ *rays can be done by 1D-FFT operations only, and with complexity of* $120N^2PS\log(NS)$ *operations.*

3.2 From Pseudo-Polar to Polar

Similar to the USFFT approach, we suggest to compute the polar-FT values based on a different grid for which fast algorithm exists, and then go to the polar coordinates via an interpolation stage. However, instead of using the cartesian grid in the first stage, we use the pseudo-polar one discussed in the previous section. Since this grid is closer to the polar destination coordinates, there is a reason to believe that this approach will lead to better accuracy and thus lower oversampling requirements. However, as we shall see next, beyond the proximity of the pseudo-polar coordinates to the polar ones, other very important benefits are the ability to perform the necessary interpolations via pure 1D operations without loosing accuracy, and the ability to manipulate the data in an orderly fashion that enables smart memory management. These properties are vital in understanding the superiority of the proposed scheme over traditional USFFT methods.

We define the polar coordinate system based on the Pseudo-Polar one, with manipulations that lay out the necessary interpolation stages discussed later on. We concentrate on the basically-vertical frequency sampling points in the pseudo-polar grid as given in (1). The polar BV ones are obtained by two operations:

Rotate the Rays: In order to obtain a uniform angle ray sampling as in the polar coordinate system, the rays must be rotated. This is done by replacing the term $2m/N$ in ξ_x above with $\tan(\pi m/2N)$. The result is a set of points organized on concentric squares as before, but the rays are spread differently with linearly growing angle instead of linearly growing slope. Figure 2 presents this step as would be done as an interpolation stage. Rotating the rays amounts to 1D operation along horizontal lines (for the BV points). A set of N uniformly spread points along this line are to be replaced by a new set of N points along the same line in different locations (marked as small squares) owing to the uniform angle sampling of the new rays.

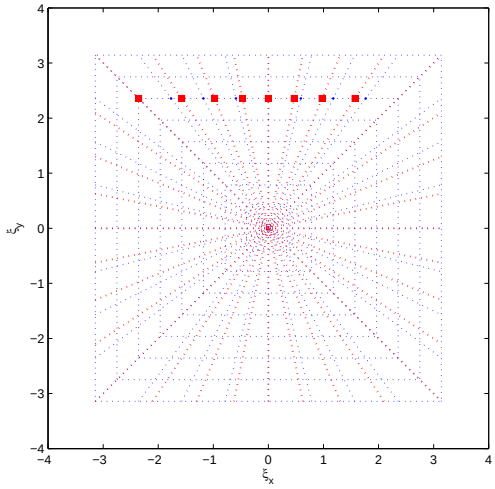

Figure 2: First Interpolation stage.

We have the following property:

Theorem 2 *The first interpolation stage from the pseudo-polar grid to the polar one is applied on trigonometric polynomials of order* N.

The implication of this observation is that this function is relatively smooth and with a moderate oversampling (factor of $2-4$) a near-perfect interpolation is expected for a small neighborhood operation.

Circle the Squares: In order to obtain concentric circles as required in the polar coordinate system, we need to circle the squares. This is done by dividing both ξ_x and ξ_y by a constant along each ray, based on its angle, and therefore a function of the parameter m, being $R[m] = \sqrt{1+\tan^2\left(\frac{\pi m}{2N}\right)}$. Figure 3 presents this step as would be done as an interpolation stage. Circling the squares amounts to 1D operation along rays, which again is a 1D operation. A set of $2N$ uniformly spread points along this line are to be replaced by a new set of $2N$ points along the same line in different locations (marked as small squares). However, this time the destination points are also uniformly spread. We have the following result:

Theorem 3 *The second interpolation stage from the pseudo-polar grid to the polar one is applied on band-limited functions with required maximal sampling period of* π/N.

This result means that if this function is sampled in this rate along the entire ray (from $-\infty$ to ∞) we have a complete representation of it that enables knowledge of its values at any location. In our case, for a limited interval $z \in [-\pi, \pi]$ we have $2N$ samples, which is the

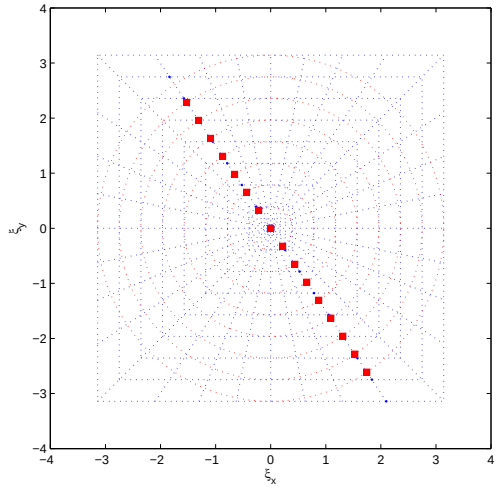

Figure 3: Second Interpolation stage.

critical sampling rate exactly. Thus, with a moderate oversampling we may represent this function very accurately. All this imply that this function is relatively smooth as well and lends itself for high accuracy interpolation.

4 Accuracy Analysis

There are many theoretical and empirical ways to study the performance of the proposed scheme, and compare it to the alternative USFFT approach. We focus here on one such possible study, and refer the reader to the more detailed report [**?**] for other comparisons.

Leaning on the linearity of the proposed transform and its approximations, and using a matrix-vector representation, we have that for a given signal $\underline{x}$ the transform error for the USFFT method is given by $(\mathbf{T}_e - \mathbf{T}_u)\underline{x}$, where $\mathbf{T}_e$ is the USFFT transform and $\mathbf{T}$ is the exact one. We solve,

$$\max_{\underline{x}} \frac{\|(\mathbf{T}_e - \mathbf{T}_u)\,\underline{x}\|_2^2}{\|\underline{x}\|_2^2}, \tag{2}$$

and this way, seek the worst-possible signal $\underline{x}$ to maximize the error, while being of unit ℓ^2-norm. Clearly, the result is the first right singular vector of the matrix $(\mathbf{T}_e - \mathbf{T}_u)$, and the error is the first singular value squared [18]. Figure 4 presents the real and imaginary parts of these worst-case signals of size 16×16 for the USFFT (oversampling $S = 9$ in each axis) and the Polar-FFT (oversampling along the rays with $S_r = 20$ and along the squares with $S_s = 4$ – overall oversampling of 80, parallel to the one adopted for the USFFT), and the absolute frequency description of this signal. The USFFT worst error is $8.9e-3$ while the Polar-FFT one is $1.92e-6$. We see that the USFFT method is weaker, its worst signal is concentrated near the frequency origin where the method is most weak. Note that the worst signal is modulated (notice the shift from the center in the spatial domain) to result with a very non-smooth frequency behavior.

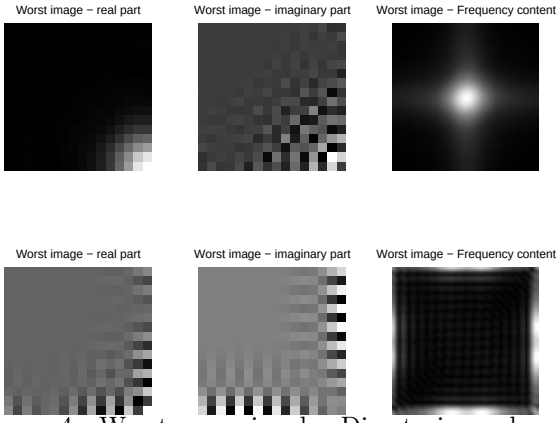

Figure 4: Worst case signal - Direct eigenvalue approach.

A problem with the above analysis is the difficulty in understanding the meaning of the error found, being a ratio between energies in the frequency and the spatial domains. An interesting alternative is the definition of worst signals by

$$\max_{\underline{x}} \frac{\|(\mathbf{T}_e - \mathbf{T}_u)\,\underline{x}\|_2^2}{\|\mathbf{T}_e\underline{x}\|_2^2} \quad \text{subject to} \quad \mathbf{F_1}\underline{x} = 0, \tag{3}$$

Put in words, we seek the worst signal such that it has a fixed energy in the polar frequency coordinates. However, if we do not restrict the solution, it will naturally concentrate its energy in the frequency domain corners for maximal effect. Thus, we add the constraint to force the signal to have zero energy in the frequency domain outside the π-radius circle. If $\mathbf{F}$ represents the regular cartesian FFT in a predetermined density, then $\mathbf{F}_1$ represents the rows from it corresponding to the frequency points outside the circle. This problem can be reposed as a generalized eigenvalue problem, and its results are shown in Figure 5. The USFFT worst error is $2.7e-6$ and with the Polar-FFT method the error we obtain is $1.0e-10$.

5 Conclusions

In this article we describe a fast high accuracy Polar FFT. For a given two-dimensional signal of size $N \times N$, the proposed algorithm produces a polar FFT with complexity of $O(N^2 \log N)$, just like in a Cartesian 2D-FFT. Two special features of this algorithm are (i)

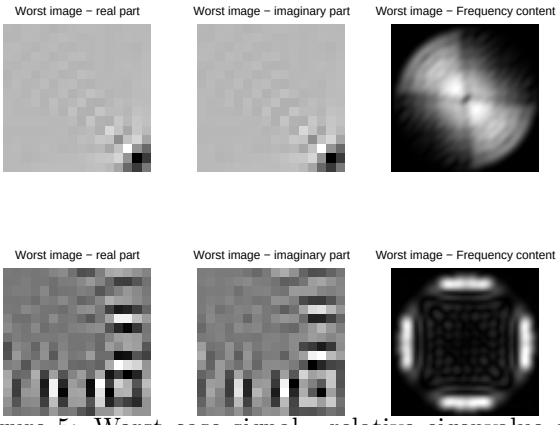

Figure 5: Worst case signal - relative eigenvalue approach with constraint.

it involves only 1-D equispaced FFT's and 1D interpolations, leading to highly efficient algorithm from the memory management point of view; and (ii) it leads to very high accuracy for moderate oversampling factors. It is shown that the presented approach is far more accurate than state-of-the-art methods known as Unequally-Spaced Fast Fourier Transform (USFFT) methods.

References

[1] Dongarra, J. and Sullivan, F. (2000) Introduction the top 10 algorithms, Special issue of *Computing in Science and Engineering*, Vol. 2, No. 1.

[2] Briggs, B. and Henson, E. (1995) *The FFT: an Owner's Manual for the Discrete Fourier Transform*, SIAM, Philadelphia.

[3] Averbuch, A., Coifman, R., Donoho, D.L., Eladm M. and Israeli, M. (2003) Accurate and Fast Discerete Polar Fourier Transform, *draft*.

[4] Suli, E. and Ware, A. (1991) A spectral method of characteristics for hyperbolic problems *SIAM J. Numer. Anal.*, Vol. 28, No. 2, pp. 423–445.

[5] Boyd, J.P. (1992) A fast algorithm for Chebyshev, Fourier, and sinc interpolation onto an irregular grid, *J. of Comp. Phys.*, Vol. 103, No. 2, pp.243–257.

[6] Beylkin, G. (1995) On the fast Fourier transform of functions with singularities, *Applied Computational Harmonics*, Vol. 2, pp. 263-381.

[7] Anderson, C. and Dahleh, M. D. (1998) Rapid computation of the discrete Fourier transform, *SIAM J. Sci. Comput.*, Vol. 17 pp. 913–919.

[8] Dutt, A. and Rokhlin, V. (1993) Fast Fourier transforms for nonequispaced data, *SIAM J. Sci. Comput.*, Vol. 14, pp. 1368–1393.

[9] Dutt, A. and Rokhlin, V. (1995) Fast Fourier transforms for nonequispaced data II, *Appl. Comput. Harmon. Anal.* Vol. 2, pp. 85–100.

[10] Ware, A. F. (1998) Fast approximate Fourier transform for irregularly spaced data, *SIAM Review*, Vol. 40, No. 4, pp. 838–856.

[11] Duijndam, A.j.W. and Schonewille, M.A. (1999), Nonuniform fast Fourier transform, *Geophysics*, Vol. 64, No. 2, pp. 539–551.

[12] Nguyen, N. and Liu, Q.H. (1999) The regular Fourier matrices and nonuniform fast Fourier Transforms, *SIAM Journal of Scientific Computing*, Vol. 21, No. 1, pp. 283–293.

[13] Fessler, J.A., and Sutton, B.P. (2003) Nonuniform Fast Fourier Transforms Using Min-Max Interpolation, *IEEE Trans. on Signal Processing*, Vol. 51, No. 2, pp. 560–574.

[14] Mersereau, R. M. and Oppenheim, A. V., (1974) Digital reconstruction of multidimensional signals from their projections, *Proc. IEEE*, Vol. 62, No. 10, pp. 1319–1338.

[15] Moraski, K. J. and Munson Jr., D.C. (1991) Fast tomographic reconstruction using chirp-z interpolation, in *Proceedings of 25th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, pp. 1052-1056.

[16] Choi, H. and Munson Jr., D.C., (1998) Direct-Fourier reconstruction in tomography and synthetic aperture radar, *Int. J. of Imaging Systems and Tech.*, Vol.9, No.1, pp. 1-13.

[17] Averbuch, A., Coifman, R., Donoho, D.L., and Israeli, M. (2003) Fast Slant Stack: A notion of Radon transform for data in a Cartesian grid which is rapidly computible, algebraically exact, geometrically faithful and invertible, to appear in *SIAM Scientific Computing*.

[18] Golub, G.H. and Van Loan C.F, (1996) *Matrix Computations*, John Hopkins University Press, Third edition.