

Interconversion Between Truncated Cartesian and Polar Expansions of Images

Wooram Park, *Student Member, IEEE*, and Gregory S. Chirikjian, *Member, IEEE*

Abstract—In this paper, we propose an algorithm for lossless conversion of data between Cartesian and polar coordinates, when the data is sampled from a 2-D real-valued function (a mapping: $\mathbb{R}^2 \mapsto \mathbb{R}$) expressed as a particular kind of truncated expansion. We use Laguerre functions and the Fourier basis for the polar coordinate expression. Hermite functions are used for the Cartesian coordinate expression. A finite number of coefficients for the truncated expansion specifies the function in each coordinate system. We derive the relationship between the coefficients for the two coordinate systems. Based on this relationship, we propose an algorithm for lossless conversion between the two coordinate systems. Resampling can be used to evaluate a truncated expansion on the complementary coordinate system without computing a new set of coefficients. The resampled data is used to compute the new set of coefficients to avoid the numerical instability associated with direct conversion of the coefficients. In order to apply our algorithm to discrete image data, we propose a method to optimally fit a truncated expression to a given image. We also quantify the error that this filtering process can produce. Finally the algorithm is applied to solve the polar-Cartesian interpolation problem.

Index Terms—Coordinate conversion, interpolation method, truncated expansions.

I. INTRODUCTION

In 2-D real space (\mathbb{R}^2), polar and Cartesian coordinates are very common and widely used. Many measurement devices such as ultrasound equipment and radar systems gather data in polar coordinates [8], [9]. On the other hand, since conventional display systems have a rectangular grid and corresponding mathematical expressions are more straightforward (e.g., the Euclidean distance between points is easy to compute and products of Fourier expansions follow naturally), Cartesian coordinates are widely used for image display and most mathematical manipulations. Therefore, in order to display data obtained from polar-coordinate-based transducers and deal with them using Cartesian-based mathematical tools, we need a concrete way of converting between polar and Cartesian coordinates. Of course, existing coordinate conversion techniques and interpolation methods have been suggested before, including: [5], [6], and [9]. However, all of these techniques

Manuscript received July 10, 2006; revised March 13, 2007. This work was supported in part by the National Institutes for Health under Grant R01GM075310 and in part by the National Science Foundation-Funded JHU Center for Computer-Integrated Surgical Systems and Technology. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Stanley J. Reeves.

The authors are with the Department of Mechanical Engineering, The Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: wpark7@jhu.edu; gregc@jhu.edu).

Digital Object Identifier 10.1109/TIP.2007.899190

are approximate in nature, and so the conversion from polar to Cartesian and back (or Cartesian to polar and back) results in the destruction of some information. Existing methods are, therefore, not a lossless process, and if one restricts the discussion to Fourier expansions (i.e., Fourier series in Cartesian coordinates and Fourier-Hankel transform in polar coordinates) then there will never be an exact conversion. Even though one can reconstruct a band-limited function to within a desired error bound using an optimal sampling window with Fourier-based analysis techniques [7], it is generally not possible to describe the same function as simultaneously bandlimited in both of these two classical bases.¹ The basis considered in this work is, therefore, special.

Conversion between polar and Cartesian coordinates is given as

$$(x, y) \Leftrightarrow (\rho, \theta) \quad \text{where} \quad x = \rho \cos \theta, \quad y = \rho \sin \theta.$$

Using this relation, we can have two representation of a 2-D real-valued function (mapping: $\mathbb{R}^2 \mapsto \mathbb{R}$) as

$$f^c(x, y) = f^p(\rho, \theta)$$

where the superscripts $(\cdot)^c$ and $(\cdot)^p$ denote the representations in the Cartesian and polar coordinate systems, respectively. If the function is continuous, the conversion is very straightforward and has no ambiguity. However, if the function is defined on a discrete and equally spaced grid, the conversion cannot be defined in the same way, because the two grid systems do not match as shown in Fig. 1. Therefore, when we display the data from a polar grid on a Cartesian grid, we have to perform interpolation computations to obtain data on the Cartesian grid using the data available on the polar grid. A similar problem can occur in tomographic reconstructions [4].

In Section II, we investigate the truncated expansions of a 2-D function both in the Cartesian and polar coordinate systems. In Section III, we show how to compute a set of coefficients that specifies the truncated expansion fit to given discrete data. The error in this filtering process is investigated. In Section IV, we show that the two sets of coefficients have a special relationship, which we use to define the lossless coordinate conversion of a 2-D function. In Section V, we show how our algorithm works with numerical examples. Section VI gives conclusions and discussions.

¹The term “band limited” is used commonly to denote truncated Fourier expansions, but can also be used more generally to describe truncated expansions in any orthogonal basis. Generally, a truncated expansion in one basis will not be truncated in another. In order to avoid any confusion, we use the word “truncated” in the context of Hermite/Laguerre–Fourier expansions, since these are generally not band limited in the Fourier sense.

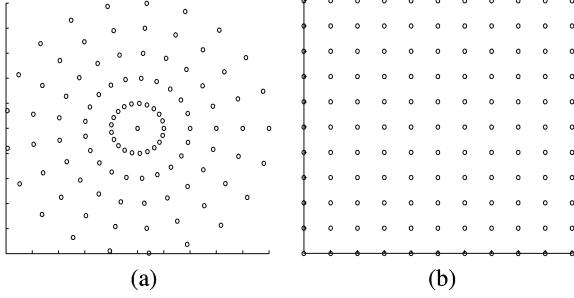


Fig. 1. Equally spaced grids. (a) Grid in polar coordinates. (b) Grid in Cartesian coordinates.

II. TRUNCATED EXPANSIONS IN CARTESIAN AND POLAR COORDINATES

In this section, we review Hermite and Laguerre functions, and how they are used to construct orthonormal expansions of functions over 2-D domains.

A. Truncated Hermite Expansions in Cartesian Coordinates

Hermite polynomials are well known in the literature [12], [13] and are generated by the Rodrigues formula

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} (e^{-x^2}).$$

The *Hermite functions* are defined as

$$h_n(x) = \frac{1}{2^{n/2} \sqrt{n! \sqrt{\pi}}} H_n(x) e^{-\frac{x^2}{2}}. \quad (1)$$

Note that the set $\{h_n(x)\}$ is orthonormal, which means $\int_{-\infty}^{\infty} h_m(x) h_n(x) dx = \delta_{mn}$. The Hermite functions satisfy the recurrence formula

$$\sqrt{n+1} h_{n+1}(x) = \sqrt{2} x h_n(x) - \sqrt{n} h_{n-1}(x).$$

This recurrence formula is very useful to evaluate the Hermite functions of high orders.

The set $\{h_n(x)\}$ for $n = 0, 1, 2, \dots$ forms a basis for the set of square-integrable functions on the line [12]. Due to this fact and their orthonormality, any $f \in L^2(\mathbb{R})$ can be expanded as

$$f(x) = \sum_{n=0}^{\infty} \hat{f}_n h_n(x)$$

where

$$\hat{f}_n = \int_{-\infty}^{\infty} f(x) h_n(x) dx.$$

Approximately, a function can be expanded as a truncated Hermite series

$$f(x) \approx \sum_{n=0}^N \hat{f}_n h_n(x)$$

where N is a large nonnegative integer.

If the domain of a function is 2-D, we can employ the 2-D Hermite series.

$$f^c(x, y) = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \hat{f}_{mn} h_m(x) h_n(y)$$

where Hermite series coefficients are defined as

$$\hat{f}_{mn} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f^c(x, y) h_m(x) h_n(y) dx dy. \quad (2)$$

Also, we can use a truncated expansion of the form

$$f^c(x, y) = \sum_{m=0}^N \sum_{n=0}^N \hat{f}_{mn} h_m(x) h_n(y) \quad (3)$$

where N is a large nonnegative integer. We will use the slightly different form as follows:

$$f^c(x, y) = \sum_{m=0}^N \sum_{n=0}^{N-m} \hat{f}_{mn} h_m(x) h_n(y) \quad (4)$$

or equivalently

$$f^c(x, y) = \sum_{m=0}^N \sum_{n=0}^m \hat{f}_{n,m-n} h_n(x) h_{m-n}(y). \quad (5)$$

The reason why we use (4) or (5) instead of (3) will be explained in Section IV.

It should be noted that finite Hermite expansions have some drawbacks. For example, they do not preserve the truncation limit under shifts, while Fourier expansions do. Furthermore, since the Hermite functions decay exponentially outside of a bounded region, it is important to center the image of interest within this region to obtain the highest resolution for a given truncation. Despite these shortcomings, the Hermite function expansions are superior for Cartesian-polar coordinate conversions for the reasons demonstrated in the rest of the paper.

B. Truncated Expansions in Polar Coordinates Using Laguerre Functions and Fourier Basis

The *associated Laguerre polynomials* are also well known and given by the Rodrigues formula

$$L_n^k(x) = \frac{e^x x^{-k}}{n!} \frac{d^n}{dx^n} (e^{-x} x^{n+k}).$$

The associated Laguerre polynomials are orthogonal over $[0, \infty)$ with respect to the weighting function $x^k e^{-x}$

$$\int_0^{\infty} x^k e^{-x} L_m^k(x) L_n^k(x) dx = \frac{(n+k)!}{n!} \delta_{m,n}.$$

On the other hand, the *Fourier series* expansion of a 1-D function with period 2π is defined as

$$f(x) = \frac{1}{\sqrt{2\pi}} \sum_{k=-\infty}^{+\infty} \hat{f}(k) e^{ikx}$$

where

$$\hat{f}(n) = \frac{1}{\sqrt{2\pi}} \int_0^{2\pi} f(x) e^{-inx} dx.$$

Using the Laguerre polynomials and the Fourier basis, one can define orthonormal basis functions in 2-D polar coordinates as follows [1]:

$$\begin{aligned} \chi_{m,n}(\rho, \theta) &= (-1)^{(m-|n|)/2} \sqrt{\frac{[(m-|n|)/2]!}{\pi [(m+|n|)/2]!}} \rho^{|n|} \\ &\times L_{(m-|n|)/2}^{|n|}(\rho^2) e^{-\rho^2/2} e^{-in\theta} \end{aligned} \quad (6)$$

where $(m - |n|)$ and $(m + |n|)$ are even numbers. If we define $Y_{mn}(r)$ and $Z_n(\theta)$ as

$$Y_{mn}(\rho) = (-1)^{(m-|n|)/2} \sqrt{\frac{2[(m-|n|)/2]!}{[(m+|n|)/2]!}} \rho^{|n|}$$

$$\times L_{(m-|n|)/2}^{|n|}(\rho^2) e^{-\rho^2/2}$$

$$Z_n(\theta) = \frac{1}{\sqrt{2\pi}} e^{-in\theta}$$

then $\chi_{m,n}(\rho, \theta)$ can be decomposed as

$$\chi_{m,n}(\rho, \theta) = Y_{mn}(\rho) Z_n(\theta).$$

A 2-D function in polar coordinates can be expressed as

$$f^p(\rho, \theta) = \sum_{m=0}^{\infty} \sum_{n=-m}^m \tilde{f}_{m,n} \chi_{m,n}^*(\rho, \theta) \quad (7)$$

where the Laguerre–Fourier coefficients are

$$\tilde{f}_{m,n} = \int_0^{2\pi} \int_0^{\infty} f^p(\rho, \theta) \chi_{m,n}(\rho, \theta) \rho d\rho d\theta \quad (8)$$

and χ^* is the complex conjugate of χ . Note that the integer variable n in (7) increases by multiples of 2. A truncated expansion can be written in the form

$$f^p(\rho, \theta) = \sum_{m=0}^M \sum_{n=-m}^m \tilde{f}_{m,n} \chi_{m,n}^*(\rho, \theta) \quad (9)$$

where M is a positive large integer.

One of the major contributions of this paper is to establish the equivalence of (4) and (9), and to find how N and M and \hat{f}_{mn} and $\tilde{f}_{m,n}$ are related.

III. FILTERED IMAGE

When data is obtained from discrete images, we cannot directly apply (2) and (8) to this data to get the truncated expansion. In this section, we propose a way to obtain truncated expansions corresponding to a discrete image. Sampling of these expansions defines a filtering² of the original image. We investigate the difference between the original image and its truncated (filtered) version. Of course, for this to be a filter, the number of coefficients must be less than or equal to the number of values specified on the grid. Otherwise, no information (or noise) would be rejected from the original image, and the excess degrees of freedom would need to be constrained in some way, since they would not be determined by the original image. A filtering method using Hermite expansions in the Cartesian coordinates is presented in the manuscript [14]. A brief summary of these results is presented here without derivation. A detailed filtering method using Laguerre–Fourier expansions in polar coordinates will be presented.

²Since this is not shift invariant, it is not a filter in the sense of Fourier-based image processing. However, we view it as a filter in the sense that it acts like a low-pass filter in the Hermite expansion.

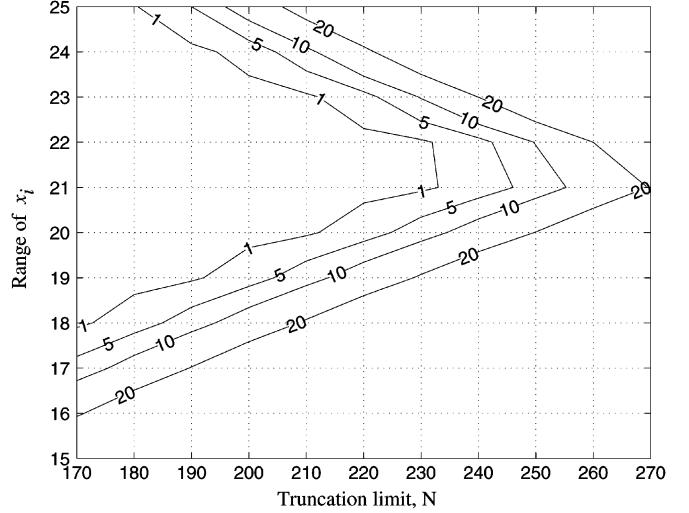


Fig. 2. Contour plot of logarithm of condition number of $H^T H$ depending on truncation limit and range of x_i ($M = 301$).

A. Filtered Images in Cartesian Coordinates

Definition: \bar{F}_{xy} is a given $M \times M$ image matrix. The Hermite-filtered image function $f(x, y)$ is given by

$$f(x, y) = \sum_{m=0}^N \sum_{n=0}^{N-m} \hat{f}_{mn} h_m(x) h_n(y)$$

where $\hat{f}_{m-1,n-1}$ is the (i, j) element of \hat{F} , and \hat{F} is given by

$$\hat{F} = (H^T H)^{-1} H^T \bar{F}_{xy} H (H^T H)^{-1}$$

where $H_{i,j} = h_{j-1}(x_i)$. The Hermite-filtered image is sampled from this function on the grid points of the original image.

The matrix H depends on the truncation limit N and a series of x_i . For a given $M \times M$ image, x_i is determined by

$$x_i = -R + 2R(i-1)/(M-1)$$

where R is the range of x_i and $i = 1, 2, \dots, M$. This means that the image is mapped to $[-R, R] \times [-R, R]$ in the Cartesian coordinates. Fig. 2 shows the logarithm of the condition number of $H^T H$ when the size of the image is 301×301 . Since the small condition number of $H^T H$ guarantees the stability in its inversion as well as the invertibility, 21 appears to be a good number for the range of x_i , making it possible to choose a larger truncation limit with which the Hermite filtered image can describe more details of the original image.

Choice of the range of x_i is related to the effect of scaling factor in the Hermite transform. Basically the Hermite transforms of the similar functions $f(x)$ and $f(10x)$ illustrate a weakness of Hermite expansions. However, when we get a truncated expansion fit to a given discrete data, we can choose the scaling factor so that the difference between the filtered image and the original image is small and the matrix inversion in the filtering process is stable. Since, in Fig. 2, the available truncation limit is maximized near the value of 21, this value can be chosen as a near-optimal scale regardless of the image of interest.

Fig. 3 shows a 301×301 brain scanning image and corresponding Hermite-filtered image with the truncation limit, 240.

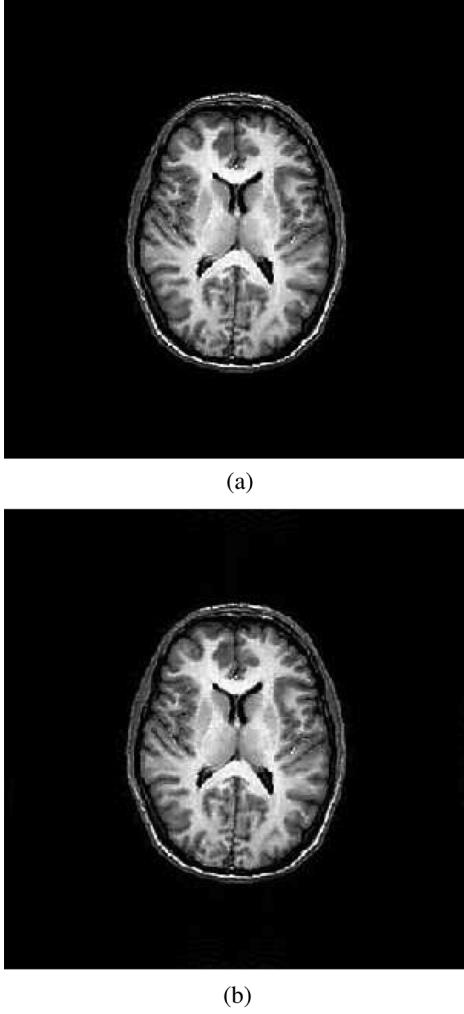


Fig. 3. Original and Hermite-filtered images. (a) Brain image (301×301). (b) Hermite-filtered image ($N = 240$, error = 5.0%). The original image is from <http://www.med.univ-angers.fr/discipline/radiologie/Intlatlas/t1ax11.html>.

The normalized RMS error between the original and the filtered version is 5.0%. For the truncation limits, 150 and 200, the errors are 10.2% and 6.9% (the images are not shown), respectively. The normalized error is defined as $\|I - I_F\|/\|I\|$, where $\|\cdot\|$ denotes the Frobenius norm, and I and I_F are the original image matrix and its Hermite-filtered version, respectively.

B. Filtered Images in Polar Coordinates

As in Cartesian coordinates, for a given array of $M_\rho \times M_\theta$ sample points on a polar grid, $\bar{f}^p(\rho_p, \theta_q)$, we minimize the following cost function to get the Laguerre–Fourier coefficients

$$\begin{aligned} C(\tilde{f}_{m,n}) &= \sum_{p=1}^{M_\rho} \sum_{q=1}^{M_\theta} \left| \bar{f}^p(\rho_p, \theta_q) \right. \\ &\quad \left. - \sum_{m=0}^N \sum_{n=-m}^m \tilde{f}_{m,n} \chi_{m,n}^*(\rho_p, \theta_q) \right|^2 \rho_p. \end{aligned}$$

For a fixed ρ_p , we consider the two functions

$$\begin{aligned} g(\theta_k) &= \bar{f}^p(\rho_p, \theta_k), \\ h(\theta_k) &= \sum_{m=0}^N \sum_{n=-m}^m \tilde{f}_{m,n} \chi_{m,n}^*(\rho_p, \theta_k) \\ &= \sum_{m=0}^N \sum_{n=-m}^m \tilde{f}_{m,n} Y_{m,n}(\rho_p) Z_n^*(\theta_k) \end{aligned}$$

where $k = 1, 2, \dots, M_\theta$. When the two functions are defined on an equally spaced grid on a circle, Parseval's equality for the discrete Fourier transform (DFT) tells us

$$\sum_{k=1}^{M_\theta} |g(\theta_k) - h(\theta_k)|^2 = \frac{1}{M_\theta} \sum_{l=0}^{M_\theta-1} |\mathcal{F}(g)_l - \mathcal{F}(h)_l|^2$$

where the DFT is defined as

$$\mathcal{F}(g)_l = \sum_{j=0}^{M_\theta-1} g(\theta_j) e^{-2\pi i j l / M_\theta}.$$

The DFT of $Z_n^*(\theta_k)$ gives

$$\begin{aligned} \mathcal{F}(Z_n^*(\theta_k))_l &= \sum_{j=0}^{M_\theta-1} Z_n^*(\theta_j) e^{-2\pi i j l / M_\theta} \\ &= \frac{1}{\sqrt{2\pi}} \sum_{j=0}^{M_\theta-1} e^{2\pi i j (n-l) / M_\theta} \\ &= \frac{M_\theta}{\sqrt{2\pi}} \delta_{l,n}. \end{aligned}$$

With these relations, the cost function can be rewritten as

$$\begin{aligned} C &= S \sum_{p=1}^{M_\rho} \rho_p \sum_{l=0}^{M_\theta-1} \left| \mathcal{F}(\bar{f}^p(\rho_p))_l \right. \\ &\quad \left. - \frac{M_\theta}{\sqrt{2\pi}} \sum_{m=0}^N \sum_{n=-m}^m \tilde{f}_{m,n} Y_{m,n}(\rho_p) \delta_{l,n} \right|^2 \\ &= S \sum_{p=1}^{M_\rho} \sum_{l=0}^{M_\theta-1} \left| \mathcal{F}(\bar{f}^p(\rho_p))_l \sqrt{\rho_p} \right. \\ &\quad \left. - \frac{M_\theta}{\sqrt{2\pi}} \sum_{m=0}^N \sum_{n=-m}^m \tilde{f}_{m,n} Y_{m,n}(\rho_p) \sqrt{\rho_p} \delta_{l,n} \right|^2 \end{aligned}$$

where S is a constant, which is not important in minimizing the cost function. Changing the order of summations gives

$$\begin{aligned} C &= S \sum_{l=0}^{M_\theta-1} \sum_{p=1}^{M_\rho} \left| \mathcal{F}(\bar{f}^p(\rho_p))_l \sqrt{\rho_p} - \frac{M_\theta}{\sqrt{2\pi}} \right. \\ &\quad \times \left. \sum_{n=-N}^N \sum_{m=|n|}^N \tilde{f}_{m,n} Y_{m,n}(\rho_p) \sqrt{\rho_p} \delta_{l,n} \right|^2. \end{aligned}$$

Note that, in the fourth summation, the variable, m is increasing by multiples of 2. For the filtering to be possible, the number of data points should be greater than or equal to the number of

coefficients, which means $M_\theta \geq N + 1$ in the θ direction. The cost function is

$$\begin{aligned} C = S \sum_{l=0}^N \sum_{p=1}^{M_\rho} & \left| \mathcal{F}(\bar{f}^p(\rho_p))_l \sqrt{\rho_p} \right. \\ & - \frac{M_\theta}{\sqrt{2\pi}} \sum_{n=-N}^N \sum_{m=|n|}^N \tilde{f}_{m,n} Y_{m,n}(\rho_p) \sqrt{\rho_p} \delta_{l,n} \left. \right|^2 \\ & + S \sum_{l=N+1}^{M_\theta-1} \sum_{p=1}^{M_\rho} |\mathcal{F}(\bar{f}^p(\rho_p))_l \sqrt{\rho_p}|^2. \end{aligned}$$

Since the second term on the right hand side is constant, the first term is all that is of interest when performing minimization. The new cost function is

$$\begin{aligned} C' = \sum_{l=0}^N \sum_{p=1}^{M_\rho} & \left| \mathcal{F}(\bar{f}^p(\rho_p))_l \sqrt{\rho_p} \right. \\ & - \frac{M_\theta}{\sqrt{2\pi}} \sum_{m=l}^N \tilde{f}_{m,l} Y_{m,l}(\rho_p) \sqrt{\rho_p} \left. \right|^2. \quad (10) \end{aligned}$$

For a fixed l , let us define the following two vectors and a matrix (shown in the equation at the bottom of the page), where $[n/2] = n/2$ if n is even and $[n/2] = (n - 1)/2$ if n is odd. The cost function is

$$C' = \sum_{l=0}^N \|b_l - A_l x_l\|^2.$$

Even though b_l and x_l consist of complex numbers, the pseudo inverse approach is still possible because A_l is a matrix of real numbers. As long as the condition number of A_l is small, the minimizer is

$$x_l = (A_l^T A_l)^{-1} A_l^T b_l.$$

The matrix A_l is determined by the truncation limit and the range of ρ , much like H was in the case of Cartesian coordinates. Fig. 4 shows the condition numbers of A_0 , when $M_\rho = 151$; 21 and 240 are the best choice for the range of ρ and the truncation limit, respectively. In that case, Fig. 5 shows the condition number of A_l with respect to l . Since the maximum condition number appears at $l = 0$, the choice is safe. Fig. 6

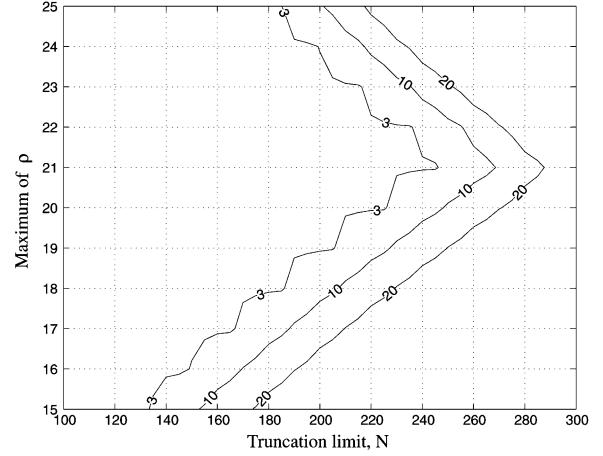


Fig. 4. Contour plot of logarithm of condition number of A_0 depending on truncation limit and range of ρ ($M_\rho = 151$).

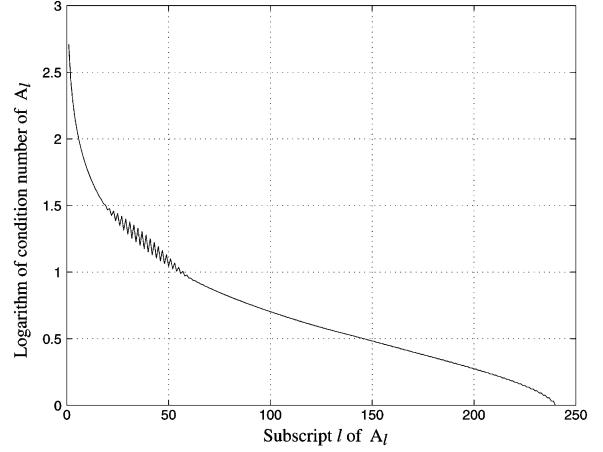


Fig. 5. Logarithm of condition number of A_l ($M_\theta = 151, N = 240$).

shows a brain image expressed in polar coordinates and corresponding Laguerre–Fourier-filtered images with different truncation limits.

IV. INTERCONVERSION BETWEEN POLAR AND CARTESIAN COORDINATES

In this section, we find the relationship between the representations of a function in polar and Cartesian coordinates when the

$$\begin{aligned} b_l &= [\mathcal{F}(\bar{f}^p(\rho_1))_l \sqrt{\rho_1} \dots \mathcal{F}(\bar{f}^p(\rho_{M_\rho}))_l \sqrt{\rho_{M_\rho}}]^T \\ x_l &= [\tilde{f}_{l,l} \tilde{f}_{l+2,l} \dots \tilde{f}_{2[(N-l)/2]+l,l}]^T \end{aligned}$$

and

$$\begin{aligned} A_l &= \frac{M_\theta}{\sqrt{2\pi}} \\ &\times \begin{pmatrix} \sqrt{\rho_1} Y_{l,l}(\rho_1) & \sqrt{\rho_1} Y_{l+2,l}(\rho_1) & \vdots \\ \sqrt{\rho_2} Y_{l,l}(\rho_2) & \ddots & \vdots \\ \cdots & \cdots & \sqrt{\rho_{M_\rho}} Y_{2[\frac{N-1}{2}]+l,l}(\rho_{M_\rho}) \end{pmatrix} \end{aligned}$$

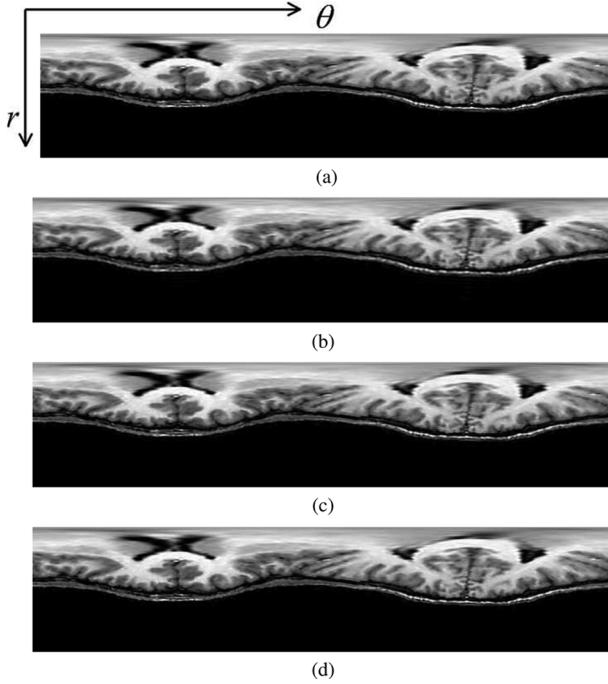


Fig. 6. Original and Laguerre–Fourier-filtered images viewed in a polar coordinate array. (a) Brain image on polar coordinates (700×151). (b) Laguerre–Fourier-filtered image ($N = 150$, error = 7.1%). (c) Laguerre–Fourier-filtered image ($N = 200$, error = 4.5%). (d) Laguerre–Fourier-filtered image ($N = 240$, error = 3.3%).

truncated expansions in (5) and (9) are used. Furthermore, we relate the truncation limits, N and M , so that there is no loss of information during conversion.

A. Laguerre–Fourier Coefficients in Terms of Hermite Coefficients

The Laguerre polynomials and the Fourier basis can be written in terms of the Hermite polynomials as follows [2]:

$$e^{im\theta} \rho^m L_n^m(\rho^2) = \frac{(-1)^n}{2^{2n+m} n!} \sum_{r=0}^n \sum_{s=0}^m i^s C_n^r C_m^s \\ \times H_{2r+m-s}(x) H_{2n-2r+s}(y) \quad (11)$$

where $\rho = \sqrt{x^2 + y^2}$ and $\theta = \text{atan}(x, y)$. The function $\text{atan}(x, y)$ returns the four-quadrant inverse tangent, which is the angle between the vectors $(x, y)^T$ and $(1, 0)^T$.

Using (1), (6), and (11), we can express $\chi_{mn}(\rho, \theta)$ as

$$\chi_{mn}(\rho, \theta) = \sum_{r=0}^{(m+n)/2} \sum_{s=0}^{-n} i^s C_{\frac{m+n}{2}}^r C_{-n}^s h_{2r-n-s}(x) \\ \times h_{m+n-2r+s}(y) \sqrt{\frac{(2r-n-s)!(m+n-2r+s)!}{2^m (\frac{m-n}{2})! (\frac{m+n}{2})!}}$$

where $n \leq 0$, $x = \rho \cos \theta$ and $y = \rho \sin \theta$. If we use a new variable, $\alpha = 2r - n - s$, then the double sum is $\sum_{r=0}^{(m+n)/2} \sum_{\alpha=2r}^{2r-n}$. If we define the combination symbols, $C_n^k = n!/(k!(n-k)!)$, to be zero when they cannot be defined (e.g., $C_2^{-1}, C_3^4 = 0$), we can extend the limits of the summations as $\sum_{r=0}^{[m/2]} \sum_{\alpha=2r}^m$, where $[m/2] = m/2$ if m is even and $[m/2] = (m-1)/2$ if m

is odd. Finally, we change the order of summations to get

$$\chi_{mn}(\rho, \theta) = \sum_{\alpha=0}^m \sqrt{\frac{\alpha!(m-\alpha)!}{2^m (\frac{m-n}{2})! (\frac{m+n}{2})!}} \\ \times \left[\sum_{r=0}^{[\alpha/2]} i^{2r-n-\alpha} C_{(m+n)/2}^r C_{-n}^{2r-n-\alpha} \right] h_\alpha(x) h_{m-\alpha}(y).$$

For $n > 0$, we use the fact that $\chi_{m,n} = \chi_{m,-n}^*$. Substituting this χ_{mn} and (5) in (8) gives

$$\tilde{f}_{p,q} = \sum_{n=0}^p G_n(p, q) \hat{f}_{n,p-n} \quad (12)$$

where

$$G_n(p, q) = \begin{cases} \sqrt{\frac{n!(p-n)!}{2^p (\frac{p+q}{2})! (\frac{p-q}{2})!}} \sum_{r=0}^{[n/2]} C_{\frac{p+q}{2}}^r C_{-q}^{m-2r} i^{2r-q-n} & \text{if } q \leq 0 \\ \sqrt{\frac{n!(p-n)!}{2^p (\frac{p+q}{2})! (\frac{p-q}{2})!}} \sum_{r=0}^{[n/2]} C_{\frac{p-q}{2}}^r C_q^{m-2r} (-i)^{2r+q-n} & \text{if } q > 0. \end{cases}$$

A similar equation has been shown in [1]. However, (12) specifically tells us how we can compute the Laguerre–Fourier coefficients using the Hermite coefficients. This equation also contains some other important properties, as described below.

First, the Laguerre–Fourier coefficients (\tilde{f}) can be computed by linear combination of the Hermite coefficients (\hat{f}). Second, $\tilde{f}_{p,q}$ is determined only by a set $\{\hat{f}_{n,p-n}\}$ ($n = 0, 1, \dots, p$). In other words, the relationships are localized. Third, it tells us how the truncation limit should be determined in order for the coordinate conversion to be lossless. Since we started with (5), p in (12) cannot exceed N . Therefore, the Laguerre–Fourier coefficients from (12) are $\tilde{f}_{p,q}$, ($p = 0, 1, \dots, N, q = -p, -p+2, \dots, p$). We reconstruct the 2-D function, $f^p(\rho, \theta)$ as

$$f^p(\rho, \theta) = \sum_{p=0}^N \sum_{q=-p}^p \tilde{f}_{p,q} \chi_{p,q}^*(\rho, \theta).$$

Note that this is directly derived from (5) without approximation. Comparing this to (9), we come to know that the truncation limits in (5) and (9), N and M , should be equal in order for there to be no loss during conversion.

B. Hermite Coefficients in Terms of Laguerre–Fourier Coefficients

The Hermite polynomials can be written in terms of the Laguerre polynomials and the Fourier basis as follows [2]:

$$H_l(x) H_{k-l}(y) = \text{Re} \left((-i)^{k-l} \sum_{r=0}^l \sum_{s=0}^{k-l} (-1)^r C_l^r C_{k-l}^s \right. \\ \left. \times (r+s)! e^{i(k-2r-2s)\theta} \rho^{(k-2r-2s)} L_{r+s}^{k-2r-2s}(\rho^2) \right) \quad (13)$$

where $C_n^k = n!/(k!(n-k)!)$, $x = \rho \cos \theta$, $y = \rho \sin \theta$, and $\text{Re}(c)$ gives a real part of a complex number, c . To evaluate

the associate Laguerre polynomial, L_m^n for $n < 0$, we use the formula [2]

$$m!\rho^n L_m^n(\rho^2) = (-1)^n(m+n)!\rho^{-n}L_{m+n}^{-n}(\rho^2). \quad (14)$$

Depending on r and s in (13), the superscript of the Laguerre polynomial may be positive or negative. Using (14), one can rewrite (13) with Laguerre polynomials having only positive superscripts. Then, applying (1) and (6) to the new version of (13), we can express the Hermite coefficients in terms of the Laguerre–Fourier coefficients as

$$\hat{f}_{l,k-l} = \operatorname{Re} \left\{ \sum_{j=0}^k D_j(l, k-l) \tilde{f}_{k,2j-k} \right\} \quad (15)$$

where

$$D_j(l, k-l) = \sum_{r=0}^l (-1)^{j-r}(-i)^{k-l} C_l^r C_{k-l}^{j-r} \sqrt{\frac{(k-j)!j!}{2^k l!(k-l)!}}.$$

This relationship is also linear and localized. In addition, the truncation limits, N and M should be equal for lossless conversion.

C. Interconversion Between Truncated Hermite and Laguerre–Fourier Expansions

Now we can conclude that the following two expressions are equivalent

$$\begin{aligned} f^p(\rho, \theta) &= \sum_{p=0}^N \sum_{q=-p}^p \tilde{f}_{p,q} \chi_{p,q}^*(\rho, \theta) \\ &\Downarrow \\ f^c(x, y) &= \sum_{m=0}^N \sum_{n=0}^m \hat{f}_{n,m-n} h_n(x) h_{m-n}(y) \end{aligned} \quad (16)$$

where $x = \rho \cos \theta$ and $y = \rho \sin \theta$, and the constraints described below are in effect.

Since f is a real-valued function, in the Cartesian coordinate expression there are $((N+1)(N+2)/2)$ coefficients that are real numbers. On the other hand, since n is increasing by 2 in the polar coordinate expression, there are $((N+1)(N+2)/2)$ coefficients that are complex numbers, which means there are $(N+1)(N+2)$ real numbers. However, since $\tilde{f}_{p,0}$ is a real number and $\tilde{f}_{p,q} = \tilde{f}_{p,-q}^*$, there are exactly $((N+1)(N+2)/2)$ independent real numbers in both expressions.

As shown in (16), once we have a truncated expansion in either of the two coordinate systems, we can easily evaluate the function on the complementary grid. This can be rewritten as

$$\begin{aligned} f^p(\rho, \theta) &= f^c(\rho \cos \theta, \rho \sin \theta) \\ &= \sum_{m=0}^N \sum_{n=0}^m \hat{f}_{n,m-n} h_n(\rho \cos \theta) h_{m-n}(\rho \sin \theta) \end{aligned} \quad (17)$$

$$\begin{aligned} f^c(x, y) &= f^p(\sqrt{x^2 + y^2}, \operatorname{atan}(x, y)) \\ &= \sum_{p=0}^N \sum_{q=-p}^p \tilde{f}_{p,q} \chi_{p,q}^*(\sqrt{x^2 + y^2}, \operatorname{atan}(x, y)). \end{aligned} \quad (18)$$

Suppose filtered data is provided on one coordinate grid (i.e., polar or Cartesian). Evaluation on the complementary coordinate grid is possible without using the truncated expansion associated with the complementary grid by simply resampling when the expansions in (16) are equivalent. It is important to realize that the justification for doing this is the lossless interconversion between these expansions when the compatible truncation limits are enforced via filtering. In other words, the mathematical invertibility of (12) and (15) in exact arithmetic is what justifies resampling.

In principle, given a truncated expansion in one coordinate system, we can find the complementary expansion by implementing (12) or (15). However, in practice, there may be an obstacle in computing them. The factorials and the combinations (C_n^m) for large arguments introduce some numerical error during computation. To overcome this, we use a resampling technique. Given a set of \tilde{f} , we sample appropriate points of $f^p(\rho, \theta)$ using (17) and then compute the Laguerre–Fourier coefficients using the method described in Section III-A. For an arbitrary image, there is loss of information in the filtering process, but applying it to resampled data from a truncated expansion gives exact coefficients (to within machine precision). The whole process and the numerical examples will be presented in the next section.

V. NUMERICAL EXAMPLES

Fig. 7(a) shows how we perform the interconversion for a given image initially defined on a Cartesian coordinate grid. For the image on a rectangular grid, we compute the Hermite coefficients using the filtering process suggested in Section III-A. The truncated Hermite expansion gives the filtered image as shown in Fig. 3. Then we evaluate the Hermite expansion on the polar grid. Since we know that the sampled values are sample values of a truncated expansion on the polar grid, we can compute the exact coefficients of the truncated Laguerre–Fourier expansion using the method in Section III-B. If we again evaluate the truncated Laguerre–Fourier expansion on a Cartesian coordinate grid, the image is the same (to within machine precision) as the original Hermite filtered image. The double arrows in Fig. 7 indicate that the process is reversible. Fig. 8(a) is resampled data on polar grid using the truncated Hermite expansion. We compute the exact Laguerre–Fourier coefficients using this resampled data. Again, if we resampled data on the Cartesian grid using the truncated Laguerre–Fourier expansion, we have the resulting image shown in Fig. 8(b), which is the same as the filtered image of Fig. 3(b). The normalized error between the two is 3.5×10^{-14} , which illustrates the lossless interconversion when the filtering process in Cartesian coordinates is applied.

Similarly, Fig. 7(b) shows the interconversion for a given image on a polar coordinate grid. For the image on the polar grid, we compute the Laguerre–Fourier coefficients using the filtering process. Fig. 9(a) is resampled data on a Cartesian grid using the truncated Laguerre–Fourier expansion. We compute the exact Hermite coefficients using this resampled data. When we resampled data on a polar grid using the truncated Hermite expansion, the resulting image shown in Fig. 9(b) is the same as the filtered image of Fig. 6(d). The normalized error between

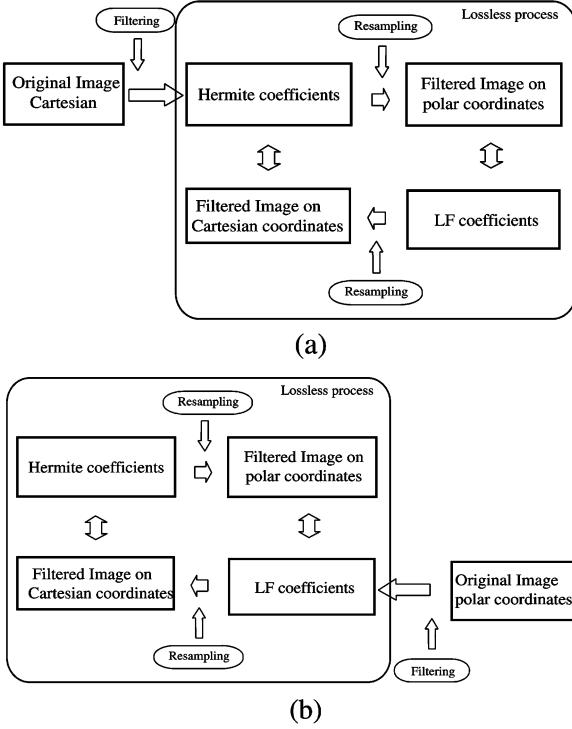


Fig. 7. Block diagram of interconversion of data given in different coordinate systems. (a) Interconversion with data on Cartesian grid. (b) Interconversion with data on polar grid.

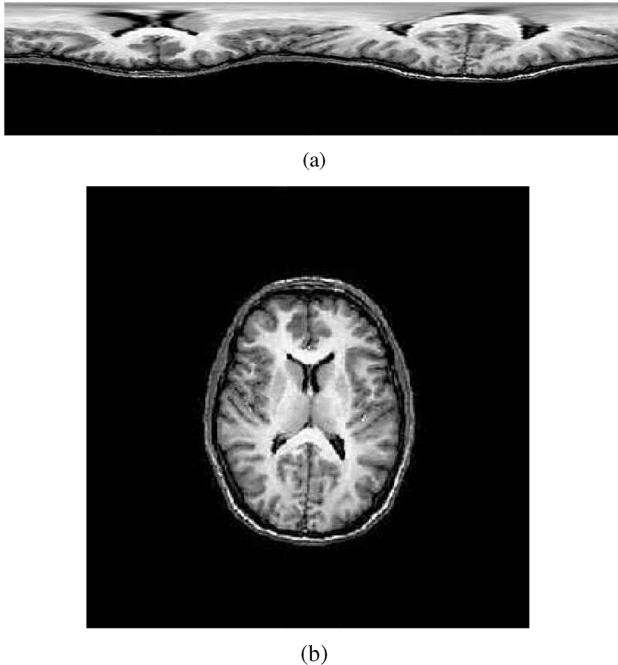


Fig. 8. Images from interconversion with data initially given on a Cartesian grid. (a) Reconstructed data on polar grid with Hermite expansion. (b) Reconstructed data on Cartesian grid with Laguerre–Fourier expansion.

the two is 3.3×10^{-14} , which illustrates the lossless interconversion when filtering in polar coordinates is applied.

As we mentioned earlier, the data obtained from a transducer or sensor in polar coordinates often needs to be displayed on a Cartesian coordinate grid. Since equally spaced grids in the

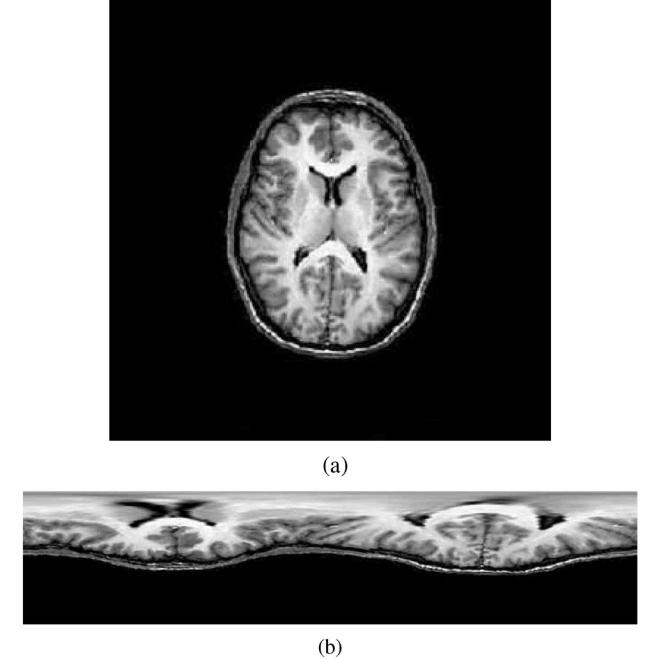


Fig. 9. Images from interconversion with data initially given on a polar grid. (a) Reconstructed data on Cartesian grid with Laguerre–Fourier expansion. (b) Reconstructed data on polar grid with Hermite expansion.

two coordinate systems do not match as shown in Fig. 1, one needs to compute a new set of data that fits on a Cartesian grid. Displaying the data from the polar grid on the Cartesian grid directly introduces two problems. First, it is hard to determine where to assign data at (r, θ) on the Cartesian grid. The data may be assigned to $(r \cos \theta, r \sin \theta)$ on the Cartesian grid, but the equally spaced Cartesian grid cannot capture $(r \cos \theta, r \sin \theta)$ completely. Second, even if one assigned the data to the approximate position, there can be some vacant pixels in areas far from the origin as shown in Fig. 10(a), and while local interpolation methods exist (e.g., bivariate linear), these use only local features and are not reversible.

With data provided on a polar coordinate grid, we compute the Laguerre–Fourier coefficients by the filtering process. Evaluating the truncated Laguerre–Fourier expansion onto a Cartesian coordinate grid gives the interpolated image as shown in Fig. 10(b). Since the interpolated image can also be viewed as being sampled from a truncated Hermite expansion, we then can apply existing image processing methods associated with the Hermite transform [10], [11].

In the strict meaning of interpolation, our algorithm has a drawback in that the filtering process introduces some error with respect to the original image, and that error persists through the coordinate conversion process. However, we showed that the error can be controlled at the expense of the truncation limit and the actual error does not leave critical artifacts on the image as shown in Figs. 3 and 6.

VI. CONCLUSION AND DISCUSSION

In this paper, we have proposed a new method of interconverting images between polar and Cartesian coordinate systems without loss of information after an initial filtering process. We

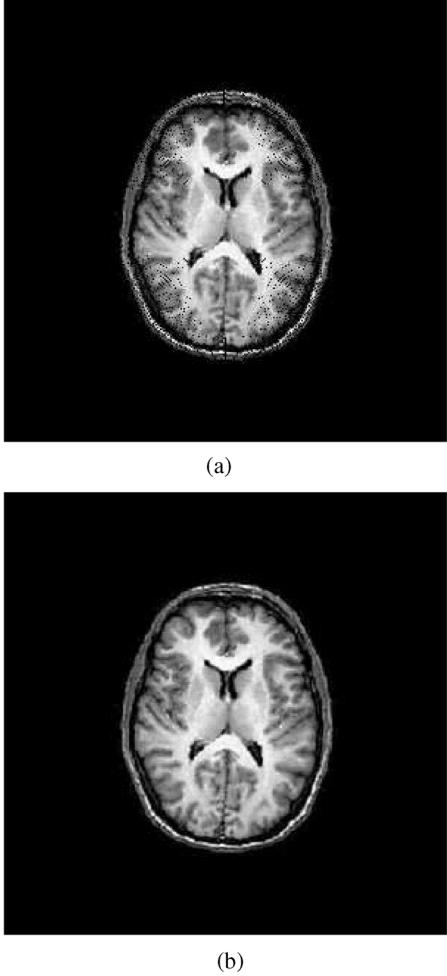


Fig. 10. Interpolation using the interconversion method. (a) Direct assignment of polar grid data onto a Cartesian grid. (b) Interpolated image.

defined the filtering process and assessed the error that the filter can produce. Using truncated expansions, we investigated the relationship between the two representations of a function in polar and Cartesian coordinates. Some special functions such as the Laguerre polynomials, the Fourier basis and the Hermite functions were used.

Conversion between the coordinate systems boiled down to conversion of the coefficients of the truncated expansions on the polar and Cartesian coordinates. We found that the two sets of coefficients are linearly and locally related. From the equation expressing the relationship, we uncovered a special truncation limit for the two expansions, which guarantees lossless conversion.

Based on the equivalence of the two truncated expansions in polar and Cartesian coordinates, we could get an image in a new coordinate system by resampling the points from the truncated expansion in the original coordinate system. Additionally, we showed that this technique can be applied to the interpolation problem. With the filtered image on the polar grid, we could obtain an image on a Cartesian grid without losing any information or having missing pixels by using a resampling technique.

In our expansions, an image (and its extension to values all over \mathbb{R}^2) is defined by $\mathcal{O}(N^2)$ coefficients. The Hermite filtering process is of $\mathcal{O}(N^3)$, because we invert a $N \times N$ matrix. Resampling the Hermite expansion on polar coordinates is

of $\mathcal{O}(N^4)$, while $\mathcal{O}(N^3)$ computations are sufficient in resampling it onto Cartesian coordinates. Our implementation of the Laguerre–Fourier filtering process is of $\mathcal{O}(N^4)$, since we invert a $N \times N$ matrix N times. Note that we invert the matrices regardless of Fourier transform. Even when employing the fast Fourier transform, minimizing (10) still uses $\mathcal{O}(N^4)$ computations. Resampling the Laguerre–Fourier expansion on the polar and Cartesian coordinates uses $\mathcal{O}(N^3)$ and $\mathcal{O}(N^4)$ computations, respectively. Therefore, the total complexity of interconversion is $\mathcal{O}(N^4)$. If the constants in (12) and (15) are precalculated, the direct conversion of those coefficients is of $\mathcal{O}(N^3)$. However, the direct computation introduces numerical error as explained in Section IV-C. Even though it is possible, the complexity of the whole interconversion process does not change because the sampling process on the complementary grid uses $\mathcal{O}(N^4)$ computations. There presently are not fast algorithms for the Hermite-function and Laguerre–Fourier transforms. This is an active area of research, and as the situation changes, our algorithm may become faster.

Finally, we showed how we can use our algorithm can be used to interpolate an image defined in the polar grid to obtain an interpolated image in Cartesian coordinates. If the original image is filtered in the way we suggested in this paper, the interpolated image does not lose information of the original filtered image. Therefore, we also propose a new filtering process that enables the interpolation between the polar and Cartesian coordinate systems.

REFERENCES

- [1] R. Massey and A. Refregier, "Polar shapelets," *Mon. Not. R. Astron. Soc.*, vol. 363, pp. 197–210, 2005.
- [2] I. Kimel and L. R. Elias, "Relations between Hermite and Laguerre Gaussian modes," *IEEE J. Quant. Electron.*, vol. 29, no. 9, pp. 2562–2567, Sep. 1993.
- [3] A. Refregier, "Shapelets: I. A method for image analysis," *Mon. Not. R. Astron. Soc.*, vol. 338, pp. 35–47, 2003.
- [4] Z. Chen and R. Ning, "Filling the radon domain in computed tomography by local convex combination," *Appl. Opt.*, vol. 42, no. 35, Dec. 2003.
- [5] C. Basoglu, Y. Kim, and V. Chalana, "A real-time scan conversion algorithm on commercially available microprocessors," *Ultrason. Imag.*, vol. 18, no. 4, pp. 241–260, Oct. 1996.
- [6] A. P. Berkhoff, H. J. Huisman, J. M. Thijssen, E. M. G. P. Jacobs, and R. J. F. Homan, "Fast scan conversion algorithms for displaying ultrasound sector images," *Ultrason. Imag.*, vol. 16, no. 2, pp. 87–108, 1994.
- [7] J. J. Knab, "The sampling window," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 1, pp. 157–159, Jan. 1983.
- [8] D. Chang, W. Wu, and J. Jeng, "An efficient architecture of ultrasonic scan conversion for implementing the cubic convolution interpolation," in *Proc. IEEE Nuclear Science Symp.*, Nov. 1997, vol. 2, pp. 1546–1550.
- [9] S. J. Julier and J. K. Uhimaun, "A consistent, debiased method for converting between polar and cartesian coordinate systems," *Proc. SPIE*, vol. 3086, pp. 110–121, 1997.
- [10] A. M. van Dijk and J. Martens, "Feature-based image compression with steered Hermite transforms," in *Proc. IEEE Int. Conf. Image Processing*, Sep. 1996, vol. 1, pp. 205–208.
- [11] J. Martens, "Local orientation analysis in images by means of the Hermite transform," *IEEE Trans. Image Process.*, vol. 6, no. 8, pp. 1103–1116, Aug. 1997.
- [12] G. S. Chirikjian and A. B. Kyatkin, *Engineering Applications of Non-commutative Harmonic Analysis*. Boca Raton, FL: CRC, 2001.
- [13] N. J. Vilenkin and A. U. Klimyk, *Representation of Lie Groups and Special Functions*. Dordrecht, The Netherlands: Kluwer, 1991, vol. 1–3.
- [14] W. Park and G. S. Chirikjian, "Lossless rotation of Hermite-filtered images," *IEEE Trans. Image Process.*, to be published.



Wooram Park (S'06) received the B.S.E. and M.S.E. degrees in mechanical engineering from Seoul National University, Seoul, Korea, in 1999 and 2003, respectively, and the M.S.E. degree in mechanical engineering from The Johns Hopkins University, Baltimore, MD, in 2005, where he is currently pursuing the Ph.D. degree in the Department of Mechanical Engineering.

His research interests include structural biology using image processing and NMR.



Gregory S. Chirikjian (M'93) received the B.S.E. degree in engineering mechanics, the M.S.E. degree in mechanical engineering, and the B.A. degree in mathematics from The Johns Hopkins University, Baltimore, MD, in 1988, and the Ph.D. degree from the California Institute of Technology, Pasadena, in 1992.

Since the summer of 1992, he has been with the Department of Mechanical Engineering, The Johns Hopkins University, where he is now Professor and Chair. His research interests include mathematical

aspects of robotics and imaging. In recent years, he has also been applying methods from robotics to model conformational transitions in biological macromolecules.

Dr. Chirikjian is a 1993 National Science Foundation Young Investigator, a 1994 Presidential Faculty Fellow, and a 1996 recipient of the ASME Pi Tau Sigma Gold Medal.