Generalized Interpolation: Higher Quality at no Additional Cost

Thierry Blu, Philippe Thévenaz and Michael Unser

Biomedical Imaging Group, DMT/IOA
Swiss Federal Institute of Technology, Lausanne
CH-1015 Lausanne EPFL, SWITZERLAND

{thierry.blu,philippe.thevenaz,michael.unser}@epfl.ch
http://bigwww.epfl.ch

Abstract— **We extend the classical interpolation method to generalized interpolation. This extension is done by replacing the interpolating function by a non-interpolating function that is applied to prefiltered data, in order to preserve the interpolation condition. We show, both theoretically and practically, that this approach performs much better than classical methods, for the same computational cost.**

I. Introduction

Interpolation techniques are essential in various applications of digital signal processing; in particular, in medical imaging: reslicing of MRI or CT data, tomographic reconstruction, unwarping, calibration, enlargement of images [1]. Other potential applications such as digital publishing also require interpolation algorithms. In biomedical imaging specifically, the emphasis is placed on high quality processing.

In this paper, we show how, what we call "generalized interpolation", outperforms classical interpolation methods which are based on compactly supported interpolating kernels. Our method gives much more flexibility for the design of the interpolator, whereas the computational cost increase remains negligible.

We give a quantitative comparison of the approximation accuracy of the classical interpolation kernels with kernels that are derived from non-interpolating functions through generalized interpolation. The results show that our method, combined with a careful kernel design, provides substantial quality increase for the same computational cost.

Most of the problems we are interested in are multidimensional by nature (images, volumes), but they can be brought back to one dimensional interpolation issues if we assume, as it will be the case here, that the interpolators are separable.

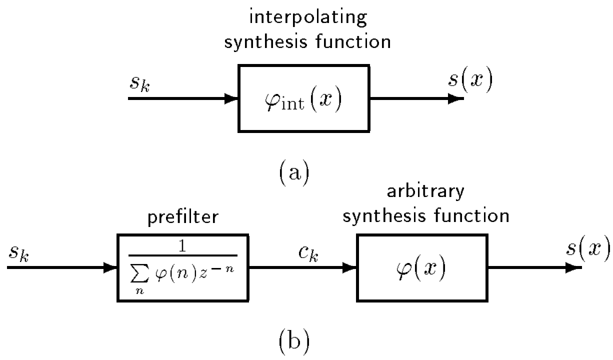

Fig. 1. Block diagram representation of the interpolation scheme: (a) classical method where φ_{int} is assumed to satisfy $\varphi_{\mathrm{int}}(n) = \delta_n$; (b) generalized interpolation method.

II. Generalized Sampling

We consider the problem of interpolating a discrete data sequence $\{s_k\}_{0\le k\le N-1}$ in a computationally efficient way. The usual method consists in defining a compactly supported interpolating function $\varphi_{\mathrm{int}}(x)$; i.e., a function that satisfies $\varphi_{\mathrm{int}}(k) = \delta_k$ for all integer k, and to build the approximating function $s(x)$ as

$$s(x) = \sum_{k\in\mathbb{Z}} s_k \varphi_{\mathrm{int}}(x-k) \tag{1}$$

This is, for example, Keys' approach [2]. Here the computational complexity depends only on the size of the support of $\varphi_{\mathrm{int}}(x)$.

The goal of this paper is to outline the computational and approximation advantages of the following slight modification of (1), namely

$$s(x) = \sum_{k\in\mathbb{Z}} c_k \varphi(x-k). \tag{2}$$

As for (1), we still require the interpolation condition $s(k) = s_k$. Thus, the c_k in (2) satisfy the following convolution equation $s_k = \varphi(k) * c_k$. This

TABLE I

Computation times for the rotation of a 512×512 image

B-spline degree	Data handling	Filter	Total	Ratio $\frac{\text{Filter}}{\text{Total time}}$
	times in seconds			
0	0.039	0	0.333	0%
1	0.039	0	0.467	0%
2	0.233	0.036	1	3.6%
3	0.233	0.036	1.217	3%
4	0.233	0.072	1.428	5%
5	0.233	0.072	1.689	4.3%
6	0.233	0.122	1.95	6.3%
7	0.233	0.122	2.261	5.4%

implies that the unknown coefficients c_k can be obtained by inverse filtering of the s_k's

$$c_k = h_k * s_k \tag{3}$$

where h_k is the impulse response of the filter

$$H(z) = \frac{1}{\sum_{k\in\mathbb{Z}} \varphi(k) z^{-k}}. \tag{4}$$

In effect, the interpolation constraint ensures that formulæ (1) and (2) are equivalent.

The interpolation procedure defined by (2) and (3) is called "generalized interpolation". Here, the function $\varphi(x)$ is still assumed to have compact support, but does not need to satisfy the interpolation constraints, which obviously provides additional degrees of freedom. We will show that this generalization can yield dramatically better approximation accuracy, at the additional cost of low-order prefiltering (3). We now give some arguments that show that this additional cost is negligible.

A. Implementation Notes

From the definition (2) it appears that the only computational addition to (1) is the prefiltering (3). This filter is symmetric if $\varphi(x)$ is chosen symmetric, which will be the case in general. However, it will also have an infinite impulse response, unless $\varphi(x)$ is interpolating: this calls for a proper extension of the samples s_k beyond the interval $0 \ldots N-1$. One extension that has most good properties (consistency with respect to symmetric filtering, "continuity") is the mirror boundary extension [3]. As a preliminary to the prefiltering step, we decompose $H(z)$ as follows

$$H(z) = C \prod_{k=1}^{K-1} \underbrace{\frac{1}{(1-z_k z)(1-z_k z^{-1})}}_{H_k(z)} \tag{5}$$

where $[-K, K]$ is the support of $\varphi(x)$. Then, we implement successively each $H_k(z)$ using a succession of *forward* and *backward* 1-pole recursions; this requires approximately 2 multiplications and 2 additions per input sample. The final cost of the prefiltering step is thus approximately $2(K-1)$ mults+adds per point, which is to be compared to the cost of evaluating φ at a $2K$ given points, as required by (1) or (2) for each output point. In general, the former cost is negligible compared to the latter, in particular, if we choose a small support function $\varphi(x)$; e.g., $K = 2$ for cubic B-splines. This cost gets even smaller in $D > 1$ dimensions, since the cost due to the prefilter increases linearly with D, while the cost of the evaluation of φ increases exponentially with D.

In support of our claim, Table I shows the repartition of this computation time for image rotation using splines of various degrees. More surprisingly, we observe that this cost is also negligible compared to the simple cost resulting from the handling of data before filtering. Note that, in our tests, this handling cost is 6 times higher for non interpolating splines (i.e., for degree ≥ 2) because the prefilter requires additional line and column manipulations.

III. Approximation Error

To evaluate the approximation error involved, we use a recent result [4] which expresses the $\mathbf{L}^2$ error as a scalar product between the squared Fourier transform of the signal and a Fourier kernel which depends on φ only. For the case where the s_k are the samples $f(k)$ of a well-behaved function $f(x)$ [4], [5], we have

$$\underbrace{\|f(x) - s(x)\|_{\mathbf{L}^2}}_{\varepsilon_f} = \underbrace{\sqrt{\int |\hat{f}(\omega)|^2 E(\omega) \frac{d\omega}{2\pi}}}_{\eta_f} + \rho_f \tag{6}$$

where

$$E(\omega) = \frac{\left| \sum_{n\in\mathbb{Z}\setminus\{0\}} \hat{\varphi}(\omega + 2n\pi) \right|^2 + \sum_{n\in\mathbb{Z}\setminus\{0\}} |\hat{\varphi}(\omega + 2n\pi)|^2}{\left| \sum_{n\in\mathbb{Z}} \hat{\varphi}(\omega + 2n\pi) \right|^2} \tag{7}$$

and where ρ_f is a usually small correction term. In fact, this term is such that $\varepsilon_f^2 - \eta_f^2$ cancels on the average when f is shifted, i.e.

$$\lim_{A\to\infty} \frac{1}{2A} \int_{-A}^{A} \varepsilon^2_{f(\cdot+\tau)}\, d\tau = \eta_f^2.$$

A similar result was also given in [6] for the particular case where $\varphi(x)$ satisfies the interpolation constraint, and where $f(x)$ is bandlimited.

As it is clear from (6), the general accuracy of the approximation is entirely given by the kernel $E(\omega)$. Thus, optimizing the approximation scheme for a given set of functions (here, the ones that satisfy Shannon's sampling theorem) is equivalent to minimize the Fourier kernel over the frequency support of those functions (here $[-\pi, \pi]$). This is the approach we adopted in a previous paper [7] where we exhibited the expression of compactly supported synthesis functions that minimize the Fourier kernel in the neighborhood of $\omega = 0$.

The plots of some Fourier kernels in Fig. 2 show that the "non-interpolating" synthesis functions perform significantly better than the interpolating ones. Note that, except for the linear spline and Dodgson's kernels, the synthesis functions have the same support and the same polynomial degree.

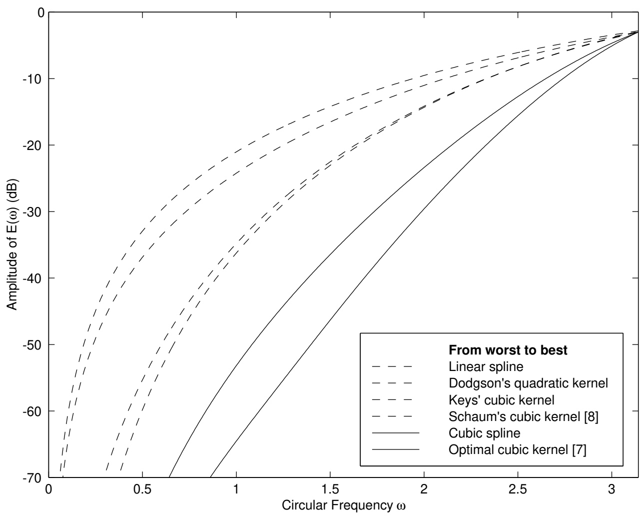

Fig. 2. Fourier approximation kernels of interpolating (dashed line) and non-interpolating (plain line) kernels. The results are significantly better (i.e., closer to the abscissa line) with generalized sampling.

The gap between the performance of generalized vs classical interpolation is even more impressive when we know that increasing the degree of the interpolating function is of little benefit to the performance of the interpolation [9]. This is not the case with generalized interpolation.

IV. Experiments

In order to test for the practical validity of the theoretical prediction of Fig. 2, we have rotated fifteen times an original 512×512 image by an angle of $\frac{2\pi}{15}$ radians. We have used the 6 interpolation methods (4 classical and 2 generalized interpolation methods) analyzed in Fig. 2 to implement the rotations. The central part of the resulting images is shown in Figs. 3 and 4 for each interpolation method. The concentric circle image is a simple but efficient test to analyze the selective frequency loss caused by the different methods: the closer to the center, the higher the frequencies. We observe that generalized sampling is characterized by a superior preservation of high frequencies. This is also manifest on Lena's image (better preservation of contours, especially on the hat).

The results are summerized in Table II: note that, especially for Lena, the SNR results give an imperfect account of the high frequency preservation

TABLE II

Numerical results of the rotation experiments

Synthesis function	SNR (dB) circles	SNR (dB) Lena
linear spline	7.1	26.9
Dodgson	9.4	27.32
Keys	15.0	27.73
Schaum	14.0	27.69
cubic spline	23.2	27.99
optimal cubic	32.7	28.08

V. Conclusion

We have shown the quality improvement brought by generalized interpolation as compared to the classical approach. What is especially interesting from a practical point of view is that this improvement can be achieved at no additional computation cost.

References

[1] J.A. Parker, R.V. Kenyon, and D.E. Troxel, "Comparison of interpolating methods for image resampling," *IEEE Trans. Med. Imag.*, vol. 2, pp. 31–39, 1983.

[2] R.G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech and Signal Process.*, vol. 29, no. 6, pp. 1153–1160, 1981.

[3] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part I—theory," *IEEE Trans. Signal Process.*, vol. 41, no. 2, pp. 821–832, February 1993.

[4] T. Blu and M. Unser, "Quantitative Fourier analysis of approximation techniques: Part I—interpolators and projectors," *IEEE Trans. Signal Process.*, 1999, To appear.

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

[6] 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, September 1982.

[7] T. Blu, P. Thévenaz, and M. Unser, "Minimum support interpolators with optimum approximation properties,"

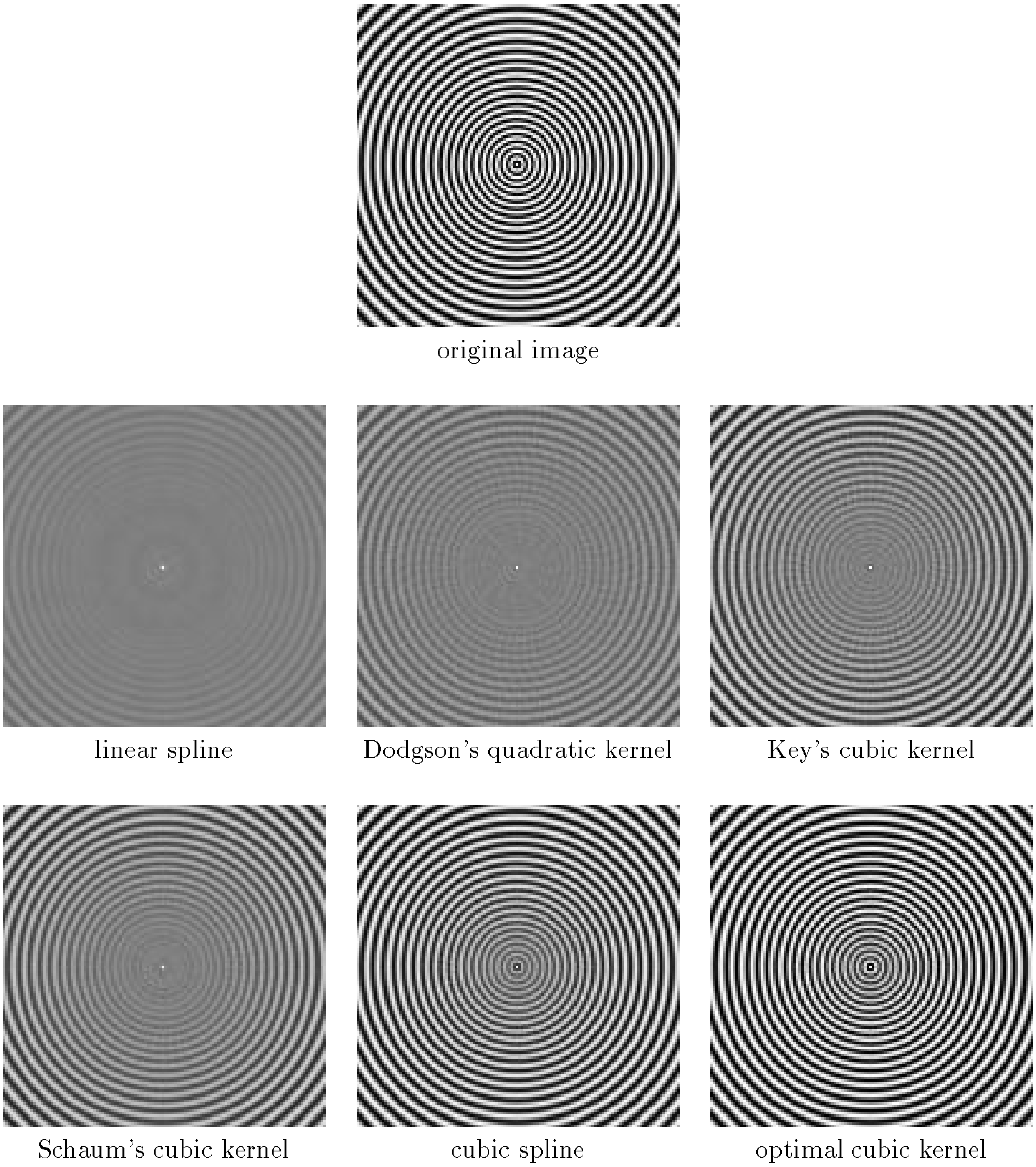

Fig. 3. Rotation experiment (15 rotations of the original image by $\frac{2\pi}{15}$ radians) using 6 different kernels: the "non-interpolating" kernels show the least noticeable artifacts

in *Proc. Int. Conf. Image Process.*, Chicago, IL, October 1998, vol. III, pp. 242–245.

[8] A. Schaum, "Theory and design of local interpolators," *Comput. Vis. Graph. Image Process.*, vol. 55, no. 6, pp. 464–481, November 1993.

[9] E.H.W. Meijering, K.J. Zuiderveld, and M.A. Viergever, "Image reconstruction by convolution with symmetrical piecewise nth-order polynomial kernels," *IEEE Trans. Image Process.*, vol. 8, no. 2, pp. 192–201, February 1999.

original image

linear spline

Dodgson's quadratic kernel

Key's cubic kernel

Schaum's cubic kernel

cubic spline

optimal cubic kernel

Fig. 4. Rotation experiment (15 rotations of the original image by $\frac{2\pi}{15}$ radians) using 6 different kernels: the "non-interpolating" kernels show the least noticeable artifacts