Computing Geometric Moments for Objects with an Exact Polygonal Representation

Mats Carlin
University of Oslo, c/o SINTEF,
Box 124 Blindern, N-0314 Oslo, Norway
Tel : +47 2206 7300, Fax : +47 2206 7350
Email : Mats.Carlin@ecy.sintef.no
http://www.ifi.uio.no/~matsc

Preprint from Vision Interface 1998, Vancouver, Canada, 18-21 June 1998.

Abstract

Computing geometric moments of a two-dimensional object is of interest since the moments have a physical and/or statistical interpretation. For objects with an exact polygon representation the geometric moments may be computed directly. This article reviews the problem and earlier efforts. We then present two new methods for computing the geometric moments for objects with a polygonal contour representation. Both methods make use of Green's theorem. The first method applies integration by parts and is sensitive to large slope values. The second method applies binomial expansion and is insensitive to large slope values, and is hence preferable.

keywords : geometric moments, computation, polygon, binomial expansion, numerical stability

1 Introduction

The method of moment invariants has been used for over 35 years for object recognition in images [3]. For cases where an exact representation is given, such as for CAD-drawings, the geometric moments are of interest since they represent certain physical and/or statistical attributes of the object. These attributes may be used as input to similarity measures for different objects or as input to physical models of a production process.

- The zeroth order moment is simply the mass or area of the object
- The first order moments may be used to locate the center of mass of an object
- The second order moments are the moments of inertia and may be used to determine the principal axes, image ellipse and radii of gyration.
- The third order moments may be used to determine the projection skewness
- The fourth order moments may be used to determine the projection kurtosis

The geometric moments may be used to compute moment invariants. These are invariant with respect to a similarity transformation of the object, e.g. invariant to translation, rotation, reflection and scale. We refer to any standard text on this subject for the conversion from simple geometric moments to moment invariants [5] or orthogonal moments such as the Legendre or Zernike moments [11]. For an excellent review on the theory of different moments, see [8]. For extensions of moments to 3 and higher dimensions, see [2, 6].

In this paper we present two previously unpublished methods for computing the geometric moments for objects with an exact polygonal representation. Both methods make use of Green's theorem. The first method applies integration by parts and is sensitive to slope values of the individual line segments. The second method applies binomial expansion and is insensitive to large slope values. The second method also forms a closed-form expression for moments of all orders. The importance of the slope insensitivity is further illustrated by an example. Slope insensitivity is of great importance when computing second order or higher order moments. In this article we will review articles related to computation of moments directly from objects with a polygonal contour representation. Two-dimensional polygonal contours may be obtained from images by representing the segmented objects by the contour pixel coordinates. In many cases, such as for two-dimensional CAD drawings, parametric contour representations are the starting point and polygonal contour approximations with a given resolution may easily be computed. This is a more convenient representation for analysis because of its simplicity.

2 Computing moments for polygons

The regular expression for geometric moments is

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x,y) dx dy \qquad (1)$$

In our case we limit ourselves to binary objects (silhouettes) using a function $f(x,y) = 1$ within the object and $f(x,y) = 0$ elsewhere.

2.1 Green's theorem

Suppose that we have an exact parametric contour representation of the object. It is then a known fact that we may use Green's theorem to reduce the double integral to a simpler curve integral :

$$\begin{aligned} m_{pq} &= \frac{1}{p+1} \oint_C x^{p+1} y^q dy \\ &= -\frac{1}{q+1} \oint_C x^p y^{q+1} dx \qquad (2) \end{aligned}$$

where $\oint_C$ is the curve integral along the contour. Tang was the first to show that a discrete version of Green's theorem applies to binary objects [10]. In our context the discretisation of the theorem is not of importance as we use the original continuous version of Green's theorem. However Tangs paper inspired other researchers to use the obvious simplification which Green's theorem represents to compute the moments from contour representations.

If we do have objects with holes, then we may compute the moments for the outer contour first and then subtract the moments of the inner contours. The only prerequisite is that all contours are represented in the same counter-clockwise direction.

2.2 A piecewise linear parametric representation and the additivity of moments

Polygonal contours are piecewise linear. The moments are additive regardless of the order and hence can be computed partially for each line segment

$$m_{pq} = \sum_{i=1}^{n} \psi_{pq}^{(i)} \qquad (3)$$

where $\psi_{pq}^{(i)}$ denotes the partial moments for line segment i. There are at least two different approaches to the computation of the partial moments. The first approach is to compute the moments for the triangles formed by the individual line segments and the origin. Leu uses this approach and by adding the triangular moments for each line segment [4]. Leu shows that it was possible to compute the total moments of the polygon. It is however not trivial to decide if a triangular moment should be positive or negative in Leu's expressions. The decision of the sign can be done at the cost of some extra computations. No closed-form expression for computing moments of all orders is provided in Leu's paper. The second approach is to compute the moments for the trapezoidal area formed by the line segment and one of the axes. In our presentation we will rely on the last manner, while comparing the results with earlier results based on triangular moments. Strachan et.al split this trapezoidal area in a triangle and a rectangle [7]. They use a rather cumbersome method for computation of the double integral of each moment for each line segment depending on 7 different classes of line segments. They seem not to have been aware of the possibility of using Green's theorem.

Each line segment has end points (x_i, y_i) and (x_{i+1}, y_{i+1}) and we may express any point on each line segment as a function of one variable $t \in [0,1]$. For notational simplicity we assume $(x_{n+1}, y_{n+1}) = (x_1, y_1)$. Any point on the i'th line segment may be expressed as $x = x_i + t(x_{i-1} - x_i)$ and $y = y_i + t(y_{i-1} - y_i)$ leading to $dy = (y_{i-1} - y_i)dt$. Then the general expression for the partial moments follows

$$\begin{aligned} \psi_{pq}^{(i)} &= \frac{(y_{i+1} - y_i)}{p+1} \int_0^1 (x_i + t(x_{i+1} - x_i))^{p+1} \cdot \\ & \quad (y_i + t(y_{i+1} - y_i))^q dt \qquad (4) \end{aligned}$$

2.3 Computing the moments based on integration by parts

One common technique for computing integrals of this type is by using integration by parts.

$$\int v \frac{du}{dt} = uv - \int u \frac{dv}{dt} \tag{5}$$

By iteratively applying integration by parts we may gradually reduce one of the powers of the expression at the cost of gradually increasing the other power. One possible solution for the partial moments is

$$\begin{aligned} \psi_{pq}^{(i)} &= \sum_{k=0}^{q} \{(-1)^k \frac{p!}{(p+k+2)!} \frac{q!}{(q-k)!} \cdot \\ & (\frac{y_{i+1}-y_i}{x_{i+1}-x_i})^{k+1} \cdot \\ & (x_{i+1}^{p+k+2} y_{i+1}^{q-k} - x_i^{p+k+2} y_i^{q-k})\} \end{aligned} \tag{6}$$

Since we have a singularity for $(x_{i+1} - x_i) = 0$ another expression has to be found [1]

$$\begin{aligned} \psi_{pq}^{(i)} &= \frac{y_{i+1}-y_i}{p+1} \int_0^1 x_i^{p+1} (y_i + t(y_{i+1} - y_i))^q dt \\ &= \frac{1}{q+1} \frac{1}{p+1} x_i^{p+1} (y_{i+1}^{q+1} - y_i^{q+1}) \end{aligned} \tag{7}$$

One of the major problems caused by using the expression 5 is that when the slope is close to vertical the expression is numerical unstable.

Of all earlier published methods, the method closest to our presentation is one presented by Jiang & Bunke [1]. Jiang & Bunke use a different reduction scheme than integration by parts.

$$\begin{aligned} \psi_{pq}^{(i)} &= \frac{1}{(p+1)(x_{i+1}-x_i)^{q+1}} \sum_{k=0}^{q} \{ \binom{q}{k} \cdot \\ & \frac{(y_{i+1}-y_i)^{k+1} (y_i x_{i+1} - x y_{i+1})^{q-k}}{p+k+2} \cdot \\ & (x_{i+1}^{p+k+2} - x_i^{p+k+2})\} \end{aligned} \tag{8}$$

If $(x_{i+1} - x_i) = 0$ then equation 7 should be used. Fast recursive expressions are also known [1]. Both these expressions are however sensitive to the slope values. This is a severe drawback.

2.4 Binomial expansion of the powers

To avoid the problem of near vertical slopes we have tried another route to solving the problem based on binomial expansion of the powers. The general expression for expansion of the powers in our case is

$$(at + b)^p = \sum_{j=0}^{p} \binom{p}{j} a^j t^j b^{p-j} \tag{9}$$

We now apply the binomial expansion of each of the two powers in our partial moment expression $(x_i + t(x_{i+1} - x_i))^{p+1} = \sum_{j=0}^{p+1} \binom{p+1}{j} (x_{i+1} - x_i)^j t^j x_i^{p-j}$ and $(y_i + t(y_{i+1} - y_i))^q = \sum_{k=0}^{q} \binom{q}{k} (y_{i+1} - y_i)^k t^k y_i^{q-k}$. The integral only applies to t^{k+j} which is the kernel of the double summation and the resulting integral $\int_0^1 t^{k+j} dt$ can easily be solved. A closed-form expression for moment computation based on binomial expansion becomes

$$\begin{aligned} \psi_{pq}^{(i)} &= \frac{y_{i+1}-y_i}{p+1} \sum_{j=0}^{p+1} \sum_{k=0}^{q} \{ \frac{\binom{p+1}{j}\binom{q}{k}}{k+j+1} \cdot \\ & (x_{i+1} - x_i)^j \, x_i^{p+1-j} \cdot \\ & (y_{i+1} - y_i)^k y_i^{q-k}\} \end{aligned} \tag{10}$$

Singer has developed a closed-form expression based on the triangular moments [9]. Singer finds his expression by looking at symmetries in a recursive formulation of the problem and dimensional analysis and uses a rather lengthy argumentation to show the validity of the expression

$$\begin{aligned} \psi_{pq}^{(i)} &= (y_{i+1} x_i - y_i x_{i+1}) \cdot \\ & \sum_{j=0}^{p} \sum_{k=0}^{q} \{ \frac{\binom{p}{j}\binom{q}{k}}{\binom{p+q}{j+k}} \frac{(p+q)!}{(p+q+2)!} \cdot \\ & x_i^j x_{i+1}^{p-j} y_i^k y_{i+1}^{q-k}\} \end{aligned} \tag{11}$$

The two above expression are not slope sensitive and there is only one single expression for all moments. It is easy to speed up computation of any of them by precomputing the binomial fractions which are repeated for each line segment. Since many of the powers are repeated in the computations of moments of different orders these may as well be precomputed and tabulated.

3 Testing the sensitivity for high slope values

To support our claims on numerical instability for high slope values of the 'integration by part' type of methods, we have performed an experiment gradually tilting a unit square by fractions of a degree. A plot of the logarithmic absolute value of the error for each method on moments of order up to the eighth order is given in figure 1. The tests were done in MATLAB which uses double precision, i.e. 64-bits for each number in the computations. Both Singers method and our binomial expansion method perform well, giving neglible errors on bit level. Both the 'integration by parts' expression and Jiang and Bunke's method give severe degradation even for the third order moments. In table 1 we have listed the minimum tilting angle of the unit square giving an error of less than 1/1000.

Moment	Singer	Bin	J & B	IBP
0	0	0	0	0
1	0	0	0	0
2	0	0	0.002	0.001

[3] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8:179–187, February 1962.

[4] Jia-Guu Leu. Computing a shape's moment from its boundary. *Pattern Recognition*, 24(10):949–957, 1991.

[5] Yajun Li. Reforming the theory of invariant moments for pattern recognition. *Pattern Recognition*, 25(7):723–730, 1992.

[6] Luren Yang, Fritz Albregtsen & Torfinn Taxt. Fast computation of three-dimensional geometric moments using a discrete divergence theorem and a generalization to higher dimensions. *Graphical Models and Image Processing*, 59(2):97–108, March 1997.

[7] Strachan, Nevadba & Allen. A method for working out the moments of a polygon using an integration technique. *Pattern Recognition Letters*, 11:351–354, May 1990.

[8] Richard J. Prokop & Anthony P. Reeves. A survey of moment-based techniques for unoccluded object representation and recognition. *Graphical Models and Image Processing*, 54(5):438–460, September 1992.

[9] Mark H. Singer. A general approach to moment calculation for polygons and line segments. *Pattern Recognition*, 26(7):1019–1028, 1993.

[10] Gregory Y. Tang. A discrete version of green's theorem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(3):242–249, May 1982.

[11] Michael Reed Teague. Image analysis via the general theory of moments. *Journal of the Optical Society of America*, 70(8):920–930, August 1980.