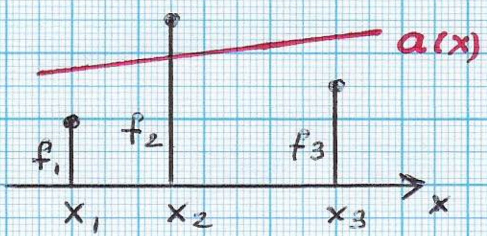
Stratovan

DISCRETE versus CONTINUOUS Least Squares Approximation

→ Relevance: BEST approximation of recorded CT projection data

i) DISCRETE Case



"Minimize sum of squared differences d_i":

$$d_i^2 = (a(x_i) - f_i)^2$$

$$E = \sum_i (a(x_i) - f_i)^2$$

Example: $a(x) = c_1 x + c_0 \Rightarrow E = \sum_i (c_1 x_i + c_0 - f_i)^2$

Minimize E: $\frac{\partial}{\partial c_i} E = 0$, all i

$$\frac{\partial}{\partial c_1} E = \sum_i 2 (c_1 x_i + c_0 - f_i) x_i = 0$$

$$\Rightarrow \sum_i (c_1 x_i^2 + c_0 x_i) = \sum_i f_i x_i$$

$$\frac{\partial}{\partial c_0} E = \sum_i 2 (c_1 x_i + c_0 - f_i) \cdot 1 = 0$$

$$\Rightarrow \sum_i (c_1 x_i + c_0) = \sum_i f_i \cdot 1$$

$\Rightarrow$ Matrix form ($\mathbb{c} = \begin{pmatrix} c_1 \\ c_0 \end{pmatrix}$):

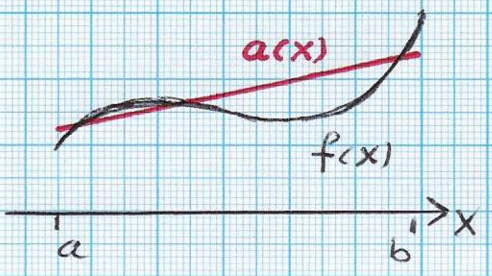
$$\begin{pmatrix} x_1 & x_2 & x_3 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \end{pmatrix} \mathbb{c} = \begin{pmatrix} x_1 & x_2 & x_3 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix}$$

$$X^T \quad X \quad \mathbb{c} = X^T \mathbb{f}$$

$$\mathbb{c} = (X^T X)^{-1} X^T \mathbb{f}$$

...

ii) CONTINUOUS Case



Here:

$$d^2(x) = \left(a(x) - f(x)\right)^2$$

$$E = \int_a^b d^2(x)\, dx$$

$$= \int \left(a(x) - f(x)\right)^2 dx$$

Example: $a(x) = \sum_{i=1}^{n} c_i b_i(x) \Rightarrow E = \int \left(\sum_i c_i b_i(x) - f(x) \right)^2 dx$

Minimize E: $\frac{\partial}{\partial c_j} E = \int 2 \left(\sum_i c_i b_i(x) - f(x) \right) b_j(x)\, dx = 0$

$$\Rightarrow \int \sum_i c_i b_i(x)\, b_j(x)\, dx = \int f(x)\, b_j(x)\, dx, \quad j = 1 \ldots n$$

$\Rightarrow$ Matrix form (with $\langle f, g \rangle = \int_a^b f(x)\, g(x)\, dx$):

$$\begin{pmatrix} \langle b_1, b_1 \rangle & \cdots & \langle b_1, b_n \rangle \\ \vdots & & \\ \langle b_n, b_1 \rangle & \cdots & \langle b_n, b_n \rangle \end{pmatrix} \mathbb{C} = \begin{pmatrix} \langle f, b_1 \rangle \\ \vdots \\ \langle f, b_n \rangle \end{pmatrix}$$

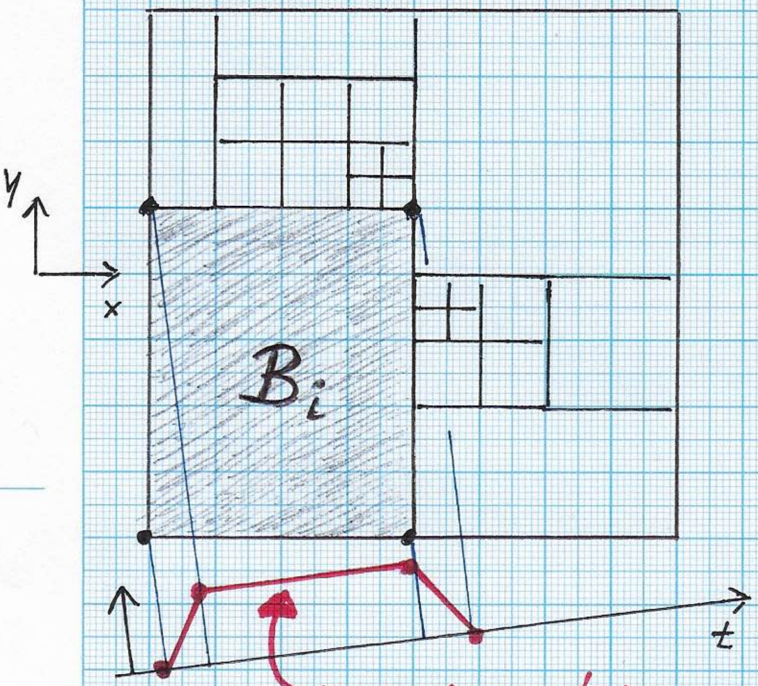
$$\mathbb{B} \quad \mathbb{C} = \mathbb{f}$$

$$\underline{\underline{\mathbb{C} = \mathbb{B}^{-1} \mathbb{f}}}$$

- To compute an approximation $a(x)$ for the recorded DISCRETE data $\{f_i\}$ one can directly compute $a(x)$, or construct a CONTINUOUS function $f(x)$ first and then compute $a(x)$.

■ ADAPTIVE MULTIRESOLUTION (AMR) RECONSTRUCTION AND LOCALIZATION

→ Examples illustrated for 2D case, parallel projection:



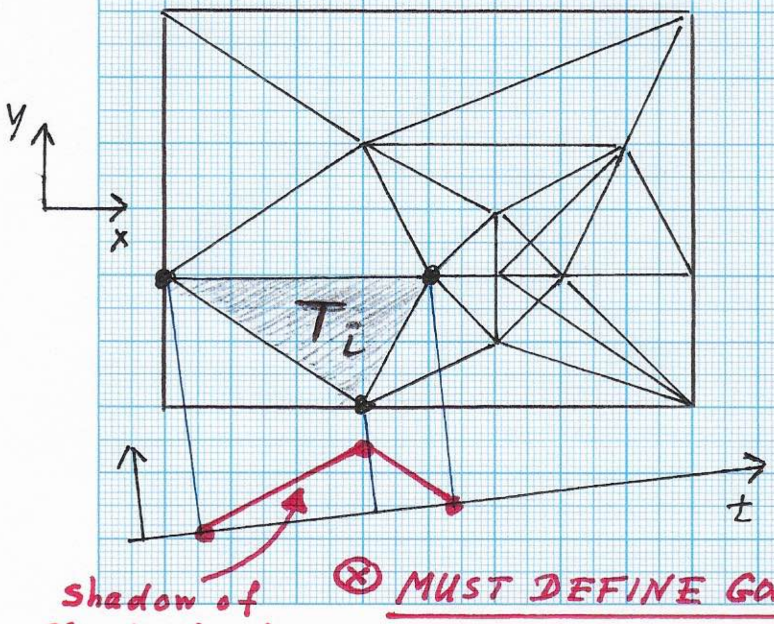
- Reconstruction domain = tessellation consisting of boxes B_i (rectangles in 2D, cuboids in 3D) or simplices T_i (triangles in 2D, tetrahedra in 3D)

- Constant box/simplex function:
  $B_i(\mathbf{x}) = 1$ if $\mathbf{x} \in B_i$ (else 0)
  $T_i(\mathbf{x}) = 1$ if $\mathbf{x} \in T_i$ (else 0)

- Piecewise constant recon.:

$$I(\mathbf{x}) = \sum_i c_i B_i(\mathbf{x}) \quad (2D)$$

$$\left(= \sum_i c_i T_i(\mathbf{x}) \ (3D) \right)$$



- Efficient refinement:
  i) Compute initial recon $I(\mathbf{x})$ using low-res. mesh
  ii) Refine mesh locally, compute new recon $I(\mathbf{x})$ — refine until $I(\mathbf{x})$ satisfies "quality threshold" ⊗

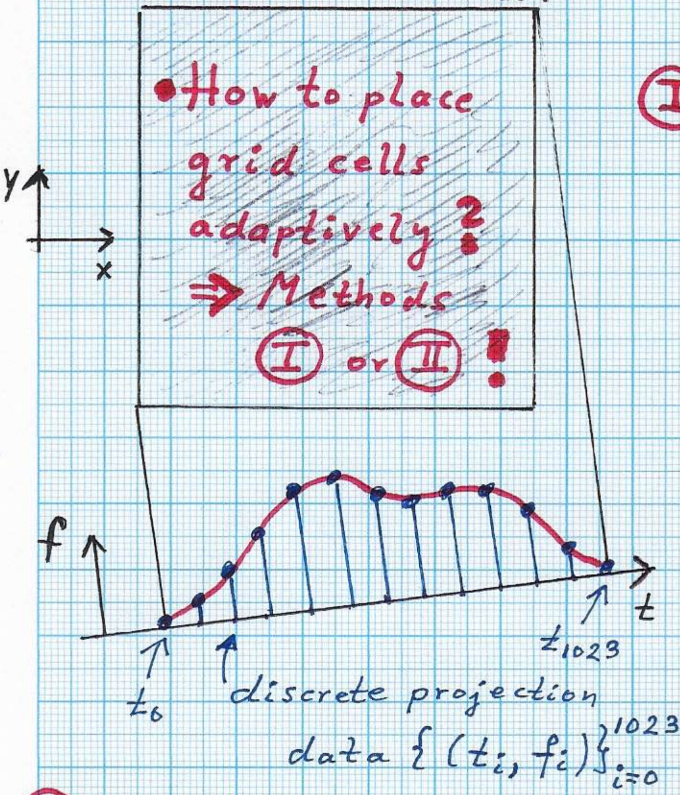
⊗ MUST DEFINE GOOD QUALITY/ERROR METRIC!

~BH

AMR RECONSTRUCTION AND LOCALIZATION – Cont'd.

→ Method for reconstruction grid with varying grid cell density and resolution needed!
(Principles apply to parallel and perspective projections.)

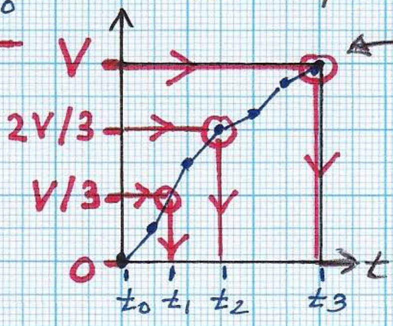
entire recon. domain



I • Compute TOTAL VARIATION of recorded projection data:

$$V = \sum_i |f_{i+1} - f_i|$$

• Goal: relatively higher resolution in regions of higher variation ⇒ "Distribute V uniformly with respect to t-axis."



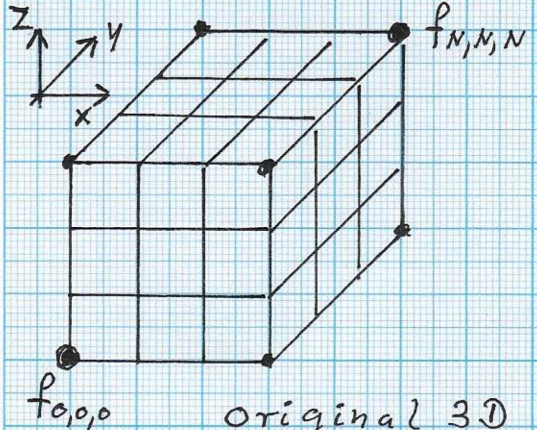
Ex: compute values t_j, $j=0..3$, such that each interval $[t_j, t_{j+1}]$ has same value $V/3$ associated with it.

• Use this distribution/density of t_j-values to "guide" grid cell density in xy-space.

• Consider all projections, e.g., 720.

II
• Use coarse initial grid and compute first reconstr.
• Forward-project reconstr. and compute error between its computed projections and the real projections
• Determine where the error is (locally) relatively larger and use this insight to refine xy-grid.
• Refine until error < ε.

■ APPROXIMATION OF 3D IMAGE DATA



original 3D image data set

Goal: Given a high-resolution 3D image data set $\{f_{i,j,k}\}$, $i,j,k = 1...N$, where the $f_{i,j,k}$-values are associated with an underlying Cartesian grid, approximate this data set via BEST approx. methods and multi-res. techniques!

• Motivating Reasons

1) DATA COMPRESSION. Less data ⇒ less storage & time.

2) CHOICE OF RESOLUTION. Can discretize an analytical approximation at arbitrary resolution.

3) HIGHER-ORDER DERIVATIVES. Can compute higher-order differential properties exactly via analytical approximation (e.g., gradient, Hessian etc.).

I) NOISE REDUCTION. Can use a multi-res. analytical approximation to remove small details and noise from data set.

II) SCALE-SPECIFIC ANALYSIS. Can use wavelet coefficients to compute scale-specific features for classification.

■ APPROXIMATION OF 3D IMAGE DATA – Cont'd.

... Reasons

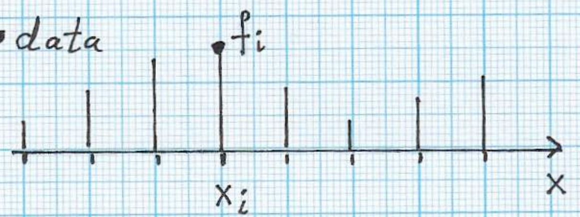
III) PROGRESSIVE ANALYSIS. Can use wavelet approximation to consider progressively more (or less) detail coefficients for more efficient processing.

A) SUPPORT OF CNN ANALYSIS. Wavelet detail coefficients result from a progressively more-and-more detail-oriented analysis done via convolution with wavelet filter masks; can use wavelet coefficients like the results produced by the convolutional network layers of a CNN.

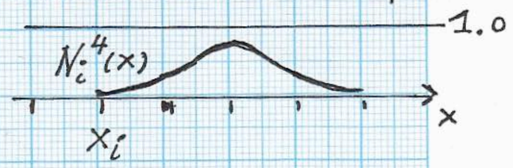
• Illustrations for 1D Case

INTERPOL.

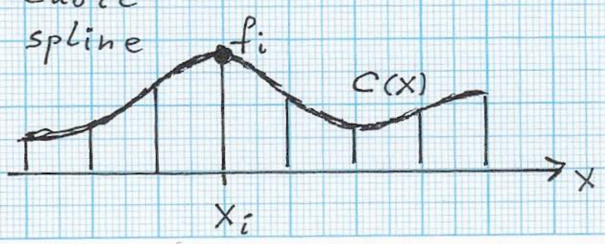
• data



• normalized cubic B-spl. function



• cubic spline



$\Rightarrow$ interpolating cubic spline: $c(x) = \sum_i d_i N_i^4(x)$

defined by conditions

$c(x_j) = \sum_i d_i N_i^4(x_j) = f_j$,

$j = 1 \ldots N$

Bernd Hamann 7/16/2020

Stratovan

■ APPROXIMATION OF 3D IMAGE DATA – Cont'd.

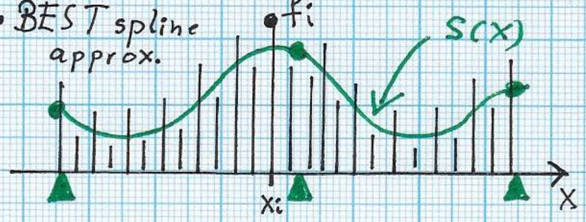
... Illustrations

⇒ interpolating TRI-cubic spline using tensor products of normalized B-spline basis functions:

$$C(x,y,z) = \sum_k \sum_j \sum_i d_{i,j,k} \, N_i^4(x) \, N_j^4(y) \, N_k^4(z)$$

conditions: $C(x_{I,J,K}) = \sum_k \sum_j \sum_i \ldots = f_{I,J,K}$

• BEST spline approx.



$\{\blacktriangle\}$ = knots of BEST cubic spl. approx.

⇒ determine no. of spline segments and location of knots

⇒ normalized cubic B-spline basis functions defined for knots $\{\blacktriangle\}$

⇒ original "function" to be approximated: piecewise constant function defined by $\{f_i\}$

⇒ optimal cubic spline approximation obtained by solving NORMAL EQUATIONS defined by inner products of N_i^4 basis functions ("left-hand side") and inner products of original function and N_i^k basis functions ("right-hand side").

⇒ $S(x) = \sum_j d_j \, N_j^4(x)$ MINIMIZES $\int (f(x) - S(x))^2 \, dx$.

original function

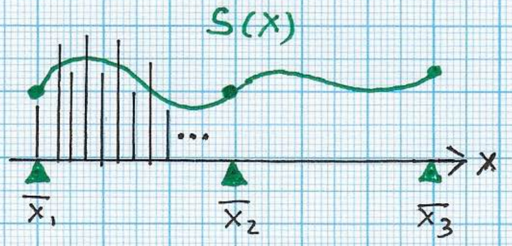
Bernd Hamann

7/16/2020

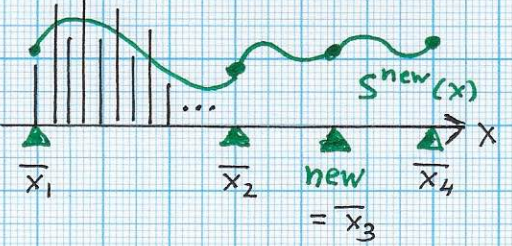
Stratovan

APPROXIMATION OF 3D IMAGE DATA – Cont'd.

... Illustrations



• iterative refinement



• Multi-res. via refinement

⟹ compute knot interval-specific local error values

$$e_j = \int_{\bar{x}_j}^{\bar{x}_{j+1}} (f(x) - s(x))^2 \, dx$$

and insert new knot(s) in interval(s) of largest local error value(s); refine until...

• Multi-res. cubic (TRI-cubic) B-spline wavelet analysis and approximation:

→ Use scaling and wavelet functions for cubic (tri-cubic) B-spline wavelet computation provided in LIBRARY OF JESUS PULIDO.

~ BH