Exact Computation of Area Moments for Spline and Wavelet Curves

M. Jacob, T. Blu and M. Unser

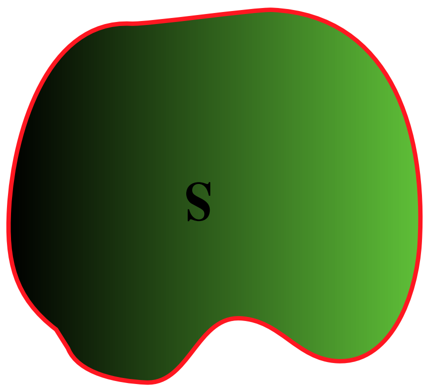
Biomedical Imaging Group*
Swiss Federal Institute of Technology,
Lausanne

6^{th} September 2000

*http://bigwww.epfl.ch

What are Area Moments ??

$$I_{m,n} = \int_{\mathbf{S}} x^m y^n \, dx dy$$



Why Moments ??

- Moment Invariants

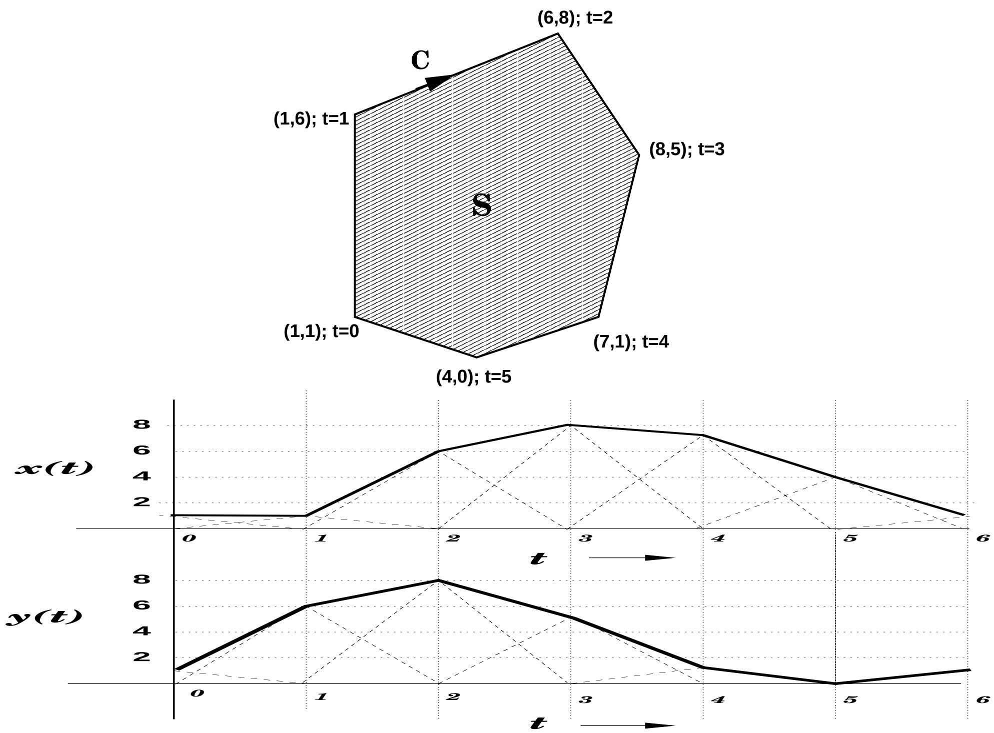
  Widely used in Pattern classification.
- Extraction of Elliptic shape parameters

  Computer Vision and Medical Imaging

How are Moments extracted ??

Image $\rightarrow$ [Segmentation] $\rightarrow$ [Moment Computation] $\rightarrow$ Features

Parametric Representation of Curves



$$x(t) = \sum_{k=0}^{M-1} c_k \, \varphi_p(t-k)$$

$$y(t) = \sum_{k=0}^{M-1} d_k \, \varphi_p(t-k),$$

where

$$\varphi_p(t) = \sum_{k=-\infty}^{\infty} \varphi(t-kM)$$

Alternate Wavelet Representation

(by Chuang et. al., Wang et. al.)

φ builds a Multiresolution analysis (Mallat)

$\Rightarrow$

$$x(t) = \sum_i \sum_k c'_{i,k}\, \psi_{i,k}$$
$$y(t) = \sum_i \sum_k d'_{i,k}\, \psi_{i,k}$$

$c'_{i,k}, d'_{i,k}$ $\rightarrow$ **Mallat's algorithm** $\rightarrow$ c_k, d_k

- Sparse Representation
- Useful in Constraining Shapes

$\Rightarrow$ Wavelet curves also fall in the framework.

Problem Addressed

To compute the moments of a region bounded by a parametric curve represented in some basis functions

Conventional Approaches

1) Pixel Based Schemes

- Loss of subpixel accuracy.
- Error dependent on the orientation.
- Indirect Approach.

2) Approximation as a polygon

- Linear splines $\rightarrow$ low approximation order.
- Impossible to compute the curvature .
- Indirect Approach.

Indirect Approach

Moments $\rightarrow$ double integral

$$I_{m,n} = \iint_{\mathbf{S}} x^m y^n dx dy$$

Direct Approach

Surface Integral $\rightarrow$ Green's theorem $\rightarrow$ Line Integral

Green's Theorem

Volume integral of Divergence of a vector field $\longrightarrow$ Surface integral of the field

$$\int_{\mathbf{V}} (\nabla.\mathbf{F}) d\mathbf{V} \; = \; \int_{\mathbf{S}} \mathbf{F}.d\mathbf{S},$$

2-D Implication

$$\iint_{\mathbf{S}} \left(\frac{\partial \mathbf{F}_x}{\partial x} + \frac{\partial \mathbf{F}_y}{\partial y}\right) dx dy \; = \; \oint_{\mathbf{C}} (\mathbf{F}_y dx - \mathbf{F}_x dy)$$

Set $\mathbf{F} = (\frac{x^m y^{n+1}}{n+1})\mathbf{e}_y$ and $\mathbf{C} = (x(t), y(t))$

$$\boxed{I_{m,n} = \int_0^M \frac{x^m(t) y^{n+1}(t) x'(t)}{n+1} dt}$$

Computation of Area

$$\mathbf{I}_{0,0} = \int_0^M y(t) x'(t) dt$$

If $x(t), y(t) \in \text{span}\{\varphi(t-k);\ k \in \mathbb{Z}\}$

$$\mathbf{I}_{0,0} = \sum_{k,l} d_k c_l \underbrace{\int_0^M \varphi_p(t-k)\varphi_p'(t-l)dt}_{g_0^p(k-l)}$$

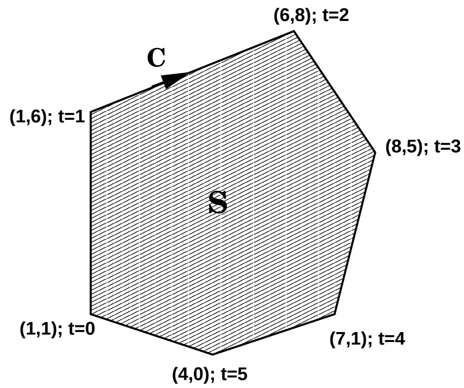
$g_0^p(m) = \sum_k g_0(m - kM)$, where

$$g_0(m) = \int_{-\infty}^{\infty} \varphi'(t)\varphi(t-m)dt$$

Hence,

$$\boxed{\mathbf{I}_{0,0} = \underbrace{\sum_{k=0}^{M} d_k^p \sum_{l=-N}^{N} c_{k+l}^p \, g_0(l)}_{\langle d, g_0^T * c^p \rangle}}$$

Example



$$c_k = 1, 6, 8, 7, 4, 1$$
$$d_k = 6, 8, 5, 1, 0, 1$$

$\varphi =$ Linear B-spline

$$g_0 \quad : \quad (0.5, 0, -0.5); \quad l \in -1, 0, 1$$

Hence,

$$\mathbf{I}_{0,0} = \frac{\langle (6, 8, 5, 1, 0, 1), (5, 7, 1, -4, -6, -3) \rangle}{2}$$
$$= 42 \text{ units}$$

Computation of Higher Order Moments

$$I_{m,n} = \frac{\langle c^p, {g_{m+n}}^T * (c^{p[m]} \otimes d^{p[n+1]}) \rangle}{n+1},$$

where

$$g_{m+n}(\mathsf{k}) = \int_{-\infty}^{\infty} \varphi'(t)\varphi(t-k_1)\ldots\varphi(t-k_{m+n+1})dt$$

and $c^{[m]} = \underbrace{c \otimes c \otimes \ldots c}_{\text{m times}}$

Computational Complexity : $M(2N-2)^{(m+n+2)}$

$$\varphi'(t) = \varphi_1(t) - \varphi_1(t-1), \text{ where}$$

$$. \quad \varphi_1 \leftrightarrow 2H(z)/(1+z^{-1})$$

$$g_m(\mathsf{k}) = f_m(\mathsf{k}) - f_m(\mathsf{k}-1),$$

where

$f_m(\mathsf{k}) = \int \varphi_1(t)\varphi(t-k_1)\ldots\varphi(t-k_m)dt$

How can you compute f_m??

$\Rightarrow$ Make use of its properties

Properties of the Kernel

- Finite support:

$$I = [-N, N-1] \times \ldots [-N, N-1] \times [-N, N-1]$$

- Symmetry:

$$f(\mathsf{k}) = f(\sigma_{\mathsf{i}}(\mathsf{k}))$$

- Two-scale relation

$$\boxed{f_m(\mathsf{k}) = \sum_{\mathsf{l} \in \mathbb{Z}^{\mathsf{m}}} \mathsf{H}_{\mathsf{m}}(\mathsf{l}) \mathsf{f}_{\mathsf{m}}(2\mathsf{k} - \mathsf{l}),}$$ where

$$\mathsf{H}_{\mathsf{m}}(z_1, z_2, .., z_m) = \frac{1}{2} H_1 \left(\prod_{k=1}^{m} z_k \right) \prod_{k=1}^{m} H(z_k^{-1})$$

Computation of f_0

$$f_0(k) \;=\; \sum_{\mathbf{l} \in \mathbb{Z}} H_0(l) f_0(2k - l)$$

$$\Rightarrow \quad (\mathbf{A}_0 - \mathbf{I})\mathbf{f}_0 \;=\; \mathbf{0}$$

$$[\mathbf{A}_0]_{k,l} = H_0(2k-l); \; \mathbf{f}_0 = [f(-N+1), \ldots f(N-2)]^T$$

Partition of Unity $\Rightarrow [1, 1 \ldots 1]\mathbf{f}_0 = 1$

$$\mathbf{B}_0 \mathbf{f}_0 = [0, 0, \ldots 1]$$

Computation of Higher Order Kernels

$$\mathbf{A_m} \mathsf{f}_m = \mathsf{f}_m,$$

where $\text{dim}(\mathbf{A_m}) = (2N-2)^m \times (2N-2)^m$

With the One - One mapping

$$\rho : [-N+1, N-2]^m \mapsto [0, (2N-2)^m - 1]$$

and

$$\mathbf{A_m} = \mathbf{H}_m(2\rho^{-1}(i) - \rho^{-1}(j)); \quad \mathsf{f}_m(i) = f\left(\rho^{-1}(i)\right)$$

Normalization Constraint $\longrightarrow \sum_i \mathsf{f}_m(i) = 1$

$$\Rightarrow \quad \mathbf{B}_m \mathsf{f}_m = [0, 0, \ldots 0, 1]^T$$

- Exact and Unique Solution.

$f_m(\mathsf{k}) \longrightarrow$ Samples of Box Spline function.

Refine with two-scale relation $\rightarrow$ Non-integer Samples.

To Summarise..

- Moments are computed as

$$\boxed{I_{m,n} = \frac{\langle c^p, g_{m+n}^T * (c^{p[m]} \otimes d^{p[n+1]}) \rangle}{n+1},}$$

where

$$g_m(\mathsf{k}) \;=\; f_m(\mathsf{k}) - f_m(\mathsf{k}-1),$$

- The Kernel f_m is precomputed by solving

$$\mathbf{B}_m \mathsf{f}_m = [0, 0, \ldots 0, 1]^T,$$

where $\dim(\mathbf{B_m}) = (2N-2)^m \times (2N-2)^m$

Example of Kernels

Linear Spline

$$f_0(k_1); k_1 \in \{-1, 0\} \quad : \quad \frac{1}{2}[\ 1 \quad 1\ ]$$

$$f_1(k_1, k_2); k_1, k_2 \in \{-1, 0\} \quad : \quad \frac{1}{6}\begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}$$

$$\begin{cases} f_2(-1, k_2, k_3); k_2, k_3 \in \{-1, 0\} & : \quad \frac{1}{12}\begin{bmatrix} 1 & 1 \\ 3 & 1 \end{bmatrix} \\ \\ f_2(0, k_2, k_3); k_2, k_3 \in \{-1, 0\} & : \quad \frac{1}{12}\begin{bmatrix} 1 & 3 \\ 1 & 1 \end{bmatrix} \end{cases}$$

Cubic Spline

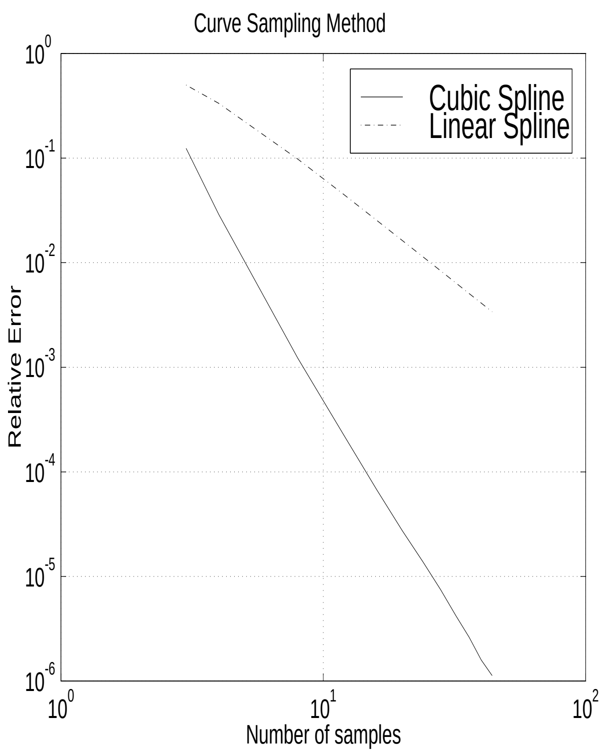
$$f_0(k_1) : \frac{1}{720}.[\ 1,\ \ 57,\ \ 302,\ \ 302,\ \ 57,\ \ 1\ ]$$

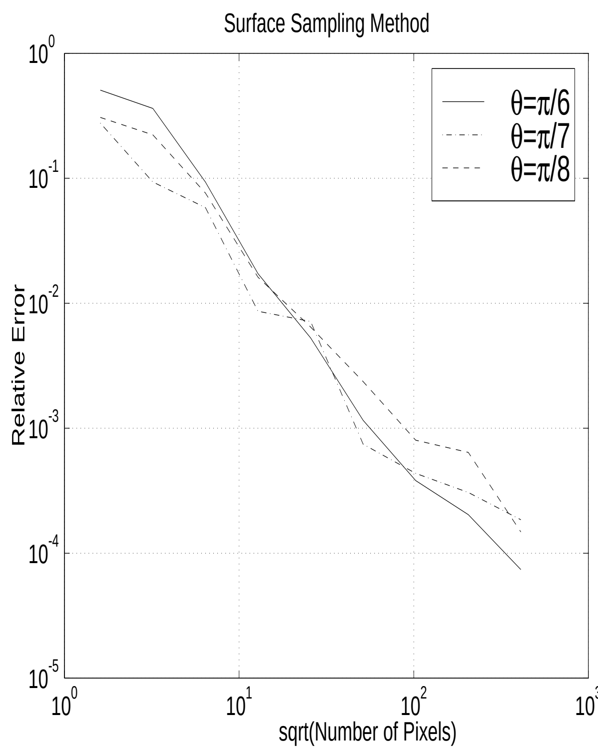
$$k_1 = -3, \ldots 2$$

Comparison of Moment Estimators

- Ellipse sampled → linear and cubic spline interpolation.

- Ellipse Rasterized.





Conclusions

The main advantages of the new method are:

- the exactness of the computation;
- the independence of the orientation
- Consistency with the snake model
- Direct Approach.
- Simple Implementation - Filtering with a kernel * followed by an inner product.

*Higher order spline kernels are available at http://bigwww.epfl.ch/jacob