Exact Computation of Area Moments for Spline and Wavelet Curves

Mathews Jacob, Thierry Blu and Michael Unser
Biomedical Imaging Group, DMT/IOA
Swiss Federal Institute of Technology, Lausanne
CH-1015 Lausanne EPFL, SWITZERLAND
Mathews.Jacob, Thierry.Blu, Michael.Unser@epfl.ch

Abstract

We present an exact algorithm for the computation of the moments of a region bounded by a curve represented in a scaling function or wavelet basis. Using Green's theorem, we show that the computation of the area moments is equivalent to applying a suitable multidimensional filter on the coefficients of the curve and thereafter computing a scalar product. We compare this algorithm with existing methods such as pixel-based approaches and approximation of the region by a polygon.

1 Introduction

Moments constitute very useful descriptors of the shape of an object [1]; they easily yield features that are invariant to translation and rotation. In the standard formulation, they are computed as surface integrals which requires raster scanning through the image. There are many instances, however, where the boundaries of objects are described by parametric curves. This is the case, for example, when the objects are detected using parametric snakes which are represented using B-spline [2] or wavelet basis functions [3]. Another simple case is when the region is described as a polygon [4].

In this paper, we address the problem of computing the area moments of objects described by such parametric curves when the basis functions are scaling functions, including wavelets. The originality of our approach is that the computation is exact, and also more direct than the more conventional pixel-based methods which requires an explicit labelling of the inner region of the curve prior computation. Also with pixel-based schemes, the subpixel information is lost in the rastering process, which results in lesser accuracy. Moreover the error in the area-based computation of moments is dependent on the orientation of the shape.

Since a polygon can be represented in terms of linear splines, the computation of moments by approximating the shape as a polygon [4, 5, 6] is a particular case of our approach. While the polygon method can be made as accurate as desirable by increasing the number of segments, the convergence is slow because of the low approximation order of linear splines. Moreover, it is not suitable for computing curvature, which is an interesting shape feature as it is invariant to rotation, translation and can be easily normalized to scale changes. This motivates us to investigate higher order schemes where the curves is represented by smoother basis functions such as splines and wavelets.

2 Evaluation of the moments

2.1 Parametric representation of a curve

A curve $\mathbf{C}$ in the $x-y$ plane can be represented in terms of an arbitrary parameter t' as $r(t') = (x(t'), y(t'))$. If the curve is a closed one, as discussed in this paper, the functions $x(t')$ and $y(t')$ are periodic functions of the parameter.

The curve $\mathbf{C}$ can be represented in terms of the scaling function basis at a scale j in terms of a new parameter $t = 2^j.t'$ as

$$x(t) = \sum_{k=0}^{M} c_k.\phi_p(t-k) \quad (1)$$

$$y(t) = \sum_{k=0}^{M} d_k.\phi_p(t-k), \quad (2)$$

where

$$\phi_p(t) = \sum_{k=-\infty}^{\infty} \phi(t-k.M) \quad (3)$$

assuming that the period ,M, is an integer.

If the curve is represented in a wavelet basis as described in [3], the scaling function coefficients at any scale can be obtained from the wavelet coefficients using Mallat's fast wavelet reconstruction equation.

2.2 Computation of moments using Green's Theorem

Green's theorem relates the volume integral of the divergence of a vector field in a closed region to the surface integral of the field :

$$\int_{\mathbf{V}} (\nabla.\mathbf{F}) d\mathbf{V} \quad = \quad \int_{\mathbf{S}} \mathbf{F}.d\mathbf{S}, \tag{4}$$

where S denotes the surface that encloses the closed region V and dS is the unit vector pointing out of the volume. Assuming that the volume has a constant cross section bounded by the curve $\mathbf{C}$, and the variation of the field along the z direction is zero, we have,

$$\int_{\mathbf{S}} (\frac{\partial \mathbf{F}_x}{\partial x} + \frac{\partial \mathbf{F}_y}{\partial y}) dx dy \quad = \quad \oint_{\mathbf{C}} (\mathbf{F}_y dx - \mathbf{F}_x dy) \tag{5}$$

The first integral is evaluated over the area $\mathbf{S}$ enclosed by the curve and the second one along the curve $\mathbf{C}$. The computation of the moments involves the evaluation of the integral $\int_{\mathbf{S}} x^m.y^n.dxdy$ and this, by (5), is equivalent to

$$\mathbf{I}_{m,n} \quad = \quad \int_0^M \frac{x^m(t).y^{n+1}(t)}{n+1} (\frac{dx(t)}{dt}) dt \tag{6}$$

when the closed curve $\mathbf{C}$ is parametrized by t and $\mathbf{F}$ in (5) is chosen as $\vec{j}.(\frac{x^m.y^{n+1}}{n+1})$ where $\vec{j}$ is the unit vector along the y direction.

Now if we assume that the curve is represented in terms of a scaling function basis as given by the equations (1) and (2), the above integral becomes,

$$\mathbf{I}_{m,n} = \frac{\sum_k \sum_{\mathbf{i},\mathbf{j}} c_k.\mathbf{c}_{\mathbf{i}}^{[m]}.\mathbf{d}_{\mathbf{j}}^{[n+1]}.\mathbf{g}_{m+n}^{p}(\mathbf{i}-k, \mathbf{j}-k)}{n+1}, \tag{7}$$

where $\mathbf{i} \in \mathbb{Z}^{m+1}$, $\mathbf{j} \in \mathbb{Z}^n$ and $\mathbf{c}^{[m]}$ stands for the m-times tensor product $c \otimes c \ldots \otimes c$. In the above equation, the kernel $\mathbf{g}_{m+n}^{p}$ is obtained from the M-periodization of

$$\mathbf{g}_{m+n}(\mathbf{a}) = \int_{-\infty}^{\infty} \phi'(t)\phi(t-a_1)..\phi(t-a_{m+n+1}).dt, \tag{8}$$

where $\mathbf{a} \in \mathbb{Z}^{m+n+1}$ and $\phi(x)$ is the non periodized scaling function.

The function $\phi'(t)$ corresponds to the differential of the scaling function and hence could be substituted by $\phi_1(t) - \phi_1(t-1)$, where $\phi_1(t)$ is the scaling function whose mask is given by $2 \times H(z)/(1+z^{-1})$; $H(z)$ is the mask of the function $\phi(t)$. Then the above equation becomes

$$\mathbf{g}_{m+n}(\mathbf{a}) = \mathbf{f}_{m+n}(\mathbf{a}) - \mathbf{f}_{m+n}(\mathbf{a}+1), \tag{9}$$

where

$$\mathbf{f}_{m+n}(\mathbf{a}) = \int_{-\infty}^{\infty} \phi_1(t)\phi(t-a_1)..\phi(t-a_{m+n+1}).dt \tag{10}$$

2.3 Properties of the kernel f

The fact that the functions $\phi(t)$ are the solutions to a finite two-scale relation imparts some nice properties to the kernel $\mathbf{f}$ obtained from it. These properties enable us to compute it exactly.

1. **Finite support:** If the scaling function is having a support $[0, N]$, then the kernel will be zero outside the region given by

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

2. **Symmetry:** As (10) is symmetric with respect to the parameters $a_1, a_2, ..$, interchanging them will not affect the value of the kernel. This implies

   $$f(\mathbf{a}) \quad = \quad f(\sigma_i(\mathbf{a})) \tag{11}$$

   where σ_i indicates all possible permutation operators which implies $(m+n+1)!$ relations. In addition to these relations, if the scaling functions are symmetric as in the case of splines, we have

   $$f(\mathbf{a}) \quad = \quad f(-\mathbf{a}), \tag{12}$$

   Both these properties together imply $2\,(m+n+1)!$ relations which could be used to accelerate the computation of the kernel as well as the moments.

3. **Two-scale relation:** It can be shown that the kernel satisfies a two-scale relation

   $$\mathbf{f}_m(\mathbf{a}) \quad = \quad \sum_{\mathbf{b}} H(\mathbf{b}).\mathbf{f}_m(2\mathbf{a}-\mathbf{b}), \tag{13}$$

   where $\mathbf{a}, \mathbf{b} \in \mathbb{Z}^{m+1}$. The mask H in the above equation is

   $$H(b_1, b_2, .., b_m) = \sum_k h_1(k).h(k-b_1)..h(k-b_m), \tag{14}$$

   This follows from the fact that the scaling functions $\phi(t)$ from which the kernel is derived satisfies the two-scale relation

   $$\phi(t) = \sum_k h(k).\phi(2t-k) \tag{15}$$

   It is this property that enables us to compute it exactly by solving a linear system of equations. The technique is analogous to the one used for computing the integer (or dyadic rational) samples of a scaling function from the transition operator [7].

3 Practical Implementation and Results

3.1 On the computation of the Moments.

In this section we analyze equation (7) and simplify it for the faster computation. Due to space limitations, we concentrate on the simplest case: the area of the region.

From the general equation (7) for the computation of the moments, we get the area bounded by the curve as

$$\mathbf{I}_{0,0} \quad = \quad \sum_{k=0}^{M} c_k^p . \sum_{l=-N}^{N} . d_{k+l}^p . g_0(l), \qquad (16)$$

where g is given by (8), and the sequences c_k^p and d_k^p are M-periodized versions of the coefficients c_k and d_k. This is because convolving a non-periodized sequence with a periodized kernel is equivalent to convolving a periodized sequence with a non-periodized kernel. The latter is easier to compute in practice. Moreover this serves to reduce the range of summation of the inner sum from $-N$ to N, which is typically much less than the range of sum from 0 to M. In a similar way, it can be seen that for the computation of the higher order moments, all the summations except the outer one is in the range $-N$ to N.

From (16), we see that the computation of the area involves just a filtering operation by $g(-l) = g^T(l)$, followed by an inner product . This can be written as,

$$\mathbf{I}_{0,0} \quad = \quad < c, g_0^T * d^p >,$$

where $< ., . >$ denotes the inner product. With a similar notation, the computation of the other moments are given as

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

$$= \quad - \frac{< d, g_{m+n}^T * (c^{p[m+1]} \otimes d^{p[n]}) >}{m+1}$$

As the n-dimensional sequence is separable, the filtering operation is much simpler than the usual n-dimensional filtering.

The computational complexity in the computation of the moment $\mathbf{I_{m,n}}$ is $M.(2N-2)^{m+n+2}$, neglecting the symmetries. Thus for small N and reasonable m and n, the complexity is quite managable.

3.2 Experiment and results.

In this section we compare the new technique with the existing ones: approximation using polygons and rastering. We try to estimate the elliptical parameters of an ellipse with known parameters and choose the relative error in the parameters as the criterion of comparison.

With the new technique we choose the cubic B-spline due to its nice approximation properties and minimum curvature properties [8]. To compare it with the approximation of the region as a polygon, the ellipse is sampled uniformly and the samples are interpolated using the two techniques (linear and cubic splines). The relative error in the centered 2nd order moments vs the number or samples are plotted in figure 1 . It can be seen from figure 1 that the relative error is much smaller for the cubic spline interpolation even at low sampling rates and that it decays at a faster rate.

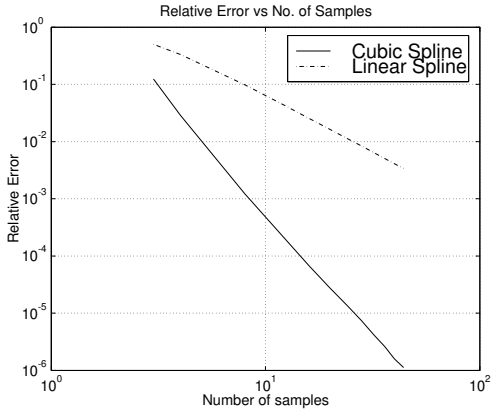

Figure 1. Comparison of moment estimators

In the traditional scanning approach, the ellipse is scanned along the x and y axes with a step size Δ and the monomials are computed at the grid points assigned to the interior of the curve. Fig 2 shows the decay of the computed error for an ellipse vs $\frac{\sqrt{\text{Area}}}{\Delta}$ for three different orientations. The plot clearly shows the dependence of the accuracy on the orientation of the ellipse.

It can be seen that to achieve a relative error of 0.1% the interior of the ellipse has to be sampled at about 3600 points, where as to achieve the same error using the cubic spline interpolation we need only around 9 points on the curve. In comparison the polygon method (linear spline) requires more than 40 samples to have a similar error. More interesting is the case when the interior of the ellipse has to be sampled at about 2.5×10^5 to achieve an error of 0.002% while the cubic splines require only 25 samples to achieve the same. In Fig 3, we show the ellipse corresponding to the 2nd order moments of the central structure in the image. The contour of the object was estimated using a snake where the curve was represented parametrically in terms of cubic B-spline basis; the moments were computed using our algorithm. Note that the fit is astonishingly good.

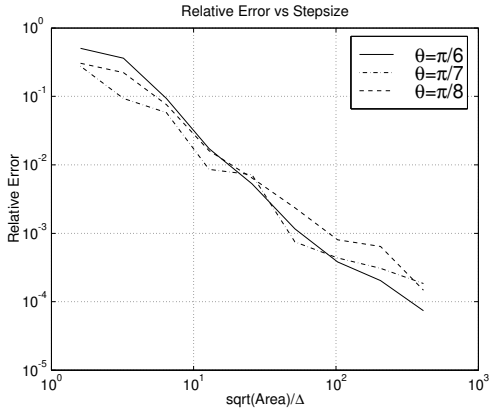

Figure 2. Variation of error vs $\frac{1}{\Delta}$ in a raster scan moment estimator

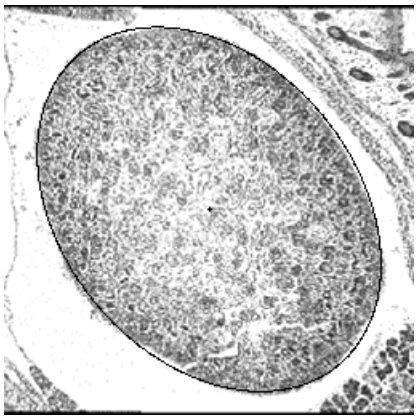

Figure 3. Estimated ellipse for a real image

4 Conclusion

In this paper we have presented a new approach for the computation of the moments of a curve described in a wavelet or scaling function basis. We have demonstrated that this scheme is more accurate than the conventional approaches in the estimation of the moments. Unlike the pixel-based schemes, it is independent of the orientation of the shape.

References

[1] S.Rad, K.C.Smith, and B.Benhabib, "Application of moment and fourier descriptors to the accurate estimation of elliptical shape parameters," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Newyork,USA, 1991, vol. 4, pp. 2465–8.

[2] P. Brigger, J. Hoeg, and M. Unser, "B-spline snakes: a flexible tool for parametric contour detection," *IEEE Trans. Image Process.*, 1999, in press.

[3] G.C.H.Chuang and J. Kuo, "Wavelet descriptor of planar curves: Theory and applications," *IEEE Trans. Image Process.*, vol. 5, pp. 56–70, 1996.

[4] M.Singer, "A general approach to moment calculation for polygons and line segments," *Pattern Recognition*, vol. 26, no. 7, pp. 1019–1028, Jan 1993.

[5] S.F.Bockman, "Generalising the formula for areas of polygons to moments," *American Mathematical Monthly*, vol. 96, no. 2, pp. 131–133, February 1989.

[6] A.R.Allan N.J.C.Strachan, P.Nesvadba, "A method for working out the moments of a polygon using an integration technique," *Pattern Recognition Letters*, vol. 11, pp. 351–354, May 1990.

[7] Gilbert Strang and Truong Q. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, 1996.

[8] M.Unser, "Splines: a perfect fit for signal and image processing," *IEEE Signal Processing Magazine*, vol. 16, pp. 22–38, 1999.