Interpolation Revisited

Philippe Thévenaz*, *Member, IEEE*, Thierry Blu, *Member, IEEE*, and Michael Unser, *Fellow, IEEE*

***Abstract*—Based on the theory of approximation, this paper presents a unified analysis of interpolation and resampling techniques. An important issue is the choice of adequate basis functions. We show that, contrary to the common belief, those that perform best are not interpolating. By opposition to traditional interpolation, we call their use generalized interpolation; they involve a prefiltering step when correctly applied. We explain why the approximation order inherent in any basis function is important to limit interpolation artifacts. The decomposition theorem states that any basis function endowed with approximation order can be expressed as the convolution of a B-spline of the same order with another function that has none. This motivates the use of splines and spline-based functions as a tunable way to keep artifacts in check without any significant cost penalty. We discuss implementation and performance issues, and we provide experimental evidence to support our claims.**

***Index Terms*—Approximation constant, approximation order, B-splines, Fourier error kernel, maximal order and minimal support (Moms), piecewise-polynomials.**

I. Introduction

THE ISSUE of quality is particularly relevant to the medical community; for ethical reasons, it is a prime concern when manipulating data. Any manipulation should result in the least amount of distortion or artifacts, so as not to influence the clinician's judgment. For practical reasons, efficiency is another prime concern. Any processing should result in the least computational effort, particularly when dealing with the large amount of data involved in volumetric medical imaging. In this paper, we analyze the tradeoff between the quality and the cost of several interpolation methods, and we introduce generalized interpolation as a means to overcome the limitations of traditional interpolation.

Interpolation is at the heart of various medical imaging applications [1]–[3]. In volumetric imaging, it is often used to compensate for nonhomogeneous data sampling. This rescaling operation is desirable to build isometric volumes [4]–[6]. Another application of this transform arises in the three-dimensional (3-D) reconstruction of icosahedral viruses [7]. In volume rendering, it is common to apply by interpolation a texture to the facets that compose the rendered object [8]. In addition, volume rendering may also require the computation of gradients, which is best done by taking the interpolation model into account [9]. In functional magnetic resonance imaging (fMRI), the relative amplitude of the difference between two conditions (e.g., active versus inactive) is very small. Registration is used to align fMRI images before they are subtracted to reveal where and how they differ [10]. However, even with perfect knowledge of the ideal geometric transformation, a deficient interpolation can wash out these tiny differences.

The essence of interpolation is to represent an arbitrary continuously defined function as a discrete sum of weighted and shifted basis functions. An important issue is the adequate choice of those basis functions. The traditional view asks that they satisfy the interpolation property, and many researchers have put a significant effort in optimizing them under this specific constraint [11]–[17]. Over the years, these efforts have shown more and more diminishing returns.

Here, instead, we introduce and advocate the use of generalized interpolation, which does away with the constraint of interpolation at the cost of an additional prefiltering step. The overall benefit is to allow for the use of a much broader class of potential basis functions, some of which enjoy, at the same time, excellent approximation properties and short support. We present a performance analysis that lies on the firm ground of approximation theory. We introduce analytical tools that allow the practitioner to determine the theoretical performance of any basis function, including noninterpolating ones, and provide an in-depth analysis of many piecewise-polynomial cases. This class is important because it contains some families of basis functions that can be shown to be the best achievable with respect to several criteria, such as maximal regularity (B-splines), or best least-squares approximation properties (o-Moms). We show both theoretically and experimentally that generalized interpolation performs better than traditional interpolation in the context of image transformations. The resulting quality can be arbitrarily high; for a given quality, generalized interpolation comes at a lower computational cost than that incurred by the traditional methods which satisfy the interpolating constraint.

The organization of this paper is as follows. Section II introduces the notations and we compare traditional interpolation with our proposition for generalized interpolation. In Section III, we discuss some desirable aspects of basis functions in the imaging context. We expose in Section IV the main contribution of this paper, where we apply to interpolation a methodology that originates in the theory of approximation. We present the decomposition theorem and develop tools that are beneficial to its application. In Section V, we analyze several piecewise-polynomial basis functions, while we analyze sinc-based ones in Section VI. In Section VII, we conduct a theoretical study of the tradeoff between speed and quality; the validity of this study is confirmed by the experiments that we present in Section VIII. Finally, we conclude in Section IX.

Manuscript received July 22, 1999; revised May 12, 2000. The Associate Editor responsible for coordinating the review of this paper and recommending its publication was Y. Censor. *Asterisk indicates corresponding author*.

*P. Thévenaz, T. Blu, and M. Unser are with the Swiss Federal Institute of Technology, Lausanne CH-1015, Switzerland.

Publisher Item Identifier S 0278-0062(00)07330-4.

II. Interpolation

A. Scope

We restrict this paper to the case where the discrete data are regularly sampled on a Cartesian grid. We also restrict the discussion to exact interpolation, where the continuous model is required to take the same values as the sampled data at the grid locations. Finally, we restrict ourselves to linear methods, such that the sum of two interpolated functions is equal to the interpolation of the sum of the two functions.

B. Traditional Interpolation

Let us express an interpolated value $f(\mathbf{x})$ at some (perhaps noninteger) coordinate $\mathbf{x}$ in a space of dimension q as a linear combination of samples $f_{\mathbf{k}}$ evaluated at integer coordinates $\mathbf{k} = (k_1, k_2, \ldots, k_q) \in Z^q$

$$f(\mathbf{x}) = \sum_{\mathbf{k} \in Z^q} f_{\mathbf{k}} \varphi_{\text{int}}(\mathbf{x} - \mathbf{k}) \qquad \forall \mathbf{x} = (x_1, x_2, \ldots, x_q) \in R^q. \tag{1}$$

The sample weights are given by the values of the function $\varphi_{\text{int}}(\mathbf{x} - \mathbf{k})$. To satisfy the requirement of exact interpolation, we ask that the function φ_{int} vanishes for all integer arguments except at the origin, where it must take a unit value. A classical example of the basis function φ_{int} is the sinc function, in which case all synthesized functions are bandlimited.

C. Generalized Interpolation

As an alternative approach, let us consider the form

$$f(\mathbf{x}) = \sum_{\mathbf{k} \in Z^q} c_{\mathbf{k}} \varphi(\mathbf{x} - \mathbf{k}) \qquad \forall \mathbf{x} \in R^q. \tag{2}$$

The crucial difference between the classical formulation (1) and the generalized formulation (2) is the introduction of coefficients $c_{\mathbf{k}}$ in place of the sample values $f_{\mathbf{k}}$. This offers new possibilities, in the sense that interpolation can now be carried in two separate steps. Firstly, the determination of coefficients $c_{\mathbf{k}}$ from the samples $f_{\mathbf{k}}$, and second, the determination of desired values $f(\mathbf{x})$ from the coefficients $c_{\mathbf{k}}$. The benefit of this separation is to allow for an extended choice of basis functions, some with better properties than those available in the restricted classical case where $c_{\mathbf{k}} = f_{\mathbf{k}}$. The apparent drawback is the need for an additional step. We will see later that this drawback is largely compensated by the gain in quality resulting from the larger selection of basis functions to choose from.

Whether the interpolation is traditional or generalized, we carry the summations all the way to—and from—infinity. Thus, it is essential to assign a specific value to those samples $f_{\mathbf{k}}$ that are unknown because they are out of the range of our data. In the context of generalized interpolation, we will soon see that any given coefficient $c_{\mathbf{k}_0}$ is dependent on all sample values $f_{\mathbf{k}}$, $\forall \mathbf{k} \in Z^q$; for this reason, it is preferable to limit the degree of arbitrariness when extrapolating data. To reduce boundary effects, we prefer to avoid the traditional extension which imposes $f_{\mathbf{k}} = 0$, $\mathbf{k} \notin \mathbf{D}$, where $\mathbf{D}$ is the known support of the data; instead, we advocate to perform implicit data mirroring. This latter solution is preferable because no special value (e.g., 0) is introduced; only already-existing values are used. When compared to the other traditional extension known as periodization, it offers the additional advantage that no abrupt transition results on the data boundaries. Furthermore, the structure of mirror-extended data is invariant under filtering provided the filter is symmetric, which yields consistency of design. In the case one-dimensional (1-D) letting the known range be $[x_0, x_1]$, the mirror-extended signal satisfies

$$\begin{cases} f(x_0 - x) = f(x_0 + x) \\ f(x_1 - x) = f(x_1 + x). \end{cases}$$

Given $f(x)$, $x \in [x_0, x_1]$, the (eventually multiple) application of the folding operation just given is sufficient to define $f(x)$, $\forall x \in R$. We explain in Appendix-A how this translates to a practical algorithm for discrete arguments, and we extend this 1-D case to higher dimensions in Section III-B.

D. Determination of the Coefficients

To enforce exact interpolation for integer arguments $\mathbf{x} = \mathbf{k}_0$, we write that

$$f_{\mathbf{k}_0} = \sum_{\mathbf{k} \in Z^q} c_{\mathbf{k}} p_{\mathbf{k}_0 - \mathbf{k}} \qquad \forall \mathbf{k}_0 \in Z^q \tag{3}$$

where $p_{\mathbf{k}} = \varphi(\mathbf{k})$. Given some function φ that is known *a priori*, this expression is nothing but a linear system of equations in terms of the unknown coefficients $c_{\mathbf{k}}$. We are now faced with a problem of the form $\mathbf{c} = \mathbf{P}^{-1}\mathbf{f}$, and a large part of the literature (e.g., [18]) is devoted to the development of efficient techniques for inverting the matrix $\mathbf{P}$ in the context of specific basis functions φ. The problem is trivial when φ is interpolating, for then $\mathbf{P}$ is the identity.

Another strategy proceeds by recognizing that (3) is equivalent to the discrete convolution [1]

$$f_{\mathbf{k}_0} = (c * p)_{\mathbf{k}_0} \qquad \forall \mathbf{k}_0 \in Z^q. \tag{4}$$

It directly follows that the infinite sequence of coefficients $\{c_{\mathbf{k}}\}$ can be obtained by convolution of the infinite sequence $\{f_{\mathbf{k}}\}$ with the convolution-inverse $(p)^{-1}$, which is uniquely defined, and which does generally exist in the cases of interest. Convolving both sides of (4) by $(p)^{-1}$, we get that

$$c_{\mathbf{k}_0} = \left((p)^{-1} * f\right)_{\mathbf{k}_0} \qquad \forall \mathbf{k}_0 \in Z^q. \tag{5}$$

Since convolution is nothing else but filtering, (5) suggests that discrete filtering can be an alternative solution to matrix inversion for the determination of the sequence of coefficients $\{c_{\mathbf{k}}\}$ needed to enforce the desirable constraint (3). To derive the proper algorithm, we start by noting that the basis function φ is always symmetric in an imaging context. Thus, we can write the z-transform of $(p)^{-1}$ as

$$P^{-1}(z) = \frac{1}{p_0 + \sum_{k \in N_*} p_k (z^k + z^{-k})} \propto z^n \prod_{i=1}^{n} (z - z_i)^{-1} (z - z_i^{-1})^{-1}$$

where $\{z_i\}$ are n out of the $(2n)$ poles of $(p)^{-1}$; those are necessarily real and come in reciprocal pairs. Thus, the convolution inverse $(p)^{-1}$ can be decomposed in a series of n filter pairs, where each pair consists of a causal (with pole z_i) and of an anticausal (with pole z_i^{-1}) IIR filter. With suitable parameters, we can then apply a very efficient algorithm which leads to a recursive in-place implementation [19], [20]. Its computational load for the popular cubic B-spline is two additions and three multiplications per produced coefficient.

E. Reconciliation

Comparing (1) with (2), it appears that classical interpolation is a special case of generalized interpolation—with $(p)_{\mathbf{k}}^{-1} = \delta_{\mathbf{k}}$, $c_{\mathbf{k}} = f_{\mathbf{k}}$ and $\varphi = \varphi_{\text{int}}$. The converse is also true. To see why, we determine the interpolating φ_{int} from its noninterpolating counterpart φ by writing

$$\begin{aligned} f(\mathbf{x}) &= \sum_{\mathbf{k}_1 \in Z^q} \left((p)^{-1} * f\right)_{\mathbf{k}_1} \varphi(\mathbf{x} - \mathbf{k}_1) \\ &= \sum_{\mathbf{k}_1 \in Z^q} \sum_{\mathbf{k}_2 \in Z^q} (p)_{\mathbf{k}_2}^{-1} f_{\mathbf{k}_1 - \mathbf{k}_2} \varphi(\mathbf{x} - \mathbf{k}_1). \end{aligned}$$

Finally, the interpolating φ_{int} that is hidden behind a noninterpolating φ is

$$\varphi_{\text{int}}(\mathbf{x}) = \sum_{\mathbf{k} \in Z^q} (p)_{\mathbf{k}}^{-1} \varphi(\mathbf{x} - \mathbf{k}). \tag{6}$$

It is crucial to understand that this equivalence allows for the exact and efficient handling of an infinite-support, interpolating basis function φ_{int} by performing all operations with a finite-support, noninterpolating basis function φ.

III. Desirable Properties

A. Generalities

The price to pay for high-quality interpolation is computation time. For this reason, it is important to select a basis function that offers the best tradeoff. There are several aspects to consider. The most important deals with the support of φ_{int} or φ, which is a measure of the smallest interval in which we have that $\varphi(\mathbf{x}) \neq 0$. The larger the support, the more the computation time. Another important aspect is the quality of approximation inherent in the basis function. Other aspects involve the ease of analytical manipulation (when useful), the ease of computation, and the efficiency of the determination of the coefficients $c_{\mathbf{k}}$ when φ is not interpolating.

B. Separability

Consider (1) or (2) in multidimensions, with $q > 1$. Let us assume that the support of the interpolating φ_{int} or the noninterpolating φ is of size S^q. This large computational burden can only be reduced by imposing restrictions on φ. An easy and convenient way is to ask that the basis function be separable, as in

$$\varphi_{\text{sep}}(\mathbf{x}) = \prod_{i=1}^{q} \varphi(x_i) \qquad \forall \mathbf{x} = (x_1, x_2, \ldots, x_q) \in R^q.$$

The very beneficial consequence of this restriction is that the data can be processed in a separable fashion, line-by-line, column-by-column, and so forth. In particular, the determination of the interpolation coefficients needed for generalized interpolation is separable, too, because the form (2) is linear. For the rest of this paper, we concentrate on separable basis functions; we describe them and analyze them in one dimension, and we use the expression above to implement interpolation efficiently in a multidimensional context.

C. Symmetry

Preserving spatial relations is a crucial issue for any imaging system. Since interpolation can be interpreted as a convolution (or equivalently, filtering) operation, it is important that the response of the involved filter does not result in any phase degradation. This consideration translates into the well-known and desirable property of symmetry such that $\varphi(\mathbf{x}) = \varphi(-\mathbf{x})$ or $\varphi_{\text{int}}(\mathbf{x}) = \varphi_{\text{int}}(-\mathbf{x})$. Symmetry is satisfied by all basis functions considered here, at the possible minor and very localized exception of nearest-neighbor interpolation.

D. Regularity

Some authors insist that the regularity of the basis function is an important issue [16]. This may be true when differentiation of f is needed, but differentiating data more than, say, once or twice, is uncommon in everyday imaging applications. Often, at most the gradient is needed; thus, it is not really necessary to limit the choice of basis functions to those that have a high degree of regularity.

IV. Approximation Theory

A. Error Kernel

Since most clinical data are available once only, at a given resolution (or sampling step), there exist no absolute truth regarding the value of f between its samples $f_{\mathbf{k}}$. It is thus necessary to resort to mathematical analysis for the assessment of the quality of interpolation. The general principle is to define an interpolated function f_h as given by a set of samples that are h units apart and that satisfy

$$f_h(\mathbf{x}) = \sum_{\mathbf{k} \in Z^q} c_{\mathbf{k}} \varphi\left(\frac{1}{h}\mathbf{x} - \mathbf{k}\right) \qquad \forall \mathbf{x} \in R^q$$

with the interpolation constraint that $f_h(h\mathbf{k}) = f(h\mathbf{k})$ for all $\mathbf{k} \in Z^q$. The difference between $f_h(\mathbf{x})$ and $f(\mathbf{x})$ for all $\mathbf{x} \in R^q$ will then describe how fast the interpolated function f_h converges to the true function f when the samples that define f_h become more and more dense, or, in other words, when the sampling step h becomes smaller and smaller.

Let us perform the following experiment:

1) Take some arbitrary square-integrable function $f(\mathbf{x})$ and select a sampling step $h > 0$;
2) Create a set of samples $f(h\mathbf{k})$;
3) From this sequence, using either (1) or (2), build an interpolated function $f_h(\mathbf{x}) = \sum c_{\mathbf{k}} \varphi((1/h)\mathbf{x} - \mathbf{k})$;
4) Compare f with f_h using some norm, for example the mean-square (or L_2) norm $\varepsilon^2(h) = \|f - f_h\|_{L_2}^2 = \int_{-\infty}^{\infty} \ldots \int_{-\infty}^{\infty} (f(\mathbf{x}) - f_h(\mathbf{x}))^2 \, dx_1 \ldots dx_q$.

When the sampling step h gets smaller, more details of f can be captured; it is then reasonable to ask that the approximation error $\varepsilon(h)$ gets smaller, too. The following formula predicts the approximation error in the Fourier domain [21]–[23]

$$\eta^2(h) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}\left|\hat{f}(\boldsymbol{\omega})\right|^2 E_{\text{int}}(\boldsymbol{\omega}h)\,d\omega_1\ldots d\omega_q \tag{7}$$

where $\hat{f}(\boldsymbol{\omega})$ is the Fourier transform of the arbitrary function $f(\mathbf{x})$, and where E_{int} is an interpolation error kernel that depends on the basis function only, and that is given by

$$E_{\text{int}}(\boldsymbol{\omega}) = \frac{\left|\sum_{\mathbf{k}\in Z_*^q}\hat{\varphi}(\boldsymbol{\omega}+2\pi\mathbf{k})\right|^2 + \sum_{\mathbf{k}\in Z_*^q}|\hat{\varphi}(\boldsymbol{\omega}+2\pi\mathbf{k})|^2}{\left|\sum_{\mathbf{k}\in Z^q}\hat{\varphi}(\boldsymbol{\omega}+2\pi\mathbf{k})\right|^2}. \tag{8}$$

The equivalence $\varepsilon = \eta$ holds for bandlimited functions. For those functions f that would not belong to that class, the estimated error $\eta(h)$ must be understood as the average error over all possible sets of samples $f(h\mathbf{k}+\boldsymbol{\Delta})$, where $\boldsymbol{\Delta} = (\Delta_1, \Delta_2, \ldots, \Delta_q)$ is some phase term with $\Delta_i \in [0, h[$. This applies to dimensions $q \geq 1$ and to interpolating as well as noninterpolating basis functions [21]–[23]. In the restricting conditions where $q = 1$, for bandlimited functions f and when the basis function φ_{int} is interpolating, this error kernel reduces to the kernel proposed in [24].

B. Order of Approximation

A decrease in the sampling step h will result in a decrease of the argument of E_{int} in (7); thus, the error kernel must vanish at the origin to ensure that the approximation error disappears altogether. The vanishing rate is controlled by the approximation order L and the constant C_{int} such that

$$\eta^2(h) = (C_{\text{int}})^2 h^{2L}\left(\frac{1}{2\pi}\int_{-\infty}^{\infty}\left|\omega^L\hat{f}(\omega)\right|^2 d\omega\right) \quad \text{as } h\to 0$$

where the parenthesized expression is recognized as being the norm of the Lth derivative of the smooth function f we started from.

Finally, for a basis function of approximation order L, we get that

$$\eta(h) = \|f - f_h\|_{L_2} = C_{\text{int}}h^L\left\|f^{(L)}\right\|_{L_2} \qquad \text{as } h\to 0. \tag{9}$$

This result expresses the fact that we can associate to any φ a number L and a constant C_{int} such that the error of approximation ε predicted by η decreases like h^L, when h is sufficiently small. The number L is called the approximation order of the basis function φ; it gives a global estimate of how fast the approximation error decays when the sampling step gets finer. The constant $C_{\text{int}} > 0$ allows one to further rank the quality of those basis functions that would have the same order L, where a smaller C_{int} corresponds to a better φ. Nevertheless, it is clear from (9) that the decay of η is dominated by h^L rather than by C_{int} as soon as $h \leq C_{\text{int}}$, which frequently happens in the cases of interest. Thus, it is important to use C_{int} for ranking basis functions of identical order only; it would be inappropriate to consider C_{int} when comparing φ_1 with φ_2 if $L_1 \neq L_2$.

Equations (7) and (8) describe the evolution of the error for every possible sampling step h; thus, the error kernel E_{int} is a key element when it comes to the comparison of basis functions, not only near the origin, but over the whole Fourier axis. The error kernel E_{int} can be understood as a way to predict the approximation error when φ is used to interpolate a sampled version of the infinite-energy function $f(x) = \sin(\omega x)$. Being a single number, but being also loaded with relevant meaning, the approximation order L is a convenient summary of this whole curve.

C. Strang–Fix Equivalence

Suppose we are interested in just the approximation order L of a basis function φ, without caring much about the details of E_{int}. In this case, the explicit computation of (8) is not necessary. Instead, Strang–Fix [25] have proposed a series of conditions that are equivalent to (9). The interest of these equivalent conditions is that they can be readily tested. They are valid for all basis functions with sufficient decay—the sinc function is one of the very rare basis where these conditions are not satisfied. We mention three equivalent 1-D Strang–Fix conditions as

1) Lth order zeros in the Fourier domain

$$\begin{cases}\hat{\varphi}(0) = 1\\ \hat{\varphi}^{(n)}(2\pi k) = 0 \quad k\in Z_* \quad n\in[0, L-1]\end{cases}$$

2) Reproduction of all monomials of degree $n \in [0, L-1]$

$$\forall n\in[0, L-1] \quad \exists\{\ldots, c_{n,-1}, c_{n,0}, c_{n,1}, \ldots\} \quad \left|\sum_{k\in Z} c_{n,k}\varphi(x-k) = x^n\right.$$

3) Discrete moments

$$\sum_{k\in Z}(x-k)^n\varphi(x-k) = \mu_n \qquad \forall n\in[0, L-1]$$

where μ_n depends on n only.

Under mild hypothesis, any of these conditions is equivalent to $\varepsilon(h) \leq \text{Const}\times h^L\|f^{(L)}\|_{L_2}$. When $L = 1$, or, equivalently, when $n = 0$, these conditions are called the partition of unity, or the reproduction of the constant. More generally, the second Strang–Fix condition implies by linearity that, apart from technical details, a basis function of order L can reproduce any polynomial of degree $L-1$ or lower exactly. Thus, the approximation of data that are smooth at scale h will be close to the original for two equivalent reasons, since one can either analyze the quality of the approximation in terms of frequency contents, or in terms of polynomials. On one hand, there will be few high-frequencies, and the fact that the cardinal basis function φ_{int} may depart from sinc is mostly irrelevant. On the other hand, the Taylor expansion of a smoothly varying function is dominated by low-degree terms, and a polynomial of corresponding degree will capture the local behavior of the function in sufficient details.

TABLE I
PROPERTIES THAT APPLY TO $\varphi(x) = (\beta^{L-1} * u)(x) \leftrightarrow \hat{\varphi}(\omega) = (\mathrm{sinc}((1/2\pi)\omega))^L \hat{u}(\omega)$

	$\varphi(x)$	$u(x)$
Order	L	0
Regularity	C^m	C^{m-L}
Support	W	$W-L$
Degree	n	$n-L$

D. Decomposition Theorem

1) Theorem: Any uniform piecewise-polynomial basis function that possesses an order $L > 0$ can be expressed as the convolution of a B-spline of order L with a uniform piecewise-polynomial distribution $u(x)$ that has the properties shown in Table I.

The factorization of $\hat{\varphi}$ as the product of two terms, one of them being the Fourier transform of a B-spline of order L, was first presented in [25] without proof and without specifying the properties of the distribution u. A rigorous proof of an extended version of this theorem was later given in [26], in a more mathematically abstract and general context than required for this paper. Our actual formulation is aimed at fitting the present task and audience; the corresponding proof will appear in a forthcoming paper [27].

A direct corollary of the decomposition theorem is that the support of any function φ of order $L > 0$ must satisfy $W \geq L$, because no distribution u can have a negative support. However, it may happen that the support of u be null. In this case, u is necessarily a combination of Dirac pulses and their derivatives, which calls for an extension to negative values of the concepts of regularity and of degree: a distribution u is said to have regularity C^{-m} with $m \geq 0$ whenever its m-times integration yields a function U that has regularity C^0. For example, a discontinuous function—say, a rectangular pulse—has regularity C^{-1}, a Dirac pulse has regularity C^{-2}, the derivative $\delta^{(1)}$ of a Dirac pulse has regularity C^{-3}, and so on. The concept of degree is extended similarly.

This theorem is very relevant to the problem of determining the order of a piecewise-polynomial basis function: its Fourier transform can necessarily be factored as the product of a $(\mathrm{sinc}((1/2\pi)\omega))^L$ term with a remaining term $\hat{u}(\omega)$, where L is the order of the original piecewise-polynomial basis function. As a simple example, let us consider the B-spline of degree n which we call β^n. Then, we conclude from $\beta^n = \beta^n * \delta$ that $u = \delta$. In turn, the regularity of u is $C^{m-L} = C^{-2}$, its support is $W - L = 0$, and its degree is $n - L = -1$. Solving for these last three equations, we rederive the well-known fact that both the order L and the support W of a B-spline is one more than its degree n, while it has C^{n-1} regularity. Other examples are given in Appendix-B.

E. Piecewise-Polynomial Analysis

Uniform piecewise-polynomials form an important class of all possible basis functions. In particular, we shall see later that the basis functions that are optimal in a precise mathematical sense are all members of this class. For this reason, we feel important to develop analytical tools that ease their Fourier analysis. Let p_k be a polynomial piece

$$p_k(x) = \begin{cases} a_{k,0} + \sum_{k_0=1}^{n} a_{k,k_0} x^{k_0}, & x \in]x_k, x_{k+1}[\\ 0, & x \notin [x_k, x_{k+1}] \\ \frac{1}{2}\left(a_{k,0} + \sum_{k_0=1}^{n} a_{k,k_0} x^{k_0}\right), & x = x_k \vee x = x_{k+1} \end{cases} \quad (10)$$

where

k index of the piece;
n overall highest polynomial degree;
a_{k,k_0} polynomial coefficients.

By convention, the piece described by p_k is valid on the interval $[x_k, x_{k+1}]$; in the following analysis we assume that $x_{k+1} = x_k + 1$. The special behavior on the boundaries is such as to allow for basis functions that are pointwise symmetric, including ones with discontinuities (the boundaries are shared by two adjacent pieces).

We now decompose this piecewise-polynomial into elementary functions for which an expression for the Fourier transform is known. Let us first introduce the one-sided power function x_+^n as being given by

$$x_+^n = \begin{cases} 0, & n = 0 \wedge x < 0 \\ \frac{1}{2}, & n = 0 \wedge x = 0 \\ 1, & n = 0 \wedge x > 0 \\ x_+^0 x^n & n > 0. \end{cases} \quad (11)$$

Although x_+^n is not a finite-energy function, we can nevertheless express its Fourier transform as

$$x_+^n \overset{\text{Fourier}}{\leftrightarrow} \frac{n!}{(j\omega)^{n+1}} + j^n \pi \delta^{(n)}(\omega)$$

where $\delta^{(n)}$ is the nth derivative of the Dirac distribution δ. Using this basic building block, we now express the truncated power function $x_{[0,1]}^n$ as being given by

$$x_{[0,1]}^n = x_+^n - \sum_{k_1=0}^{n} \binom{n}{k_1} (x-1)_+^{k_1}.$$

This function satisfies $x_{[0,1]}^n = x^n$ for $x \in]0, 1[$ and $x_{[0,1]}^n = 0$ for $x \notin [0, 1]$. The boundaries are such that $x_{[a,b]}^n + x_{[b,c]}^n = x_{[a,c]}^n$, including at the particular argument $x = b$. The Fourier transform of this finite-energy function is

$$x_{[0,1]}^n \overset{\text{Fourier}}{\leftrightarrow} \frac{n! - \Gamma(n+1, j\omega)}{(j\omega)^{n+1}}$$

where $\Gamma(n+1, z)$ is the incomplete Γ-function given by the finite sum

$$\Gamma(n+1, z) = n! e^{-z} \sum_{k_1=0}^{n} \frac{z^{k_1}}{k_1!}.$$

Finally, we rewrite each polynomial piece in terms of truncated power functions. For this, we build a new set of polynomial coefficients as

$$b_{k,\,k_0} = a_{k,\,k_0} + \sum_{k_1=k_0+1} \binom{k_1}{k_0} a_{k,\,k_1}(x_0+k)^{k_1-k_0}.$$

Then, we observe that

$$p(x) = \sum_{k=0}^{m-1} p_k(x) = \sum_{k=0}^{m-1} q_k(x-x_0-k) \qquad (12)$$

where m is the number of polynomial pieces, and where q_k is a truncated polynomial given by

$$q_k(x) = \sum_{k_0=0}^{n} b_{k,\,k_0} x_{[0,\,1]}^{k_0}.$$

By contrast with (10), the important property of (12) is that both support membership and special treatment of the piece boundaries are now implicit. Letting $\hat{b}_{k_0}$ be the (2π)-periodic continuous Fourier transform of the discrete sequence of coefficients $b_{k,\,k_0}$

$$\hat{b}_{k_0}(\omega) = \sum_{k=0}^{m-1} b_{k,\,k_0} e^{-j\omega k}$$

we can finally express the Fourier transform of the piecewise-polynomial as

$$\hat{p}(\omega) = e^{-j\omega x_0} \sum_{k_0=0}^{n} \hat{b}_{k_0}(\omega) \frac{k_0! - \Gamma(k_0+1,\, j\omega)}{(j\omega)^{k_0+1}}.$$

F. Example

The analysis above facilitates the calculation of the error kernel (8) for many basis functions, because the latter are often piecewise-polynomials themselves. Moreover, it allows for an easy determination of the order L. For illustration purposes, we propose to analyze the quadratic Schaum basis function (results for many more cases are available in Appendix-B). Let this function be given by

$$\varphi(x) = \begin{cases} 0, & x < \frac{-3}{2} \\ \frac{-1}{16}, & x = \frac{-3}{2} \\ 1 + \frac{3}{2}x + \frac{1}{2}x^2, & \frac{-3}{2} < x < \frac{-1}{2} \\ \frac{9}{16}, & x = \frac{-1}{2} \\ 1 - x^2, & \frac{-1}{2} < x < \frac{1}{2} \\ \frac{9}{16}, & x = \frac{1}{2} \\ 1 - \frac{3}{2}x + \frac{1}{2}x^2, & \frac{1}{2} < x < \frac{3}{2} \\ \frac{-1}{16}, & x = \frac{3}{2} \\ 0, & \frac{3}{2} < x \end{cases}$$

which can be summarized by

$$[a_{k,\,k_0}] = \begin{bmatrix} 1 & \frac{3}{2} & \frac{1}{2} \\ 1 & 0 & -1 \\ 1 & \frac{-3}{2} & \frac{1}{2} \end{bmatrix} \qquad \text{with } x_0 = \frac{-3}{2},\ m = 3, \text{ and } n = 2.$$

From that, we build

$$[b_{k,\,k_0}] = \begin{bmatrix} \frac{-1}{8} & 0 & \frac{1}{2} \\ \frac{3}{4} & 1 & -1 \\ \frac{3}{8} & -1 & \frac{1}{2} \end{bmatrix}$$

and

$$\left[\hat{b}_{k_0}\right] = e^{-3j\omega/2}\left[\tfrac{1}{4}\, e^{j\omega/2}(3+\cos\omega - 2j\sin\omega)\,,\ 2j\sin\frac{\omega}{2},\ -2e^{j\omega/2}\sin^2\frac{\omega}{2}\right]$$

which results in

$$\begin{aligned} \hat{\varphi}(\omega) &= \frac{1}{4}\, e^{j\omega/2}(3+\cos\omega - 2j\sin\omega)\frac{1-e^{-j\omega}}{j\omega} \\ &\quad + 2j\sin\frac{\omega}{2}\,\frac{1-e^{-j\omega}(1+j\omega)}{(j\omega)^2} \\ &\quad - 2e^{j\omega/2}\sin^2\frac{\omega}{2}\,\frac{2-2e^{-j\omega}\left(1+j\omega-\frac{1}{2}\omega^2\right)}{(j\omega)^3} \\ &= \left(\frac{\sin\frac{\omega}{2}}{\frac{\omega}{2}}\right)^3 \left(1 + \frac{-1}{8}(j\omega)^2\right). \end{aligned}$$

From its explicit expression, we see that the quadratic Schaum basis function has the support $W = 3$ and the degree $n = 2$. Referring to the decomposition theorem, we determine from the expression of its Fourier transform that its order is $L = 3$; thus, the polynomial u resulting from factoring out a B-spline must have the support $W - L = 0$. It is made of the sum of a Dirac with unit weight (which corresponds to 1 in the Fourier domain), and of a second derivative of a Dirac with weight $(-1/8)$ [which corresponds to $(-1/8)(j\omega)^2$ in the Fourier domain]. Thus, in short notation we can write that $\varphi = \beta^2 - (1/8)\ddot{\beta}^2$. Also, the degree of u is $n - L = -1$ and it has $C^{r-L} = C^{-4}$ regularity, which implies that φ has C^{-1} regularity. This can be checked by inspection of the explicit expression of the quadratic Schaum basis function.

After some tedious algebra (which can be handled by most current symbolic manipulation software), the introduction of the expression above into (8) yields

$$E_{\text{int}}(\omega) = 2 - \frac{\sin^3\frac{\omega}{2}}{\left(\frac{\omega}{2}\right)^3}\left(2 + \left(\frac{\omega}{2}\right)^2\right) - \frac{17}{60}\sin^4\frac{\omega}{2}.$$

Taking the order $L = 3$ into consideration, the corresponding approximation constant of a quadratic Schaum basis function is finally given by

$$C_{\text{int}} = \lim_{\omega\to 0} \frac{\sqrt{E_{\text{int}}(\omega)}}{\omega^L} = \sqrt{\frac{407}{241\,920}} \cong 0.0410.$$

V. Specific Piecewise Polynominal Examples

A. Nearest-Neighbor

The basis function associated to nearest-neighbor interpolation is the simplest of all, since it is made of a square pulse. It satisfies the partition of unity, provided a slight asymmetry is introduced at the edges of the square pulse. Its expression is given by

$$\varphi^0(x) = \begin{cases} 0, & x < \frac{-1}{2} \\ 1, & \frac{-1}{2} \le x < \frac{1}{2} \\ 0, & \frac{1}{2} \le x. \end{cases}$$

The main interest of this basis function is its simplicity, which results in the most efficient of all implementations. In fact, for any coordinate $\mathbf{x}$ where it is desired to compute the value of the interpolated function f, there is only one sample $f_{\mathbf{k}}$ that contributes, no matter how many dimensions q are involved. The price to pay is a severe loss of quality.

B. B-Splines

There is a whole family of basis functions made of B-splines β^n. These functions are given by [28]

$$\beta^n(x) = \sum_{k=0}^{n+1} \frac{(-1)^k (n+1)}{(n+1-k)!k!} \left(\frac{n+1}{2} + x - k\right)_+^n \qquad \forall x \in R, \quad \forall n \in N$$

where $(x)_+^n$ is the one-sided power function defined in (11). From Table I, it is easy to conclude that these B-spline functions enjoy the maximal order of approximation for a given support; conversely, they enjoy the minimal support for a given order of approximation. In addition, they are maximally continuous. They have many other interesting properties as well [28], [29]; most fall outside the scope of this paper, except perhaps the fact that a B-spline derivative can be computed recursively by

$$\frac{d}{dx}\beta^n(x) = \beta^{n-1}\left(x + \frac{1}{2}\right) - \beta^{n-1}\left(x - \frac{1}{2}\right) \qquad n > 0.$$

Then, computing the exact gradient of a signal given by a discrete sequence of interpolation coefficients $\{c_k\}$ can be done as follows:

$$\begin{aligned}\frac{d}{dx} f(x) &= \sum_{k \in Z} c_k \frac{d}{dx}\beta^n(x-k) \\ &= \sum_{k \in Z} (c_k - c_{k-1}) \beta^{n-1}\left(x - k + \tfrac{1}{2}\right)\end{aligned}$$

where the n-times continuous differentiability of B-splines ensures that the resulting function is smooth when $n \geq 3$, continuous when $n \geq 2$, and bounded when $n \geq 1$.

- Degree $n = 0$: The B-spline of smallest degree $n = 0$ is almost identical to the nearest-neighbor basis function. They differ from one another only at the transition values, where we ask that β^0 be symmetric with respect to the origin (φ^0 is not), and at the same time that it satisfies the partition of unity. Thus, contrary to the nearest-neighbor case, it happens in some exceptional cases (evaluation at half-integers) that interpolation with β^0 requires the computation of the average between two samples.
- Degree $n = 1$: The B-spline function β^1 is also called linear interpolation. It enjoys a large popularity because the complexity of its implementation is very low, just above that of the nearest-neighbor; moreover, some consider that it satisfies Occam's razor principle by being the simplest interpolating basis function f one can think of that builds a continuous function out of a sequence of discrete samples $\{f_k\}$.
- Degrees $n > 1$: No B-spline β^n of degree $n > 1$ benefits from the property of being interpolating; thus, no such high-degree B-spline should ever be used in the context of Equation (1). Equation (2) must be used instead. Unfortunately, some authors have failed to observe this rule, which led them to claim that B-splines typically blur data. Therefore, such claims are misleading, even though they can be repeatedly found in the published literature. A cubic B-spline is often used in practice. Its expression is given by

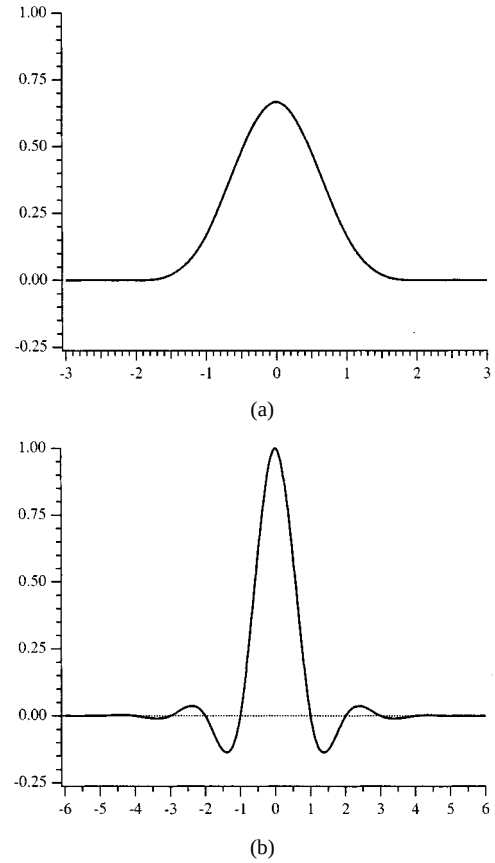


Fig. 1. B-spline of third degree. (a) Function shape. (b) Equivalent interpolant.

$$\beta^3(x) = \begin{cases} \frac{2}{3} - \frac{1}{2}|x|^2(2 - |x|), & 0 \leq |x| < 1 \\ \frac{1}{6}(2 - |x|)^3, & 1 \leq |x| < 2 \\ 0, & 2 \leq |x|. \end{cases}$$

This basis function is not interpolating. As explained in Equation (6), it is nonetheless possible to exhibit an infinite-support basis function $\varphi_{\text{int}} = \beta^3_{\text{card}}$ that allows one to build exactly the same interpolated function f. To give a concrete illustration of this fact, we show in Fig. 1 both the noninterpolating cubic B-spline β^3 along with its interpolating equivalent basis function. The latter is named a cubic cardinal spline β^3_{card}. Graphically, the B-spline looks similar to a Gaussian; this is not by chance, since a B-spline can be shown to converge to a Gaussian when its degree increases. Already for a degree as small as $n = 3$, the match is amazingly close since the maximal relative error between a cubic B-spline and a Gaussian with identical variance is only about 3.5%.

On the bottom part of Fig. 1, the cardinal spline displays decaying oscillations that are reminiscent of a sinc function. This is not by chance either, since a cardinal spline can be shown to

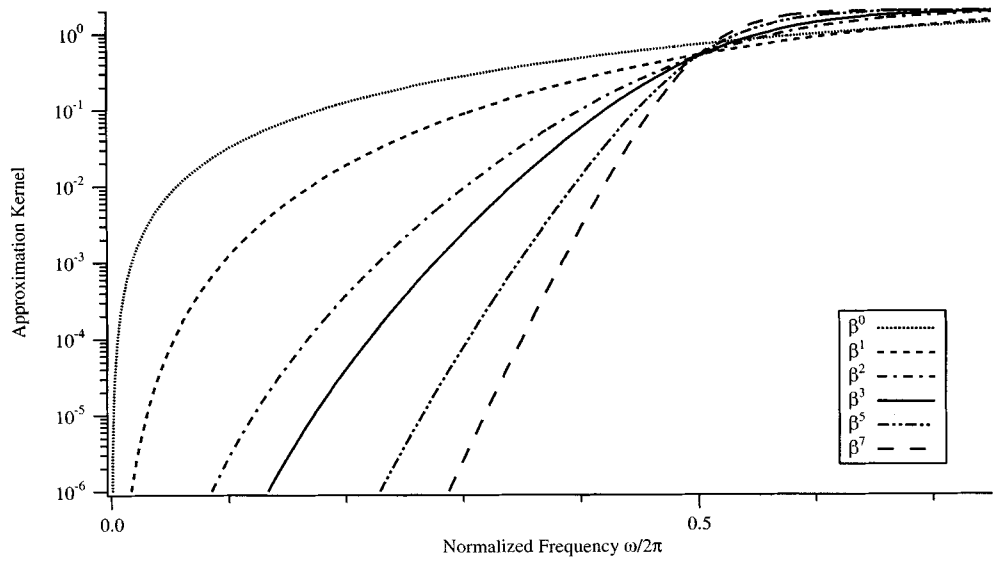


Fig. 2. B-spline basis functions. Approximation kernel for several degrees.

converge to a sinc function when its degree increases [30], [31]. Throughout the paper, we have given numerous reasons why it is more profitable to approximate a true sinc by the use of non-interpolating basis functions rather than by apodization. Even for moderate degrees, the spline-based approximations offers a sensible quality improvement over apodization for a given computational budget. In Fig. 2, we give the approximation kernel defined in (8) for B-splines of several degrees.

C. o-Moms

The family of functions that enjoy Maximal Order and Minimal Support is called Moms (or splines of minimal support in the terminology of [26]). It can be shown that any of these functions can be expressed as the weighted sum of a B-spline and its derivatives [26], [32], such that the distribution u of the decomposition theorem has a vanishing support

$$\text{Moms}^n(x) = \beta^n(x) + \sum_{m=1}^{n} c_m \frac{d^m}{dx^m}\beta^n(x).$$

B-splines are those Moms functions that are maximally differentiable. There exist several other interesting classes in this family; in particular, the o-Moms functions [32] are such that their least-squares approximation constant C_{opt} is minimal. This constant is closely related to C_{int}, because we can write that

$$C_{\text{opt}} = \frac{1}{L!}\sqrt{\sum_{k\in Z_*}\left|\hat{\varphi}^{(L)}(2\pi k)\right|^2}$$

while we have that

$$C_{\text{int}} = \frac{1}{L!}\sqrt{\sum_{k\in Z}\left|\hat{\varphi}^{(L)}(2\pi k)\right|^2} = \sqrt{C_{\text{opt}}^2 + \left|\frac{1}{L!}\hat{\varphi}^{(L)}(0)\right|^2}$$
$$= \lim_{\omega\to 0}\frac{\sqrt{E_{\text{int}}(\omega)}}{\omega^L}.$$

The constant C_{opt} corresponds to the least-squares error kernel given by

$$E_{\text{opt}}(\boldsymbol{\omega}) = 1 - \frac{|\hat{\varphi}(\boldsymbol{\omega})|^2}{\sum_{\mathbf{k}\in Z^q}|\hat{\varphi}(\boldsymbol{\omega}+2\pi\mathbf{k}|^2}.$$

After substitution of E_{int} by E_{opt} in (7), one is left with an approximation error that corresponds to the least-squares projection of a signal f onto the set of shifted basis functions φ. This error is the smallest achievable in a quadratic sense, and requires that the set of discrete constraints expressed in (3) be replaced by a continuous constraint [33]. While it is also possible to find Moms functions such that their asymptotic approximation C_{int} constant reaches its absolute minimum, experiments have shown that these particular functions are in fact less favorable for interpolation than are o-Moms.

The o-Moms functions are indexed by their polynomial degree n and they are symmetric. Their knots are identical to those of the B-spline they descend from. Moreover, they have the same support as β^n, that is, $n+1$; this support is the smallest achievable for an approximation order $n+1$. Although their order is identical to that of a B-spline of same degree, their approximation error constant C_{int} is much smaller. By construction, their approximation error constant C_{opt} is the smallest possible.

These functions are not interpolating; thus, they need a way to compute the sequence of coefficients $\{c_k\}$ required for the implementation of (2). Fortunately, the same algorithm than for the B-splines can be used. The o-Moms functions of degree zero and one are identical to β^0 and β^1, respectively. The o-Moms functions of higher degree can be determined recursively in Fourier [32]; we give here the expression for $n=3$

$$\text{o-Moms}^3(x) = \beta^3(x) + \frac{1}{42}\frac{d^2}{dx^2}\beta^3(x)$$
$$= \begin{cases} \frac{1}{2}|x|^3 - |x|^2 + \frac{1}{14}|x| + \frac{13}{21}, & 0 \le |x| < 1 \\ \frac{-1}{6}|x|^3 + |x|^2 - \frac{85}{42}|x| + \frac{29}{21}, & 1 \le |x| < 2 \\ 0, & 2 \le |x|. \end{cases}$$

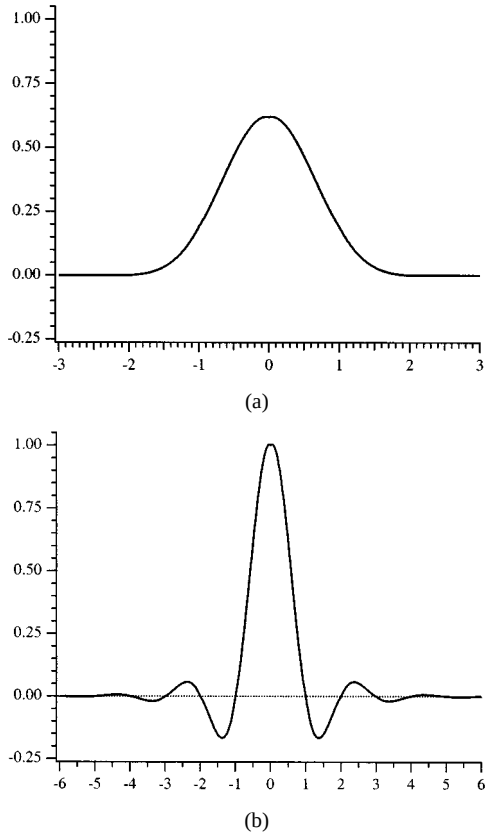


Fig. 3. o-Moms of third degree. (a) Function shape. (b) Equivalent interpolant.

As a curiosity, we point out in Figs. 3 and 4 that this basis function has a slope discontinuity at the origin; thus, its regularity is C^0 (in addition, it has other slope discontinuities for $|x| = 1$ and $|x| = 2$). It is nevertheless optimal in the sense described.

D. Schaum

Like the o-Moms, the pseudo-Lagrangian basis functions proposed by Schaum can also be represented as a weighted sum of B-splines and of their even-order derivatives [17]. They have same order and same support as B-splines and o-Moms. Their main interest is that they are interpolating. Their disadvantage with respect to both o-Moms and B-splines is a worse approximation constant C_{int}: for example, considering an approximation order $L = 4$, the value reached by a cubic o-Moms is $C_{\text{int}} = 0.000627$; the constant for the cubic spline is more than twice with $C_{\text{int}} = 0.00166$, while the cubic Schaum loses an additional order of magnitude with $C_{\text{int}} = 0.01685$. They are discontinuous for even degrees, and are C^0 for odd degrees.

E. Dodgson

Dodgson has proposed a basis function built out of quadratic polynomials [12]. Unfortunately, it fails to be a member of the family of Moms. While it has a support $W = 3$, its approximation order is $L = 2$, which is no higher than in the linear interpolation case that has the smaller support $W = 2$. In return, its constant of approximation C_{int} is about five times smaller than in the linear case.

F. Keys

The principal reason for the popularity enjoyed by the family of Keys functions is the fact that they perform better than linear interpolation, while being interpolating [15]. Thus, they do not require the determination of interpolation coefficients, and the classical equation (1) can be used. These functions are made of piecewise cubic polynomials and depend on a parameter a. Their general expression is

$$\varphi_a(x) = \begin{cases} (a+2)|x|^3 - (a+3)|x|^2 + 1, & 0 \le |x| < 1 \\ a|x|^3 - 5a|x|^2 + 8a|x| - 4a, & 1 \le |x| < 2 \\ 0, & 2 \le |x|. \end{cases}$$

Comparing this expression with that of the cubic spline, it is apparent that both require the computation of piecewise-polynomials of the same support. However, their approximation order differ: the best order that can be reached by a Keys function is $L = 3$, for the special value $a = (-1/2)$, while the cubic spline has order $L = 4$. This extra order for β^3 comes at the cost of the computation of a sequence $\{c_k\}$, for use in (2). However, by using a recursive filtering approach, this cost can be made negligible. Altogether, the gain in speed offered by Keys function is not enough to counterbalance the loss in quality when comparing β^3 with $\varphi_{(-1/2)}$. Moreover, the regularity of Keys is C^1, which is one less than that of the cubic spline.

G. Meijering

Meijering *et al.* [16] have designed piecewise-polynomial basis functions that all have the same approximation order $L = 3$ and that are interpolating. What is distinguished between them are their support and their polynomial degree n. The benefit of higher values for these characteristics is to allow for a stronger regularity C^m; also, the constant of approximation C_{int} gets smaller. With regard to approximation properties nevertheless, and even with a support as large as $W = 8$, the Meijering basis function of degree $n = 7$ is no better than an o-Moms function of the much smaller support $W = 3$ and much smaller degree $n = 2$.

We give in Fig. 5 the error kernel of several basis functions that all have the identical order of approximation $L = 3$. The corresponding basis functions have a wide range of polynomial degree (from $n = 2$ up to $n = 7$). They also have a wide range of support (from $W = 3$ up to $W = 8$), which translates into some very large differences with respect to computational requirements. Nevertheless, the approximation quality is essentially the same for all these kernels (compare to the broader selection of performances to choose from in Fig. 2). One possible selection criterion is the constant of approximation C_{int}; we give it in Table II.

H. German

German [13] has proposed an interpolating basis function with an approximation order $L = 5$. Its support is larger than necessary, which leaves some freedom to the designer. In this case, this freedom has been used to increase the regularity to C^1. Note that the quartic B-spline enjoys the same order $L = 5$

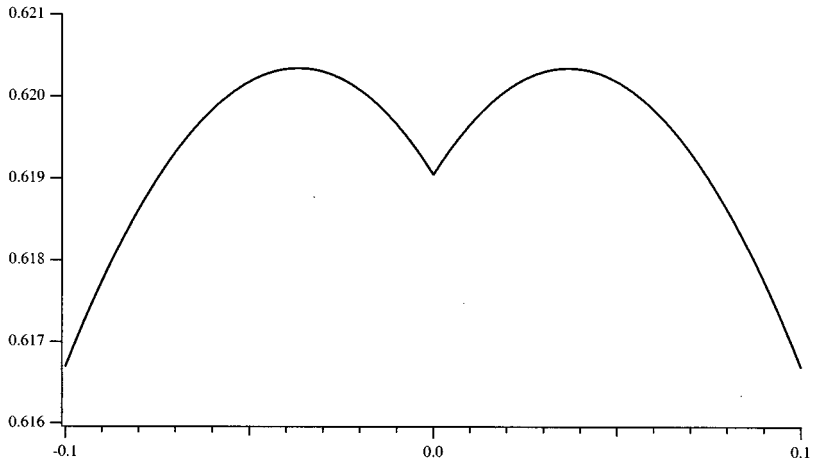


Fig. 4. o-Moms of third degree (central part).

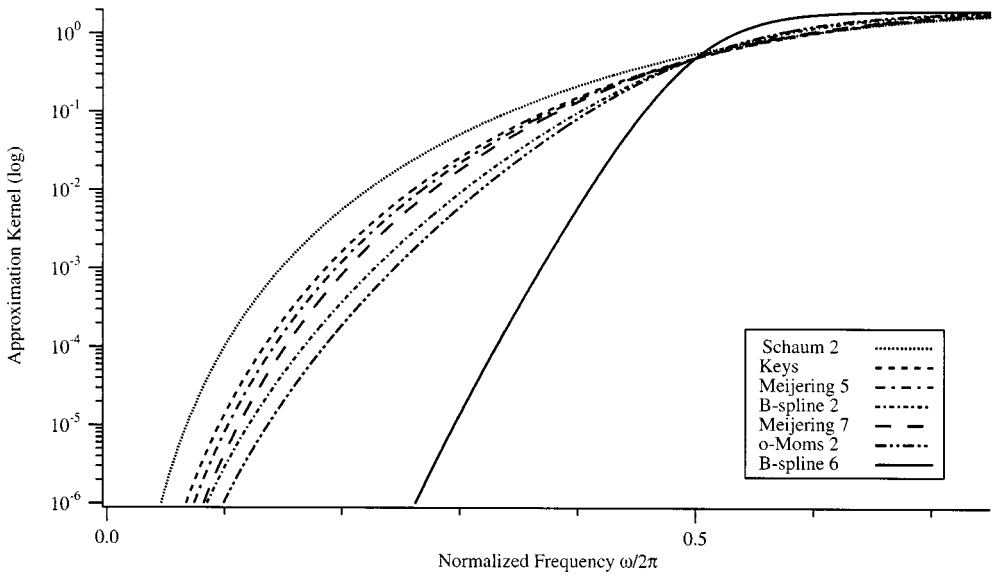


Fig. 5. Various basis functions of same approximation order $L = 3$. A sextic B-spline of degree $n = 6$ and order $L = 7$ has been added for comparison, because it offers much better performance than any of the third-order basis functions, while it has essentially the same experimental computational cost as one of them (septimic Meijering).

TABLE II
CONSTANT OF APPROXIMATIONS FOR VARIOUS BASIC FUNCTIONS WITH $L = 3$

	C_{int}	n	W
Quadratic Schaum	$\sqrt{\frac{407}{241920}} \cong 0.0410$	2	3
Keys ($a = \frac{-1}{2}$)	$\sqrt{\frac{1}{7560}} \cong 0.0115$	3	4
Quintic Meijering	$\sqrt{\frac{35}{608256}} \cong 0.0075$	5	6
Quadratic B-spline	$\sqrt{\frac{1}{30240}} \cong 0.0057$	2	3
Septimic Meijering	$\sqrt{\frac{311647}{15478779888}} \cong 0.0044$	7	8
Quadratic o-Moms	$\sqrt{\frac{1}{100800}} \cong 0.0031$	2	3

for a shorter support, while it has the maximal regularity C^4. Moreover, its approximation is much better than that of German, since we have that $C_{\text{int}} = \sqrt{(823/79833600)} \cong 0.00321$ for the fifth-order German basis function, and that $C_{\text{int}} = \sqrt{(1/47900160)} \cong 0.00014$ for the quartic B-spline.

VI. SINC-BASED BASIS FUNCTIONS

A. Bandlimitedness

For a long time, sinc interpolation—which corresponds to ideal filtering—has been the holy grail of geometric operations. Nowadays, researchers acknowledge that, while interpolation can be realized under special circumstances (e.g., translation of a periodic signal through discrete Fourier transform operations), in general, it can only be approximated. Another drawback of the function is that it decays only slowly, which tends to spread out ringing-associated effects.

The sinc function provides error-free interpolation of the bandlimited functions. There are two difficulties associated with this statement. The first one is that the class of bandlimited functions represents but a tiny fraction of all possible functions; moreover, they often give a distorted view of the physical reality in an imaging context—think of the transition air/matter in a CT scan: as far as classical physics is concerned, this transi-

tion is abrupt and cannot be expressed as a bandlimited function. Further, there exist obviously no way at all to perform any kind of antialiasing filter on physical matter (before sampling). Most patients would certainly object to any attempt of the sort.

The second difficulty is that the support of the sinc function is infinite. An infinite support is not too bothering, provided an efficient algorithm can be found to implement interpolation with another equivalent basis function that has a finite support. This is exactly the trick we used with B-splines and o-Moms. Unfortunately, no function can be at the same time bandlimited and finite-support, which precludes any hope to find an equivalent finite-support basis function φ for use in (6). Thus, the classical solution is simply to truncate sinc itself by multiplying it with a finite-support window; this process is named apodization. A large catalog of apodizing windows is available in [14], along with their detailed analysis.

B. Dirichlet Apodization

Dirichlet apodization is perhaps the laziest approach, since the window of total width W is simply an enlarged version of β^0, which requires no more computational effort than a test to indicate support membership. The apodized basis function is given by

$$\mathrm{sinc}_W^D(x) = \frac{\sin(\pi x)}{\pi x}\,\beta^0\left(\frac{x}{W}\right)$$

where W is an even integer. The price to pay for laziness is bad quality. First, the regularity of this function is low since it is not differentiable. More important, its approximation order is as bad as $L = 0$. This means that a reduction of the sampling step does not necessarily result in a reduction of the interpolation error.

C. Hanning Apodization

Apodization, being defined as the multiplication of a sinc function by some window, corresponds in Fourier to a convolution-based construction. The Hanning window is one out of several attempts to design a window that has favorable properties in Fourier. The result is

$$\mathrm{sinc}_W^H(x) = \mathrm{sinc}_W^D(x)\left(\frac{1}{2} + \frac{1}{2}\cos\left(\frac{2\pi x}{W}\right)\right).$$

With $L = 0$, the order of approximation of Hanning interpolation is no better than that of Dirichlet interpolation; the constant C_{int} is significantly improved, though. Whereas it was $C_{\mathrm{int}} = 0.1076$ for sinc_4^D, it is now $C_{\mathrm{int}} = 0.0153$ for sinc_4^H. Being continuously differentiable, Hanning is also more regular than Dirichlet.

VII. Cost-Performance Analysis

A. Generalities

The single most influential parameter that dictates the computational cost is the size W of the support of the basis function φ. Second to it, we find the cost of evaluating $\varphi(x-k)$ for a series of arguments $(x-k)$. Last, there is a small additional cost associated to the computation of interpolation coefficients $c_{\mathbf{k}}$ in the context of (2). We want to mention here that the importance of this overhead is negligible, especially in the practical case where it needs to be computed once only before several interpolation operations are performed. This situation arises often in the context of iterative algorithms, and in the context of interactive imaging; moreover, it disappears altogether when the images are stored directly as a set of coefficients $\{c_{\mathbf{k}}\}$ rather than a set of samples $\{f_{\mathbf{k}}\}$. Thus, we will ignore this overhead in the theoretical performance analysis that follows.

B. Cost

Let us assume that we want to compute the interpolated value $f(\mathbf{x})$ of a 2-D image f at argument $\mathbf{x}$, using a separable basis function of finite-support W. For each output value, we first need to perform W multiplications and additions to compute $c_{k_2} = \sum c_{k_1,k_2}\varphi(x_1 - k_1)$. This computation is embedded in a similar outer loop that is also executed W times. Finally, we need W^2 multiplications and additions in 2-D; more generally, we need $2W^q$ operations in q dimensions, where we consider that a multiplication is worth an addition.

To this cost, one must add (qW)-times the cost of the evaluation of the separable basis function. When the latter is piecewise-polynomial, on average we need $W/4$ tests to determine which of the polynomial piece to use. Once a polynomial is selected, evaluation by the Horner's scheme further requires $(W-1)$ multiplications and additions under the favorable hypothesis that $n = W - 1$. Putting these results together, the magnitude of the global cost of all operations for a piecewise-polynomial basis function is $O(W^q)$, more precisely $2W^q + qW((9/4)W - 2)$.

In the case of the sinc family, each evaluation requires the computation of a transcendental function and the multiplication by the apodization window. This cost does not depend on the support W. Hence, the magnitude of the global cost of all operations for an apodized sinc basis function is $O(W^q)$; more precisely, $2W^q + \lambda qW$ where $\lambda = 12$ operations are spent in the evaluation of a Hanning apodization window (we consider that the transcendental functions sine or cosine are worth two multiplications each), $\lambda = 9$ for a Bartlet window and $\lambda = 6$ in the Dirichlet case.

It follows from these theoretical considerations that the support for which a sinc-based basis function comes at a lesser computational cost than a polynomial-based one, is about $W = 6$ in two dimensions. For images or volumes, where $q > 1$, it is important to realize that this result does not imply that the use of sinc functions is more efficient than that of polynomials, because sinc typically requires a much larger support than polynomials to reach the same quality.

C. Performance

In Fig. 6, we present a comparison of the error kernel for several basis functions of same support $W = 4$. It includes cubic B-spline, cubic o-Moms, and cubic Schaum as examples of polynomial functions, and Dirichlet and Hanning as examples of apodized sinc. We observe that the sinc-based basis functions do not reproduce the constant. Since most of the energy of virtually any image is concentrated at low frequencies, it is easy to predict that these functions will perform poorly when compared

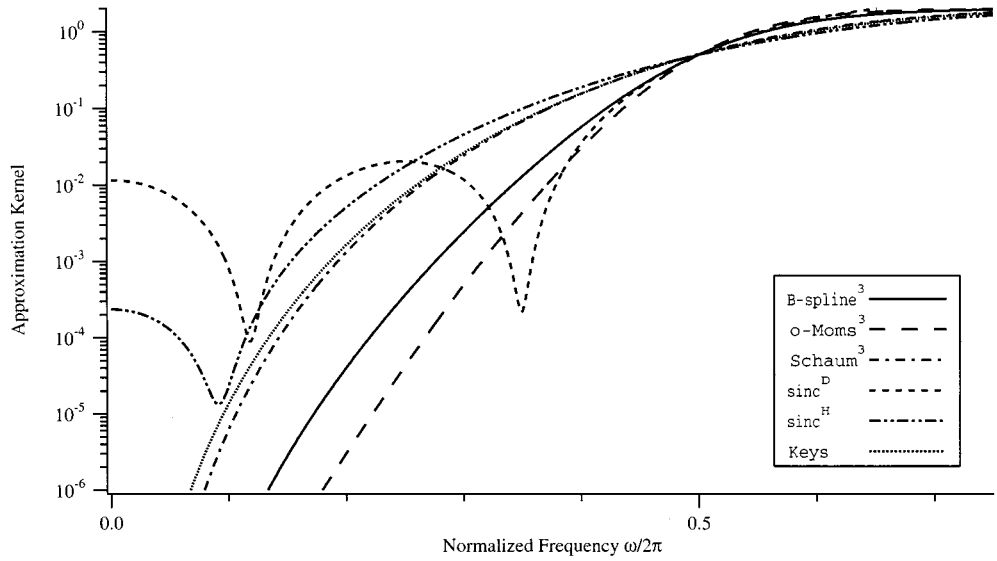


Fig. 6. Comparison of basis functions of same support $W = 4$.

TABLE III
PERFORMANCE INDEX FOR WHITE NOISE

Basis function	ε^2 (dB)	L
Nearest-neighbor	5.94	1
Linear	9.23	2
Quadratic Schaum	9.83	3
Dodgson	9.98	2
Cubic Schaum	10.99	4
Keys	11.02	3
Quintic Meijering	11.19	3
Septimic Meijering	11.39	3
German	11.75	5
Quadratic spline	12.11	3
Quadratic o-Moms	12.46	3
Cubic spline	13.14	4
Cubic o-Moms	14.03	4

to polynomial-based basis functions. We will see in the experimental section that this prediction is fulfilled; for now, we limit our analysis to that of the more promising polynomial cases.

On the grounds of (7), we can select a specific function f to sample-and-interpolate, and we can predict the amount of resulting squared interpolation error. As a convenient simplification, we now assume that this function f has a constant-value power spectrum; in this case, it is trivial to obtain the interpolation error by integrating the curves in Fig. 6. Table III gives the resulting values as a signal-to-noise ratio (SNR) expressed in dB, where the integration has been performed numerically over the domain $\omega \in [-\pi, \pi]$, and where we have set the natural sampling interval $h = 1$. These results have been obtained by giving the same weight to all frequencies up to Nyquist's rate; if low frequencies are considered more important than high frequencies, then the order of approximation L and its associated constant C_{int} are the most representative quality indexes.

D. Tradeoff

Fig. 7 presents the combination of the theoretical results of computation time and of those of interpolation quality. In order to show the versatility of the approximation kernel, we have changed the function f from bandlimited white noise $\hat{f} = 1$ (Table III) to a function that corresponds to a Markov model which captures the correlation of data such that $\int_{-\infty}^{\infty} f(x)f(x+\tau)\,dx = \rho^{-|\tau|}$. Thus, we have that

$$\left|\hat{f}(\omega)\right|^2 = \frac{-2\log(\rho)}{\omega^2 + (\log(\rho))^2}.$$

With $\rho = 0.9$, this power spectrum is representative of a large variety of real images [34]. We have performed its integration against the approximation error kernel over the domain $\omega \in [-\pi, \pi]$ only, such as to conform to bandlimitedness.

VIII. EXPERIMENTS

A. Protocol

To magnify the degradation that results from interpolation, we adopt the following strategy that has for goal the highlighting of—as opposed to the avoidance of—the deleterious effects of interpolation: we apply a succession of $r = 15$ rotations of $(2\pi/15) = 24°$ each to some image, such that the output of any given step is used as input for the next step. We then compare the initial image to the final output. To limit potential boundary effects, we perform the final comparison on some central portion of the image only. Also, we avoid any interaction between interpolation and quantization by performing all computations in a floating-point format. Fig. 8 shows the central portion of the image we want to rotate.

B. Nearest-Neighbor and Linear Interpolation

Since the circular pattern of Fig. 8 is characterized by a radial sinusoidal chirp with higher frequencies in the center than in the periphery, the effect of the interpolation-associated filtering can be read visually as increases or—more often—decreases in the

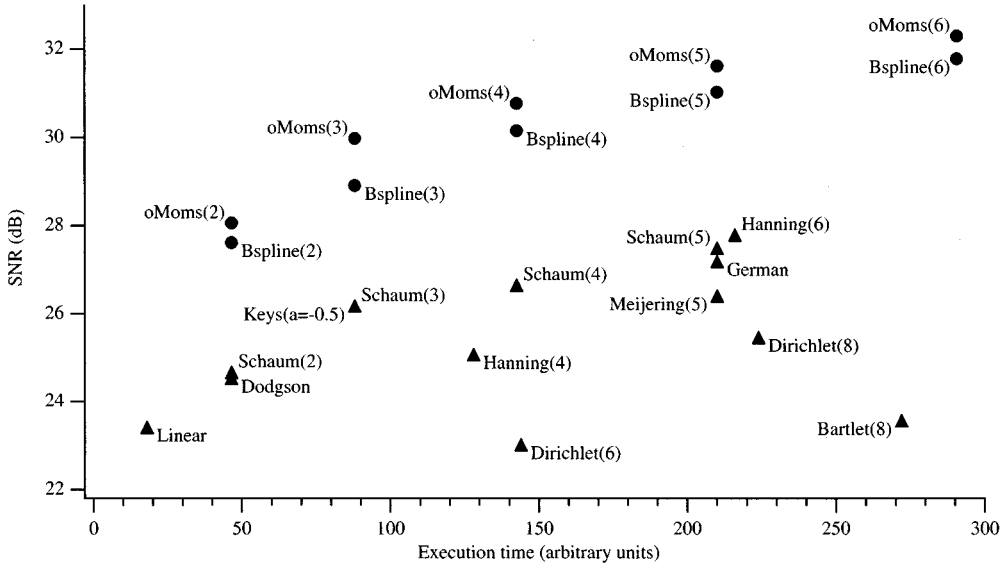
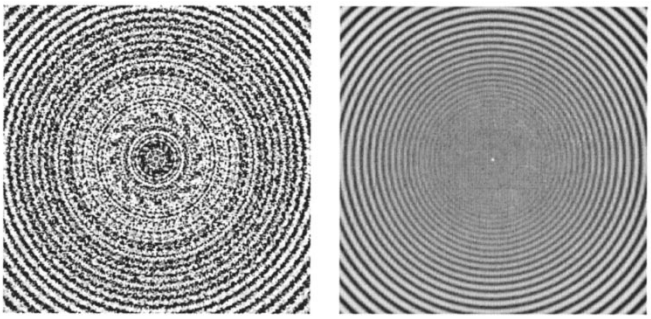


Fig. 7. Theoretical performance index for a Markov-like power spectrum. Triangles correspond to interpolating basis functions, circles to noninterpolating ones.

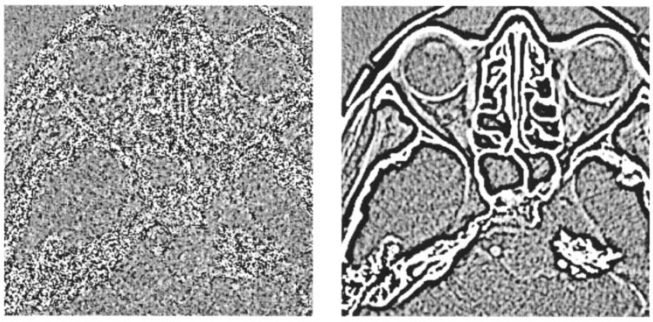
(a) (b) (c)

Fig. 8. Central portion of test images. (a) CT scan. (b) Lena. (c) Synthetic circular pattern.



(a) (b)

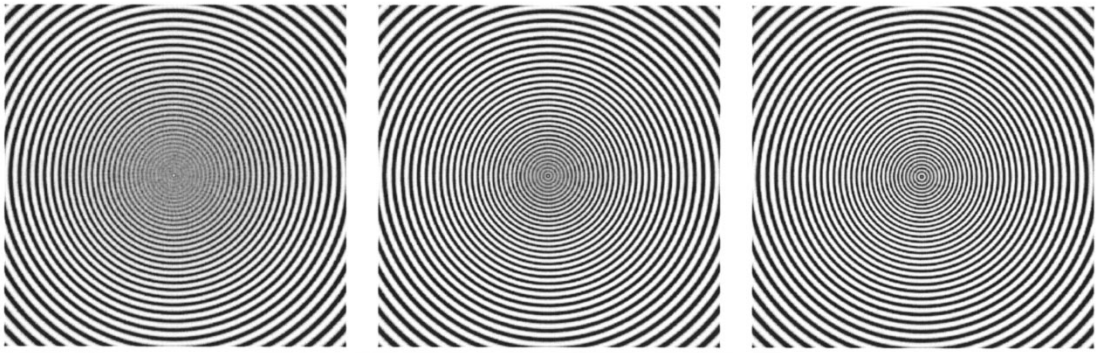
Fig. 9. Rotation of a circular pattern. (a) Nearest-neighbor. (b) Linear interpolation.



(a) (b)

Fig. 10. Difference between the original and the several-times rotated image of a CT scan. Positive values are light and negative values are dark. A zero difference is represented by middle gray. (a) Nearest-neighbor interpolation. (b) Linear interpolation.

degree of modulation of those spatial frequencies. Fig. 9 shows the effect of the rotation experiment when using the two most commonly found interpolants. Nearest-neighbor interpolation results in a lot of data shuffling, and the image is highly cluttered. Linear interpolation results in the loss of high frequencies, which corresponds to strong blurring. These losses cannot be compensated; they correspond to the prediction made in Fig. 2, according to which linear interpolation (equivalently, β^1) performs poorly when compared to other cases. Fig. 10 shows the differential effect of the rotation experiment on the CT scan. In this case, we represent error images—the result of subtracting the rotated image from the original one. We observe that the data shuffling aspect of nearest-neighbor interpolation results

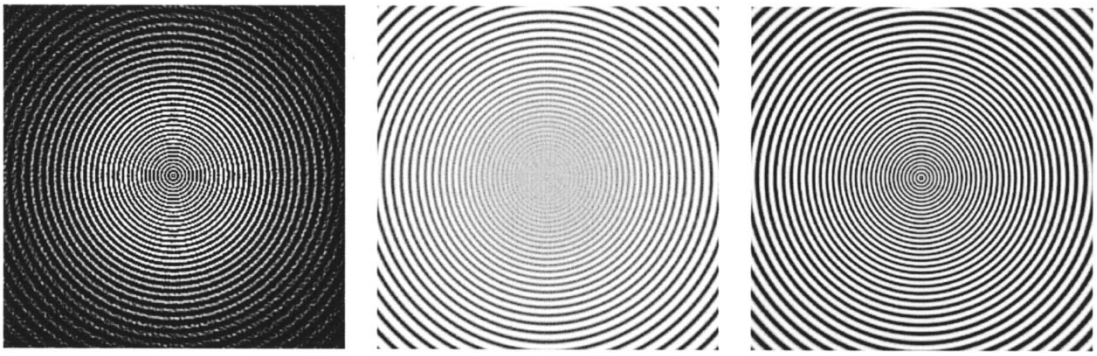


(a) (b) (c)

Fig. 11. Rotation of a circular pattern. (a) Keys. (b) Cubic spline. (c) Cubic o-Moms.

(a) (b) (c)

Fig. 12. Difference image for the CT scan. (a) Keys. (b) Cubic spline. (c) Cubic o-Moms.



(a) (b) (c)

Fig. 13. Rotation of a circular pattern. (a) Dirichlet $(W = 4)$. (b) Hanning $(W = 4)$. (c) Sextic spline ($W = 7$, less computation time than Hanning with $W = 4$).

in widespread noise, while the low-pass aspect of linear interpolation results in artifacts mostly near edges.

C. Cubic Interpolation

Figs. 11 and 12 propose three basis functions of identical support which have essentially the same computational cost. On the left, despite the use of the optimal parameter $a = (-1/2)$, Keys offers the poorest visual performance since the central part of the figure is blurred. In addition, close inspection (particularly on a monitor screen) discloses blocking artifacts that betray themselves as moiré patterns. Those are absent with cubic spline and cubic o-Moms interpolation, although patterns unrelated to interpolation may eventually be present on paper, in reason of the dithering process inherent in printing these figures. More important, cubic spline interpolation results in less blurring, and cubic o-Moms in even less. Fig. 11 shows the resulting image for the circular pattern, while Fig. 12 shows the difference image for the CT scan.

D. Sinc-Based Interpolation

Fig. 13(a) and (b) shows the result of using two different truncated and apodized approximations of sinc, where the support is the same as in the functions of Figs. 11 and 12. The test images of Fig. 8 have a nonnull average; since an apodized version of sinc does not reproduce this constant value faithfully, each incremental rotation results in the drift of the average value of the

TABLE IV
RESULTS IN NUMERICAL FORM

Basis Function	Time [s]	CT [dB]	Circular [dB]	Lena [dB]
Nearest-neighbor	**0.24**	**19.44**	**4.96**	**18.57**
Linear	**0.36**	**24.21**	**7.13**	**21.98**
Dodgson	**0.52**	**28.11**	**9.49**	**24.23**
Quadratic Schaum	**0.70**	**39.88**	**16.33**	**28.79**
(Cubic spline	*0.88*	*50.77*	*23.22*	*31.98)*
(Cubic o-Moms	*0.88*	*62.05*	*32.76*	*34.29)*
Keys	**0.90**	**38.74**	**15.00**	**28.16**
Cubic Schaum	**0.92**	**37.51**	**14.09**	**27.61**
Quadratic spline	**0.95**	**45.52**	**19.65**	**30.56**
Quadratic o-Moms	**0.95**	**46.71**	**20.37**	**30.88**
Quartic Schaum	**1.11**	**48.64**	**21.29**	**31.08**
Cubic o-Moms	**1.17**	**62.05**	**32.76**	**34.29**
Cubic spline	**1.17**	**50.77**	**23.22**	**31.98**
German	**1.33**	**16.66**	**15.89**	**16.48**
Quintic Schaum	**1.35**	**46.17**	**19.07**	**30.13**
Quintic Meijering	**1.35**	**39.71**	**15.65**	**28.53**
Quartic spline	**1.38**	**60.07**	**30.25**	**33.90**
Quartic o-Moms	**1.40**	**62.62**	**33.36**	**34.53**
Sinc Dirichlet W=4 (-)	**1.45**	**0.39**	**-0.54**	**0.34**
Sextic Schaum	**1.63**	**54.57**	**25.02**	**32.35**
Quintic spline	**1.64**	**63.61**	**35.01**	**34.81**
Quintic o-Moms	**1.67**	**65.15**	**39.06**	**35.40**
Sinc Dirichlet W=4 (+)	**1.70**	**-4.04**	**-27.73**	**-12.48**
Sinc Bartlet W=4 (-)	**1.72**	**0.41**	**0.38**	**0.41**
Septimic Meijering	**1.90**	**40.92**	**16.41**	**28.94**
Sextic spline	**1.92**	**65.40**	**40.17**	**35.54**
Sinc Bartlet W=4 (+)	**1.97**	**39.09**	**15.27**	**28.32**
Sextic o-Moms	**1.97**	**65.97**	**43.82**	**35.96**
Sinc Hamming W=4 (-)	**2.08**	**18.12**	**12.58**	**17.66**
Sinc Hanning W=4 (-)	**2.08**	**6.81**	**7.13**	**6.76**
Sinc Dirichlet W=6 (-)	**2.09**	**-15.21**	**-13.62**	**-15.24**
Sinc Hamming W=4 (+)	**2.33**	**33.18**	**12.35**	**26.50**
Sinc Hanning W=4 (+)	**2.33**	**29.30**	**10.16**	**24.81**
Sinc Dirichlet W=6 (+)	**2.46**	**19.34**	**5.69**	**19.36**
Sinc Bartlet W=6 (-)	**2.50**	**1.08**	**1.03**	**1.08**
Sinc Bartlet W=6 (+)	**2.87**	**39.27**	**20.77**	**31.10**
Sinc Hamming W=6 (-)	**3.04**	**24.83**	**22.11**	**24.06**
Sinc Hanning W=6 (-)	**3.05**	**19.61**	**18.59**	**19.32**

image. This drift manifests itself as images that appear too dark (negative drift) or too light (positive drift). We conclude that sinc performs extremely poorly when compared to other basis functions of the same support, and not only drift of the mean value, but also both blocking and excessive blurring artifacts are present.

Even though the support of the sinc-based basis functions presented in Fig. 13 is short, the associated computation time is already substantial. Fig. 13(c) shows that much higher quality can be achieved in less time with a more efficient, noninterpolating basis function (here, sextic B-spline). Fig. 13(c) is visually indistinguishable from Fig. 8(c).

E. Discussion

Table IV presents in succinct form the numeric results of these experiments, along with some additional ones. In particular, we also provide the results for the standard Lena test image, and for the synthetic circular test pattern of Fig. 8. The execution time is given in seconds and corresponds to the duration of a single rotation of a square image 512 pixels on a side. The computer is a Power Macintosh 9600/350 and the measure of the SNR is defined as

$$\mathrm{SNR} = 10\log\left(\sum_{\mathbf{k}\in Z^2} f_{\mathbf{k}}^2 \Bigg/ \sum_{\mathbf{k}\in Z^2} (f_{\mathbf{k}} - g_{\mathbf{k}})^2\right)$$

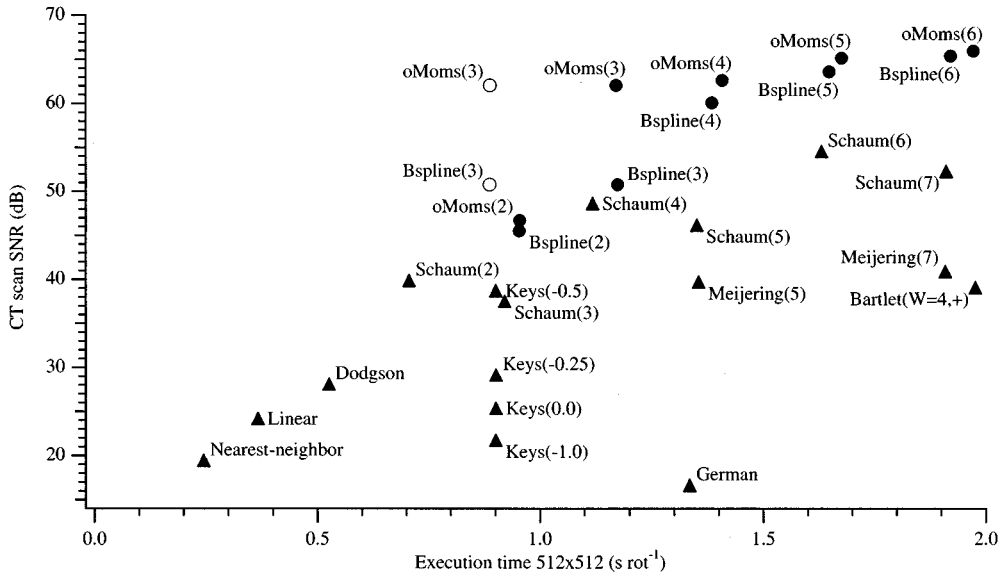


Fig. 14. Summary of the main experimental results for the CT scan. Triangles correspond to interpolating basis functions, circles to noninterpolating ones. The hollow circles give the computational time for an accelerated implementation.

where f is the original (nonnull average) data, and where g is given by the r-times chaining of the rotation.

These results point out some of the difficulties associated with the analysis of the performance of a basis function φ. For example, the computation time should ideally depend on the number of mathematical operations only. In reality, the optimization effort put into implementing each variation with one basis function or another, has also some influence. For instance, we have a fast implementation of the cubic spline and of the cubic o-Moms (in italic in Table IV and in white circles in Fig. 14) that runs in shorter time than the normal implementation (in bold in Table IV and in black circles in Fig. 14). We have, nevertheless, shown the result of all implementations because this corresponds to a somewhat unified level of optimization. The time spent for the determination of the interpolation coefficients—with a filtering algorithm implemented according to Section II-D—is included in the reported results.

We have provided results for two classes of sinc functions. The class labeled (-) represents traditional apodization, while the class labeled (+) represents an apodized-normalized version. In the latter case, we have modified the basis function as

$$\mathrm{sinc}_W^w(x) = \frac{\dfrac{\sin(\pi x)}{\pi x} w\left(\dfrac{x}{W}\right)}{\displaystyle\sum_{k\in Z} \frac{\sin(\pi(x-k))}{\pi(x-k)} w\left(\frac{x-k}{W}\right)}$$

where w is the original apodization window. The purpose of this operation is to restore some better approximation order to the sinc-based cases, at the cost of additional processing. As can be seen in Table IV, this is successful only for large supports and incurs an unbearably long computation time. Meanwhile, the resulting quality is obviously suboptimal when compared to the much better piecewise-polynomial cases.

Fig. 14 proposes a graphic summary of the most relevant results (CT scan, quality better than 20 dB, execution time shorter than 2 s). It is interesting to compare this figure with Fig. 7; the similarity between them confirms that our theoretical ranking of basis functions was justified. The difference between the interpolation methods is more pronounced in the experimental case because it has been magnified by the number of rotations performed.

IX. Conclusion

We have analyzed two approaches to the exact interpolation of data given by regular samples. In classical interpolation, the basis functions must be interpolating, while noninterpolating basis functions are allowed in generalized interpolation. We have tried to dispel the too-commonly held belief according to which noninterpolating functions (typically, cubic B-splines) should be avoided. This misconception, present in many books or reports on interpolation, arose because practitioners have sometimes attempted to use noninterpolating basis functions without the prefiltering step that is required to achieve a consistent implementation of the generalized interpolation. We have provided a unified framework for the theoretical analysis of the performance of both methods. We have applied this analysis to specific cases that involve piecewise-polynomial functions as well as sinc-based interpolants. We have performed 2-D experiments that support the 1-D theory.

We conclude from both theoretical and practical concerns that the most important index of quality is the approximation order of the basis function, its support being the most important parameter with respect to efficiency. Thus, the class of Moms functions, stands apart as the best achievable compromise between quality and speed. We have observed that many formerly proposed basis functions, such as Dodgson, Keys, Meijering, German, and any of the apodized versions of a sinc, do not belong to this class. Experiments have confirmed that these basis functions do indeed perform poorly. In particular, no sinc-based interpolation results in an acceptable quality with regard to its computational demand. In addition, they are difficult to handle analytically, which leads to unnecessary

complications for simple operations such as differentiation or integration.

The more favorable class of Moms functions can be further divided into subclasses, the most relevant being B-splines, Schaum, and o-Moms. Of those three, the Schaum functions are the only representatives that are interpolating. Nevertheless, experiments have shown that this strong constraint is detrimental to the performance; we observe that the time spent in computing the interpolation coefficients required by B-splines and o-Moms is a small, almost negligible investment that offers a high payoff in terms of quality. For this reason, we discourage the use of Schaum and advocate generalized interpolation instead, with noninterpolating basis functions such as B-splines and o-Moms.

Finally, comparing B-splines with o-Moms, we conclude that the lack of regularity of the latter makes them less suitable than B-splines for imaging problems that require the computation of derivatives, for example to perform operations such as edge detection or image-based optimization (e.g., snake contouring, registration). These operations are very common in medical imaging. Thus, despite a poorer objective result than o-Moms, B-splines are very good candidates for the construction of an image model. Moreover, they enjoy additional properties such as easy analytical manipulation, several recursion relations, the m-scale relation (of great importance for wavelets, a domain that has strong links with interpolation [29], [35]), minimal curvature for cubic B-splines, easy extension to inexact interpolation (smoothing splines, least-squares [33]), simplicity of their parametrization (a single number—their degree—is enough to describe them), and possible generalization to irregular sampling.

APPENDIX I

A. Mirror Extension

We want to extend a finite, discrete signal $\{f_0, \ldots, f_{N-1}\}$ of length N into an infinite-length, discrete signal by using a mirror extension. To obtain an explicit sample value for any argument $k \in Z$, we write $f(k) = f(s(q_k))$, where

$$s(q_k) = \begin{cases} q_k, & 0 \le q_k < N \\ 2N-2-q_k, & N \le q_k < 2N-2 \end{cases}$$

and where

$$q_k = \begin{cases} 0, & N = 1 \\ |k| - (2N-2)\left\lfloor \dfrac{|k|}{2N-2} \right\rfloor, & 1 < N. \end{cases}$$

B. Piecewise-Polynomial Basis

We give here a technical summary of the principal characteristics of some uniform piecewise-polynomial basis functions.[1] We draw the attention of the reader on the following facts:

- the higher the approximation order, the better the quality;
- for basis functions of identical approximation order, the smaller the approximation constant, the better the quality. It is, however, meaningless to compare approximation constants associated to basis functions that would differ in approximation order;
- the smaller the support, the less the computational burden;
- never use a noninterpolating basis function in the context of (1). Use (2) instead, and determine the interpolation coefficients as explained in Section II-D.

[1]Only those error kernels that have an expression with a manageable size are reported.

1) First-Order B-spline (Symmetric Nearest-Neighbor):

Piecewise-polynomial representation
$[a]_{1\times 1} = [1]$, $x_0 = -1/2$.
Degree $n = 0$, regularity C^{-1}, support $W = 1$, approximation constant $C_\varphi = \sqrt{1/12} \cong 0.288$.
Order $L = 1$, interpolating, decomposition $\hat{u}(\omega) = 1$.
Error kernel

$$E_{\text{int}}(\omega) = 2 - 2\,\frac{\sin\dfrac{\omega}{2}}{\dfrac{\omega}{2}}.$$

2) Second-Order B-spline (Linear):

Piecewise-polynomial representation

$$[a]_{2\times 2} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \qquad x_0 = -1.$$

Degree $n = 1$, regularity C^0, support $W = 2$, approximation constant $C_\varphi = \sqrt{1/120} \cong 0.0912$.
Order $L = 2$, interpolating, decomposition $\hat{u}(\omega) = 1$.
Error kernel

$$E_{\text{int}}(\omega) = 2 - 2\,\frac{\sin^2\dfrac{\omega}{2}}{\left(\dfrac{\omega}{2}\right)^2} - \frac{2}{3}\sin^2\frac{\omega}{2}.$$

3) Second-Order Dodgson (Quadratic):

Piecewise-polynomial representation

$$[a]_{3\times 3} = \begin{bmatrix} \frac{3}{2} & \frac{5}{2} & 1 \\ 1 & 0 & -2 \\ \frac{3}{2} & \frac{-5}{2} & 1 \end{bmatrix}, \quad x_0 = \frac{-3}{2}.$$

Degree $n = 2$, regularity C^0, support $W = 3$, approximation constant $C_\varphi = \sqrt{1/320} \cong 0.0559$.
Order $L = 2$, interpolating.
Decomposition

$$\hat{u}(\omega) = 2\,\frac{\sin\dfrac{\omega}{2}}{\dfrac{\omega}{2}} - \cos\frac{\omega}{2}.$$

Error kernel

$$E_{\text{int}}(\omega) = 2 - \frac{\sin^2\dfrac{\omega}{2}}{\left(\dfrac{\omega}{2}\right)^2}\left(4\,\frac{\sin\dfrac{\omega}{2}}{\dfrac{\omega}{2}} - 2\cos\frac{\omega}{2}\right) - \left(\frac{2}{5} - \frac{1}{15}\cos\omega\right)\sin^2\frac{\omega}{2}.$$

4) Third-Order Schaum (Quadratic):

Piecewise-polynomial representation

$$[a]_{3\times 3} = \begin{bmatrix} 1 & \frac{3}{2} & \frac{1}{2} \\ 1 & 0 & -1 \\ 1 & \frac{-3}{2} & \frac{1}{2} \end{bmatrix}, \qquad x_0 = \frac{-3}{2}.$$

Degree $n = 2$, regularity C^{-1}, support $W = 3$, approximation constant $C_\varphi = \sqrt{407/241\,920} \cong 0.0410$.
Order $L = 3$, interpolating, decomposition $\hat{u}(\omega) = 1 + (1/2)(\omega/2)^2$.
Error kernel

$$E_{\text{int}}(\omega) = 2 - \frac{\sin^3 \frac{\omega}{2}}{\left(\frac{\omega}{2}\right)^3}\left(2 + \left(\frac{\omega}{2}\right)^2\right) - \frac{17}{60}\sin^4\frac{\omega}{2}.$$

5) Third-Order Keys (Cubic):

Piecewise-polynomial representation

$$[a]_{4\times 4} = \begin{bmatrix} 2 & 4 & \frac{5}{2} & \frac{1}{2} \\ 1 & 0 & \frac{-5}{2} & \frac{-3}{2} \\ 1 & 0 & \frac{-5}{2} & \frac{3}{2} \\ 2 & -4 & \frac{5}{2} & \frac{-1}{2} \end{bmatrix}, \qquad x_0 = -2.$$

Degree $n = 3$, regularity C^1, support $W = 4$, approximation constant $C_\varphi = \sqrt{1/7560} \cong 0.0115$.
Order $L = 3$, interpolating.
Decomposition

$$\hat{u}(\omega) = 3\,\frac{\sin\frac{\omega}{2}}{\frac{\omega}{2}} - 2\cos\frac{\omega}{2}.$$

Error kernel

$$E_{\text{int}}(\omega) = 2 - 2\,\frac{\sin^3\frac{\omega}{2}}{\left(\frac{\omega}{2}\right)^3}\left(3\,\frac{\sin\frac{\omega}{2}}{\frac{\omega}{2}} - 2\cos\frac{\omega}{2}\right) - \left(\frac{16}{35} - \frac{2}{35}\cos\omega\right)\sin^4\frac{\omega}{2}.$$

6) Third-Order Meijering (Quintic):

Piecewise-polynomial representation

$$[a]_{6\times 6} = \begin{bmatrix} \frac{-243}{32} & \frac{-891}{64} & \frac{-81}{8} & \frac{-117}{32} & \frac{-21}{32} & \frac{-3}{64} \\ \frac{61}{32} & \frac{165}{64} & \frac{-7}{8} & \frac{-85}{32} & \frac{-21}{16} & \frac{-13}{64} \\ 1 & 0 & \frac{-17}{8} & 0 & \frac{63}{32} & \frac{27}{32} \\ 1 & 0 & \frac{-17}{8} & 0 & \frac{63}{32} & \frac{-27}{32} \\ \frac{61}{32} & \frac{-165}{64} & \frac{-7}{8} & \frac{85}{32} & \frac{-21}{16} & \frac{13}{64} \\ \frac{-243}{32} & \frac{891}{64} & \frac{-81}{8} & \frac{117}{32} & \frac{-21}{32} & \frac{3}{64} \end{bmatrix},$$

$$x_0 = -3.$$

Degree $n = 5$, regularity C^3, support $W = 6$, approximation constant $C_\varphi = \sqrt{35/608256} \cong 0.00758$.
Order $L = 3$, interpolating.
Decomposition (see (A-1) at the bottom of the page).

7) Third-Order B-spline (Quadratic):

Piecewise-polynomial representation

$$[a]_{3\times 3} = \begin{bmatrix} \frac{9}{8} & \frac{3}{2} & \frac{1}{2} \\ \frac{3}{4} & 0 & -1 \\ \frac{9}{8} & \frac{-3}{2} & \frac{1}{2} \end{bmatrix}, \qquad x_0 = \frac{-3}{2}.$$

Degree $n = 2$, regularity C^1, support $W = 3$, approximation constant $C_\varphi = \sqrt{1/30240} \cong 0.00575$.
Order $L = 3$, noninterpolating, decomposition $\hat{u}(\omega) = 1$.
Error kernel

$$E_{\text{int}}(\omega) = 2 - \frac{\sin^3\frac{\omega}{2}}{\left(\frac{\omega}{2}\right)^3}\left(\frac{8}{3+\cos\omega} + \frac{7}{30}\,\frac{\omega^3\sin\frac{\omega}{2}}{(3+\cos(\omega))^2}\right).$$

8) Third-Order Meijering(Septimic):

Piecewise-polynomial representation (see (A-2) at the bottom of the next page).
Degree $n = 7$, regularity C^5, support $W = 8$, approximation constant $C_\varphi = \sqrt{311647/15478779888} \cong 0.00448$.
Order $L = 3$, interpolating.
Decomposition (see (A-3) at the bottom of the next page).

9) Third-Order o-Moms (Quadratic):

Piecewise-polynomial representation

$$[a]_{3\times 3} = \begin{bmatrix} \frac{137}{120} & \frac{3}{2} & \frac{1}{2} \\ \frac{43}{60} & 0 & -1 \\ \frac{137}{120} & \frac{-3}{2} & \frac{1}{2} \end{bmatrix}, \qquad x_0 = \frac{-3}{2}.$$

Degree $n = 2$, regularity C^{-1}, support $W = 3$, approximation constant $C_\varphi = \sqrt{1/100800} \cong 0.00314$.
Order $L = 3$, noninterpolating, decomposition $\hat{u}(\omega) = 1 - (1/60)\omega^2$.
Error kernel

$$E_{\text{int}}(\omega) = 2 - \frac{\sin^3\frac{\omega}{2}}{\left(\frac{\omega}{2}\right)^3}\left(\frac{120 - 2\omega^2}{43 + 17\cos\omega} + \frac{85}{2}\,\frac{\omega^3\sin\frac{\omega}{2}}{(43+17\cos\omega)^2}\right).$$

$$\hat{u}(\omega) = \frac{3}{4}\,\frac{95\sin\left(\frac{\omega}{2}\right) + 15\sin\left(\frac{3\omega}{2}\right) - 67\omega\cos\left(\frac{\omega}{2}\right) - 3\omega\cos\left(\frac{3\omega}{2}\right)}{\omega^3} \tag{A-1}$$

10) Fourth-Order Schaum (Cubic):

Piecewise-polynomial representation

$$[a]_{4\times 4} = \begin{bmatrix} 1 & \frac{11}{6} & 1 & \frac{1}{6} \\ 1 & \frac{1}{2} & -1 & \frac{-1}{2} \\ 1 & \frac{-1}{2} & -1 & \frac{1}{2} \\ 1 & \frac{-11}{6} & 1 & \frac{-1}{6} \end{bmatrix}, \quad x_0 = -2.$$

Degree $n = 3$, regularity C^0, support $W = 4$, approximation constant $C_\varphi = \sqrt{103/362880} \cong 0.0168$.
Order $L = 4$, interpolating, decomposition $\hat{u}(\omega) = 1 + \frac{1}{6}\omega^2$.
Error kernel

$$E_{\text{int}}(\omega) = 2\frac{\sin^4\frac{\omega}{2}}{\left(\frac{\omega}{2}\right)^4}\left(2 + \frac{1}{3}\omega^2\right) - \left(\frac{524}{945} - \frac{62}{945}\cos\omega\right)\sin^4\frac{\omega}{2}.$$

11) Fourth-Order B-spline (Cubic):

Piecewise-polynomial representation

$$[a]_{4\times 4} = \begin{bmatrix} \frac{4}{3} & 2 & 1 & \frac{1}{6} \\ \frac{2}{3} & 0 & -1 & \frac{-1}{2} \\ \frac{2}{3} & 0 & -1 & \frac{1}{2} \\ \frac{4}{3} & -2 & 1 & \frac{-1}{6} \end{bmatrix}, \quad x_0 = -2.$$

Degree $n = 3$, regularity C^2, support $W = 4$, approximation constant $C_\varphi = \sqrt{1/362880} \cong 0.00166$.
Order $L = 4$, noninterpolating, decomposition $\hat{u}(\omega) = 1$.
Error kernel

$$E_{\text{int}}(\omega) = 2 - \frac{\sin^4\frac{\omega}{2}}{\left(\frac{\omega}{2}\right)^4}\left(\frac{12 + \frac{1}{35}\omega^4 + \left(6 - \frac{1}{280}\omega^4\right)\cos\omega}{(2+\cos\omega)^2}\right).$$

12) Fourth-Order o-Moms (Cubic):

Piecewise-polynomial representation

$$[a]_{4\times 4} = \begin{bmatrix} \frac{29}{21} & \frac{85}{42} & 1 & \frac{1}{6} \\ \frac{13}{21} & \frac{-1}{14} & -1 & \frac{-1}{2} \\ \frac{13}{21} & \frac{1}{14} & -1 & \frac{1}{2} \\ \frac{29}{21} & \frac{-85}{42} & 1 & \frac{-1}{6} \end{bmatrix}, \quad x_0 = -2.$$

Degree $n = 3$, regularity C^0, support $W = 4$, approximation constant $C_\varphi = \sqrt{1/2540160} \cong 0.000627$.
Order $L = 4$, noninterpolating, decomposition $\hat{u}(\omega) = 1 - (1/42)\omega^2$.
Error kernel

$$E_{\text{int}}(\omega) = 2 - \frac{\sin^4\frac{\omega}{2}}{\left(\frac{\omega}{2}\right)^4} \cdot \left(\frac{\left(546 - 13\omega^2 + \frac{1}{12}\omega^4\right) + \left(336 - 8\omega^2 + \frac{73}{120}\omega^4\right)\cos\omega}{(13 + 8\cos\omega)^2}\right).$$

13) Fifth-Order German (Quartic):

Piecewise-polynomial representation

$$[a]_{8\times 5} = \begin{bmatrix} \frac{-1}{3} & \frac{-5}{18} & \frac{-11}{44} & \frac{-1}{144} & 0 \\ \frac{-1}{3} & \frac{-1}{36} & \frac{5}{18} & \frac{7}{48} & \frac{1}{48} \\ \frac{13}{6} & \frac{145}{36} & \frac{17}{8} & \frac{29}{144} & \frac{-1}{16} \\ 1 & 0 & \frac{-335}{144} & \frac{-185}{144} & \frac{1}{24} \\ 1 & 0 & \frac{-335}{144} & \frac{185}{144} & \frac{1}{24} \\ \frac{13}{6} & \frac{-145}{36} & \frac{17}{8} & \frac{-29}{144} & \frac{-1}{16} \\ \frac{-1}{3} & \frac{1}{36} & \frac{5}{18} & \frac{-7}{48} & \frac{1}{48} \\ \frac{-1}{3} & \frac{5}{18} & \frac{-11}{44} & \frac{1}{144} & 0 \end{bmatrix}, \quad x_0 = -3.$$

$$[a]_{8\times 8} = \begin{bmatrix} \frac{9088}{867} & \frac{49984}{2601} & \frac{13064}{867} & \frac{5680}{867} & \frac{8875}{5202} & \frac{923}{3468} & \frac{213}{9248} & \frac{71}{83232} \\ \frac{114355}{3468} & \frac{1983989}{20808} & \frac{99572}{867} & \frac{521045}{6936} & \frac{301115}{10404} & \frac{45703}{6936} & \frac{79}{96} & \frac{403}{9248} \\ \frac{259}{3468} & \frac{-109739}{20808} & \frac{-243}{17} & \frac{-33145}{2312} & \frac{-5815}{867} & \frac{-9583}{6936} & \frac{-477}{9248} & \frac{1127}{83232} \\ 1 & 0 & \frac{-3611}{1734} & 0 & \frac{16775}{10404} & 0 & \frac{-22039}{27744} & \frac{-22013}{83232} \\ 1 & 0 & \frac{-3611}{1734} & 0 & \frac{16775}{10404} & 0 & \frac{-22039}{27744} & \frac{22013}{83232} \\ \frac{259}{3468} & \frac{109739}{20808} & \frac{-243}{17} & \frac{33145}{2312} & \frac{-5815}{867} & \frac{9583}{6936} & \frac{-477}{9248} & \frac{-1127}{83232} \\ \frac{114355}{3468} & \frac{-1983989}{20808} & \frac{99572}{867} & \frac{-521045}{6936} & \frac{301115}{10404} & \frac{-45703}{6936} & \frac{79}{96} & \frac{-403}{9248} \\ \frac{9088}{867} & \frac{-49984}{2601} & \frac{13064}{867} & \frac{-5680}{867} & \frac{8875}{5202} & \frac{-923}{3468} & \frac{213}{9248} & \frac{-71}{83232} \end{bmatrix}, \quad x_0 = -4 \tag{A-2}$$

$$\hat{u} = \frac{7\left(8594\sin\left(\frac{\omega}{2}\right) + 3769\sin\left(\frac{3\omega}{2}\right) + 71\sin\left(\frac{5\omega}{2}\right)\right) - \omega\left(62512\cos\left(\frac{\omega}{2}\right) - 8313\cos\left(\frac{3\omega}{2}\right) - 71\cos\left(\frac{5\omega}{2}\right)\right)}{\frac{289}{5}\omega^5} \tag{A-3}$$

Degree $n = 4$, regularity C^1, support $W = 6$, approximation constant $C_\varphi = \sqrt{823/79833600} \cong 0.00321$.
Order $L = 5$, interpolating.
Decomposition

$$\hat{u}(\omega) = \left(1 - \frac{5}{9}\omega^2\right)\cos\left(\frac{\omega}{2}\right) - \omega\left(\frac{1}{36}\omega\cos\left(\frac{3\omega}{2}\right) - \frac{57}{36}\sin\left(\frac{\omega}{2}\right) - \frac{1}{12}\sin\left(\frac{3\omega}{2}\right)\right).$$

ACKNOWLEDGMENT

The authors thank the anonymous reviewers for having brought to their attention various sources for the demonstration of the decomposition theorem of Section IV-D.

REFERENCES

[1] E. Maeland, "On the comparison of interpolation methods," *IEEE Trans. Med. Imag.*, vol. 7, no. 3, pp. 213–217, Sept. 1988.
[2] J. L. Ostuni, A. K. S. Santha, V. S. Mattay, D. R. Weinberger, R. L. Levin, and J. A. Frank, "Analysis of interpolation effects in the reslicing of functional MR images," *J. Comput. Assist. Tomogr.*, vol. 21, no. 5, pp. 803–810, 1997.
[3] R. W. Parrot, M. R. Stytz, P. Amburn, and D. Robinson, "Toward statistically optimal interpolation for 3-D medical imaging," *IEEE Eng. Med. Biol.*, vol. 12, no. 5, pp. 49–59, Sept.–Oct. 1993.
[4] M. Haddad and G. Porenta, "Impact of reorientation algorithms on quantitative myocardial SPECT perfusion imaging," *J. Nucl. Med.*, vol. 39, no. 11, pp. 1864–1869, Nov. 1998.
[5] B. Migeon and P. Marche, "*In vitro* 3D reconstruction of long bones using B-scan image processing," *Med. Biol. Eng. Comput.*, vol. 35, no. 4, pp. 369–372, July 1997.
[6] M. R. Smith and S. T. Nichols, "Efficient algorithms for generating interpolated (zoomed) MR images," *Magn. Reson. Med.*, vol. 7, no. 2, pp. 156–171, June 1988.
[7] S. D. Fuller, S. J. Butcher, R. H. Cheng, and T. S. Baker, "Three-dimensional reconstruction of icosahedral particles—The uncommon line," *J. Struct. Biol.*, vol. 116, no. 1, pp. 48–55, Jan.–Feb. 1996.
[8] F. M. Weinhaus and V. Devarajan, "Texture mapping 3D models of real-world scenes," *ACM Comput. Surv.*, vol. 29, no. 4, pp. 325–365, Dec. 1997.
[9] T. Möller, R. Machiraju, K. Mueller, and R. Yagel, "Evaluation and design of filters using a Taylor series expansion," *IEEE Trans. Visual. Comput. Graph.*, vol. 3, no. 2, pp. 184–199, Apr.–June 1997.
[10] P. Thévenaz, U. E. Ruttimann, and M. Unser, "A pyramid approach to sub-pixel registration based on intensity," *IEEE Trans. Image Processing*, vol. 7, no. 1, pp. 27–41, Jan. 1998.
[11] C. R. Appledorn, "A new approach to the interpolation of sampled data," *IEEE Trans. Med. Imag.*, vol. 15, no. 3, pp. 369–376, June 1996.
[12] N. A. Dodgson, "Quadratic interpolation for image resampling," *IEEE Trans. Image Processing*, vol. 6, no. 9, pp. 1322–1326, Sept. 1997.
[13] I. German, "Short kernel fifth-order interpolation," *IEEE Trans. Signal Processing*, vol. 45, no. 5, pp. 1355–1359, May 1997.
[14] F. J. Harris, "On the use of windows for harmonic analysis with the discrete Fourier transform," *Proc. IEEE*, vol. 66, no. 1, pp. 51–83, Jan. 1978.
[15] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, no. 6, pp. 1153–1160, Dec. 1981.
[16] E. H. W. Meijering, K. J. Zuidervel, and M. A. Viergever, "Image reconstruction with symmetrical piecewise nth-order polynomial kernels," *IEEE Trans. Image Processing*, vol. 8, no. 2, pp. 192–201, Feb. 1999.
[17] A. Schaum, "Theory and design of local interpolators," *CVGIP: Graph. Models Image Processing*, vol. 55, no. 6, pp. 464–481, Nov. 1993.
[18] M.-L. Liou, "Spline fit made easy," *IEEE Trans. Comput.*, vol. C-25, no. 5, pp. 522–527, May 1976.
[19] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part I—Theory," *IEEE Trans. Signal Processing*, vol. 41, no. 2, pp. 821–832, Feb. 1993.
[20] ——, "B-spline signal processing: Part II—Efficient design and applications," *IEEE Trans. Signal Processing*, vol. 41, no. 2, pp. 834–848, Feb. 1993.
[21] T. Blu and M. Unser, "Approximation error for quasi-interpolators and (multi)-wavelet expansions," *Appl. Comput. Harmonic Anal.*, vol. 6, no. 2, pp. 219–251, Mar. 1999.
[22] ——, "Quantitative Fourier analysis of approximation techniques: Part I—Interpolators and projectors," *IEEE Trans. Signal Processing*, vol. 47, no. 10, pp. 2783–2795, Oct. 1999.
[23] ——, "Quantitative Fourier analysis of approximation techniques: Part II—Wavelets," *IEEE Trans. Signal Processing*, vol. 47, no. 10, pp. 2796–2806, Oct. 1999.
[24] S. K. Park and R. A. Schowengerdt, "Image sampling, reconstruction, and the effect of sample-scene phasing," *Appl. Opt.*, vol. 21, no. 17, pp. 3142–3151, Sept. 1982.
[25] G. Strang and G. Fix, "A Fourier analysis of the finite element variational method," in *Constructive Aspects of Functional Analysis*. Erice, Italy: Edizioni Cremonese, Centro Internazionale Matematico Estivo, June 27–July 7 1971, pp. 796–830.
[26] A. Ron, "Factorization theorems for univariate splines on regular grids," *Israel J. Math.*, vol. 70, no. 1, pp. 48–68, 1990.
[27] T. Blu, "Complete parametrization of piecewise polynomial interpolators,", submitted for publication.
[28] I. J. Schoenberg, "Contribution to the problem of approximation of equidistant data by analytic functions," *Quart. Appl. Math.*, vol. 4, no. 2, pp. 45–99 and 112–141, 1946.
[29] M. Unser, "Ten good reasons for using spline wavelets," in *Proc. SPIE, Wavelet Applications in Signal and Image Processing V*, vol. 3169, San Diego, CA, July 30–Aug. 1 1997, pp. 422–431.
[30] A. Aldroubi and M. Unser, "Sampling procedures in function spaces and asymptotic equivalence with Shannon's sampling theorem," *Num. Function Anal. Optim.*, vol. 15, no. 1 and 2, pp. 1–21, 1994.
[31] A. Aldroubi, M. Unser, and M. Eden, "Cardinal spline filters: Stability and convergence to the ideal sinc interpolator," *Signal Processing*, vol. 28, no. 2, pp. 127–138, Aug. 1992.
[32] T. Blu, P. Thévenaz, and M. Unser, "Minimum support interpolators with optimum approximation properties," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Chicago, IL, Oct. 4–7, 1998, pp. 242–245.
[33] M. Unser and I. Daubechies, "On the approximation power of convolution-based least squares versus interpolation," *IEEE Trans. Signal Processing*, vol. 45, no. 7, pp. 1697–1711, July 1997.
[34] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
[35] M. Unser and A. Aldroubi, "Polynomial splines and wavelets—A signal processing perspective," in *Wavelets—A Tutorial in Theory and Applications*, C. K. Chui, Ed. San Diego, CA: Academic, 1992, pp. 91–122.