Abstract

In this work we review the definition and basic properties of the fractional Fourier transform, along with its most common applications. We completely describe the numerical implementation of this linear transformation and use it to restore degraded images that are not suited for other typical image processing methods.

Fractional Fourier Transform for Optimal Image Filtering

Procesamiento Digital de Imágenes

Manuel Guizar Sicairos, David Said Martínez

Instituto Tecnológico y de Estudios Superiores de Monterrey

May 10, 2005

Contents

1 Introduction

Image processing and enhancement has been a popular subject of study since the development of digital picture capturing devices. The digital format of captured images permitted the automatic and accurate detection and measurement of features, pattern recognition, color and texture enhancement and segmentation of regions of interest [1]. This proved to be a very valuable tool in quality inspection for industrial production.

When the degradation is space-invariant, a Wiener filter may be applied with excellent results [2], however the current popular approach to restore images affected by a space-variant degradation is slow and computationally demanding [3].

In particular, Fourier-domain filtering has proven to be a very powerful tool for removing space-invariant degradations and other types of linear degradations, but it has very limited or no restoration power when used on images distorted by space-dependant degradations.

This work reviews typical properties and applications of the fractional Fourier transform and explores its capabilities as a tool for image filtering and restoration, to attack the areas where the traditional Fourier transform is insufficient. The remainder of this work presents a brief theoretical framework of the fractional Fourier transform, as well as its properties and a computer implementation algorithm. Additionally we describe space-variant degradation models and the appropriate filters to apply in any fractional domain. The optimal restoration filter is applied in fractional domains to sample images to illustrate the superior results from those obtained by filtering in the ordinary Fourier domain.

In the appendices we define the technical terms that are used throughout this manuscript and provide the complete computer code that was implemented.

2 Theoretical Framework

The Fourier series, named after Jean Baptiste Joseph Fourier is an integral transform that provides the frequency spectrum of a periodic function, decomposing it in an infinite sum of sinusoidal functions with discrete increases in their frequency. It was presented in 1807 and later published in 1822 in the *Théorie Analytique de la Chaleur* (The Analyitic Theory of Heat) with, according to the modern standards, informal conclusions [1]. Since then, Fourier's work has inspired a great number of transformations with a wide range of applications .

Since its publication, the fourier series, inspired a great number of breakthroughs including the Fourier transform, the fast Fourier transform algorithm and the fractional Fourier transform.

2.1 Fourier Transform

The Fourier transform is one of the best known integral transform, since its introduction it has been used in innumerable applications and has led to the development of other important transforms [4], [5].

The Fourier transform of a continuous function $f(x)$ is defined by [6]

$$F(u) = \mathcal{F}\{f(x)\} = \int_{-\infty}^{\infty} f(x) \exp(-i2\pi ux) dx, \tag{1}$$

where u is the Cartesian coordinate in the Fourier domain, often referred to as the frequency domain since the Fourier transform of a function represents its frequency content distribution. This interpretation is particularly valuable for noise removal where the noise signal often has a high frequency.

Since the Fourier transform is separable in Cartesian coordinates the two-dimensional version, which is mostly used for optical and image processing applications, can be straight-

forwardly obtained

$$F(u,v) = \mathcal{F}\{f(x,y)\} = \int\int_{-\infty}^{\infty} f(x,y)\exp(-i2\pi[ux+vy])dxdy. \tag{2}$$

This two-dimensional version is often used in image processing and restoration to remove noise and degradation from a given image [7]. If a digitized image is corrupted by a linear position-invariant degradation and additive noise, frequency domain filtering provides very good results by means of Wiener filtering [1]. With these degradation, the observed image $o(x,y)$ can be expressed as

$$o(x,y) = h(x,y) \otimes f(x,y) + \eta(x,y), \tag{3}$$

where $h(x,y)$ is the space-invariant degradation kernel, $\eta(x,y)$ is the additive noise term and $\otimes$ stands for the convolution operation.

The Wiener filtering approach is founded considering images and noise as uncorrelated random processes, and attempting to find an estimate $\hat{f}(x,y)$ of the uncorrupted image such that the error between them is minimized. In this approach the Fourier transform of the estimate is given by [1]

$$\hat{F}(u,v) = \left[\frac{1}{H(u,v)}\frac{|H(u,v)|^2}{|H(u,v)|^2+K}\right] O(u,v), \tag{4}$$

where $O(u,v) = \mathcal{F}\{o(x,y)\}$ is the Fourier transform of the observed image, $H(u,v) = \mathcal{F}\{h(x,y)\}$ is the transform of the degradation kernel and K is a specified constant. The actual estimate $\hat{f}(x,y)$ is obtained after an inverse Fourier transform of $\hat{F}(u,v)$.

This filtering approach has been improved by iterative application of the filter and is nowadays widely used in many image improvement applications [2], [8], [9]. However, the Weiner filtering method is only suitable for space-invariant degradations, where the degradation process can be expressed as a convolution, and generates considerable errors if applied on images that are affected by a space-dependant kernel.

2.2 Fractional Fourier Transform

The fractional Fourier transform is a linear integral transform, commonly used in signal processing and optics, it is a generalization of the Fourier transform and provides a smooth transition from the spatial coordinates to the spatial frequency domain [10].

The fractional Fourier transform has been used to solve partial differential equations to provide an extra degree of freedom [4], in digital watermarking for multimedia copyright protection [11], beam forming [12], target detection [13], signal reconstruction [14], [15], representation [16] and multiplexing [17].

The fractional powers $\mathcal{F}^a$ of the ordinary Fourier transform operation $\mathcal{F}$ correspond to rotation by angles $a\pi/2$ in the space-frequency plane, see figure 1. An ordinary Fourier transform is depicted in the phase plane as a $\pi/2$ rotation. The fractional domains correspond to oblique axes in the time-frequency plane, and thus the fractional Fourier transform is directly related to the Radon transform of the Wigner distribution [18], [19]. Of particular interest from image processing perspective is the concept of filtering in fractional Fourier domains.

The ath order fractional Fourier transform is a linear operation defined by the integral [20]

$$f_a(u) \equiv \mathcal{F}^a \{f(u)\} \equiv \int_{-\infty}^{\infty} K_a(u,u') f(u') du', \tag{5}$$

where

$$\begin{aligned} K_a(u,u') &\equiv A_\alpha \exp\left[i\pi\left(u^2 \cot\alpha - 2uu' \csc\alpha + u'^2 \cot\alpha\right)\right], && \text{(6a)} \\ A_\alpha &\equiv \sqrt{1 - i\cot\alpha}, && \text{(6b)} \\ \alpha &\equiv \frac{a\pi}{2}, && \text{(6c)} \end{aligned}$$

The ordinary Fourier transform can be used to compute the propagation pattern observed in the Fraunhoffer or far-field region, when you are far enough from the light source so that

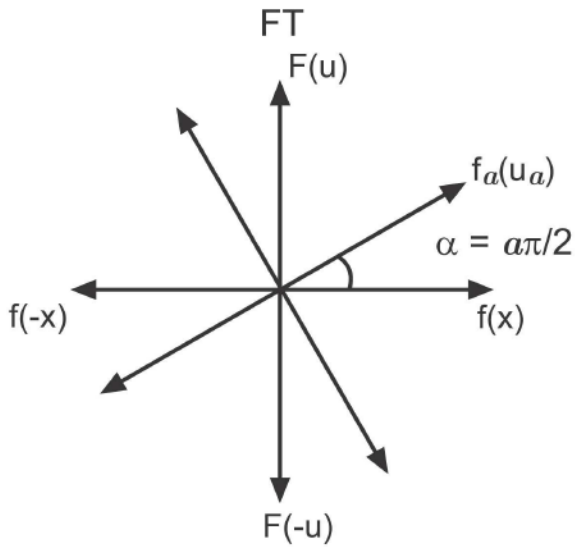

Figure 1: Graphical interpretation of the fractional Fourier transform.

its radiated field wave can be considered planar [21]. This property of the frequency domain is well-understood and is often used in optics to study and characterize optical beams [22].

Physically the fractional Fourier transform is intimately related to near-field or Fresnel diffraction in wave and beam propagation, when the radiated field must be considered mathematically spherical rather than planar, thus this transformation permits the computation of propagation patterns at any given point in the near-field region [23].

Figure 2 schematically depicts the optical representation of the fractional Fourier transform, the 0th-order transform is actually the identity or null operator and retrieves the wavefront at the radiation source, notice that upon applying the limit $a \to \infty$ in equations (6) the integral kernel becomes a limit definition of the Dirac delta function $K(u, u') = \delta(u - u')$.

The field at an intermediate point in the Fresnel diffraction region may be retrieved by computing a fractional transform of the field on the source plane, as the propagation distance increases so does the order of the transformation. In the limiting case of propagation beyond the near-field region, the transformation order is $a = 1$ and the field distribution is retrieved

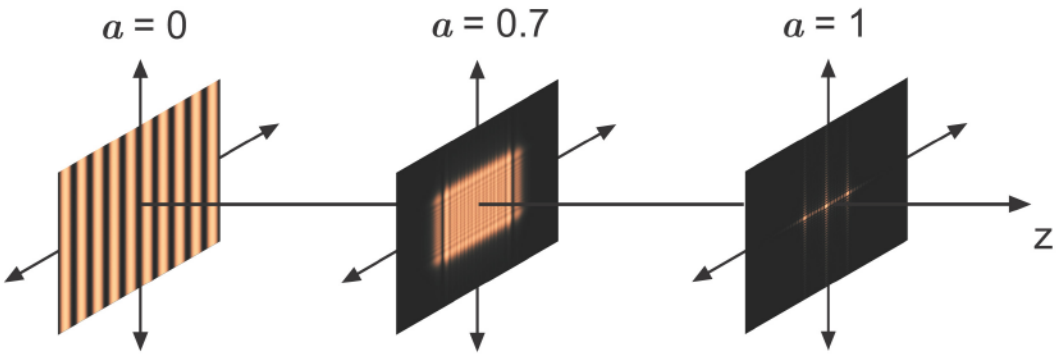

Figure 2: Optical interpretation of the fractional Fourier transform.

by using a Fourier transform, notice that in this case the kernel in equation (6a) is exactly that of the ordinary Fourier transform [21].

The fractional Fourier transform is evidently linear but not shift-invariant, several properties of this transformation have been mentioned or derived previously [20]. The properties of the fractional Fourier transform that are more relevant for the analysis in subsequent sections are listed in equations (7),

$$\text{Index additivity} \qquad \mathcal{F}^{a+b}\{f(u)\} = \mathcal{F}^{a}\left\{\mathcal{F}^{b}\{f(u)\}\right\}, \tag{7a}$$

$$\text{Null operator} \qquad \mathcal{F}^{0}\{f(u)\} = f(u), \tag{7b}$$

$$\text{Index periodicity} \qquad \mathcal{F}^{a+4n}\{f(u)\} = \mathcal{F}^{a}\{f(u)\}, \tag{7c}$$

$$\text{Parity operator} \qquad \mathcal{F}^{a+2}\{f(u)\} = f_{a+1}(-u), \tag{7d}$$

$$\text{Convolution} \qquad \mathcal{F}^{1}\{f(u) \otimes g(u)\} = f_{a+1}(u) g_{a+1}(u). \tag{7e}$$

We will take advantage of these properties when numerically implementing the optimal filtering for image restoration.

3 Computation of One-Dimensional Fractional Fourier Transform

In this section we will describe the implemented algorithm for computation of the one-dimensional fractional Fourier transform, along with the numerical tests that we performed to ensure proper representation of a function in the fractional domains and their results.

All computer codes were written in Matlab®, mainly because of its clear programming language and wealth of integrated mathematical functions that allow direct manipulation of images and clear display of results [24].

3.1 Computer Implementation

The numerical calculation of this transformation is of fundamental importance for the actual application of filtering techniques in fractional domains with computer programs of dedicated hardware. The most widely used algorithm for its computation is named the fast convolution algorithm, pioneered by Ozaktas *et al.* [25]. Although many more accurate discrete approaches of this transformation have been proposed [26]–[29], the fast convolution algorithm is faster, requires less computational effort and has an acceptable accuracy for most applications.

In the fast convolution algorithm, first we take advantage of the index periodicity property, equation (7c) to reduce the continuous order interval to the finite range $a \in [0, 4]$, to achieve this we simply take the modulus of the input transformation order after division by four.

The transformation order is further reduced to the interval $a \in [0, 2]$ by using the parity operator property shown in equation (7d), if the required transformation is in the range $a \in [2, 4]$ we can invert the input function and reduce the transformation order by two,

namely

$$\mathcal{F}^a\left\{f(x)\right\} = \mathcal{F}^{a-2}\left\{f(-x)\right\}, \tag{8}$$

this can be straightforwardly derived from equation (7d).

To retrieve accurate results, the fast convolution algorithm still requires an additional range decrease, since the convolution is performed between chirped functions that have their highest frequency values near even numbers of a. In order to perform the operation as far as possible from transformation orders $a = 0$ and $a = 2$, the original algorithm proposed that the transformation order should be reduced to the main interval [25] $a \in [0.5, 1.5]$. However, later the main interval was optimized through a theoretical analysis [30] and it was shown that a reduction of transformation error can be achieved if the main interval is $a \in [0.366, 1.366]$.

To reduce the interval $a \in [2, 4]$ to the main interval $a \in [0.366, 1.366]$ we take advantage of the index additivity property, equation (7a), and the fast Fourier transform algorithm, namely

$$\mathcal{F}^a\left\{f(x)\right\} = \mathcal{F}^{\mp 1}\left\{\mathcal{F}^{a\pm 1}\left\{f(x)\right\}\right\}. \tag{9}$$

The definition of the fractional Fourier transform, as expressed in equation (5), can be rewritten as a convolution operation

$$\begin{aligned} F_a(u) = \mathcal{F}^a\left\{f(x)\right\} &= A_\alpha B'_a(u) \int_{-\infty}^{\infty} f(x) B'_a(x) \exp\left[i\pi(u-x)^2 \csc\alpha\right] dx & (10)\\ &= A_\alpha B'_a(u) \left[f'_a(x) \otimes B_a(x)\right], & (11) \end{aligned}$$

where

$$\begin{aligned} f'_a(x) &= f(x) B'_a(x), & (12)\\ B_a(x) &= \exp\left(i\pi x^2 \csc\alpha\right), & (13)\\ B'_a(x) &= \exp\left[-i\pi \tan(\alpha/2) x^2\right], & (14) \end{aligned}$$

the operator $\otimes$ stands for convolution operation, and the functions $B_a(x)$ and $B'_a(x)$ are chirp functions.

By using the convolution theorem of the Fourier transform, equation (11) can be rewritten as a conventional multiplication

$$F_a(u) = \mathcal{F}^a\left\{f(x)\right\} = A_\alpha B'_a(u)\mathcal{F}^{-1}\left\{\mathcal{F}^1\left\{f'_a\right\}\mathcal{F}^1\left\{B_a(x)\right\}\right\} \tag{15}$$

The discrete convolution that is required for computation of the fractional transform can be performed by using the well-assessed fast Fourier transform algorithm. The Fourier transform of the chirp function $B_a(x)$ has an analytic closed form given by

$$\mathcal{F}^1\left\{B_a(x)\right\} = \sqrt{\frac{i}{\csc\alpha}}\exp\left(\frac{-i\pi\omega^2}{\csc\alpha}\right), \tag{16}$$

where the variable ω is used to clearly distinguish between the frequency domain and fractional Fourier domains. This analytic function was used, instead of computing the discrete Fourier transform of $B_a(x)$ to improve the algorithm accuracy.

3.2 Numerical Tests and Results

We numerically tested the accuracy and stability of the implemented one-dimensional algorithm. A fractional Fourier transform of the rectangular function was computed for several transformation orders using the previously described implementation. The rectangular function is defined as

$$\mathrm{rect}(x) = \begin{cases} 1 & |x| <= \frac{1}{2}, \\ 0 & |x| > \frac{1}{2}, \end{cases} \tag{17}$$

and has no analytic expression for its fractional transform, thus the results were directly compared to previously published numerical approximations [20]. Figure 3 show the magnitude and phase distribution of the computed transformations, they are remarkably similar to the results found in the literature, and they explicitly depict the smooth transition between the original function and the integer order Fourier transform.

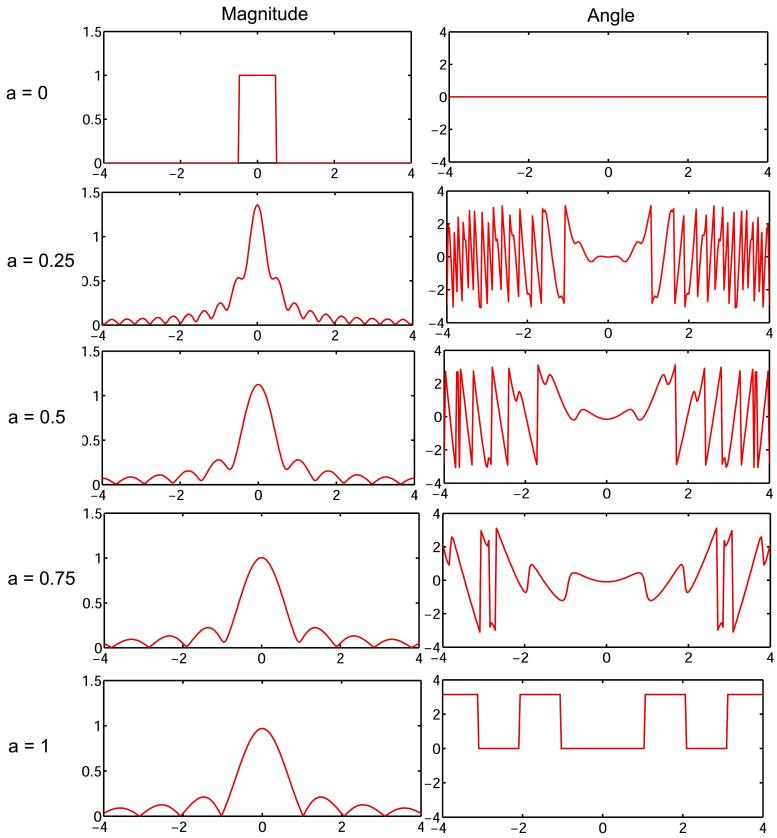

Figure 3: Magnitude and phase distribution for computed fractional Fourier transforms of the rectangular function.

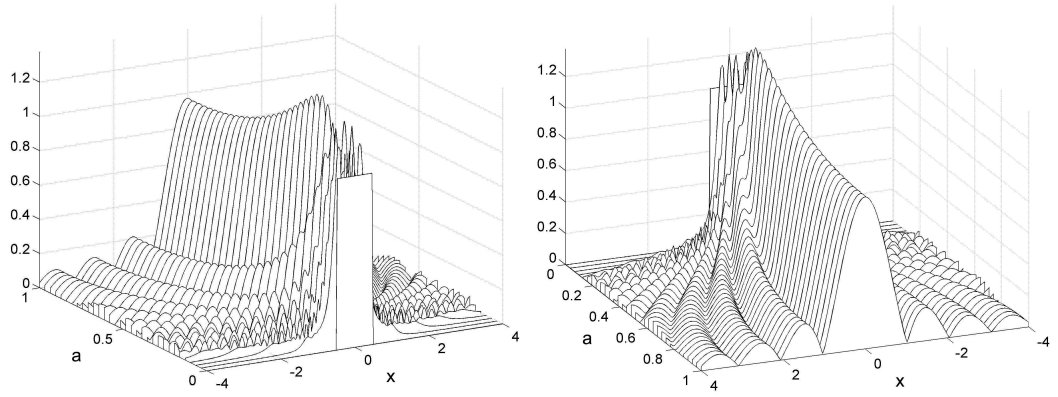

Figure 4: Iterative transformation of the rectangular function.

To test the method stability we implemented iterative transformations of the rectangular function with $a = 0.025$. After 40 successive transformations, due to the fractional transform additive property, the sine cardinal function should be retrieved, that is, the Fourier transform of the original input. The results of this test are shown in figure 4, where the a-th transforms of the input function are explicitly shown. Notice that after 40 transformations with $a = 0.025$ the retrieved function has the exact form of the magnitude of a sine cardinal function; also notice the resemblance of these results to the actual optical propagation of a finite square aperture.

The results from the performed numerical tests of the implemented one-dimensional gave excellent results, the computed transforms have a remarkable resemblance to previously published material [20], [23], [25].

4 Computation of Two-Dimensional Fractional Fourier Transform

For optical and image processing applications, the two-dimensional fractional Fourier transform is often needed. Of particular interest for noise removal and image restoration is the

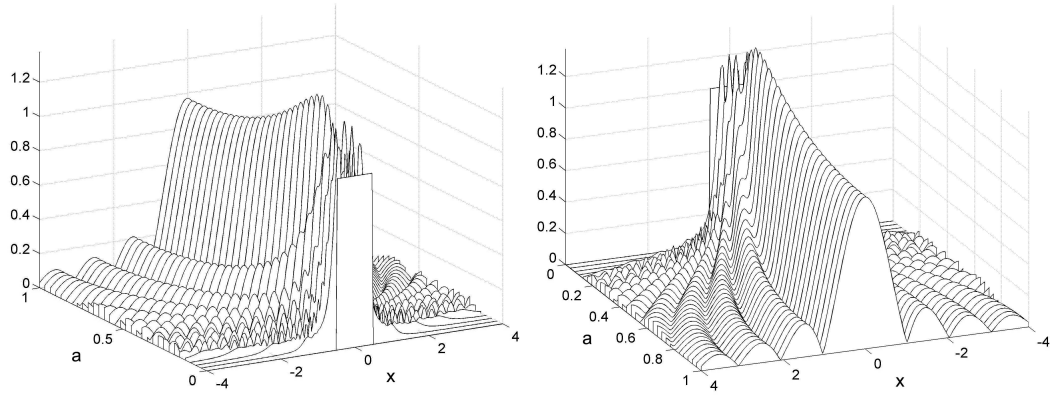

Figure 4: Iterative transformation of the rectangular function.

To test the method stability we implemented iterative transformations of the rectangular function with $a = 0.025$. After 40 successive transformations, due to the fractional transform additive property, the sine cardinal function should be retrieved, that is, the Fourier transform of the original input. The results of this test are shown in figure 4, where the a-th transforms of the input function are explicitly shown. Notice that after 40 transformations with $a = 0.025$ the retrieved function has the exact form of the magnitude of a sine cardinal function; also notice the resemblance of these results to the actual optical propagation of a finite square aperture.

The results from the performed numerical tests of the implemented one-dimensional gave excellent results, the computed transforms have a remarkable resemblance to previously published material [20], [23], [25].

4 Computation of Two-Dimensional Fractional Fourier Transform

For optical and image processing applications, the two-dimensional fractional Fourier transform is often needed. Of particular interest for noise removal and image restoration is the

application of the this transformation with different orders along the vertical and horizontal dimensions.

Later in this work it will became clear that the optimal domain for image processing is often a fractional domain with different transformation orders for each Cartesian coordinate, and that the results obtained are superior to those of the conventional Fourier transform.

4.1 Computer Implementation

The integral kernel of the two-dimensional fractional transform is defined as

$$f_{a_x,a_y}(x,y) = \int_{\infty}^{\infty}\int_{\infty}^{\infty} K_{a_x,a_y}(x,y;x',y')f(x',y')dx'dx', \tag{18}$$

where a_x and a_y denote independent transformation orders along the horizontal and vertical coordinate respectively.

The transformation kernel $K_{a_x,a_y}(x,y;x',y')$ is separable in Cartesian coordinates, namely

$$K_{a_x,a_y}(x,y;x',y') = K_{a_x}(x,x')K_{a_y}(y,y'). \tag{19}$$

This permits computing the two-dimensional transformation by independently integrating along the y axis followed by a corresponding integration along the x axis. With this approach, we take advantage of the one-dimensional algorithm described in section 3 to compute an a_y order transform along the columns of the input array and then apply an a_x order transform to the rows of the resulting matrix.

4.2 Numerical Tests and Results

To prove the method accuracy when computing transforms of different orders for each Cartesian dimension, we computed the $a_x = 0.8$ and $a_y = 0.4$ fractional transformation of a Gaussian function.

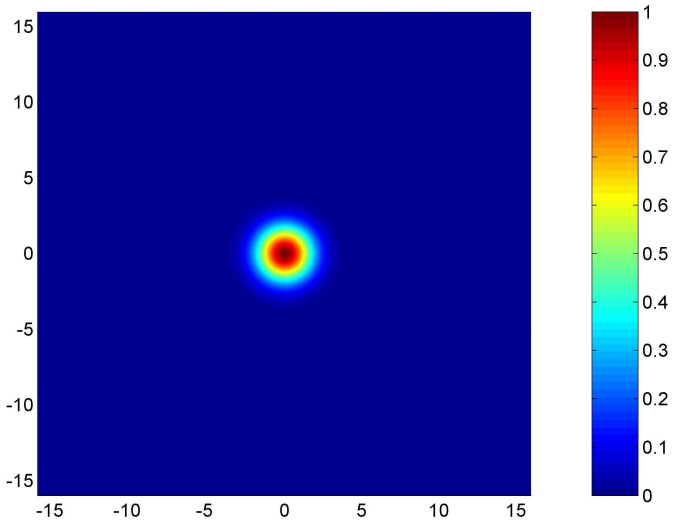

Figure 5: Magnitude of the Gaussian function.

The implemented input function $f(x, y)$ is given by

$$f(x, y) = \exp\left[\sigma\left(x^2 + y^2\right)\right], \tag{20}$$

where $\sigma = 0.1$ is a free parameter that defines the width of the function, its amplitude distribution is shown in figure 5.

Equation (20) can be expressed as a multiplication of two functions that depend exclusively in one of the Cartesian dimensions, and the one-dimensional transform of a Gaussian function is known analytically, thus we can directly compare the results of our numerical implementation with exact analytic solutions.

The transformation of equation (20) can be expressed as

$$\begin{aligned} f_{a_x,a_y}(x, y) &= \mathcal{F}^{a_x,a_y}\left\{f(x, y)\right\} = \mathcal{F}^{a_x,a_y}\left\{f(x)f(y)\right\} &(21)\\ &= \mathcal{F}^{a_x}\left\{f(x)\right\}\mathcal{F}^{a_y}\left\{f(y)\right\} = f_{a_x}(x)f_{a_y}(y), &(22)\end{aligned}$$

where the one-dimensional transform of a Gaussian function is given by

$$f_{a_x}(x) = \sqrt{\frac{1 - i\cot\alpha_x}{\sigma - i\cot\alpha_x}}\exp\left[i\pi x^2\frac{(\sigma^2 - 1)\cot\alpha_x}{\sigma^2 + \cot^2\alpha_x}\right]\exp\left[-\pi x^2\frac{\sigma\csc^2\alpha_x}{\sigma^2 + \cot^2\alpha_x}\right], \tag{23}$$

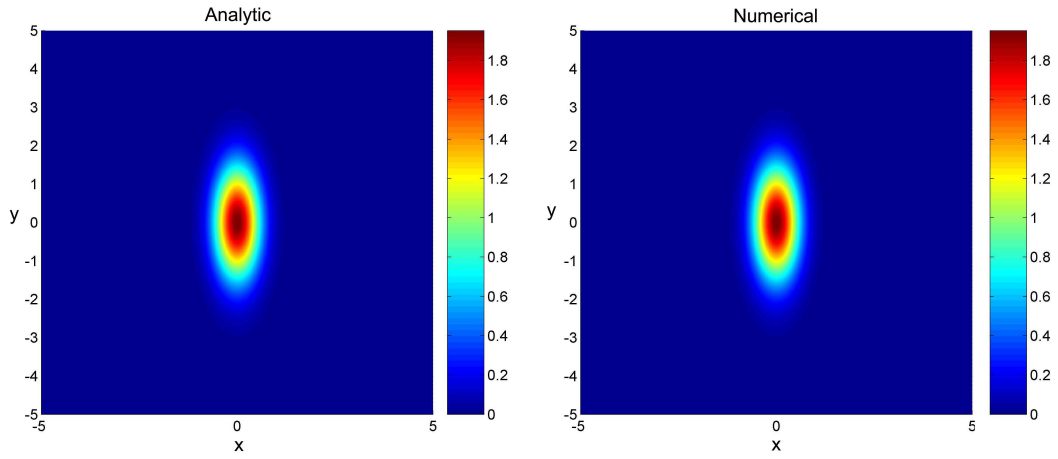

Figure 6: Magnitude comparison between the analytic expression of a fractional transform of a Gaussian function and that computed with the described algorithm.

which can be straightforwardly identified as a chirped Gaussian function. Notice that although the input function is purely real, its transform is complex and contains an oscillatory phase that linearly increases in frequency. This particular characteristic of chirped functions makes them very difficult to represent numerically and provides an ideal benchmark to evaluate the method limitations.

A comparison between the magnitude of the analytic fractional Fourier transform of the Gaussian function and that of the numerically retrieved is shown in figure 6. Figure 7 depicts an equivalent comparison between the phase of these functions. An excellent qualitative and quantitative agreement is achieved in these comparisons, notice that the phase of the numerically computed function is accurately retrieved within an horizontal range, beyond this position the oscillating frequency of the signal is larger than the sampling Nyquist frequency and cannot be properly modeled.

The amplitude and phase comparison between the analytical expression and that obtained from the described algorithm gave excellent results. The different implemented transformation orders for the x and y directions ensure that proper computation of the expression

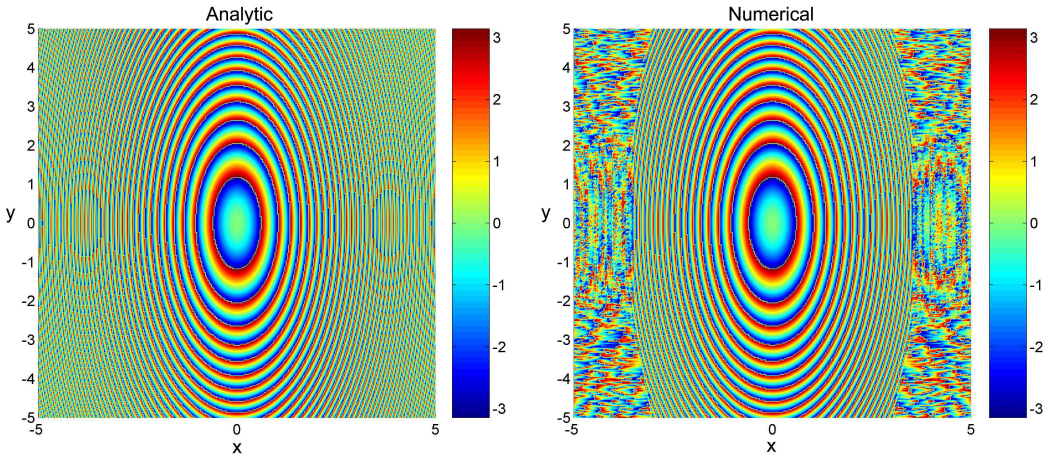

Figure 7: Phase comparison between the analytic expression of a fractional transform of a Gaussian function and that computed with the described algorithm.

appearing in equation (18) is achieved through separate computation of the horizontal and vertical integrals.

An important feature that any algorithm that approximates the fractional Fourier transform must fulfill, is the original function retrieval after an inverse transformation. This is particularly important in image processing, since accurate inverse transformation is crucial to obtain the desired result after filtering. We performed numerical tests on a sample image to ensure a good quality of the retrieved data.

Figure 8 shows the results of image retrieval. Zero padding is a widely used technique when processing in the frequency domain, the discrete Fourier transform assumes that the input signal is periodical and may incur in serious errors if not properly padded with zeros. With this technique, the original image is expanded with zeros around it, The number of zeros must be equal to the total number of pixels in the image. In the case of the fractional Fourier transform we found that zero padding is crucial to avoid aliasing effects in the retrieved image, this occurs because in the computation we need to perform a convolution and we implement it as a multiplication in the Fourier domain.

Figure 8: Results of image retrieval with the fractional Fourier transform.

Notice in figure 8, that the aliasing effects are practically eliminated in the image were the image transform and retrieval were computed using zero padding. Still in this image, there are some artifacts that can be observed near the image borders upon close examination, this are completely removed by applying a trim operation before performing the inverse transform. This trimming consists on forcing all points of the original function padding to have a value of zero. Excellent results are obtained for the image retrieval with zero padding and trimmed where the artifacts are completely removed.

5 Optimal Image Restoration

In this section the concept of filtering in fractional Fourier domains is applied to the problem of estimating images with space varying statistics in the presence of atypical degradation and noise. An optimal filter for each set of fractional domains is estimated with a statistical

Figure 8: Results of image retrieval with the fractional Fourier transform.

Notice in figure 8, that the aliasing effects are practically eliminated in the image were the image transform and retrieval were computed using zero padding. Still in this image, there are some artifacts that can be observed near the image borders upon close examination, this are completely removed by applying a trim operation before performing the inverse transform. This trimming consists on forcing all points of the original function padding to have a value of zero. Excellent results are obtained for the image retrieval with zero padding and trimmed where the artifacts are completely removed.

5 Optimal Image Restoration

In this section the concept of filtering in fractional Fourier domains is applied to the problem of estimating images with space varying statistics in the presence of atypical degradation and noise. An optimal filter for each set of fractional domains is estimated with a statistical

Figure 9: Image corrupted by additive chirp noise signal.

approach to restore an image degraded by the general model

$$o(x,y) = \int\int h(x,y;x',y')f(x',y')dx'dy' + \eta(x,y), \tag{24}$$

where $o(x,y)$ is the observed image, $f(x,y)$ is the image we wish to recover, $h(x,y;x',y')$ is the kernel of the degradation model and $\eta(x,y)$ is an additive noise term, usually originated during digital capture of data.

5.1 Degradation Models

To illustrate the performance of fractional Fourier domain filtering we apply two different degradation mechanisms to sample images and then attempt to recover the original signal.

The original image shown in figure 9 was corrupted by means of two chirp waveforms. Although the ordinary Fourier transform is best suited to process images with periodic noise of a single frequency, this particular noise has a spatially varying frequency and orientation. Thus, best results are expected by processing in fractional domains.

A second degradation model that we implemented is a spatially dependant motion blur

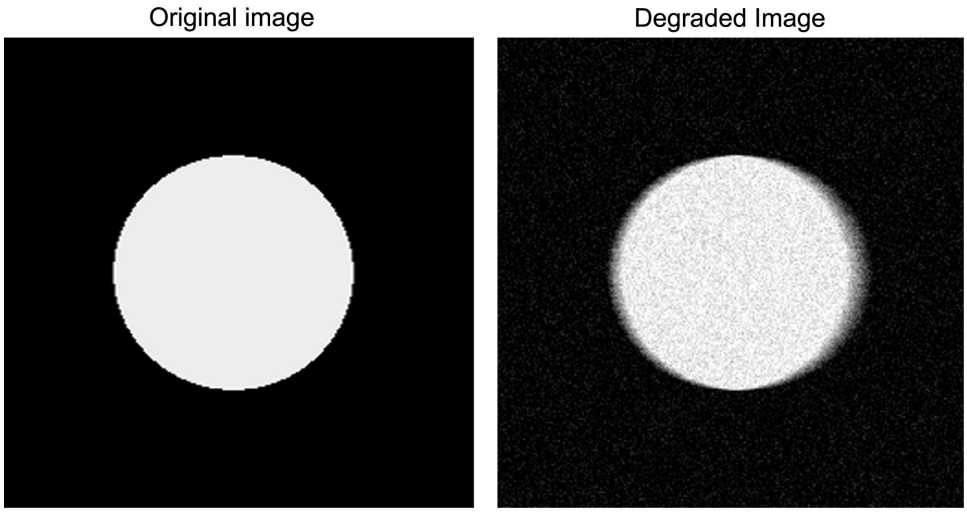

Figure 10: Image corrupted by spatially dependant motion blur and additive Gaussian noise.

that was applied to the original image. For this type of degradation the kernel in equation (24) is given by

$$h(x, y; x', y') = \frac{1}{\alpha x + \alpha_0} \operatorname{rect}\left(\frac{x - x'}{\alpha x + \alpha_0} - \frac{1}{2}\right) \delta(y - y'), \tag{25}$$

where α and α_0 are the parameters of the distortion model and correspond to the growing rate and initial window size of the rectangular function. This particular degradation creates a motion effect on the original image, where the magnitude of the motion degradation depends on the pixel position.

For the particular case of $\alpha = 0$ in equation (25), the motion blur becomes independent of spatial coordinates and the optimum domain is that of the conventional Fourier transform. In any other case the frequency domain will not be suited to represent the degradation as a multiplication and filtering in fractional domains will give better results than the ordinary Fourier transform.

We also added white Gaussian noise of signal to noise ratio $SNR = 4$ and zero mean to the images that were degraded with the spatially dependant blur. This noise is typically introduced by digital capture. Figure 10 and 11 show this degradation implemented on two

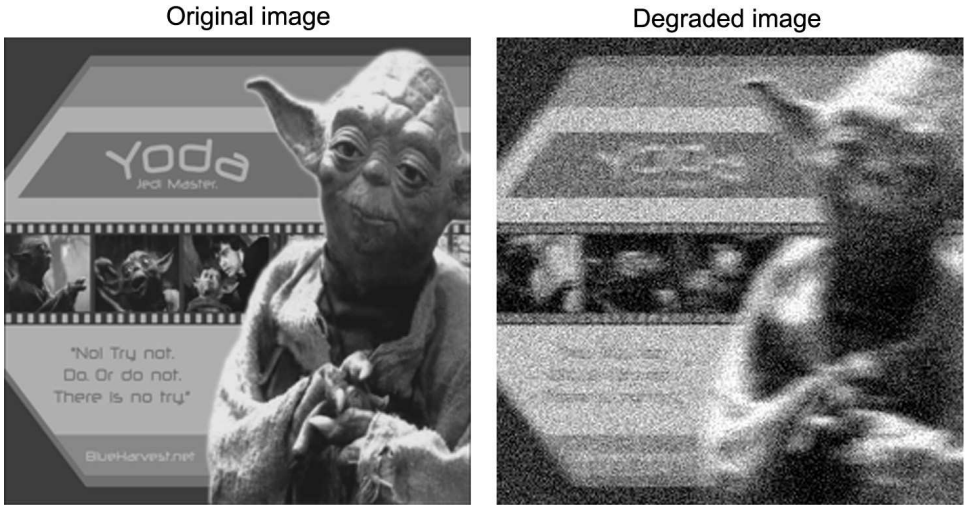

Figure 11: Image corrupted by spatially dependant motion blur and additive Gaussian noise.

different images, notice how the blur degradation is stronger on the right side of the sample images.

5.2 Optimum Filter Implementation

The optimum image restoration technique for this kind of degradation is well known but has a high computational cost. We choose to implement a multiplicative filter in the fractional domain in order to speed up the filter execution time. Optimal noise signal separation can sometimes be better achieved in a fractional domain than in the ordinary frequency domain [31].

It was shown by Alper [3] that the optimal fractional-domain multiplicative filter is described by

$$m_{opt}(x,y) = \frac{R_{f_{a_x,a_y},o_{a_x,a_y}}}{R_{o_{a_x,a_y},o_{a_x,a_y}}} = \frac{f_{a_x,a_y}(x,y) \otimes o^*_{a_x,a_y}(x,y)}{o_{a_x,a_y}(x,y) \otimes o^*_{a_x,a_y}(x,y)}, \tag{26}$$

where $(*)$ stands for complex conjugation. The operator $R_{f_{a_x,a_y},o_{a_x,a_y}}$ represents the cross-correlation of the original and the observed images [32].

This operation can be greatly accelerated by taking advantage of the index additivity

properties of the fractional Fourier transform, shown in equation (7a). Therefore, the filter can be re-written as follows:

$$m_{opt}(x,y) = \frac{\mathcal{F}^{-1,-1}\left\{f_{a_x+1,a_y+1}(x,y)o^*_{a_x+1,a_y+1}(x,y)\right\}}{\mathcal{F}^{-1,-1}\left\{o_{a_x+1,a_y+1}(x,y)o^*_{a_x+1,a_y+1}(x,y)\right\}}, \tag{27}$$

where $o(x,y)$ is the degraded image and $f(x,y)$ is the original image. Notice that with equation (27) we can compute the optimum filter with an element by element multiplication instead of a convolution, with this simple modification the numerical requirements are greatly relaxed.

The estimate of the original function can be straightforwardly obtained using the filter defined in equation (27), namely

$$\hat{f}(x,y) = \mathcal{F}^{-a_x,-a_y}\left\{o_{a_x,a_y}(x,y)m_{opt}(x,y)\right\}. \tag{28}$$

The main advantage of the optimal filter approach, equation (28), over the Weiner filter shown in expression (4) is that the former can account for spatially-dependant degradation and the filter may be applied in any combination of horizontal and vertical fractional domains. Of course this suggests that we should search for the fractional domains where optimal results are obtained.

5.3 Optimum Order Search

Even with the optimum filter, as shown in equation (27), the processed image can have a significant amount of noise if processed in an arbitrary fractional domain. Since the error of the estimated function is modified by the chosen fractional domain there should exist a combination of a_x and a_y domains that minimizes the error between the original and the estimated images.

We propose an exhaustive search of the minimum error filtering domain, by performing the optimum filtering approach in evenly spaced samples for the a_x and a_y transformation orders in the [-1,1] range.

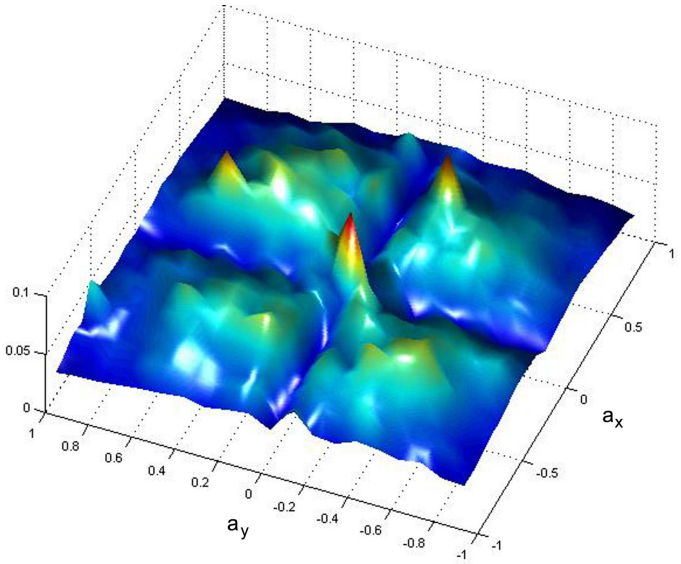

Figure 12: Error topology for the fractional domain search for optimum filtering

With this methodology, the optimum order search is achieved by finding the best-performing filtering domain within this grid. We used the mean square error to test the filter performance in the fractional domains. This error measurement is defined for a digital image as

$$\text{MSE} = \sum_x \sum_y |o(x,y) - f(x,y)|^2, \tag{29}$$

The topology of the mean square error computation, for different a_x and a_y filtering domains is shown in figure 12. The optimum filtering domain lies wherever the deepest valley in this topology is.

5.4 Results

The optimum fractional domain for multiplicative filtering was searched for each of the degraded images shown in section 5.1. The optimal restoration filter appearing in equation (27) was applied to the degraded images in each fractional domain of the domain space, seeking the highest reduction of mean square error.

Figure 13: Results of optimum filtering in the Fourier domain and in the optimal fractional domain.

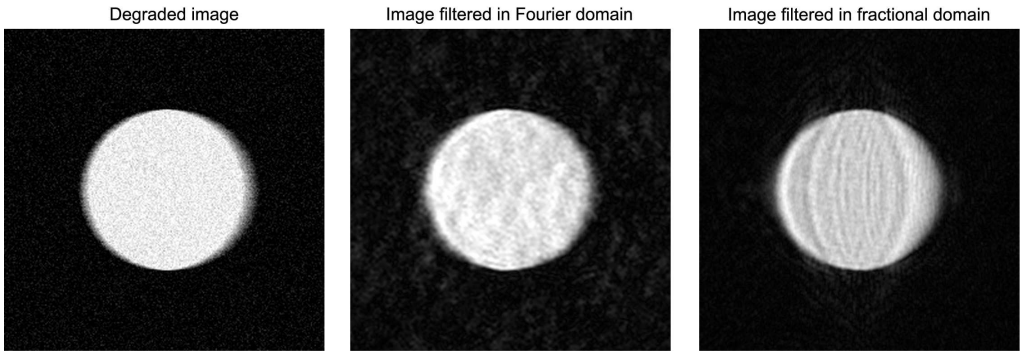

Figure 14: Results of optimum filtering in the Fourier domain and in the optimal fractional domain.

The image that was corrupted by a chirp additive noise and the obtained restoration results are shown in figure 13. The original error of the corrupted image is MSE $= 5 \times 10^{-3}$, this quantity was reduced to MSE $= 3.4 \times 10^{-3}$ in the $a_x = 1.0$, $a_y = 0.9$ fractional domain. The result obtained with the conventional Fourier transform is also shown in figure 13 the retrieved error with this transformation was MSE $= 4.6 \times 10^{-3}$. A reduction of the computed mean square error of 32% was achieved by searching for the optimal fractional domain, 24% more than filtering in the ordinary frequency domain.

Two additional degraded images were processed with the described methodology, these

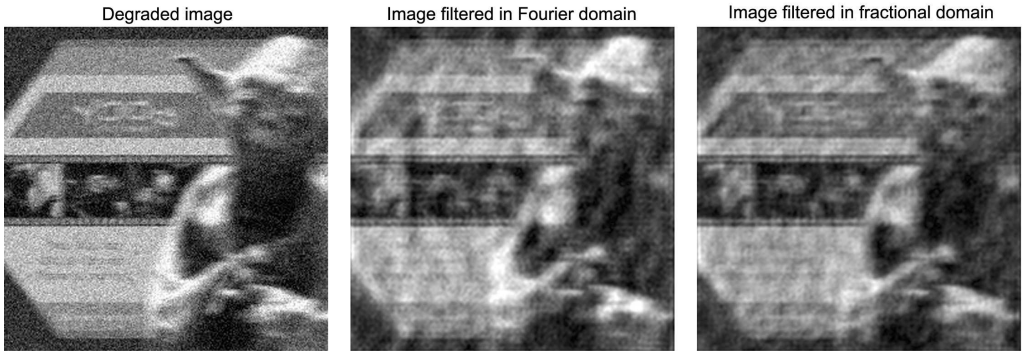

Figure 15: Results of optimum filtering in the Fourier domain and in the optimal fractional domain.

images were corrupted using a spatially dependant blur with additive Gaussian noise. The degraded image of a circle and the results obtained are shown in figure 14. The original error of the corrupted image is MSE $= 13 \times 10^{-3}$, this quantity was reduced to MSE $= 9 \times 10^{-3}$ in the $a_x = -0.5$, $a_y = 0.7$ fractional domain. The result obtained with the conventional Fourier transform is also shown in figure 14, the retrieved error with this transformation was MSE $= 4.6 \times 10^{-3}$. A reduction of the computed mean square error of 31% was achieved by searching for the optimal fractional domain, 16% more than filtering in the ordinary frequency domain.

The results obtained for the second sample image corrupted with the spatially dependant blur and additive Gaussian noise are shown in figure 15. The mean square error of the corrupted image is MSE $= 26 \times 10^{-3}$, this quantity was reduced to MSE $= 17 \times 10^{-3}$ in the $a_x = -0.9$, $a_y = 1.0$ fractional domain. The result obtained with the conventional Fourier transform is also shown in figure 15, the retrieved error with this transformation was MSE $= 17 \times 10^{-3}$. A reduction of the computed mean square error of 35% was achieved by searching for the optimal fractional domain, 12% more than filtering in the ordinary frequency domain.

Although the results shown in figure 15 did not improve visually, the restored images are

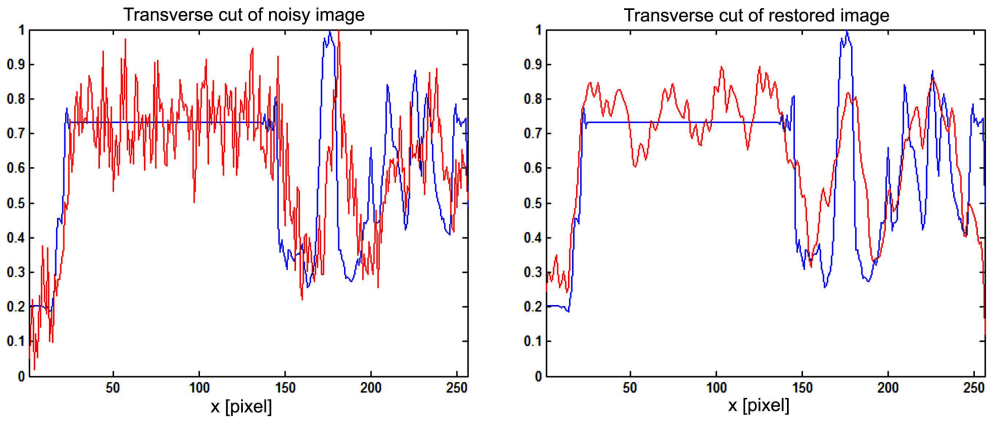

Figure 16: Comparison of image values along an horizontal line for the degraded and restored image.

better suited for automated processing or recognition. The gray level values of the degraded image and the restored image are shown in figure 16 for better visualization of the results. The red lines depict the degraded and restored values, while the blue lines depict the original uncorrupted image. Notice how upon filtering in the optimal fractional domain the noise amplitude and frequency are greatly reduced, and the gray level abrupt changes, that seemed to be shifted due to the motion blur, are closer to their original value in the restored image. Although this effects are hard to notice from direct inspection of the retrieved image, their effects will be noticeable when trying to automatically retrieve useful information from it.

6 Conclusion

Although the results showed in section 5.4 demonstrate a significant reduction in the image mean square error, they do not reflect an important improvement in visual quality. This indicates that the implemented error parameter is not a good indicator for visual quality. Nevertheless, we found optimal filter in fractional domains is a good alternative to restore image parameters before segmentation or image recognition, due to the high percentage of

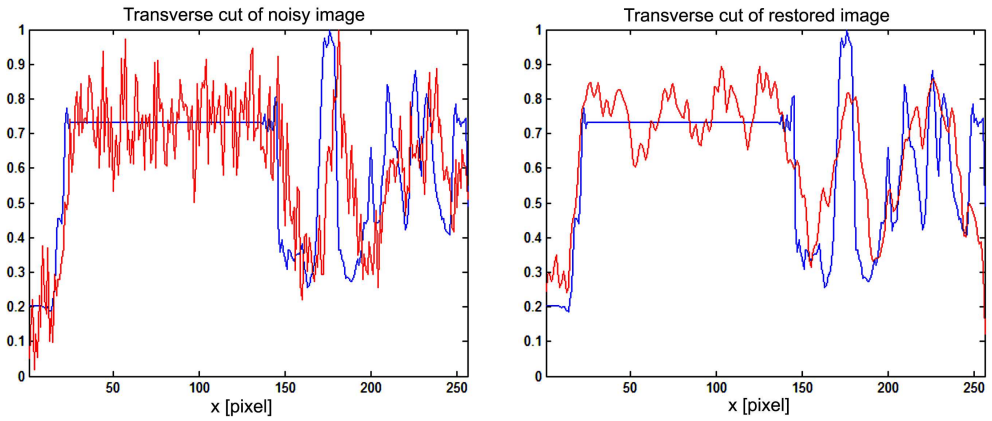

Figure 16: Comparison of image values along an horizontal line for the degraded and restored image.

better suited for automated processing or recognition. The gray level values of the degraded image and the restored image are shown in figure 16 for better visualization of the results. The red lines depict the degraded and restored values, while the blue lines depict the original uncorrupted image. Notice how upon filtering in the optimal fractional domain the noise amplitude and frequency are greatly reduced, and the gray level abrupt changes, that seemed to be shifted due to the motion blur, are closer to their original value in the restored image. Although this effects are hard to notice from direct inspection of the retrieved image, their effects will be noticeable when trying to automatically retrieve useful information from it.

6 Conclusion

Although the results showed in section 5.4 demonstrate a significant reduction in the image mean square error, they do not reflect an important improvement in visual quality. This indicates that the implemented error parameter is not a good indicator for visual quality. Nevertheless, we found optimal filter in fractional domains is a good alternative to restore image parameters before segmentation or image recognition, due to the high percentage of

error decrease that was observed. We are confident that if an image quality descriptor was available, the same algorithm presented in this work would render better visual results.

Additionally, our work further demonstrates that traditional Fourier filtering is not always the best technique to restore a heavily degraded image, since it can be significantly outperformed by fractional Fourier filtering. In the implemented examples, a greater reduction in the image error was achieved by fractional domain filtering. We found that filtering in fractional domains outperforms traditional frequency filtering, specially when removing additive chirp noise, as shown in figure 13, where by filtering in fractional domains noise was reduced 24% more than with traditional Fourier filtering.

Filtering in fractional domain renders better results because the noise signal can be optimally separated from the image signal, the fractional domains provide additional parameters that can be optimized to achieve superior restoration results.

A Computer Code

All implemented computer codes are included in this section for completeness and future reference.

One-Dimensional Fractional Fourier Transform

The algorithm realization of the fractional transformation in one dimension is included here. This code receives a matrix of arbitrary number of elements and performs the transformation on the columns of the array.

```
% F = frft(f, a, deltay) returns the ath order fractional fourier
% transform of the one-dimensional functions in the columns of f, using the
% convolution algorithm.
%
% f :: The columns of f represents one-dimensional samples of a signal,
% which should be sampled in an even amount of points. Several conditions
% should be met for f:
%
%               1. The amount of samples of f should be even;
%                  The best performance is accomplished when
%                  the amount of samples if a power of 2.
%               2. f should be pre-processed with a reasonable
%                  amount of zero-padding to avoid aliasing.
%
% a :: any positive real number or 0 is admitted for a, which is the order for
% the fractional transform to perform.
%
% deltay :: Is the sampling period for the dependent variable. This is
% needed to sample an analytic form of the convolution-chirp.

function F = frft(f,a,deltay); pack
%Defines the lower limit for the fractional fourier transform
h = 0.366;

%Calculate the number of sample points and the number of received columns
[Ny,Nx] = size(f);

%If the number of sample points is odd, return an error message.
```

```
if( mod(Ny, 2) )
    error( sprintf('Sample points should be even, %d points received', Ny ) );
    F = []
    return
end;

%FFT shift functions.
chess = (-1).^[0:Ny-1]'; chessmat = repmat(chess,1,Nx);

%Wrap for a, fractional fourier transforms are periodic with a period of 4.
%When a = 5 it is equivalent to perform a fractional transform of order 1.
a = mod(a,4);

%Validations for a to avoid unnecessary computing, taking advantage of some
%properties
switch a
    case 0
        %Zero transform of a function is itself.
        F = f;
        return
    case 1
        %First order fractional fourier transform is equal to the fourier
        %transform.
        F = (1/sqrt(Ny))*fft(f.*chessmat).*chessmat;
        return
    case 2
        %Second order transform is the inverted version of the function.
        f = flipud(f);
        F = [ zeros(1, Nx); f(1:Ny-1, :) ];
        return
    case 3
        %Third order is the inverted direct fourier transform.
        f = flipud(f);
        f = [ zeros(1, Nx); f(1:Ny-1, :) ];
        F = (1/sqrt(Ny))*fft(f.*chessmat).*chessmat;
        return
    otherwise
        %Otherwise, further processing is required.
        if a > 2
            f = flipud(f);
            f = [ zeros(1, Nx); f(1:Ny-1, :) ];
            a = a-2;
```

```
        end
end

%Backup of the validated order for final processing...
a2 = a;

%Another wrapping to fit in the range (h, 1+h) as described in <reference>
if a > 1+h
    a = a-1;
elseif a < h
    a = a+1;
end

%%% Begin of actual fractional fourier transforming.

alfa = a*pi/2;
% Reconstruction of the sampling vector for Bprime
y = repmat( deltay*( [1:Ny]' - (Ny/2 + 1) ), [1 Nx] );

%Sampling of an analytic version of the tranformation of the chirp.
Bprime = exp(-i*pi*tan(alfa/2)*y.^2 ); fprime = f.*Bprime;

%Calculation of Fprime
Fprime = fft(fprime.*chessmat);

%Reconstruction of a sampling vector for the fourier domain.
v = y/(Ny*deltay^2);

Balfa = sqrt(i/csc(alfa))*exp( -i*pi*v.^2/csc(alfa) );

% Convolution
convol = ifft(Fprime.*Balfa).*chessmat;

%Inclution of scaling parameters and other required calculations.
F = sqrt(1-i*cot(alfa))*Bprime.*convol;

%Final validation
if a2 > 1+h
    F = (1/sqrt(Ny))*fft(F.*chessmat).*chessmat;
elseif a2 < h
    F = sqrt(Ny)*ifft(F.*chessmat).*chessmat;
end
```

Two-Dimensional Fractional Fourier Transform

With this code we compute two-dimensional fractional Fourier transform using the one-dimensional code previously presented.

```
function F = frft2(f, ay, ax, deltay, deltax)

F = frft(f, ay, deltay); F = F.'; F = frft(F, ax, deltax);

F = F.';
```

Optimal Filtering and Fractional Order Search

We applied this computer code to filter the degraded image in each fractional domain. Search for the optimal fractional domain is achieved by computing and applying the optimal filter in a predetermined grid of the domain space.

```
clear all; close all;
% Search parameters, (2/step)^2 grid points
step = 0.2;

% Read images from file
blurname ='test3j.bmp'; originalname ='test3o.bmp';%
f = im2double(imread(blurname));%
fo = im2double(imread(originalname));
% Image red channel is chosen for processing
f=f(:,:,1); fo=fo(:,:,1);
% Image size is obtained and the search grid is defined
[jlim, ilim] = size(f); [X,Y]=meshgrid([1:ilim],[j:jlim]);

% Zero padding
f2  = zeros(2*jlim, 2*ilim); fo2 = zeros(2*jlim, 2*ilim);
f2(ilim/2+1:3*ilim/2, jlim/2+1:3*jlim/2)=f;%
fo2(ilim/2+1:3*ilim/2, jlim/2+1:3*jlim/2)=fo;
%The mask for image trimming is defined
mask = zeros(2*jlim, 2*ilim); mask(ilim/2+1:3*ilim/2,
jlim/2+1:3*jlim/2) = 1;

% Computation of deltax and deltay
[Nx,Ny]=size(f2); deltax = 1/sqrt(Nx); deltay = 1/sqrt(Ny);

% Definition of best estimate variables
MSE  = zeros(2/step); TIME = zeros(2/step); Mbest = 1000000000;
secs = 0;

% Monitor variables
h = waitbar(0, '0 porciento'); k = 0;

% Search begins
for ii = 1:(2/step),
    for jj = 1:(2/step),
        tic;
```

```
    ax = 3 + ii*step;
    ay = 3 + jj*step;

    F =  frft2(f2,ay,ax,deltay,deltax).*mask;
    Fo = frft2(fo2,ay,ax,deltay,deltax).*mask;

    F = frft2(F,1,1,deltay,deltax);
    Fo = frft2(Fo,1,1,deltay,deltax);
% Computation of optimum filter
    Num = frft2((Fo).*(F),3,3,deltay,deltax);
    Den = frft2((F).*(F),3,3,deltay,deltax);
    Mopt = (Num./Den);
    clear Num
    clear Den

    Moptpad = zeros(Ny, Nx);
    Moptpad(ilim/2+1:3*ilim/2, jlim/2+1:3*jlim/2) =...
                        (1/4)*(...
                                Mopt(1:2:Ny,1:2:Nx)+...
                                Mopt(2:2:Ny,1:2:Nx)+...
                                Mopt(1:2:Ny,2:2:Nx)+...
                                Mopt(2:2:Ny,2:2:Nx)...
                               );

    Ffil = frft2(F,3,3,deltay,deltax);
% Filter application
    est = frft2(Ffil.*Moptpad.*mask,4-ay,4-ax,deltay,deltax);

    MSE(jj,ii) = mean(mean(abs(abs( ...
    est(ilim/2+1:3*ilim/2, jlim/2+1:3*jlim/2) )...
    /max(max(abs(est(ilim/2+1:3*ilim/2, ...
    jlim/2+1:3*jlim/2))))  - fo).^2));

    if( Mbest > MSE(jj,ii) )
        Mbest = MSE(jj,ii);
        axbest = ax;
        aybest = ay;
        best = abs(est(ilim/2+1:3*ilim/2, jlim/2+1:3*jlim/2))...
        /max(max(abs(est(ilim/2+1:3*ilim/2, jlim/2+1:3*jlim/2))));
        tbest = TIME(jj,ii);
    end
    TIME(jj,ii) = toc
```

```
        secs = secs + TIME(jj,ii);
        k = k+1;
        waitbar( k/((2/step)^2), h, ...
        sprintf('%3.2f porciento %f segundos', k/((2/step)^2), secs) );
    end
end

% Search variables are saved to file
savename = sprintf('%s_%dx%d', blurname, 2/step,2/step);
save(savename);

% Display of results
figure; imshow( best );
title( sprintf('Optimo: MSE:%1.5f ax:%1.2f ay:%1.2f tiempo:%f segundos',...
 Mbest, axbest, aybest, tbest) );
figure; imshow( f ); title( 'Imagen con blur' ); figure; imshow(
fo ); title( 'Imagen original' );

axisax = [3+step:step:5]; figure; surf(axisax,axisax',MSE),
shading interp, lighting phong, view(2)

'MSE original' mean(mean(abs(f  - fo).^2))
```

B Glossary of Technical Terms

Sine cardinal: Also called the "sampling function". The sine cardinal is a function that arises frequently in signal processing and the theory of Fourier transforms.

Chirp: A signal that linearly ramps up (or down) in frequency.

Convolution: A convolution is an integral that expresses the amount of overlap of one function as it is shifted over another function.

Digitized: A term for information that has been converted into binary digits for computer processing.

Dirac delta: The delta function is a generalized function that can be defined as the limit of a class of delta sequences. The delta function is sometimes called the "impulse function" for its schematization as an infinitely narrow function with unitary area.

Fraunhoffer region: Also called the far-field region. When you get far enough from an electromagnetic source so that its radiated field wave can be considered planar.

Fresnel region: Also called the near-field region. When you are close enough to an electromagnetic source so that its radiated field must be considered mathematically spherical rather than planar.

Gaussian: The normal or Gaussian distribution is a continuous symmetric distribution that follows the familiar bell-shaped curve. The distribution is uniquely determined by its mean and variance.

Nyquist frequency: The Nyquist frequency, also called the Nyquist limit, is the highest frequency that can be coded at a given sampling rate in order to be able to fully reconstruct the signal.

Radon transform: The Radon transform is an integral transform whose inverse is used to reconstruct images from medical computer tomography scans.

Uncorrelated: In probability theory and statistics, to call two real-valued random variables uncorrelated means that their correlation is zero, or, equivalently, their covariance is zero.

Watermark: Bits altered within an image to create a pattern that indicates proof of ownership. Unauthorized use of a watermarked image can then be traced.

Wigner distribution: Time-frequency representation of a signal.

C Acknowledgements

We acknowledge the advice and support received from professor Carlos Hinojosa during the development of this project.

References

[1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. New Jersey: Prentice Hall, 2002.

[2] M. K. Ozkan, A. T. Erdem, M. I. Sezan and A. M. Tekalp, "Efficient multiframe Wiener restoration of blurred and noisy image sequences," *IEEE Transactions on Image Processing* [online] 1 (4), pp. 453–476, 1992. IEEE Xplore, http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 23 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

Radon transform: The Radon transform is an integral transform whose inverse is used to reconstruct images from medical computer tomography scans.

Uncorrelated: In probability theory and statistics, to call two real-valued random variables uncorrelated means that their correlation is zero, or, equivalently, their covariance is zero.

Watermark: Bits altered within an image to create a pattern that indicates proof of ownership. Unauthorized use of a watermarked image can then be traced.

Wigner distribution: Time-frequency representation of a signal.

C Acknowledgements

We acknowledge the advice and support received from professor Carlos Hinojosa during the development of this project.

References

[1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. New Jersey: Prentice Hall, 2002.

[2] M. K. Ozkan, A. T. Erdem, M. I. Sezan and A. M. Tekalp, "Efficient multiframe Wiener restoration of blurred and noisy image sequences," *IEEE Transactions on Image Processing* [online] 1 (4), pp. 453–476, 1992. IEEE Xplore, http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 23 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

Radon transform: The Radon transform is an integral transform whose inverse is used to reconstruct images from medical computer tomography scans.

Uncorrelated: In probability theory and statistics, to call two real-valued random variables uncorrelated means that their correlation is zero, or, equivalently, their covariance is zero.

Watermark: Bits altered within an image to create a pattern that indicates proof of ownership. Unauthorized use of a watermarked image can then be traced.

Wigner distribution: Time-frequency representation of a signal.

C Acknowledgements

We acknowledge the advice and support received from professor Carlos Hinojosa during the development of this project.

References

[1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. New Jersey: Prentice Hall, 2002.

[2] M. K. Ozkan, A. T. Erdem, M. I. Sezan and A. M. Tekalp, "Efficient multiframe Wiener restoration of blurred and noisy image sequences," *IEEE Transactions on Image Processing* [online] 1 (4), pp. 453–476, 1992. IEEE Xplore, http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 23 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[3] M. A. Kutay and H. M. Ozaktas, "Optimal image restoration with the fractional Fourier transform," *Journal of the Optical Society of America A* [online] 15 (4), pp. 825–833, 1998. Optics Infobase http://www.opticsinfobase.org (Accessed: 2 April 2005).

[4] A. D. Poularikas, *The Transforms and Applications Handbook*, Boca Raton: CRC Press LLC, 2000. MATHnetBASE [online] http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=mty.mathnetbase (Accessed: 22 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[5] E. Hecht, *Optics*, 4th ed. San Francisco: Adison Wesley, 2002.

[6] J. Rosenblatt and S. Bell, *Mathematical Analysis for Modeling*, Boca Raton: CRC Press LLC, 1999. MATHnetBASE [online] http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=mty.mathnetbase (Accessed: 14 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[7] L. Lucchese and G. M. Cortelazzo, "Motion analysis and displacement estimation in the frecuency domain," in *Digital Image Sequence Processing, Compression, and Analysis*, T. R. Reed, Boca Raton: CRC Press LLC, 2005. Engineering Electronic Library [online] http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=mty.engnetbase (Accessed: 21 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[8] A. D. Hillery and R. T. Chin, "Iterative Wiener filters for image restoration," *IEEE Transactions on Signal Processing* [online] 39 (8), pp. 1892–1899, 1991. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 20 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[9] M. A. King, P. W. Doherty, R. B. Schwinger and B. C. Penney, "A Wiener filter for nuclear medicine images," *Medical Physics* [online] 10 (6), pp. 876-880, 1983. Medical Physics Online http://scitation.aip.org (Accessed: 20 April 2005).

[10] A. I. Zayed, "On the relationship between the Fourier and fractional Fourier transforms," *IEEE Signal Processing Letters* [online] 3 (12), pp. 310–311, 1996. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 22 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[11] I. Djurović, S. Stanković and I. Pitas, "Digital watermarking in the fractional Fourier transformation domain," *Journal of Network and Computer Applications* [online] 24, pp. 167–173, 2001. Google Scholar http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=google_scholar (Accessed: 22 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[12] I. S. Yetik and A. Nehorai, "Beamforming using the fractional Fourier transform," *IEEE Transactions on Signal Processing* [online] 51 (6), pp. 1663–1668, 2003. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 22 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[13] H. B. Sun, G. S. Liu, H. Gu and W. M. Su, "Application of the fractional Fourier transform to moving target detection in airborne SAR," *IEEE Transactions on Aerospace and Electronic Systems* [online] 38 (4), pp. 1416–1424, 2002. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 20 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[14] T. Alieva, M. J. Bastiaans and L. Stankovic, "Signal reconstruction from two close fractional Fourier power spectra," *IEEE Transactions on Signal Processing* [online] 51 (1), pp. 112–123, 2003. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 22 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[15] M. F. Erden, M. A. Kutay and H. M. Ozaktas, "Repeated filtering in consecutive fractional Fourier domains and its application to signal restoration," *IEEE Trans-*

actions on Signal Processing [online] 47 (5), pp. 1458–1462, 1999. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 22 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[16] S. C. Pei, M. H. Yeh and T. L. Luo, "Fractional Fourier series expansion for finite signals and dual extension to discrete-time fractional Fourier transform," *IEEE Transactions on Signal Processing* [online] 47 (10), pp. 2883–2888, 1999. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 22 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[17] M. Martone, "A multicarrier system based on the fractional Fourier transform for time-frequency-selective chanels," *IEEE Transactions on Communications* [online] 49 (6), pp. 1011–1020, 2001. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 22 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[18] T. Alieva and M. J. Bastiaans, "Wigner distribution and fractional Fourier transform for two-dimensional symmetric optical beams," *Journal of the Optical Society of America A* [online] 17 (12), pp. 2319–2323, 2000. Google Scholar http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=google_scholar (Accessed: 3 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[19] A. W. Lohmann, "Image rotation, Wigner rotation, and the fractional Fourier transform," *Journal of the Optical Society of America A* [online] 10 (10), pp. 2181–2186, 1993. Optics Infobase, http://www.opticsinfobase.org (Accessed: 2 April 2005).

[20] H. M. Ozaktas, Z. Zalevsky, M. A. Kutay, *The Fractional Fourier Transform: With Applications in Optics and Signal Processing*, New Jersey: John Wiley & Sons, 2001.

[21] J. W. Goodman, *Introduction to Fourier Optics*, New York: McGraw-Hill, 1988.

[22] C. López, "Propagation of rotating optical fields," M.S. thesis, Instituto Tecnológico y de Estudios Superiores de Monterrey, Monterrey, Mexico, 2003. Documentos Tec [online] http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=doctec (Accessed: 22 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[23] H. M. Ozaktas, "Fourier transforms of fractional order and their optical interpretation," *Optics Communications* [online] 101 (3), pp. 163–169, 1993. Science Direct www.sciencedirect.com (Accessed: 22 April 2005).

[24] J. T. Manassah, *Elementary Mathematical and Computational Tools for Electrical and Computer Engineers Using Matlab®*, Boca Raton: CRC Press LLC, 2001. Engineering Electronic Library [online] http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=mty.engnetbase (Accessed: 11 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[25] H. M. Ozaktas, O. Arikan and M. A. Kutay, "Digital computation of the fractional Fourier transform," *IEEE Transactions on Signal Processing*, [online] 44 (9), pp. 2141–2150, 1996. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 20 March 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[26] M. H. Yeh and S. C. Pei, "A method for the discrete fractional Fourier transform computation," *IEEE Transactions on Signal Processing* [online] 51 (3), pp. 889–891, 2003. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 2 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[27] S. C. Pei, M. H. Yeh and C. C. Tseng, "Discrete fractional Fourier transform based on orthogonal projections," *IEEE Transactions on Signal Processing* [online] 47 (5), 1335–1348, 1999. IEEE Xplore

http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 2 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[28] C. Candan, M. A. Kutay, and H. M. Ozaktas, "Discrete fractional Fourier transform," *IEEE Transactions on Signal Processing* [online] 48 (5), pp. 1329–1337, 2000. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 2 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[29] S. C. Pei and J. J. Ding, "Closed-form discrete fractional and affine Fourier transforms," *IEEE Transactions on Signal Processing* [online] 48 (5), pp. 1338–1353, 2000. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 2 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[30] X. Yang, Q. Tan, X. Wei, Y. Xiang, Y. Yan and G. Jin, "Improved fast fractional-Fourier-transform algorithm," *Journal of the Optical Society of America A*, [online] 21 (9), pp. 1677–1681, 2004. Optics Infobase http://www.opticsinfobase.org (Accessed: 7 April 2005).

[31] M. A. Kutay, H. M. Ozaktas, O. Arikan and L. Onural, "Optimal filtering in fractional Fourier domains," *IEEE Transactions on Signal Processing* [online] 45 (5), pp. 1129–1143, 1997. IEEE Xplore http://biblioteca.itesm.mx/nav/contenidos_salta2.php?col_id=ieeexplore (Accessed: 5 April 2005 through Biblioteca Digital®, Tecnológico de Monterrey).

[32] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 2nd ed. New York: McGraw-Hill, 1984.

Datos generales del trabajo

Profesor: Dr. Carlos Manuel
E-mail: Hinojosa Espinosa

Materia: Procesamiento de imágenes

Categoría: Postgrado

Calificación: 100

Estándar utilizado: IEEE Style (Electrical & CS Engineering Citing & Referencing)
http://www.lib.monash.edu.au/tutorials/citing/styles-common.html

Idioma: Inglés

Autores

Nombre: Manuel Guizar Sicairos

Nombre: David Said Martínez