Linear Interpolation Revitalized

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

***Abstract*—We present a simple, original method to improve piecewise-linear interpolation with uniform knots: we shift the sampling knots by a fixed amount, while enforcing the interpolation property. We determine the theoretical optimal shift that maximizes the quality of our shifted linear interpolation. Surprisingly enough, this optimal value is nonzero and close to $1/5$.**

We confirm our theoretical findings by performing several experiments: a cumulative rotation experiment and a zoom experiment. Both show a significant increase of the quality of the shifted method with respect to the standard one. We also observe that, in these results, we get a quality that is similar to that of the computationally more costly "high-quality" cubic convolution.

***Index Terms*—Approximation methods, error analysis, interpolation, piecewise linear approximation, recursive digital filters, spline functions.**

I. Introduction

STANDARD piecewise-linear interpolation, which dates back to the Babylonians [1], is by far the most popular solution for many applications such as computer vision, digital photography, computer graphics, postscript optimization for printers, image calibration and registration, textures, and re-sampling. It is reasonably fast and does not suffer from the obvious blocking artifacts of nearest-neighbor interpolation. However, when quality is an important concern, methods based on higher-degree interpolation kernels have been developed [2]: Keys' cubic convolution method [3] has become a standard in the field, even though recent studies have shown that, for the same computational cost, cubic-spline and cubic-OMOMS kernels provide a substantial gain in quality [4]–[7].

It has been previously shown that the quality of the *interpolation* method is noticeably lower than that of the *projection* method, especially for piecewise-linear approximation [8], [9]. This suggests that there remains some significant margin of gain by optimizing linear interpolation. The goal of this paper is to explore one possible approach, by shifting the standard linear interpolation kernel. Note that, what we mean here is definitely not adapting the shift to the contents of the signal to interpolate, but instead using a *uniform* shift that is genuinely *independent* of this signal. This idea seems to have been initiated by Plonka [10] in the more general case of spline interpolation, although the main problem addressed in [10] was the *stability* of the interpolation operator, and not the *quality* of the resulting interpolation. Unsurprisingly, it was concluded that the most stable configuration is reached in the absence of shifting. Apart from [10], we do not know of any other research done on shifted interpolation, although there were other shift-based attempts (nonlinear and data-dependent) to improve the quality of standard interpolation methods [11].

Counterintuitively, we show here that there exists an optimal, nontrivial shift value (close to $1/5$) for linear interpolation, for which our new shifted interpolation improves considerably the quality of the standard—nonshifted—method. A quality as high as that of the orthogonal projection may even be attained in the limit of small sampling steps—or, equivalently, of very lowpass functions.

To predict the quality of shifted linear interpolation, we rely on the theoretical tools developed in [12]. We verify our claims by experiments that show that our method provides a quality that is much better than standard piecewise-linear interpolation, for a similar computational cost.

The paper is structured as follows: First, we review basic notions about interpolation—not necessarily with shifted linear splines—which will be needed for our analysis. In particular, we describe a theoretical method to estimate the quality of an interpolator using mathematical tools such as a Fourier approximation kernel and an asymptotic interpolation constant. Next, in Section III, we present shifted linear interpolation and show that a nontrivial value of the shift results in a quality that is optimal in an objective sense. We also evaluate the computational cost of our shifted linear method and show how to minimize it even further. Finally, we provide practical results that point out the deficiencies of standard linear interpolation and that show that its optimally-shifted version may even rival the standard high-quality (but more costly) method, namely, Keys' cubic convolution.

II. Interpolation

A. Piecewise-Linear Interpolation

Given a sequence of samples $f_n = f(nT)$ that originate from the uniform sampling of a function $f(x)$ with step T, the standard method of linear interpolation builds a function $f_T(x)$ through the following process: for $x \in [(n-1)T, nT]$, $f_T(x) = a_n x + b_n$, where a_n and b_n are chosen such that $f_T((n-1)T) = f_{n-1}$ and $f_T(nT) = f_n$. This can be shown to be equivalent to

$$f_T(x) = \sum_{n \in \mathbb{Z}} f_n \Lambda\left(\frac{x}{T} - n\right) \tag{1}$$

where $\Lambda(x)$ is the "hat" function, or linear B-spline: $\Lambda(x) = 1 - |x|$ for $|x| \leq 1$ and $\Lambda(x) = 0$ for $|x| > 1$ [13]. The Fourier

Manuscript received March 25, 2002; revised November 17, 2003. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Zixiang Xiong.

The authors are with the Biomedical Imaging Group FSTI/IOA, Swiss Federal Institute of Technology Lausanne (EPFL), CH-1015 Lausanne EPFL, Switzerland (e-mail: thierry.blu@epfl.ch; philippe.thevenaz@epfl.ch; michael.unser@epfl.ch).

Digital Object Identifier 10.1109/TIP.2004.826093

1057-7149/04$20.00 © 2004 IEEE

transform of this function, which will be useful in the sequel, is given by

$$\hat{\Lambda}(\omega) = \mathrm{sinc}^2\left(\frac{\omega}{2\pi}\right) = \left(\frac{\sin \omega/2}{\omega/2}\right)^2. \tag{2}$$

B. Generalized Interpolation

In a more general way, uniform interpolation is the process of building a function $f_T(x)$ through the formula

$$f_T(x) = \sum_{n\in\mathbb{Z}} c_n \varphi\left(\frac{x}{T} - n\right) \tag{3}$$

where the coefficients c_n are chosen so as to satisfy the interpolation condition $f_T(nT) = f_n$ [2], [4]. Here, $\varphi(x)$ might be any function with $\int \varphi(x)\,\mathrm{d}x = 1$. In particular, $\varphi(x)$ does not need to be continuous, interpolating, or symmetric. As can readily be seen, the interpolation condition is equivalent to the following filtering relation:

$$f_k = \sum_{n\in\mathbb{Z}} \varphi(k-n)c_n.$$

The coefficients c_n can be obtained by convolving the samples f_n with a filter H, the z-transform of which is $H(z) = 1/(\sum_n \varphi(n)z^{-n}) = \sum_n h_n z^{-n}$.

The interpolation formula (3) can equivalently be rewritten in a more traditional form in terms of the samples f_n [4]:

$$f_T(x) = \sum_{n\in\mathbb{Z}} f_n \varphi_{\mathrm{int}}\left(\frac{x}{T} - n\right)$$
$$\text{where } \varphi_{\mathrm{int}}(x) = \sum_{n\in\mathbb{Z}} h_n \varphi(x-n)$$

where the equivalent interpolation function $\varphi_{\mathrm{int}}(x)$ is obtained by applying the filter weights to the basis functions. The Fourier transform of φ_{int} takes the simple form

$$\hat{\varphi}_{\mathrm{int}}(\omega) = H(e^{j\omega})\hat{\varphi}(\omega). \tag{4}$$

The 1D formula (3) can easily be used for image interpolation by replacing $\varphi(x)$ by the tensor product $\varphi(x)\varphi(y)$ in (3): $f_T(x,y) = \sum_{k,l} c_{k,l}\varphi(x/T-k)\varphi(y/T-l)$. This expression, as well as the prefiltering step to obtain the $c_{k,l}$, can be realized very efficiently in a separable manner (e.g., by successive processing of the lines and columns) [4].

C. Measuring the Interpolation Error

For $f_T(x)$ to be a good approximation of $f(x)$, the quality needs to improve as T gets smaller. The rate of this improvement is called the order of approximation. Usually, we assume that $f \in \mathbf{W}_2^{1/2+\varepsilon}$ (with $\varepsilon > 0$), so as to ensure that f and f_T belong to $\mathbf{L}^2(\mathbb{R})$ [12]. A natural measure for the distance between $f_T(x)$ and $f(x)$ is then $\|f - f_T\|_{\mathbf{L}^2(\mathbb{R})}$. Obviously, this approximation error is bounded from below by $\|f - \mathrm{P}_T f\|_{\mathbf{L}^2(\mathbb{R})}$, where $\mathrm{P}_T f$ is the orthogonal projection of f onto the function space made of linear combinations of $\varphi(x/T-n)$, $n \in \mathbb{Z}$.

We have shown in [4], [12] that both the interpolation error $\varepsilon_{\mathrm{int}}(T) = \|f - f_T\|_{\mathbf{L}^2(\mathbb{R})}$ and the orthogonal projection error $\varepsilon_{\mathrm{min}}(T) = \|f - \mathrm{P}_T f\|_{\mathbf{L}^2(\mathbb{R})}$ can be estimated very accurately by

$$\eta(T) = \sqrt{\frac{1}{2\pi}\int |\hat{f}(\omega)|^2 E(\omega T)\,\mathrm{d}\omega} \tag{5}$$

where the function $E(\omega)$ is what we call a Fourier kernel of approximation. This function depends only on $\varphi(x)$, in the following ways.

i) For the orthogonal projection P_T, $E(\omega) = E_{\mathrm{min}}(\omega)$ with

$$E_{\mathrm{min}}(\omega) = \frac{\sum_{k\in\mathbb{Z}\setminus\{0\}} |\hat{\varphi}(\omega+2k\pi)|^2}{\sum_{k\in\mathbb{Z}} |\hat{\varphi}(\omega+2k\pi)|^2}. \tag{6}$$

ii) For interpolation, $E(\omega) = E_{\mathrm{int}}(\omega)$ with (7), as shown at the bottom of the page.

To be more precise, the identity between the approximation error and $\eta(T)$ holds whenever $f(x)$ is bandlimited in $[-\pi/T, \pi/T]$. In the nonbandlimited case, this identity holds *on average*—when one computes the average over all possible ξ of the error resulting from the approximation of $f(x-\xi)$ [12, Thm.2].

The above formulæ still hold for higher dimensional signals [4]—images, for instance. It is just a matter of replacing the single integral and the single summations by multiple ones, $\mathbb{Z}$ by $\mathbb{Z}^d$ and $\{0\}$ by $\{0,0,\ldots 0\}$ in (5), (6), and (7).

D. Asymptotic Interpolation Error

When the sampling step T tends to 0, we want that $f_T(x) \to f(x)$. One can easily see from the Expression (5) that this is the case when $E(0) = E_{\mathrm{int}}(0) = 0$ under relatively weak conditions on $\varphi(x)$ (for instance, see [14]); or equivalently, when $\hat{\varphi}(2k\pi) = 0$ for all $k \in \mathbb{Z}\setminus\{0\}$ and $\hat{\varphi}(0) = 1$, as is obvious from (7). More generally, it is known that $\|f - \mathrm{P}_T f\|_{\mathbf{L}^2(\mathbb{R})}$ tends to zero as T^L if and only if $\varphi(x)$ satisfies the Strang-Fix conditions [15]: $\hat{\varphi}^{(l)}(2n\pi) = 0$ for $n \in \mathbb{Z}\setminus\{0\}$ and $l = 0 \ldots L-1$. This equivalence still holds for $\|f - f_T\|_{\mathbf{L}^2(\mathbb{R})}$. The integer L is called the approximation order of $\varphi(x)$. For instance, the approximation order of $\Lambda(x)$ is $L = 2$ because of (2).

More precisely, we have shown in [12] that

$$\|f - \mathrm{Q}_T f\|_{\mathbf{L}^2(\mathbb{R})} \approx C_\varphi \|f^{(L)}\|_{\mathbf{L}^2(\mathbb{R})} T^L$$

as $T \to 0$, where Q_T is any integer-shift-invariant linear approximation operator, such as interpolation, or the orthogonal

$$E_{\mathrm{int}}(\omega) = \frac{\left|\sum_{k\in\mathbb{Z}\setminus\{0\}} \hat{\varphi}(\omega+2k\pi)\right|^2 + \sum_{k\in\mathbb{Z}\setminus\{0\}} |\hat{\varphi}(\omega+2k\pi)|^2}{\left|\sum_{k\in\mathbb{Z}} \hat{\varphi}(\omega+2k\pi)\right|^2}. \tag{7}$$

projection. The asymptotic constant $C_\varphi = \sqrt{E^{(2L)}(0)/2L!}$ may also be expressed as follows:

i) For the orthogonal projection [16]

$$C_\varphi^{\min} = \frac{1}{L!}\sqrt{\sum_{k\neq 0} |\hat{\varphi}^{(L)}(2k\pi)|^2}. \tag{8}$$

ii) For interpolation [4], ([17], when $\varphi(x)$ is interpolating)

$$C_\varphi^{\text{int}} = \sqrt{\frac{1}{L!^2}\left|\sum_{k\neq 0}\hat{\varphi}^{(L)}(2k\pi)\right|^2 + \left(C_\varphi^{\min}\right)^2}. \tag{9}$$

Comparing (8) to (9), it clearly appears that it is the quantity $\sum_{k\neq 0}\hat{\varphi}^{(L)}(2k\pi)$ that tells apart interpolation from orthogonal projection. In particular, this quantity is substantial for piecewise-linear interpolation. In the sequel, we will show that we can use a shift to make it vanish.

The result can also be extended for higher dimensional signals. For instance, in two dimensions, we can apply a Taylor analysis and prove that the interpolation error behaves asymptotically like

$$\begin{aligned}\|f - \mathrm{Q}_T f\|^2_{\mathbf{L}^2(\mathbb{R})} \approx \Big[&\left(C_\varphi^{\min}\right)^2\left(\left\|\partial_x^L f\right\|^2_{\mathbf{L}^2(\mathbb{R})} + \left\|\partial_y^L f\right\|^2_{\mathbf{L}^2(\mathbb{R})}\right)\\ &+ \left(\left(C_\varphi^{\text{int}}\right)^2 - \left(C_\varphi^{\min}\right)^2\right)\left(\left\|\partial_x^L f\right\|^2_{\mathbf{L}^2(\mathbb{R})}\right.\\ &+ \left.\left\|\partial_y^L f\right\|^2_{\mathbf{L}^2(\mathbb{R})} + 2\left\langle\partial_x^L f, \partial_y^L f\right\rangle\right)\Big] T^L\end{aligned}$$

where $C_\varphi^{\min}$ and C_φ^{int} are given by (8) and (9). Similarly to the 1D case, it is $\sum_{k\neq 0}\hat{\varphi}^{(L)}(2k\pi)$ that makes the difference between the interpolation and orthogonal projection constants. This justifies that, from now on, we will only be considering the 1D case in our optimization.

III. Shifted Linear Interpolation

Instead of building $f_T(x)$ using line segments between $(n-1)T$ and nT as in Section II, we draw them between $(n-1+\tau)T$ and $(n+\tau)T$ for some $\tau \in [0, 1/2[$, as exemplified in Fig. 1. More precisely, we consider the following process: for $x \in [(n-1+\tau)T, (n+\tau)T]$, $f_T(x) = a_n x + b_n$, where a_n and b_n are chosen such that $f_T(nT) = f_n$, and where $f_T(x)$ is continuous at $x = (n-1+\tau)T$.

Theorem 1: The τ-shifted linear interpolation of the samples f_n can be expressed as

$$f_T(x) = \sum_{n\in\mathbb{Z}} c_n \Lambda\left(\frac{x}{T} - n - \tau\right) \tag{10}$$

with

$$c_n = \sum_{k\geq 0}\frac{(-1)^k}{1-\tau}\left(\frac{\tau}{1-\tau}\right)^k f_{n-k}. \tag{11}$$

The coefficients c_n satisfy the induction equation

$$c_n = -\frac{\tau}{1-\tau}c_{n-1} + \frac{1}{1-\tau}f_n. \tag{12}$$

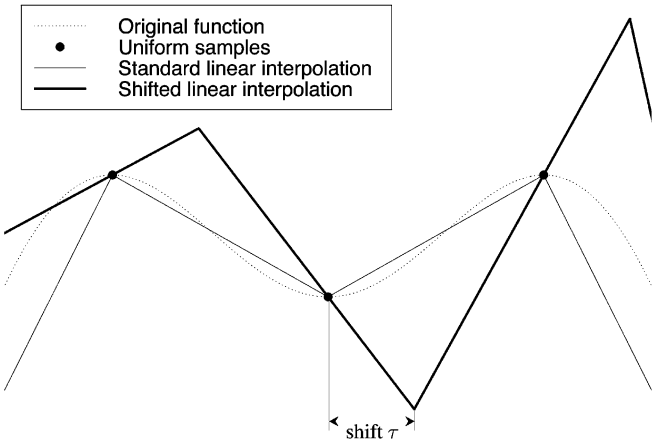

Fig. 1. "Shifted" versus "standard" linear interpolation. For clarity purpose, we have chosen $\tau = 0.4$, which is far from the optimum (15).

Proof: In order to show that $f_T(x)$ is a τ-shifted linear interpolation, we have to verify that it is piecewise-linear, continuous, and that it satisfies the interpolation condition $f_T(kT) = f_k$. Continuity and linearity are obvious since $\Lambda(x/T - \tau)$ is itself continuous and piecewise-linear. The last assertion is checked directly:

$$\begin{aligned} f_T(kT) &= \sum_{n\in\mathbb{Z}} c_n \Lambda(k - n - \tau)\\ &= c_k\Lambda(-\tau) + c_{k-1}\Lambda(1-\tau)\\ &= (1-\tau)c_k + \tau c_{k-1}\\ &= f_k \quad (\text{using}(12)).\end{aligned}$$

■

We observe that the expression (11) relates f_n to c_n through convolution by the *causal* filter H_τ with z-transform

$$H_\tau(z) = \frac{1}{1-\tau+\tau z^{-1}} = \sum_{k\geq 0}\frac{(-1)^k}{1-\tau}\left(\frac{\tau}{1-\tau}\right)^k z^{-k}.$$

Because $\tau \in [0, 1/2[$, this filter is *stable*. However, unlike standard linear interpolation ($\tau = 0$), its impulse response is *of infinite length* even if, in practice, its impulse response can be considered to be of finite support (exponential decay, see Fig. 2).

The Fourier transform of the equivalent interpolant implied by (10) is obtained by substituting the filter's response in (4):

$$\hat{\varphi}_{\text{int}}(\omega) = e^{-j\omega\tau}H_\tau(e^{j\omega})\,\mathrm{sinc}^2\left(\frac{\omega}{2\pi}\right). \tag{13}$$

Intuitively, it is expected that a small shift bring a flatter frequency amplitude because the prefilter H_τ acts like a high-pass filter that partially compensates the quadratic decreasing behavior of $\mathrm{sinc}^2(\omega/2\pi)$.

It is obvious that the computation of the coefficients c_n can be carried out very efficiently using the recursion (12): only two multiplications and one addition are necessary per data point. In order to be complete, we must initialize (12) at, say, $n = 0$. A simple choice is to assume that $f_n = f_0$ for all $n < 0$ which in turn implies the initial condition $c_0 = f_0$, by applying the nonrecursive expression (11).

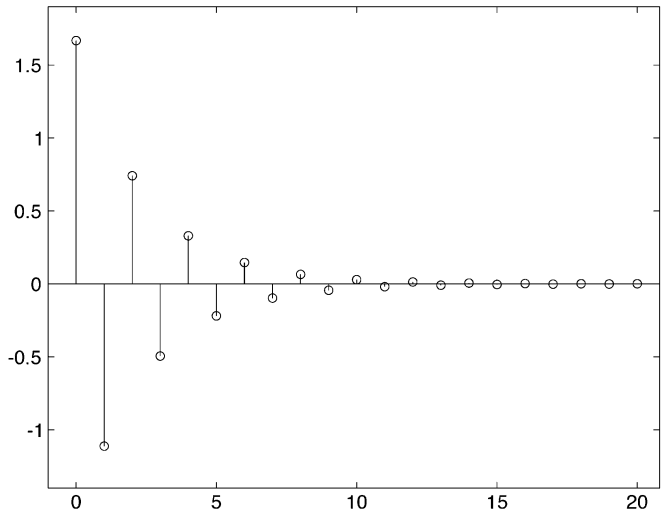

Fig. 2. Impulse response of the prefilter of the shifted linear interpolation of Fig. 1.

A. Optimal Shift

We may now apply the theory of Section II to the specific case where $\varphi(x)$ is the shifted hat function $\Lambda(x-\tau)$ which, according to Theorem 1, is the basis building block of τ-shifted linear interpolation.

Theorem 2: The asymptotic interpolation constant of the τ-shifted linear interpolation is given by

$$C_\tau^{\text{int}} = \sqrt{\frac{1}{4}\left(\tau^2-\tau+\frac{1}{6}\right)^2+\frac{1}{720}}. \tag{14}$$

Proof: We know from Section II that $\varphi(x)=\Lambda(x-\tau)$ is of order $L=2$ since it satisfies the Strang-Fix conditions of order 2. According to (8) and (9), we have to compute $\sum_{k\neq 0}\hat{\varphi}''(2k\pi)$ and $C_\varphi^{\min}$. The first quantity can be evaluated as follows:

$$\begin{aligned}\sum_{k\neq 0}\hat{\varphi}''(2k\pi) &= -\hat{\varphi}''(0)+\sum_k \hat{\varphi}''(2k\pi)\\ &= -\hat{\varphi}''(0)-\sum_n n^2\varphi(n) \quad \text{(using Poisson's formula)}\\ &= \tau^2+\frac{1}{6}-\tau \quad \text{(replacing } \varphi(x) \text{ by } \Lambda(x-\tau)).\end{aligned}$$

Then, we evaluate the asymptotic constant $C_\varphi^{\min}$. Since a shift contributes only as a phase term in the Fourier transform, we claim that $C_\varphi^{\min}$ does not depend on τ (see the expression of the orthogonal projection kernel (6)). Thus, $C_\varphi^{\min}=C_\Lambda^{\min}=\sqrt{2\zeta(4)}/((2\pi)^2=1/\sqrt{720}$. Putting things together in (9) gives (14). ∎

Surprisingly, this expression is not minimized for the standard linear interpolation $\tau=0$. Instead, the optimal choice is

$$\tau_{\text{opt}} = \frac{1}{2}\left(1-\frac{\sqrt{3}}{3}\right)\approx 0.21. \tag{15}$$

This minimizes the interpolation constant and reaches the lower value of the *optimal* $\mathbf{L}^2(\mathbb{R})$ approximation (orthogonal projection). This is remarkable because interpolation is never optimal in the least-squares sense—we must keep in mind here that our result is valid only in the asymptotic regime (e.g., smooth function $f(x)$, or small sampling step T).

We have plotted in Fig. 3 the frequency response of the equivalent interpolant (13), and we have compared it with the piecewise-linear interpolant. In the Nyquist band, the amplitude response of the shifted linear interpolant is much closer to an ideal filter than in the standard piecewise-linear case. This is obtained at the price of a slight phase distortion, and larger ripples in the aliasing bands.

Using the definition of the asymptotic constant, we see that the gain of shifted over standard linear interpolation is about 8 dB asymptotically, as the sampling step tends to 0. Obviously, this performance should degrade as the frequency content of the function to interpolate gets richer, that is, when the energy at higher frequencies becomes more significant. In particular, when $f(x)$ is the step function (a limit case that does not belong to $\mathbf{L}^2(\mathbb{R})$, but for which we can still test our interpolation method), the shifted linear interpolation gives rise to a Gibbs phenomenon—unlike the standard method (see Fig. 4).

Using the Fourier kernel of approximation (7), we can be more precise about the range of frequencies over which shifted linear interpolation outperforms standard linear interpolation.

Proposition 3: The Fourier kernel of approximation of τ-shifted linear interpolation is given by

$$\begin{aligned}E_\tau(\omega) = 1+\frac{1}{3}\frac{2+\cos\omega}{|1-\tau+\tau e^{-j\omega}|^2}\\ -2\,\text{sinc}^2\left(\frac{\omega}{2\pi}\right)\text{Re}\left(\frac{e^{-j\tau\omega}}{1-\tau+\tau e^{-j\omega}}\right).\end{aligned} \tag{16}$$

Proof: Expanding (7), we obtain

$$\begin{aligned}E_\tau(\omega) = 1+\frac{\sum_{k\in\mathbb{Z}}|\hat{\varphi}(\omega+2k\pi)|^2}{\left|\sum_{k\in\mathbb{Z}}\hat{\varphi}(\omega+2k\pi)\right|^2}\\ -2\text{Re}\left(\frac{\hat{\varphi}(\omega)}{\sum_{k\in\mathbb{Z}}\hat{\varphi}(\omega+2k\pi)}\right).\end{aligned}$$

Then, Poisson's summation formula applied to $\varphi(x)=\Lambda(x-\tau)$ yields

$$\begin{aligned}\sum_k\hat{\varphi}(\omega+2k\pi) &= \sum_n\varphi(n)e^{-jn\omega}\\ &= \Lambda(-\tau)+\Lambda(1-\tau)e^{-j\omega}\\ &= 1-\tau+\tau e^{-j\omega}\end{aligned}$$

and

$$\begin{aligned}\sum_k|\hat{\varphi}(\omega+2k\pi)|^2 &= \sum_k|\Lambda(\omega+2k\pi)|^2\\ &= \sum_n\Lambda*\Lambda(n)e^{-jn\omega}\\ &= \sum_n\beta^3(n)e^{-jn\omega}\\ &= \frac{2+\cos\omega}{3}\end{aligned}$$

where $\beta^3(x)$ is the B-spline of degree 3, the Fourier transform of which is $\text{sinc}^4(\omega/2\pi)$ [13]. Putting things together we find (16). ∎

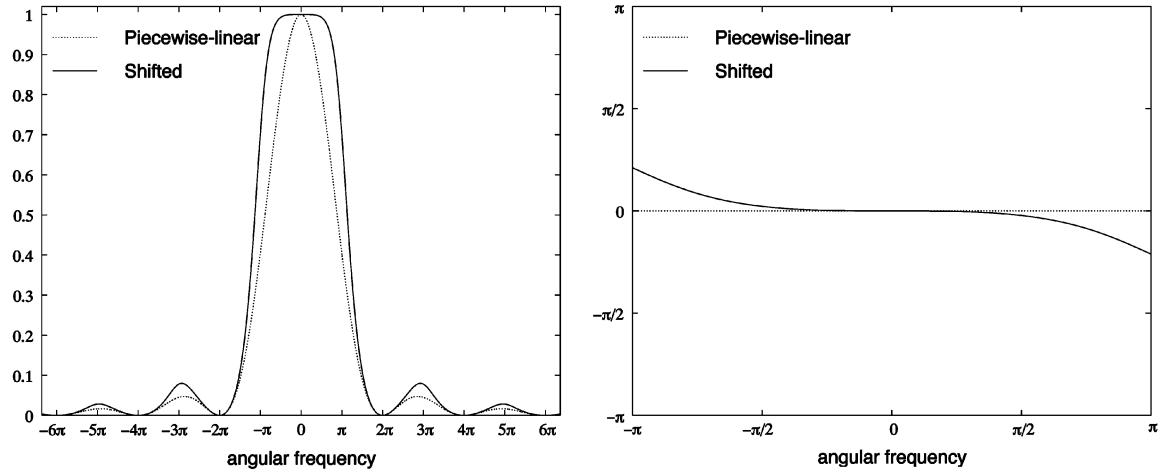

Fig. 3. Frequency plot of the amplitude (left) and of the in-band phase distortion (right) of the standard and optimally shifted linear interpolant.

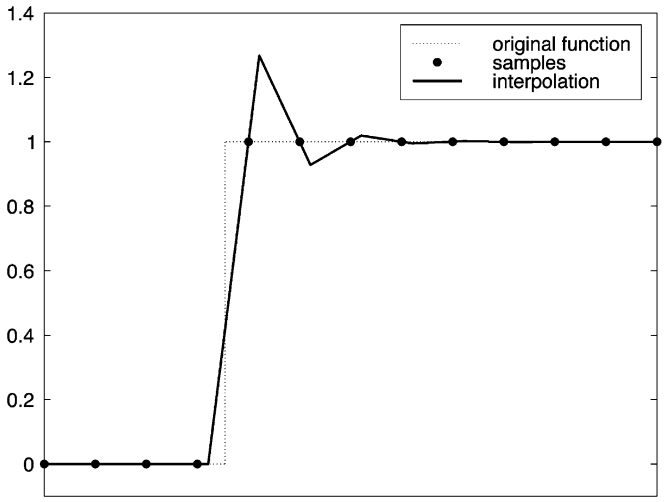

Fig. 4. Gibbs phenomenon caused by the optimally shifted linear interpolation of a unit-step function.

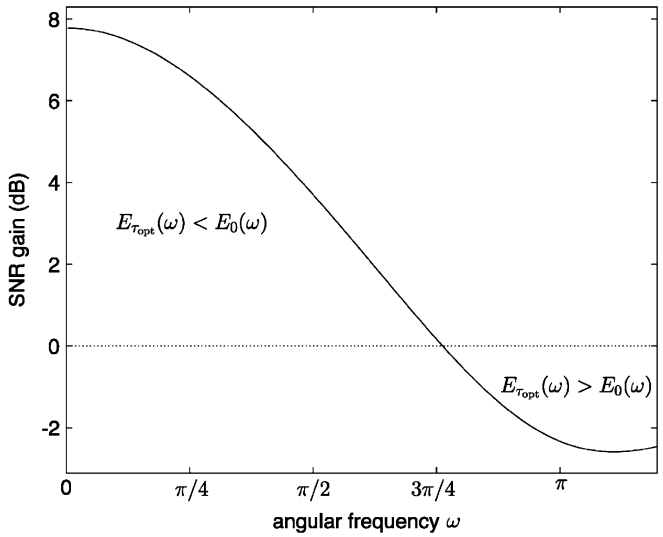

Fig. 5. Plot of the gain (17) $\gamma_{\tau_{\rm opt}}(\omega)$ as a function of ω: for $\omega < 3\pi/4$, the optimally shifted method outperforms the nonshifted one, by up to 8 dB.

The approximation kernel evaluated at the frequency ω_0 can also be interpreted as the average power of the approximation—or here, interpolation—error of the pure unit sinusoid $e^{j\omega_0 x}$ ([12] Thm.3). Thus, the quantity $E_\tau(\omega_0)/E_0(\omega_0)$ quantifies the reduction or increase of the interpolation error caused by the τ-shifted method compared to the standard one at the angular frequency ω_0. We define this gain by

$$\gamma_\tau(\omega) = 10\log_{10}\left(\frac{E_0(\omega)}{E_\tau(\omega)}\right). \tag{17}$$

This function is shown in Fig. 5 for the optimal shift $\tau = \tau_{\rm opt}$. Not only does this graph confirm that this gain should reach almost 8 dB in the neighborhood of $\omega = 0$, it also shows that the shifted linear interpolation should perform better than its nonshifted counterpart for signals with frequency range up to $3/4$th of the sampling bandwidth.

In principle, the same shifting trick may also be used for other kernels than $\Lambda(x)$ such as higher order splines; the optimal shift has then to be determined on a case-by-case basis.

B. Computational Cost and Implementation Options

The computational cost of an interpolation method characterized by (3) can be decomposed into two parts:

The Pre-Computation of the Coefficients c_k: This cost depends on the number $N_{\rm in}$ of *input* samples and the size of the prefilter H only. Typically, H is of size W where W is the length of the support of φ. If we consider image interpolation, then the full cost amounts to $2WN_{\rm in}$ multiplications and additions (using the recursive implementation).

Note that this amount is usually a negligible overhead. In addition, for applications requiring intensive interpolation, it is also possible to do the preprocessing in advance and to work with the c_k instead of the initial data. This is especially indicated in the context of iterative algorithms, and for interactive imaging and vizualization;

The Computation of $f_T(x)$: This cost depends only on the number $N_{\rm out}$ of *output* samples x and on the cost of computing $\varphi(x-k)$ for all values of k such that $(x-k)$ belongs to the

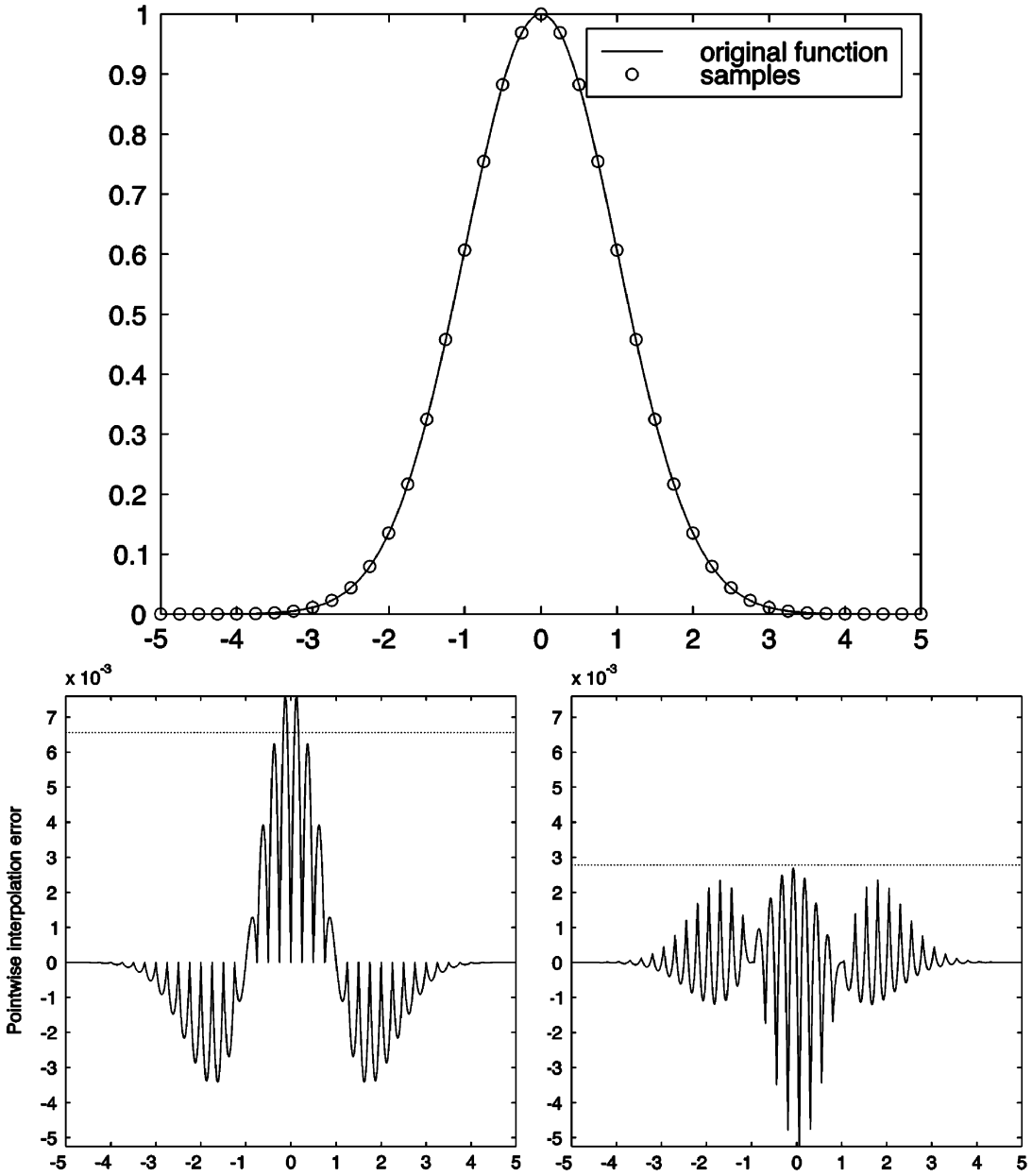

Fig. 6. Interpolation error $f(x) - f_T(x)$ for the Gaussian function $f(x) = e^{-x^2/2}$ (top) using: standard linear interpolation (bottom left) and optimally shifted linear interpolation (bottom right). The dotted lines indicate the $\mathbf{L}^2$ interpolation error: in this example, the gain of the shifted method is close to 7.5 dB. See Section IV-A for more details.

support of φ. Typically, when $\varphi(x)$ is piecewise-polynomial of degree n, the evaluation of (3) requires

- $W^2 + W$ additions and multiplications (in the 2D separable implementation);
- $2W$-times the evaluation of the 1D basis kernel, which amounts to n additions and multiplications (using Horner's scheme).

The total cost is therefore of $N_{\text{out}}(W^2 + (2n+1)W)$ additions and multiplications, for given c_k.

In the case of piecewise-linear interpolation ($n = 1$ and $W = 2$), the full cost is less than what is estimated above (which is $10N_{\text{out}}$ multiplications and additions) because the computation of the kernel Λ at $(x - k)$ requires only one addition on the whole. Finally, we need $6N_{\text{out}}$ multiplications and $8N_{\text{out}}$ additions in 2D. In the case of shifted linear interpolation, the full cost amounts to a little less than $4N_{\text{in}} + 8N_{\text{out}}$ multiplications and additions.

For comparison purposes, a higher-order interpolation method like Keys' cubic interpolation ($W = 4$ and $n = 3$) would cost $44N_{\text{out}}$, which is, in general, much more than shifted linear interpolation, in particular when $N_{\text{in}} \leq N_{\text{out}}$ (e.g., zoom ins, rotations).

In fact, a great advantage of our shifted method is to allow one to take advantage of existing optimized hard- and software solutions for *standard* piecewise-linear interpolation. This can be done at the mere cost of the prefiltering step. If we consider that a linear interpolator has two inputs (the uniform samples

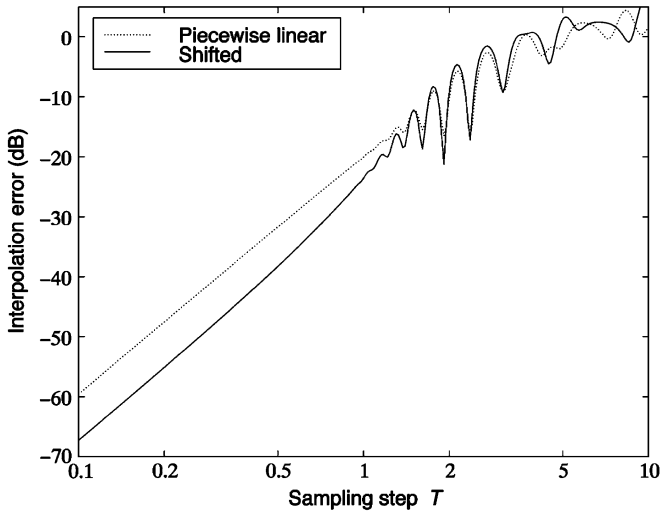

Fig. 7. Interpolation error $||f(x) - f_T(x)||_{\mathbf{L}^2(\mathbb{R})}$ for $f(x) = e^{-x^2/2}$ using standard linear and optimally shifted linear interpolation. The asymptotic gain is close to 8 dB when the sampling step tends to 0.

f_n and the resampling points x_k) and one output (the resampled values $\sum_n f_n \Lambda(x_k/T - n)$), then shifted linear interpolation can be implemented as shown in Fig. 10.

Morever, it is possible to further reduce the computational cost of the coefficients c_n. To this end, we notice that the value of the optimal shift (15) is close to the fraction $1/5$. Thus the use of $\tau = 1/5$ should result in the same pratical performance as $\tau = \tau_{\text{opt}}$. The interesting aspect brought by this value is that the induction (12) becomes

$$c_n = -2^{-2} c_{n-1} + (1 + 2^{-2}) f_n.$$

Hence, c_n can be obtained (and encoded) using finite-precision arithmetics and without a multiplication, since 2^{-2} amounts to a genuine register shift.

IV. Simulations and Practical Results

Although shifted linear interpolation is counterintuitive (in particular, because it dissymmetrizes a method that is naturally symmetric), in practice it does behave better than the nonshifted method, as we will see in this section.

A. Distribution of the Errors

One of the reasons for its better performance is, apparently, that it tends to distribute more evenly the interpolation errors. In fact, standard piecewise-linear interpolation is *systematically biased* within intervals of same convexity. This can be illustrated by performing the linear interpolation of a function $f(x)$ that would be convex in $x \in [n_0 T \leq x \leq n_1 T]$. Here, n_0, n_1 are integers, and T is assumed to be small enough so that at least two samples lie inside this segment. Hence the convexity inequality

$$\forall x \in [n_0 T, n_1 T], \quad f(x) \leq f_T(x).$$

This implies that the interpolation error is always *negative* on a *convex segment*. Similarly, the interpolation error is always *positive* on a *concave segment*. In other words, the interpolation error does not cancel on average over each of these convexity intervals; as a consequence, it adds a—significant—bias to the average square error. As the sampling step T decreases, this behavior gets even more pronounced if we assume that $f(x)$ is roughly bandlimited, because the transitions between convex and concave segments, which have length T, tend to weight less ans less.

This is exemplified in Fig. 6 where we approximate a Gaussian: the two methods have roughly the same dynamics (peak-to-peak values) within each of the convex/concave part of the graph of $f(x)$. However, the standard linear method exhibits a systematic bias within each of these intervals, whereas the shifted method does not.

We have repeated this experiment with several values of the sampling step and we have computed the $\mathbf{L}^2$ interpolation error for each method, the result of which is plotted in Fig. 7. Clearly, the shifted linear method outperforms the linear one, even for large sampling steps like $T = 1$.

B. Rotation Experiments

In order to validate our theory on practical data, we designed a compounded-rotation experiment of the ubiquitous Lena image (512×512 pixels). Let $f(x, y)$ denote this original image. We have access to its samples $\{f(k, l)\}_{k,l \in \mathbb{Z}}$ only. We first interpolate them—in a separable fashion (see Section II.B)—to get $f_T(x, y)$ (here, $T = 1$); then, we rotate f_T by the angle $\theta = (2\pi/15)$, which provides $g(x, y) = f_T(x \cos\theta + y \sin\theta, -x \sin\theta + y \cos\theta)$; finally, we resample $g(x, y)$ on the original uniform grid, which gives the "rotated" image $\{g(k, l)\}_{k,l \in \mathbb{Z}}$. Iterating this procedure 15 times provides an image that has been rotated by $15 \times 24° = 360°$, and that can be readily compared to the original image.

The advantage of such an experiment is that it is likely to amplify the interpolation errors so that it is easier to rank different interpolation methods.

As is apparent from Fig. 8, the standard linear interpolation suffers from blurring, an effect that is avoided in the shifted method which provides much more details. More surprisingly, the shifted method appears to reach a quality that is comparable to that of the higher-order, more costly Keys' cubic interpolation [3], which is the reference high-quality method.

On our website [18], we have put a similar java demo which lets the reader try various rotation angles and shifts τ on several images (256×256 pixels) that differ in their high-frequency content. The results are consistent with those of Fig. 8. Note that, in this web demo, we have chosen to perturbate the rotation center by a noninteger displacement at every iteration, in order to enforce true interpolation for every angle, including for 90° rotations.

C. Zoom Experiments

We have also tested these interpolation methods in a zoom experiment, a test that is more realistic than the compounded-rotation experiment. The drawback is that it is not objective

Fig. 8. Fifteen successive rotations by 24° of Lena using standard, shifted linear, and Keys' interpolations; notice the sharpness of the result of our shifted method, as compared to the two others.

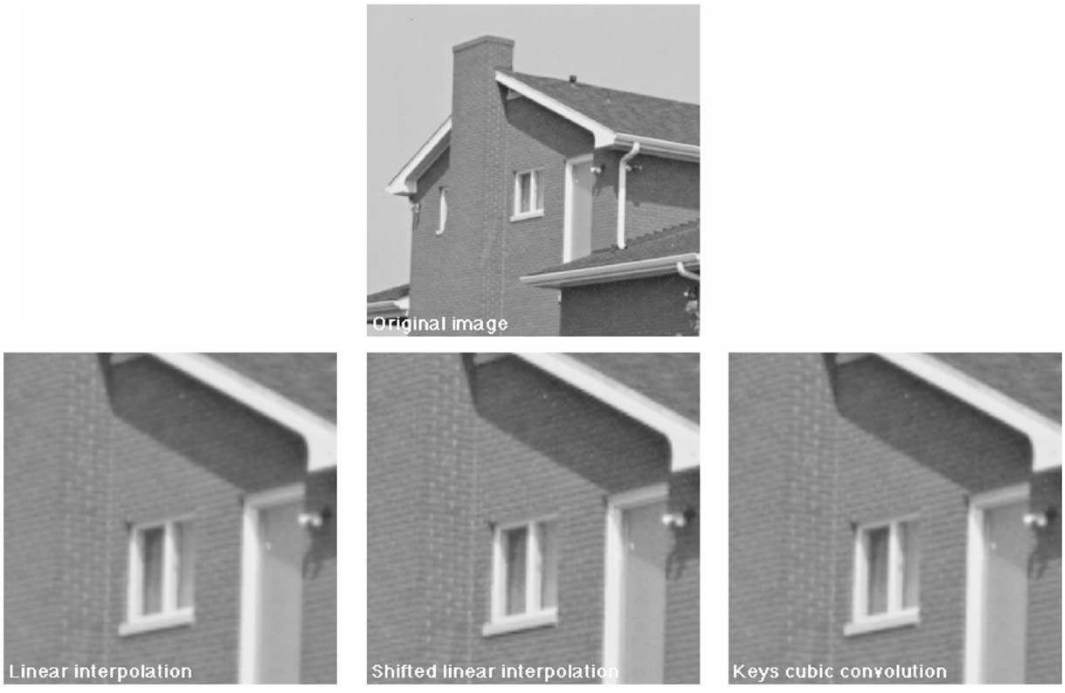

Fig. 9. Magnification by $\sqrt{5}$ of the top image using the two piecewise-linear methods and Keys' cubic convolution method. Once again, notice the sharpness of the result obtained with the optimally shifted method.

anymore, because we cannot compute Signal-to-Noise ratios. We have chosen the "House" image for its texture content (bricks). The result of a magnification by $\sqrt{5}$ is shown in Fig. 9.

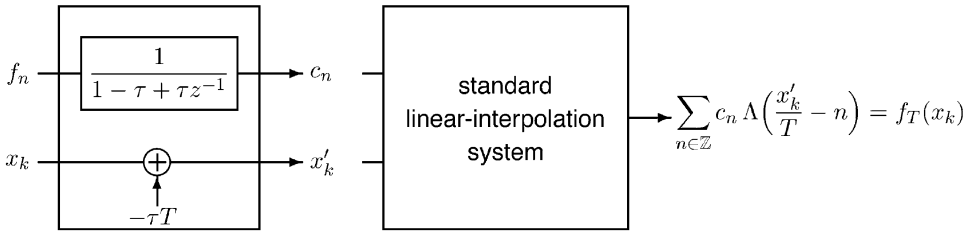

Fig. 10. Proposed implementation of τ-shifted linear interpolation using standard soft/hardware: the data are first preprocessed, which yields modified sample values c_n and shifted sampling points $x'_k = x_k - \tau T$; then, these parameters are fed into a standard linear interpolator which outputs the same result as (10).

Clearly, the standard piecewise-linear method blurs the zoomed image much more than the two other methods. A closer inspection also shows that the result of the shifted linear method is perceptually sharper than the result obtained using Keys' cubic kernel.

V. Conclusion

We have presented a simple, powerful method for improving the performance of standard linear interpolation. We proved its—asymptotic—optimality, and, more generally, we evaluated its performance using approximation-theoretical tools that we had developed in previous papers. Theory and practice are in good agreement, as illustrated by using synthetic data, by compounded-rotation experiments, and by zooming of real-life images.

For efficient implementation, we have proposed to precompute the model coefficients in a preprocessing step (simple recursive filtering), which amounts to replace the initial data by a resampled version at the shifted knot location. With such a set-up, the method can be implemented directly via standard linear interpolation, so that we can readily take advantage of existing software or of specialized hardware solutions.

References

[1] E. H. W. Meijering, "A chronology of interpolation: From ancient astronomy to modern signal and image processing," *Proc. IEEE*, vol. 90, no. 3, pp. 319–342, Mar. 2002.

[2] *Handbook of Medical Imaging, Processing and Analysis*, I. N. Bankman, Ed., Academic, New York, 2000, pp. 393–420. Image interpolation and resampling.

[3] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 29, pp. 1153–1160, Dec. 1981.

[4] P. Thévenaz, T. Blu, and M. Unser, "Interpolation revisited," *IEEE Trans. Med. Imag.*, vol. 19, pp. 739–758, July 2000.

[5] E. H. W. Meijering, W. J. Niessen, and M. A. Viergever, "Quantitative evaluation of convolution-based methods for medical image interpolation," *Med. Image Anal.*, vol. 5, no. 2, pp. 111–126, June 2001.

[6] T. Lehmann, C. Gönner, and K. Spitzer, "Addendum: B-spline interpolation in medical image processing," *IEEE Trans. Med. Imag.*, vol. 20, pp. 660–665, July 2001.

[7] T. Blu, P. Thévenaz, and M. Unser, "MOMS: Maximal-order interpolation of minimal support," *IEEE Trans. Image Processing*, vol. 10, pp. 1069–1080, July 2001.

[8] M. Unser, A. Aldroubi, and M. Eden, "Enlargement or reduction of digital images with minimum loss of information," *IEEE Trans. Image Processing*, vol. 4, pp. 247–258, Mar. 1995.

[9] A. Muñoz, T. Blu, and M. Unser, "Least-squares image resizing using finite differences," *IEEE Trans. Image Processing*, vol. 10, pp. 1365–1378, Sept. 2001.

[10] G. Plonka, "Optimal shift parameters for periodic spline interpolation," *Numer. Algorithms*, vol. 6, no. 3–4, pp. 481–510, Mar. 1994.

[11] G. Ramponi, "Warped distance for space-variant linear image interpolation," *IEEE Trans. Image Processing*, vol. 8, pp. 629–639, May 1999.

[12] T. Blu and M. Unser, "Quantitative Fourier analysis of approximation techniques: Part I—Interpolators and projectors," *IEEE Trans. Signal Processing*, vol. 47, no. 10, pp. 2783–2795, Oct. 1999.

[13] M. Unser, "Splines: A perfect fit for signal and image processing," *IEEE Signal Process. Mag.*, vol. 16, pp. 22–38, Nov. 1999.

[14] T. Blu and M. Unser, "Approximation error for quasiinterpolators and (multi-) wavelet expansions," *Appl. Comput. Harmon. Anal.*, vol. 6, no. 2, pp. 219–251, Mar. 1999.

[15] G. Strang and G. Fix, "A Fourier analysis of the finite element variational method," in *Constructive Aspects of Functional Analysis*, G. Geymonat, Ed, Rome: Edizioni Cremonese, 1973, pp. 793–840.

[16] M. Unser, "Approximation power of biorthogonal wavelet expansions," *IEEE Trans. Signal Processing*, vol. 44, pp. 519–527, Mar. 1996.

[17] 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.

[18] http://bigwww.epfl.ch/demo/jshiftlinear/

Thierry Blu (M'96) was born in Orléans, France, in 1964. He received the "Diplôme d'ingénieur" degree from École Polytechnique, France, in 1986 and from Télécom Paris (ENST), France, in 1988. In 1996, he received the Ph.D. degree in electrical engineering from ENST for a study on iterated rational filterbanks, applied to wideband audio coding. His research interests are (multi)wavelets, multiresolution analysis, multirate filterbanks, approximation and sampling theory, psychoacoustics, optics, and wave propagation.

He is with the Biomedical Imaging Group at the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, on leave from France Télécom National Center for Telecommunications Studies (CNET), Issy-les-Moulineaux, France.

Dr. Blu is currently serving as an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING.

Philippe Thévenaz (M'95) was born in 1962 in Lausanne, Switzerland. He graduated from the Lausanne Swiss Federal Institute of Technology (EPFL) in January 1986, with a diploma in microengineering. He then joined the Institute of Microtechnology (IMT) of the University of Neuchâtel, Switzerland, where he worked in the domain of image processing (optical flow) and in the domain of speech processing (speech coding and speaker recognition). He received the Ph.D. degree in June 1993, with a thesis on the use of the linear prediction residue for text-independent speaker recognition.

He then worked as a Visiting Fellow with the Biomedical Engineering and Instrumentation Program, National Institutes of Health (NIH), Bethesda MD, where he developed research interests that include splines and multi-resolution signal representations, geometric image transformations, and biomedical image registration. Since 1998, he has been with the Lausanne Swiss Federal Institute of Technology as First Assistant.

Michael Unser (M'89–SM'94–F'99) received the M.S. (summa cum laude) and Ph.D. degrees in electrical engineering in 1981 and 1984, respectively, from the Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland.

From 1985 to 1997, he was with the Biomedical Engineering and Instrumentation Program, National Institutes of Health, Bethesda, MD. He is now Professor and Head of the Biomedical Imaging Group at the EPFL. His main research area is biomedical image processing. He has a strong interest in sampling theories, multiresolution algorithms, wavelets, and the use of splines for image processing. He is the author of 100 published journal papers in these areas.

Dr. Unser is Associate Editor-in-Chief for the IEEE TRANSACTIONS ON MEDICAL IMAGING. He has been on the editorial boards of the IEEE TRANSACTIONS ON IMAGE PROCESSING (1992–1995) and IEEE SIGNAL PROCESSING LETTERS (1994–1998). He serves as regular chair for the SPIE conference on wavelets, held annually since 1993. He was general co-chair for the first IEEE International Symposium on Biomedical Imaging (ISBI'2002), Washington, DC, July 7–10, 2002. He received the 1995 best paper award and the 2000 Magazine Award from the IEEE Signal Processing Society.