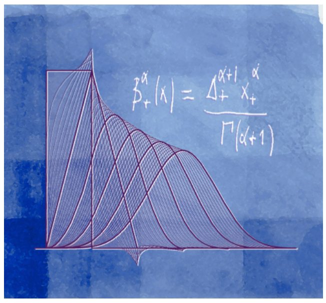
ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Sampling and interpolation for biomedical imaging: Part I

Michael Unser

Biomedical Imaging Group
EPFL, Lausanne
Switzerland



ISBI 2006, Tutorial, Washington DC, April 2006

INTRODUCTION

- **Fundamental issue in biomedical imaging**

  Linking the *discrete* and the *continuous*

- **Mismatch between theory and practice**
  - Theory : Shannon's sampling theorem
  - Practice: nearest neighbor, linear interpolation

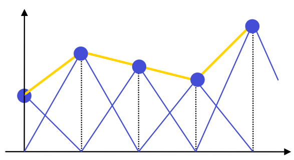
- **Limitations of Shannon sampling theory**
  - Ideal lowpass filters do not exist
  - Incompatible with finite support signals
  - Gibbs oscillations
  - Slow decay of $\mathrm{sinc}(x)$

- **Basic problem**

  How do you interpolate a signal ?

Acquisition

Algorithm design

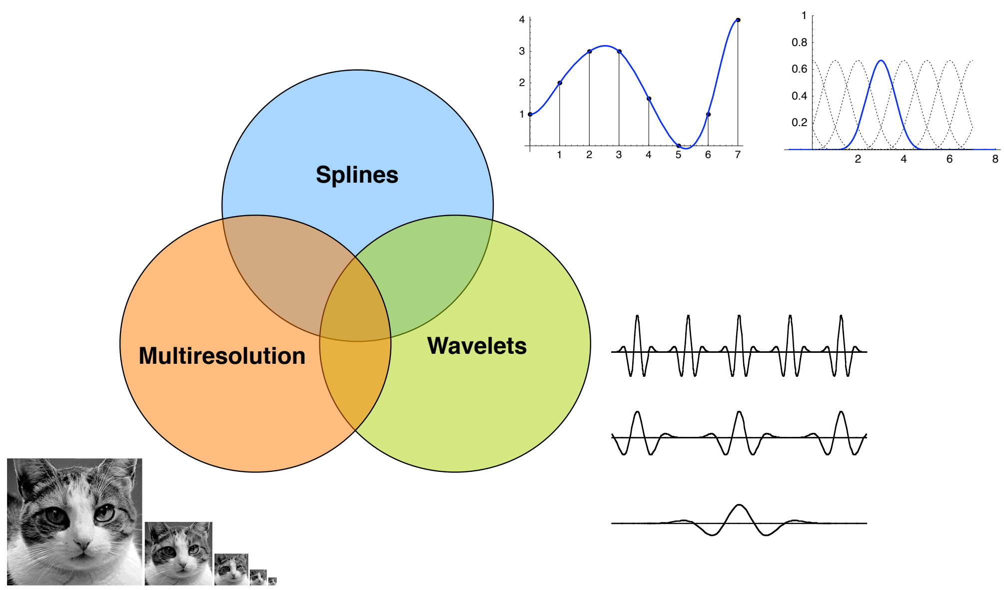


Interpolation and biomedical imaging

Image processing task	Specific operation	Imaging modality
Tomographic reconstruction	• Filtered backprojection • Fourier reconstruction • Iterative techniques • 3D + time	Commercial CT (X-rays) EM PET, SPECT Dynamic CT, SPECT, PET
Sampling grid conversion	• Polar-to-cartesian coordinates • Spiral sampling • k-space sampling • Scan conversion	Ultrasound (endovascular) Spiral CT, MRI MRI
Visualization	*2D operations* • Zooming, panning, rotation • Re-sizing, scaling	All
	• Stereo imaging • Range, topography	Fundus camera OCT
	3D operations • Re-slicing • Max. intensity projection • Simulated X-ray projection	CT, MRI, MRA
	Surface/volume rendering • Iso-surface ray tracing • Gradient-based shading • Stereogram	CT MRI
Geometrical correction	• Wide-angle lenses • Projective mapping • Aspect ratio, tilt • Magnetic field distortions	Endoscopy C-Arm fluoroscopy Dental X-rays MRI
Registration	• Motion compensation • Image subtraction • Mosaicking • Correlation-averaging • Patient positioning • Retrospective comparisons • Multi-modality imaging • Stereotactic normalization • Brain warping	fMRI, fundus camera DSA Endoscopy, fundus camera, EM microscopy Surgery, radiotherapy CT/PET/MRI
Feature detection	• Contours • Ridges • Differential geometry	All
	Contour extraction • Snakes and active contours	MRI, Microscopy (cytology)

Splines: a unifying framework

Linking the discrete and the continuous …..



Splines: bad press phenomenon

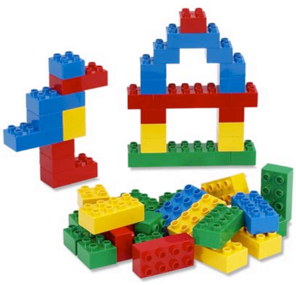
- Classical review article on interpolation, IEEE TMI, 1983
  Comparison of four interpolators:
  "The cubic B-spline provides the most smoothing."
- Classical book on Digital Image Processing, 1991 (2^{nd} ed)
  About high-order B-splines:
  "[out-of-band] interpolation error reduces significantly for higher-order interpolation functions, but at the expense of resolution error [i.e., distortion]"

⋮

- Recent book on Volume Rendering, 1998
  "The results of scaling the original image using [cubic] B-spline interpolation are shown in Figure 5.20. You can see the blurring effects ….."

CONTINUOUS/DISCRETE REPRESENTATION

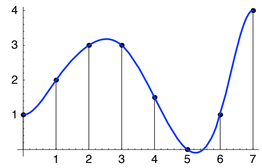
- Splines: definition
- Basic atoms: B-splines
- Riesz bases



Splines: definition

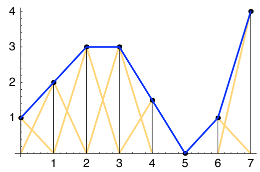
Definition: A function $s(x)$ is a polynomial spline of degree n with knots $\cdots < x_k < x_{k+1} < \cdots$ iff. it satisfies the following two properties:

- Piecewise polynomial:
  $s(x)$ is a polynomial of degree n within each interval $[x_k, x_{k+1})$;
- Higher-order continuity:
  $s(x), s^{(1)}(x), \cdots, s^{(n-1)}(x)$ are continuous at the knots x_k.



- Effective degrees of freedom per segment:

  $(n+1)$ (polynomial coefficients) $-$ n (constraints) $=$ 1



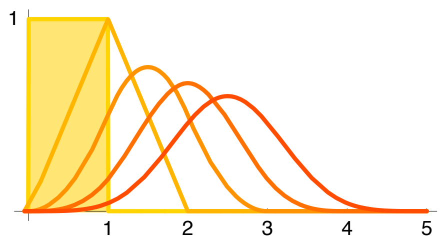
- **Cardinal splines** = unit spacing and infinite number of knots

The right framework for signal processing !

Polynomial B-splines

- B-spline of degree n

$$\beta_+^n(x) = \underbrace{\beta_+^0 * \beta_+^0 * \cdots * \beta_+^0(x)}_{(n+1)\text{ times}}$$

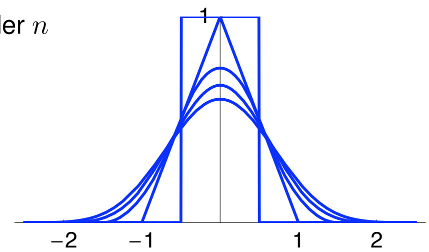


$$\beta_+^0(x) = \begin{cases} 1, & x \in [0,1) \\ 0, & \text{otherwise.} \end{cases}$$

- Key properties
  - Compact support: shortest polynomial spline of degree n
  - Positivity
  - Piecewise polynomial
  - Smoothness: Hölder-continuous of order n

- Symmetric B-spline

$$\beta^n(x) = \beta_+^n\left(x + \tfrac{n+1}{2}\right)$$



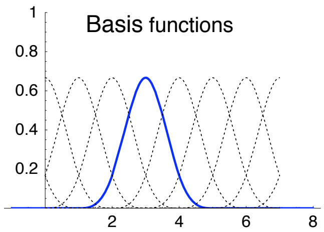
B-spline representation

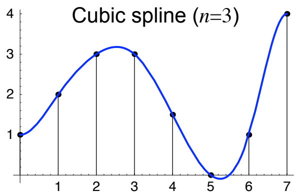
Theorem (Schoenberg, 1946)
Every cardinal polynomial spline $s(x)$ has a unique and stable representation in terms of its B-spline expansion

$$s(x) = \sum_{k\in\mathbb{Z}} c[k]\,\beta_+^n(x-k)$$

analog signal

discrete signal
(B-spline coefficients)



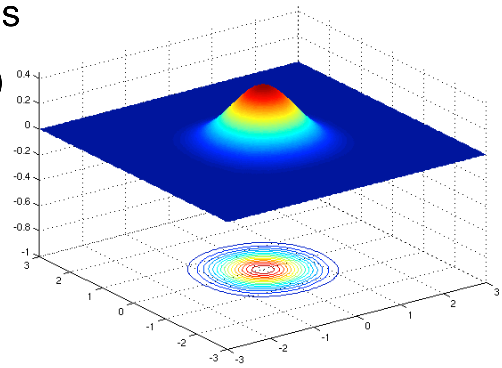


In modern terminology: $\{\beta_+^n(x-k)\}_{k\in\mathbb{Z}}$ forms a Riesz basis.

B-spline representation of images

- Symmetric, tensor-product B-splines

$$\beta^n(x_1,\cdots,x_d) = \beta^n(x_1)\times\cdots\times\beta^n(x_d)$$



- Multidimensional spline function

$$s(x_1,\cdots,x_d) = \sum_{(k_1,\cdots k_d)\in\mathbb{Z}^d} c[k_1,\cdots,k_d]\;\beta^n(x_1-k_1,\cdots,x_d-k_d)$$

continuous-space image

image array
(B-spline coefficients)

Compactly supported
basis functions

Riesz basis

Definition: Let $V = \mathrm{span}\{\varphi_k\}_{k\in\mathbb{Z}}$ be a subspace of a Hilbert space H. Then, $\{\varphi_k\}_{k\in\mathbb{Z}}$ is a Riesz basis of V iff. there exist two constants $A > 0$ and $B < +\infty$ s.t.

$$\forall c \in \ell_2,\quad A \cdot \|c\|_{\ell_2} \le \underbrace{\left\|\textstyle\sum_{k\in\mathbb{Z}} c_k\varphi_k\right\|_H}_{\|f\|_H} \le B \cdot \|c\|_{\ell_2}$$

Unique representation of a function $f \in V$: $\quad f = \sum_{k\in\mathbb{Z}} c_k\varphi_k$

Properties

- Linear independence
  Consequence of lower Riesz bound: $\quad f = 0 \Rightarrow c_k = 0$
- Stability
  Perturbation: $\quad c + \Delta c \longrightarrow f + \Delta f$
  Consequence of upper Riesz bound: $\quad \|\Delta c\|_{\ell_2}$ bounded $\Rightarrow$ $\|\Delta f\|_H$ bounded
- Norm equivalence
  The basis is orthonormal iff. $A = B = 1$, in which case, $\|c\|_{\ell_2} = \|f\|_H$

Shift-invariant spaces

Integer-shift-invariant subspace associated with a generating function φ (e.g. B-spline):

$$V(\varphi) = \left\{ f(\boldsymbol{x}) = \sum_{\boldsymbol{k}\in\mathbb{Z}^p} c[\boldsymbol{k}]\varphi(\boldsymbol{x}-\boldsymbol{k}) : c \in \ell_2(\mathbb{Z}^p) \right\}$$

Generating function: $\quad \varphi(\boldsymbol{x}) \quad \overset{\mathcal{F}}{\longleftrightarrow} \quad \hat{\varphi}(\boldsymbol{\omega}) = \int_{\boldsymbol{x}\in\mathbb{R}^p} \varphi(\boldsymbol{x}) e^{-j\langle\boldsymbol{\omega},\boldsymbol{x}\rangle} \mathrm{d}x_1 \cdots \mathrm{d}x_p$

Proposition. $V(\varphi)$ is a subspace of $L_2(\mathbb{R}^p)$ with $\{\varphi(\boldsymbol{x}-\boldsymbol{k})\}_{\boldsymbol{k}\in\mathbb{Z}^p}$ as its Riesz basis iff.

$$0 < A^2 \le \sum_{\boldsymbol{n}\in\mathbb{Z}^p} |\hat{\varphi}(\boldsymbol{\omega} + 2\pi\boldsymbol{n})|^2 \le B^2 < +\infty \quad \text{(almost everywhere)}$$

Hint for the proof (in 1D):

$$\|c\|_{\ell_2}^2 = \frac{1}{2\pi}\int_0^{2\pi} |C(e^{j\omega})|^2 \mathrm{d}\omega \quad \text{(Parseval)}$$

$$\begin{aligned} \|f\|_{L_2}^2 &= \frac{1}{2\pi}\int_{\omega\in\mathbb{R}} |C(e^{j\omega})|^2 |\hat{\varphi}(\omega)|^2 \mathrm{d}\omega \\ &= \frac{1}{2\pi}\sum_{n\in\mathbb{Z}}\int_0^{2\pi} |C(e^{j\omega})|^2 |\hat{\varphi}(\omega+2\pi n)|^2 \mathrm{d}\omega = \frac{1}{2\pi}\int_0^{2\pi} |C(e^{j\omega})|^2 \sum_{n\in\mathbb{Z}} |\hat{\varphi}(\omega+2\pi n)|^2 \mathrm{d}\omega \end{aligned}$$

INTERPOLATION REVISITED

- Classical interpolation
- Generalized interpolation
- Interpolation: filtering solution
- Application

Classical image interpolation

Discrete image data
$f[\boldsymbol{k}], \quad \boldsymbol{k} = (k_1, \cdots, k_p) \in \mathbb{Z}^p$

$\Longleftrightarrow$

Continuous image model
$f(\boldsymbol{x}), \quad \boldsymbol{x} = (x_1, \cdots, x_p) \in \mathbb{R}^p$

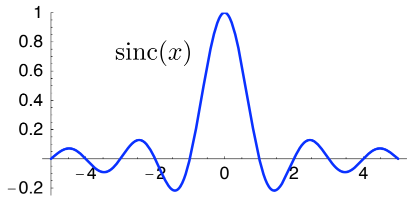
- Interpolation formula: $f(\boldsymbol{x}) = \sum_{\boldsymbol{k} \in \mathbb{Z}^p} f[\boldsymbol{k}]\, \varphi_{\text{int}}(\boldsymbol{x} - \boldsymbol{k})$
  - $f[\boldsymbol{k}]$: pixel values at location $\boldsymbol{k}$
  - $\varphi_{\text{int}}(\boldsymbol{x})$: continuous-space interpolation function
  - $\varphi_{\text{int}}(\boldsymbol{x} - \boldsymbol{k})$: interpolation function translated to location $\boldsymbol{k}$
- Interpolation condition

  At the grid points $\boldsymbol{x} = \boldsymbol{k}_0$: $f(\boldsymbol{k}_0) = \sum_{\boldsymbol{k} \in \mathbb{Z}^p} f[\boldsymbol{k}]\, \varphi_{\text{int}}(\boldsymbol{k}_0 - \boldsymbol{k})$

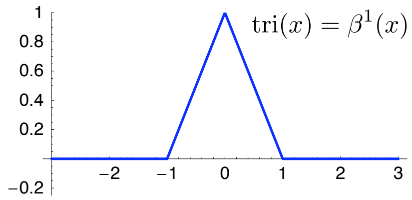
  Only possible for all f iff. $\varphi_{\text{int}}(\boldsymbol{k}) = \begin{cases} 1, & \boldsymbol{k} = \boldsymbol{0} \\ 0, & \text{otherwise} \end{cases}$

Examples of popular interpolation functions

- Bandlimited



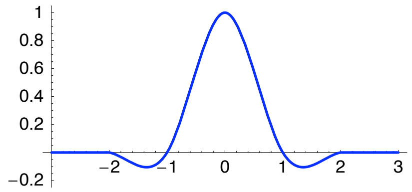
- Piecewise linear



Interpolation condition:

$$\varphi_{\text{int}}(k) = \delta_k = \begin{cases} 1, & k = 0 \\ 0, & \text{otherwise} \end{cases}$$

- Cubic convolution



[Keys, 1981; Karup-King 1899]

Generalized image interpolation

- Desired features for the interpolation kernel
  - short (to minimize computations)
  - simple expression (e.g., polynomial)
  - smooth (to avoid model discontinuities)
  - good approximation properties: reproduction of polynomials

- Generalized interpolation formula: $f(\boldsymbol{x}) = \sum_{\boldsymbol{k} \in \mathbb{Z}^p} c[\boldsymbol{k}]\, \varphi(\boldsymbol{x} - \boldsymbol{k})$
  - Simple shift-invariant structure
  - simple expression (e.g., polynomial)
  - φ selected freely (not interpolating and much shorter)

⇨ Faster interpolation formulas!

but one new difficulty:

How to pre-compute the coefficients $c[\boldsymbol{k}]$?

- Separable basis functions: $\varphi(\boldsymbol{x}) = \varphi(x_1) \cdot \varphi(x_2) \cdots \varphi(x_p)$

⇨ Further acceleration

Interpolation: filtering solution

Interpolation problem: Given the samples $\{f[\boldsymbol{k}]\}$, find the (B-spline) expansion coefficients $\{c[\boldsymbol{k}]\}$

- Interpolation condition: $f(\boldsymbol{x})|_{\boldsymbol{x}=\boldsymbol{k}} = f[\boldsymbol{k}] = \sum_{\boldsymbol{k}_1 \in \mathbb{Z}^p} c[\boldsymbol{k}_1]\varphi(\boldsymbol{k}-\boldsymbol{k}_1)$

⇨ Discrete convolution equation: $f[\boldsymbol{k}] = (b * c)[\boldsymbol{k}]$

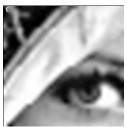
with $b[\boldsymbol{k}] \triangleq \varphi(\boldsymbol{k}) \quad \overset{z}{\longleftrightarrow} \quad B(\mathbf{z}) = \sum_{\boldsymbol{k}\in\mathbb{Z}^p} b[\boldsymbol{k}]\mathbf{z}^{-\boldsymbol{k}}$

- Inverse filtering solution

$f[\boldsymbol{k}]$ → Digital filter → $c[\boldsymbol{k}] = (h_{\text{int}} * f)[\boldsymbol{k}]$ with $H_{\text{int}}(\mathbf{z}) = \dfrac{1}{B(\mathbf{z})} = \dfrac{1}{\sum_{\boldsymbol{k}\in\mathbb{Z}^p}\varphi(\boldsymbol{k})\mathbf{z}^{-\boldsymbol{k}}}$

Note: $\varphi(\boldsymbol{x})$ separable $\Rightarrow$ $h_{\text{int}}[\boldsymbol{k}]$ separable

One-to-one continuous/discrete representation



Continuously defined signal

$f(\boldsymbol{x}) = \sum_{\boldsymbol{k}\in\mathbb{Z}^p} c[\boldsymbol{k}]\varphi(\boldsymbol{x}-\boldsymbol{k})$

⇔ Riesz-basis property

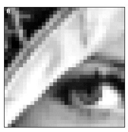
B-spline coefficients

$c[\boldsymbol{k}]$

Digital filtering

$* \, b$ (FIR) ↓ ↑ $* \, h_{\text{int}}$ (IIR)

Sampling: $f(\boldsymbol{x})|_{\boldsymbol{x}=\boldsymbol{k}}$ → $f[\boldsymbol{k}]$



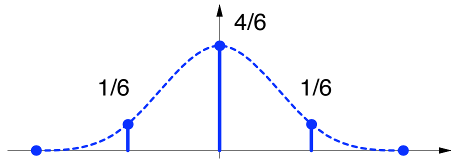
Discrete signal

In principle, all φ's are equally acceptable, but...

Example: cubic-spline interpolation

- Cubic B-spline

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



- Discrete B-spline kernel: $B(z) = \dfrac{z + 4 + z^{-1}}{6}$

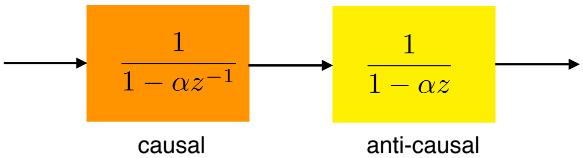
- Interpolation filter

$$\frac{6}{z+4+z^{-1}} = \frac{(1-\alpha)^2}{(1-\alpha z)(1-\alpha z^{-1})} \quad \stackrel{z}{\longleftrightarrow} \quad h_{\text{int}}[k] = \left(\frac{1-\alpha}{1+\alpha}\right)\alpha^{|k|}$$

(symmetric exponential)

$$\alpha = -2 + \sqrt{3} = -0.171573$$

Cascade of first-order recursive filters



Generic C-code (splines of any degree n)

- Main recursion

```
void  ConvertToInterpolationCoefficients (
          double c[ ],  long DataLength,   double  z[ ],  long NbPoles,  double  Tolerance)
{double Lambda = 1.0; long n, k;
   if (DataLength == 1L) return;
   for (k = 0L; k < NbPoles; k++) Lambda = Lambda * (1.0 - z[k]) * (1.0 - 1.0 / z[k]);
   for (n = 0L; n < DataLength; n++) c[n] *= Lambda;
   for (k = 0L; k < NbPoles; k++) {
      c[0] = InitialCausalCoefficient(c, DataLength, z[k], Tolerance);
      for (n = 1L; n < DataLength; n++) c[n] += z[k] * c[n - 1L];
      c[DataLength - 1L] = (z[k] / (z[k] * z[k] - 1.0))
            * (z[k] * c[DataLength - 2L] + c[DataLength - 1L]);
      for (n = DataLength - 2L; 0 <= n; n--) c[n] = z[k] * (c[n + 1L]- c[n]); }
}
```

- Initialization

```
double  InitialCausalCoefficient (
          double  c[ ],  long   DataLength,  double  z,  double  Tolerance)
{  double Sum, zn, z2n, iz; long n, Horizon;
   Horizon = (long)ceil(log(Tolerance) / log(fabs(z)));
   if (DataLength < Horizon) Horizon = DataLength;
   zn = z; Sum = c[0];
   for (n = 1L; n < Horizon; n++) {Sum += zn * c[n]; zn *= z;}
   return(Sum);
}
```

Interpolating basis function

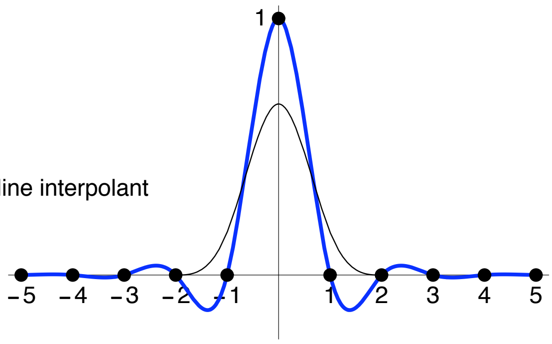
- Equivalent interpretation of generalized interpolationn

$$f(x) = \sum_{k\in\mathbb{Z}} c[k]\varphi(x-k) = \sum_{k\in\mathbb{Z}} (f[k] * h_{\text{int}}[k])\,\varphi(x-k)$$
$$= \sum_{k\in\mathbb{Z}} f[k]\;\varphi_{\text{int}}(x-k)$$

- Interpolation basis function

$$\varphi_{\text{int}}(x) = \sum_{k\in\mathbb{Z}} h_{\text{int}}[k]\;\varphi(x-k)$$

Example: cubic-spline interpolant



Finite-cost implementation of an infinite impulse response interpolator !

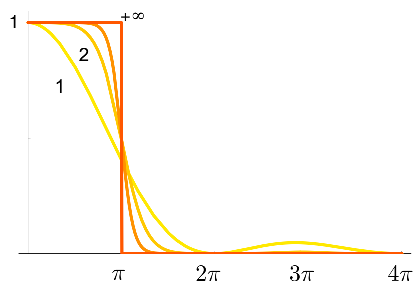
Limiting behavior (splines)

- Spline interpolator

Impulse response

Frequency response

$$\varphi^n_{\text{int}}(x) \quad \stackrel{\mathcal{F}}{\longleftrightarrow} \quad \hat{\varphi}^n_{\text{int}}(\omega) = \left(\frac{\sin(\omega/2)}{\omega/2}\right)^{n+1} H^n_{\text{int}}(e^{j\omega})$$



- Asymptotic property

The cardinal spline interpolators converge to the sinc-interpolator (ideal filter) as the degree goes to infinity:

$$\lim_{n\to\infty} \varphi^n_{\text{int}}(x) = \text{sinc}(x), \quad \lim_{n\to\infty} \hat{\varphi}^n_{\text{int}}(\omega) = \text{rect}\left(\frac{\omega}{2\pi}\right)$$ (in all L_p-norms)

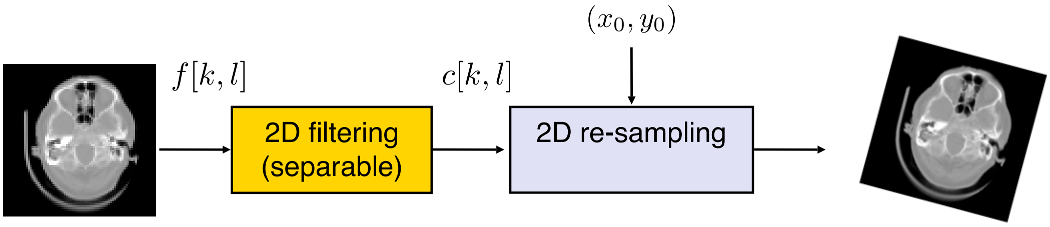
(Aldroubi et al., *Sig. Proc.*, 1992)

Includes Shannon's theory as a particular case !

Geometric transformation of images

- 2D separable model

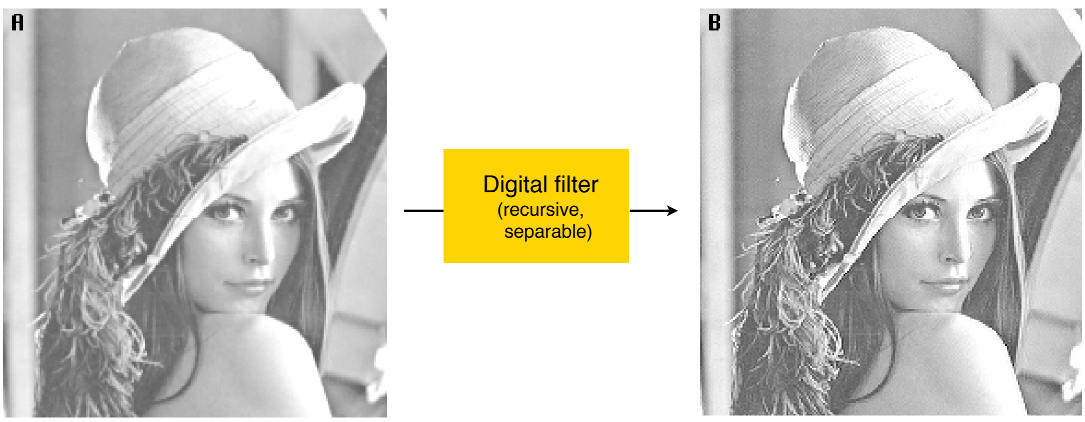
$$f(x_0, y_0) = \sum_{k=k_0(x_0)}^{k_0+n+1} \sum_{l=l_0(y_0)}^{l_0+n+1} c[k,l]\, \varphi(x_0 - l)\, \varphi(y_0 - l)$$



- Applications

zooming, rotation, re-sizing, re-formatting, warping

Cubic-spline coefficients in 2D



Pixel values $f[k,l]$

B-spline coefficients $c[k,l]$

Interpolation benchmark

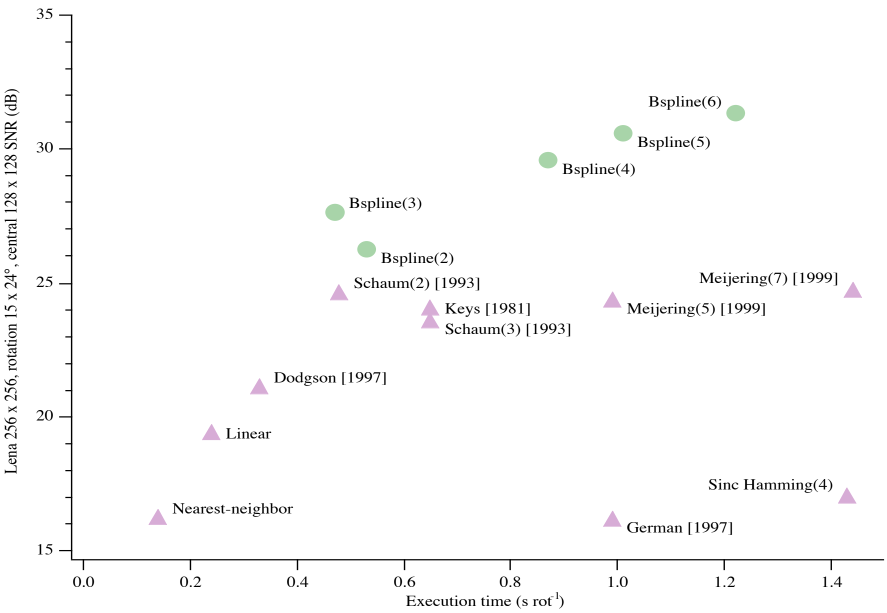
Cumulative rotation experiment: the best algorithm wins !


Bilinear


Windowed-sinc


Cubic spline

High-quality image interpolation



Demo

Thévenaz et al., *Handbook of Medical Image Processing*, 2000

MINIMUM-ERROR SIGNAL APPROXIMATION

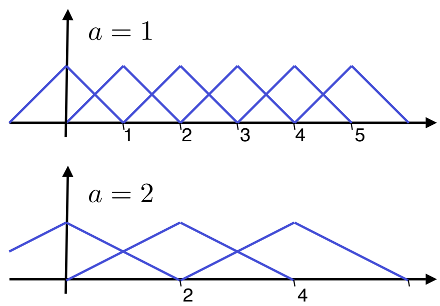
- Least-squares approximation
  - Orthogonal projection
- Image pyramids

Least-squares fit: multi-scale approximation

- Shift-invariant space at scale a

$$V_a(\varphi) = \left\{ s(x) = \sum_{k \in \mathbb{Z}} c[k]\varphi_a(x - ak) : c[k] \in \ell_2 \right\}$$

  - Rescaled basis function: $\varphi_a(x) \triangleq a^{-1/2}\varphi\left(\frac{x}{a}\right)$



- Minimum-error approximation at scale a

Continuous-space input $f(x)$

$c[k] = \langle f, \tilde{\varphi}_a(\cdot - ak)\rangle_{L_2}$

such that $\min_{s \in V_a} \|f - s\|^2_{L_2}$

Biorthogonality condition: $\tilde{\varphi}_a \in V_a(\varphi)$ such that $\langle \varphi_a(\cdot),\ \tilde{\varphi}_a(\cdot - ak)\rangle_{L_2} = \delta_k$

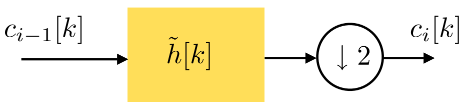
Image pyramids

- Successive approximations at dyadic scales

$$V_{2^i}(\varphi) = \left\{ s(x) = \sum_{k \in \mathbb{Z}} c_i[k] \varphi_{2^i}(x - 2^i k) : c_i[k] \in \ell_2 \right\}$$

Rescaled basis function: $\varphi_{2^i}(x) \triangleq 2^{-i/2} \varphi\left(\frac{x}{2^i}\right)$

- Repeated application of REDUCE operator



- Optimal prefilter

$$c_1[k] = \langle \sum_{l \in \mathbb{Z}} c_0[l] \varphi(\cdot - l),\ \tilde{\varphi}_2(\cdot - 2k) \rangle = (c_0 * \tilde{h})[2k]$$

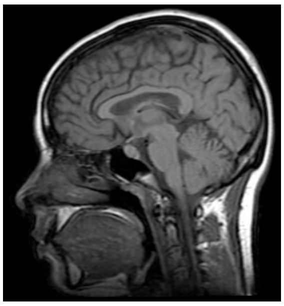
$$\Rightarrow \quad \tilde{h}[k] = \langle \varphi(\cdot),\ \tilde{\varphi}_2(\cdot + k) \rangle$$

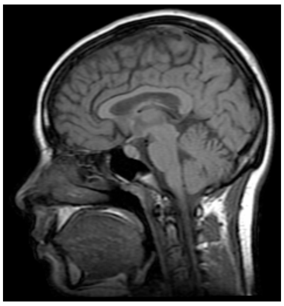
SPLINES: IMAGING APPLICATIONS

- Sampling and interpolation
  - Interpolation, re-sampling, grid conversion
  - Image reconstruction
  - Geometric correction
- Feature extraction
  - Contours, ridges
  - Differential geometry
  - Shape and active contour models
- Image matching
  - Stereo
  - Image registration (multimodal, rigid-body or elastic)
  - Optical flow

Spline approximation: LS resizing

Approximation at arbitrary scales: differential approach using splines





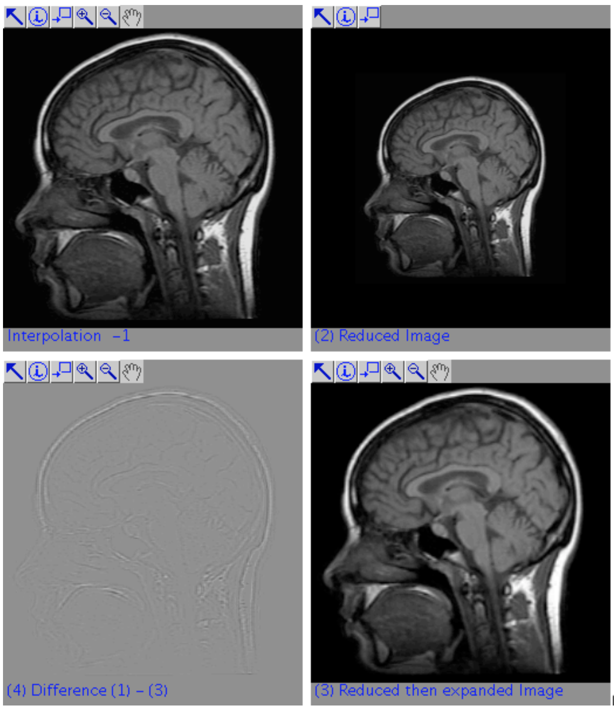
Orthogonal projection onto V_a (cubic spline)

$$a = 1 \to 10$$

Application: image resizing

- Resizing algorithm
  - Interpolation
  - Linear splines
  - scaling= 70%

SNR=22.94 dB

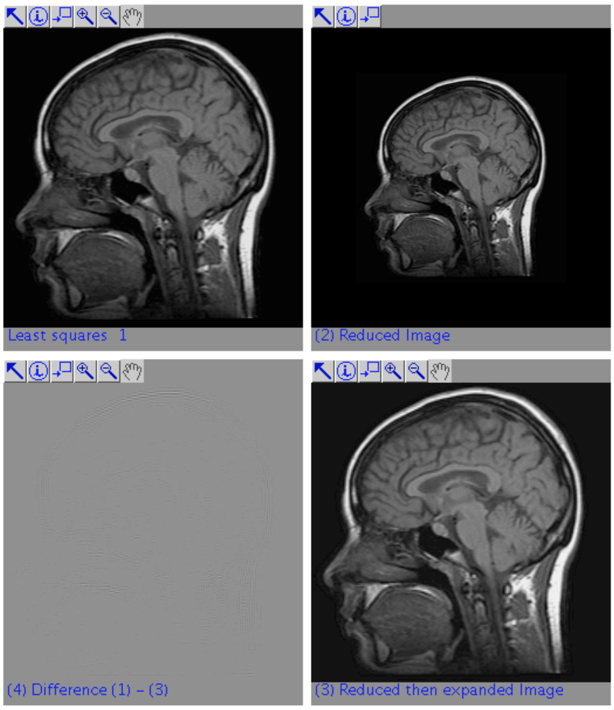


Application: image resizing (LS)

- Resizing algorithm
  - Orthogonal projector
  - Linear splines
  - scaling= 70%

SNR=28.359 dB

+ 5.419 dB



(Munoz et al., *IEEE Trans. Imag. Proc*, 2001)

B-spline derivatives

- Derivative operator

$$\mathrm{D}f(x) = \frac{df(x)}{dx} \quad \stackrel{\mathcal{F}}{\longleftrightarrow} \quad (j\omega) \times \hat{f}(\omega)$$

- Finite-difference operator (centered)

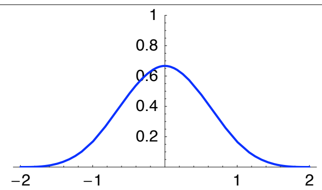
$$\Delta f(x) \stackrel{\Delta}{=} f(x + \tfrac{1}{2}) - f(x - \tfrac{1}{2}) \quad \stackrel{\mathcal{F}}{\longleftrightarrow} \quad (e^{j\omega/2} - e^{-j\omega/2}) \times \hat{f}(\omega)$$

- Derivative of a B-spline (exact)

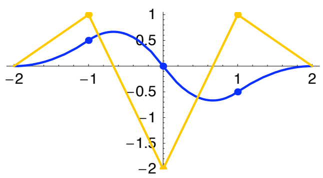
$$\mathrm{D}^m \beta^n(x) = \Delta^m \beta^{n-m}(x)$$

Discrete operator

Reduction of degree



Example: cubic B-spline



Sketch of proof:

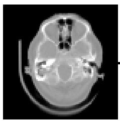
$$\hat{\beta}^n(\omega) = \mathrm{sinc}\left(\frac{\omega}{2\pi}\right)^{n+1} = \left(\frac{e^{j\omega/2} - e^{-j\omega/2}}{j\omega}\right)^{n+1}$$

$$\Rightarrow (j\omega)^m \times \hat{\beta}^n(\omega) = (e^{j\omega/2} - e^{-j\omega/2})^m \times \left(\frac{e^{j\omega/2} - e^{-j\omega/2}}{j\omega}\right)^{n+1-m}$$

Cubic-spline image differentials

- Convolution-based implementation

$f(k,l)$



→ 2D filtering (separable) → $c[k,l]$ → Differential mask → $\frac{\partial^{p+q}}{\partial x^p \partial y^q} f(k,l)$

Differential mask: $\left.\frac{\partial^{p+q}}{\partial x^p \partial y^q}\varphi(x,y)\right|_{x=k,y=l}$

JAVA code available: http://bigwww.epfl.ch/

- Laplacian

$$\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2} : \frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

- Hessian masks

$$\frac{\partial^2}{\partial x^2} : \frac{1}{6}\begin{bmatrix} 1 & -2 & 1 \\ 4 & -8 & 4 \\ 1 & -2 & 1 \end{bmatrix} \qquad \frac{\partial^2}{\partial x \partial y} : \frac{1}{2\cdot 2}\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

$$\frac{\partial^2}{\partial y^2} : \frac{1}{6}\begin{bmatrix} 1 & 4 & 1 \\ -2 & -8 & -2 \\ 1 & 4 & 1 \end{bmatrix}$$

- Gradient masks

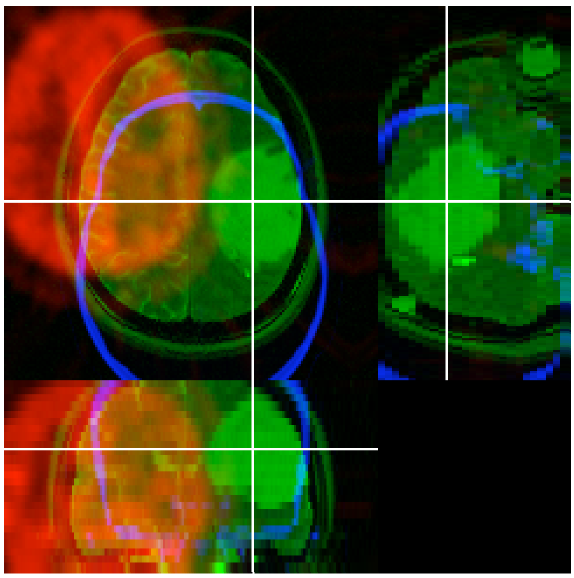
$$\frac{\partial}{\partial x} : \frac{1}{6\cdot 2}\begin{bmatrix} -1 & 0 & 1 \\ -4 & 0 & 4 \\ -1 & 0 & 1 \end{bmatrix}$$

$$\frac{\partial}{\partial y} : \frac{1}{6\cdot 2}\begin{bmatrix} -1 & -4 & -1 \\ 0 & 0 & 0 \\ 1 & 4 & 1 \end{bmatrix}$$

Multi-modal image registration

Specificities of the approach

- Criterion: mutual-information
- Cubic-spline model
  - high quality
  - sub-pixel accuracy
- Multiresolution strategy
- Marquardt-Levenberg-like optimizer
  - Speed
  - Robustness



Thévenaz and Unser, *IEEE Trans. Imag Proc*, 2000

CONCLUSION

- Generalized interpolation
  - Same as standard interpolation, except for a **prefiltering** step
  - Offers more flexibility
  - Best cost/performance tradeoff (splines)
  - Infinite-support interpolator at finite cost
- Special case of polynomial splines
  - Simple to manipulate
  - Smooth and well-behaved
  - Excellent approximation properties
  - Multiresolution properties
- Unifying formulation for continuous/discrete image processing
  - Tools: digital filters, convolution operators
  - Efficient recursive filtering solutions
  - Flexibility: piecewise-constant to bandlimited

Splines: the end of the tunnel

- Survey article on interpolation, *IEEE TMI*, 2000
  Comparison of 31 interpolation algorithms:
  "It [the cubic B-spline interpolator] produces one of the best results in terms of similarity to the original images, and of the top methods, it runs fastest."
- Addendum on spline interpolation, *IEEE TMI*, 2001
  "Therefore, high-degree B-splines are preferable interpolators for numerous applications in medical imaging, particularly if high precision is required."
  (Lehmann et *al*)
- Recent evaluation of interpolation, *Med. Image Anal.*, 2001
  Comparison of 126 interpolation algorithms:
  " The results show that spline interpolation is to be preferred over all other methods, both for its accuracy and its relatively low cost."

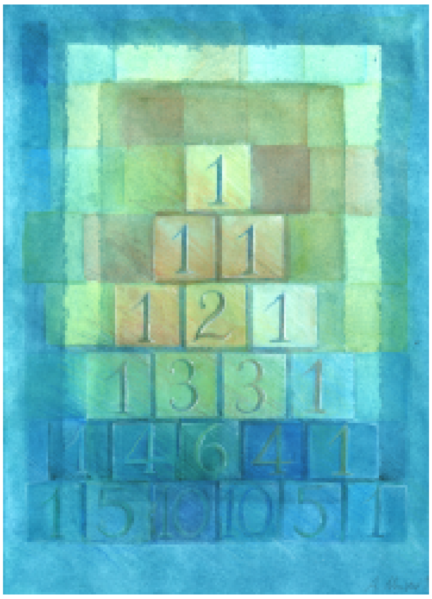
(Meijering et *al*)

Acknowledgments

Many thanks to

- Dr. Thierry Blu
- Prof. Akram Aldroubi
- Prof. Murray Eden
- Dr. Philippe Thévenaz
- Annette Unser, Artist

\+ many other researchers, and graduate students



Bibliography

- Spline tutorial
  - M. Unser, "Splines: A Perfect Fit for Signal and Image Processing," *IEEE Signal Processing Magazine*, vol. 16, no. 6, pp. 22-38, 1999.
- Pyramids and resizing
  - M. Unser, A. Aldroubi, M. Eden, "The L_2-Polynomial Spline Pyramid," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 364-379, April 1993.
  - A. Muñoz Barrutia, T. Blu, M. Unser, "Least-Squares Image Resizing Using Finite Differences," *IEEE Trans. Image Processing*, vol. 10, no. 9, pp. 1365-1378, September 2001.
- Generalized interpolation
  - P. Thévenaz, T. Blu, M. Unser, "Interpolation Revisited," *IEEE Trans. Medical Imaging*, vol. 19, no. 7, pp. 739-758, July 2000.
- Preprints and demos: **http://bigwww.epfl.ch/**