Direct Computation of Differential Invariants of Image Contours from Shading

Liangyin Yu Charles R. Dyer

Computer Sciences Department
University of Wisconsin
Madison, Wisconsin 53706 USA

Abstract

In this paper we present a framework combining differential geometry and scale-space to show that local geometric invariants of image contours such as tangent, curvature and derivative of curvature can be computed directly and stably from the raw image itself.

To solve the problem of noise amplification by differential operations, scale-parameterized local kernels are used to replace differential operations by integral operations, which can be carried out accurately when we adopt a continuous image model. We also show that tangent estimation along contours can be made quite accurately using only eight tangent estimators (a $\pi/4$ quantization) when contour location is known, and high precision and efficiency in computation can be achieved for each of the invariants regardless of the differential order involved.

1 Introduction

Image contours exhibit good correspondence between raw images and the physical world at early stages of visual perception. However, it is known that the structure of an image contour is not unique and can be defined at various scales. Generally the processes of locating and representing image contours consist of (i) identifying the location of contours, and (ii) modeling the structure of contours and computing the modeling parameters.

In order to make the structure of image contours explicit, two methods are commonly employed: (i) local edge detection followed by global curve tracing [10] and (ii) global interpolation or energy optimization [6]. The major problem with the first approach is its strictly 1-D sequential processing model and data dependency, e.g., the estimation of curvature depends exclusively on the current edge locations and estimated tangents along the edge, which, in turn, depends on the resolution provided by the edge detector. In this model any error produced in early stages will propagate to and is amplified by all the later stages. Hence higher order geometric invariants of image contours (e.g., curvature) are considered noisy and unstable computationally. The alternative approach generates closed-form curve models by either interpolating edge points or by globally minimizing error-energy functions. The former inherits the errors generated in the edge-detection process and the latter requires a careful design of energy functions to stabilize the results, and both do not perform well across tangent or curvature discontinuities.

To solve these problems, we propose first to replace the discrete image model by a continuous one based on the *sampling theorem* (a continuous signal can be fully recovered from its discrete counterpart if the sampling frequency is higher than the *Nyquist rate*). This continuous model is predicated by always using imaging devices with a resolution higher than the highest resolution that will ever be needed for the computation. Next, by applying scale-space theory [11], the differential operations can be replaced by integral operations so that the inherent problem of noise amplification can be avoided (as previously observed by [3, 8]). This advantage comes with the expense of expanding the spatial range of computation as the order of differentiation increases, but because of the adoption of the continuous image model all the operations can be carried out without quantization error.

With the help of the theoretical framework, we can derive expressions directly relating various local geometric invariants to the raw image. We show that these expressions also embody algorithms for highly efficient and stable computations for the invariants. The invariants being considered include tangent, curvature and derivative of curvature along a contour. We also show how these invariants might be used by a local model of curves such as *local canonical form* to locally represent image contours.

2 Theoretical Framework

2.1 Image Model

An image is modeled traditionally by $I(x, y)$ with x and y taking only integer values in the range of $(0, N-1)$. This discrete model is not adequate in computing local differential invariants and should be replaced by an ensemble of images parameterized by a scale-space parameter σ as follows:

$$I_\sigma(x, y) = \psi_{00}(x, y; \sigma) * I(x, y), \qquad (1)$$

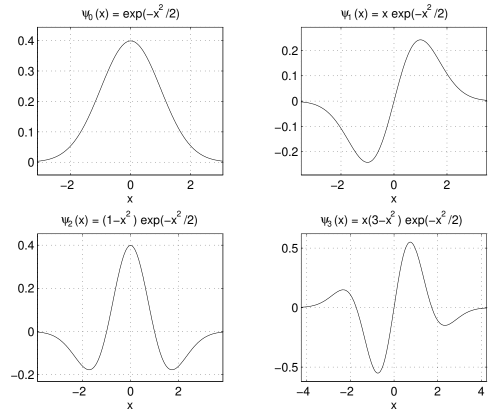

Figure 1: ψ kernel and its differentiations.

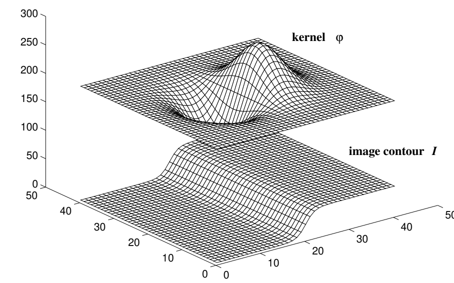

Figure 2: Contour defined by the response of image to ψ kernel at a particular orientation.

where $\psi_{00}(x, y; \sigma)$ is the 2-D Gaussian kernel and $*$ is the convolution operator. The raw image $I(x, y)$ can be viewed as corresponding to the finest resolution (the *inner scale* [3]) available from the optical front-end (e.g., retina) [7]. The 1-D Gaussian kernel ψ_0 is defined as: $\psi_0(x; \sigma) = \exp\left(-\frac{x^2}{2\sigma^2}\right) / \sqrt{2\pi}\sigma$. The i-th order differentiation of $\psi_0(x; \sigma)$ will be denoted by $\psi_i(x; \sigma)$. Since the 2-D Gaussian kernel is separable, we have $\psi_{ij}(x, y; \sigma) = \psi_i(x; \sigma)\psi_j(y; \sigma)$. The kernel ψ_0 and its first-, second- and third-order differentiations are depicted in Figure 1.

The Gaussian as the kernel of scale-space has the important property:

$$\frac{\partial^{i+j}}{\partial x^i \partial y^j}[\psi_{00}(x, y; \sigma) * I(x, y)] = \psi_{ij}(x, y; \sigma) * I(x, y). \tag{2}$$

It is this convolution property that allows us to replace differential operations on the image by integral operations when the image is parameterized into a scale-space.

2.2 Contour Model

Given a well-defined curve $\boldsymbol{c}(s)$ parameterized by the curve length s in 2-D Euclidean space, the Serret-Frenet formula holds in the neighborhood of a given s :

$$\boldsymbol{t}'(s) = \kappa \boldsymbol{n}(s), \ \boldsymbol{n}'(s) = -\kappa \boldsymbol{t}(s), \tag{3}$$

where $\boldsymbol{t}(s)$ and $\boldsymbol{n}(s)$ are the tangent and normal vectors of $c(s)$ at s, and κ is the *curvature* of the curve at s. Using Eq. (3) the first three terms of the Taylor expansion of $\boldsymbol{c}(s)$ (up to the second-order differentiation) can be expressed directly in terms of κ, κ', $\boldsymbol{t}(s)$ and $\boldsymbol{n}(s)$:

$$\begin{aligned} \boldsymbol{c}(s_0 + \epsilon) \quad = \quad & \boldsymbol{c}(s_0) + \left(\epsilon - \kappa^2\epsilon^3/3!\right) \boldsymbol{t}(s_0) \\ & + \left(\kappa\epsilon^2/2 + \kappa'\epsilon^3/3!\right) \boldsymbol{n}(s_0) + R. \end{aligned} \tag{4}$$

This is the *local canonical form* of the curve $\boldsymbol{c}$. The implication of this form is that the curve can be decomposed locally into components along the Frenet frame $(\boldsymbol{t}, \boldsymbol{n})$ and these components, up to a third-order approximation, can be expressed in terms of κ and κ' (derivative of κ with respect to s). In fact, the *fundamental theorem of the local theory of curves* asserts that $\kappa(s)$ is all we need to specify the curve uniquely (up to a rigid transform) [2].

3 Computation of Contour and Invariants

3.1 Contours

Let $x^r = x\cos\theta + y\sin\theta$ and $y^r = -x\sin\theta + y\cos\theta$. Define a 2-D antisymmetric receptive field [9] with orientation θ as

$$P(x, y, \theta; \sigma) \stackrel{\triangle}{=} -\psi_{01}(x^r, y^r; \sigma). \tag{5}$$

A local image contour with orientation θ at scale σ is defined to be a distribution of irradiance $I(x, y)$ such that

$$\begin{aligned} & [(\nabla P(x, y, \theta; \sigma) \cdot \boldsymbol{n}) * I(x, y)] = 0 \\ \text{and} \quad & \\ & [P(x, y, \theta; \sigma) * I(x, y)] \neq 0, \end{aligned} \tag{6}$$

where $\boldsymbol{n} = (\sin\theta, -\cos\theta)$ is the normal vector to the contour. The term $(\nabla P \cdot \boldsymbol{n})$ is the directional derivative of $P(x, y, \theta; \sigma)$ in the direction of $\boldsymbol{n}$ and has the explicit form: $\psi_{02}(x^r, y^r; \sigma) = \psi_0(x^r; \sigma)\psi_2(y^r; \sigma)$. This definition originates from the fact that the response of the image to kernel P has maximum rate of change when moving in the direction orthogonal to θ (see Figure 2). The additional condition is there to exclude those uniform contrast areas of the image. Note that $\psi_{02}(x^r, y^r; \sigma)$ is in the form of a Gabor filter [1]. The location (x, y) defined by Eq. (6) is the maximum response of the antisymmetric kernel P along the direction $\boldsymbol{n}$ and is analogous to the output of an oriented edge detector using the Gabor kernel.

In the following we will drop the σ term in various expressions when deemed appropriate with the understanding that we are dealing with a particular scale σ.

3.2 Tangential Field Along Contours

The tangent vector along an image contour at (x, y) and scale σ is defined as the unit vector with orientation θ such that

$$\Phi(x, y, \theta; \sigma) \overset{\triangle}{=} \frac{\partial P(x, y, \theta; \sigma)}{\partial \theta} * I(x, y) = 0 \quad (7)$$

This definition is motivated by the property that when kernel $P(x, y, \theta)$ is aligned with the local image contour (with orientation θ), the response of convolving the kernel with the image will be maximum. Define the kernel associated with $\Phi(x, y, \theta)$ as

$$\phi(x, y, \theta) = \frac{\partial P(x, y, \theta)}{\partial \theta} = \psi_{10}(x^r, y^r). \quad (8)$$

For a given point (x, y), the *orientation space* at this point is defined as:

$$\Psi(\theta; \sigma) = \phi(x, y, \theta; \sigma) * I(x, y). \quad (9)$$

Since θ is a continuous parameter, the orientation of the tangent can only be estimated by quantizing the orientation space, i.e., we need to determine the resolution of the orientation space in order to locate zero points accurately.

Physiological evidence suggests a quantization resolution of $\pi/18$ (36 quantizations) for the mammalian vision system [5]. We will show that a resolution of $\pi/4$ (8 quantizations) is sufficient if we assume a step edge model.

The orientation space at (0, 0) for a horizontal step edge is given by

$$\Psi(\theta; \sigma) = \int_{-\infty}^{0} \int_{-\infty}^{\infty} \phi(x, y, \theta; \sigma)\, dx\, dy = \frac{\sin\theta}{\sqrt{2\pi}\sigma}, \quad (10)$$

where the edge is going from 1 to 0 when crossing from the negative y-axis to the positive y-axis. The above expression is the output of applying the local kernel $\phi(x, y, \theta)$ to the step edge image. We would like to find the θ that defines the tangential field without the knowledge of the closed-form solution (which is $\sin\theta$ for step edge but unknown otherwise).

Since the sinusoidal function is linear around the zero point, we can estimate the resolution needed by estimating the linearity of the sinusoidal function in the range $(-\pi, \pi)$. If we denote the two points around zero as θ^- and θ^+ for negative and positive orientation samples, then the error between the linear approximation and the actual sinusoid will be (see Figure 3)

$$\epsilon = \frac{\theta^- \sin\theta^+ - \theta^+ \sin\theta^-}{\sin\theta^+ - \sin\theta^-}. \quad (11)$$

Hence, by keeping θ^- and θ^+ within $\pi/4$ we can keep the estimated error of the zero point within 1.3° for this case. The estimated zero point $\theta(x, y)$ is

$$\theta(x, y) = \frac{\Phi(x, y, \theta^+)\theta^- - \Phi(x, y, \theta^-)\theta^+}{\Phi(x, y, \theta^+) - \Phi(x, y, \theta^-)}. \quad (12)$$

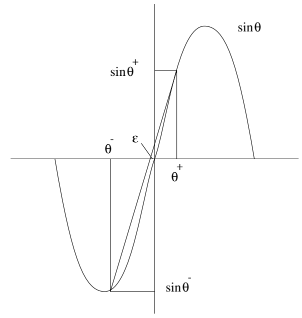

Figure 3: Response of a step edge to ϕ kernel, with linear estimation of the zero-crossing point.

3.3 Curvature Along Contours

By definition, the curvature κ is $d\theta/ds$, where s is the natural parameter (curve length). For a given tangential field, $\theta(x, y)$, and using the chain rule and implicit differentiation, we have $\kappa = \nabla\theta \cdot (\cos\theta, \sin\theta)$. Similar formula holds for $\Phi(x, y, \theta)$ (defined in Eq. 7) and it is straightforward to show that $\nabla\theta = -\nabla\Phi/\Phi_\theta$. Hence $\kappa = -\nabla\Phi \cdot \boldsymbol{t}/\Phi_\theta$.

For simplicity we will use subscripts to denote derivatives, for example, Φ_θ for $\partial\Phi/\partial\theta$. Since $\Phi(x, y, \theta)$ is defined as $\phi(x, y, \theta) * I(x, y)$, we can directly associate Φ with the kernel ϕ and, subsequently, ψ_{ij}. The explicit expressions for $(\nabla\Phi \cdot \boldsymbol{t})$ is $\psi_{20}(x^r, y^r) * I(x, y)$, which is invariant with respect to rotations. We then have the explicit form of curvature at (x_0, y_0), which can be directly used for computation:

$$\kappa(x_0, y_0) = \frac{\psi_{20}(x^r, y^r) * I(x, y)}{\psi_{01}(x^r, y^r) * I(x, y)} \quad (13)$$

A similar formulation of curvature is proposed by Koenderink *et al.* [8] for image blob boundary defined by iso-luminance (the neighborhood around a point on an image contour can indeed be approximated by an iso-luminance contour). However the tangent orientation cannot be computed accurately in their formulation and their expression of curvature for image contours requires a third-order differential to approximate.

3.4 Derivative of Curvature Along Contours

The same method used to derive curvature can also be applied to formulate higher-order geometric invariants. In particular, we are interested in the derivative of curvature, κ', since it is also part of the expression of the local canonical form. The differentiation of curvature with respect to curve length is

$$\begin{aligned}\frac{d\kappa}{ds} &= \theta_{xx}\cos^2\theta + \theta_{yy}\sin^2\theta + 2\theta_{xy}\sin\theta\cos\theta \\ &\quad + (-\theta_x \sin\theta + \theta_y \cos\theta)\kappa. \quad (14)\end{aligned}$$

Figure 4: Image of two synthetic shapes.

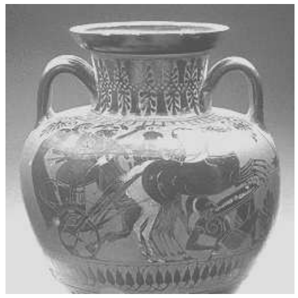

Figure 5: A vase from Smithsonian archive.

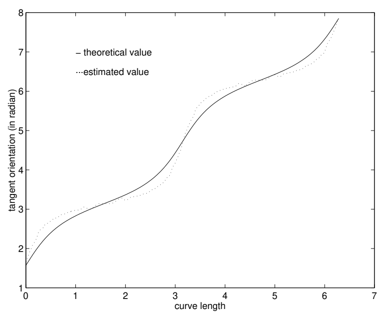

Figure 6: Comparison between theoretical and computed tangent of an ellipse.

We have already derived (θ_x, θ_y) as $-(\Phi_x, \Phi_y)/\Phi_\theta$. Similarly, we can derive θ_{xx}, θ_{yy} and θ_{xy} in terms of various orders of differentiation of Φ. The derivative of curvature can then be expressed in terms of first and second order differentiation of $\Phi(x, y, \theta)$. Using Eq. (8) we can directly compute ϕ_{xx}, ϕ_{xy} and ϕ_{yy}. If we define $\lambda = -\nabla\Phi \cdot \boldsymbol{n}/\Phi_\theta$. it then can be shown

$$\frac{d\kappa}{ds} = \kappa\lambda - \frac{1}{\Phi_\theta}\left(\psi_{30}(x^r, y^r) * I(x, y)\right). \qquad (15)$$

This formula can also be derived by applying the directional derivative of κ in the $\boldsymbol{t}$ direction.

4 Implementation and Examples

We use the images in Figure 4 and 5 to illustrate the implementation. The scale-orientation space is initially partitioned into 4×4 cells, i.e., using four scale partitions spanning from $\sigma = 1.5$ to 4 and four orientation partitions from $\theta = 0$ to 2π. This partition scheme enables us to locate contours through operations in Fourier domain, which is equivalent to performing operations uniformly in the spatial domain. At this stage θ is treated as a quantized parameter and does not get estimated. Next, the orientation space is repartitioned into eight cells and the tangent field is estimated along the contours. In this second pass θ is treated as a continuous parameter, and the computation is conducted in the spatial domain at those contour points.

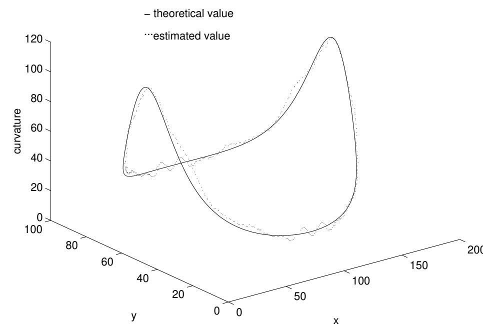

Figure 7: Comparison between theoretical and estimated curvature along an ellipse.

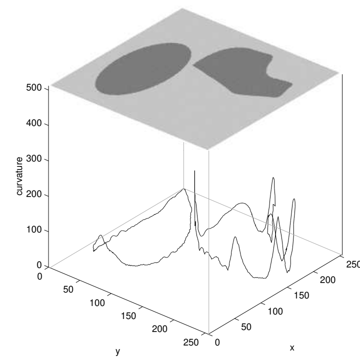

Figure 8: Curvature for shapes image.

For the ellipse in the image of shapes, the theoretical and estimated tangents are shown in Figure 6. The orientation is plotted against the curve length along the ellipse. Similar comparison is also done for curvature and is shown in Figure 7. The result of curvature computation for this image is shown in Figure 8. We also compute the curvature for the vase image along part of the boundary and the top (an ellipse) of the vase and the result is shown in Figure 9. The highest peak of the curvature comes from the concave discontinuity near the vase handle.

5 Discussion

We have shown that for each of the invariants in the local canonical form of an image contour, a set of local kernels can be derived to compute the invariant directly from the raw image. The steps are (i) compute image contours using the kernel $(\nabla P \cdot \boldsymbol{n})$ (Eq. (6)), (ii) compute the vector tangential fields for $I(x, y)$ and express them in the form of (x, y, θ), where the vector field $\boldsymbol{t}$ is $(\cos\theta, \sin\theta)$ (Eq. (7)) and (iii) for points where the tangential field is non-vanishing, compute curvature (Eq. (13))

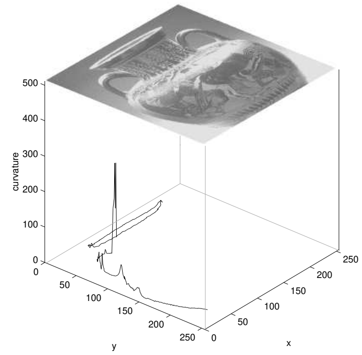

Figure 9: Curvature on boundary of vase.

and then, derivative of curvature (Eq. (15)).

The kernels derived above for computing local geometric invariants along image contours can all be found in biological systems [5, 9] though the antisymmetric kernels are not as populated as the symmetric ones, and they can all be derived from the Gaussian kernel and expand their kernel sizes as the differential order increases. These properties suggest possible connections between the computation of geometric information and the hierarchical organization of natural visual systems. In the rest of this section we consider some details of the computation in our approach.

5.1 Scale and Size of Kernels

When a signal is considered continuous in the modeling process, scales are bounded only by the object systems being modeled. However, when the model is converted to the sampling domain, the range of scales is also dictated by the conversion process, and the kernels of the receptive fields will increase in size due to this conversion.

The expressions of $\psi_i(x;\sigma)$ should all be normalized so that the total area is unity. This is important since they function as filters on images. In order to maintain this property, the normalization factor is proportional to the order of differentiation. This implies an expansion of the filter size if a constant numerical precision is to be kept. This increase of kernel size also constrains the range of scales because the bandwidth of the sampled image is constrained by the Nyquist rate. In fact, the image dimension determines both the upper and lower bound of the scales. If the scale is taken to be multiples of σ (e.g., scale = $\alpha\sigma$, with $\alpha \in I$) in the Gaussian kernel, then the upper bound is $\sigma_{max} = N/2\alpha$, where the image size is $N \times N$. On the other hand, taking N as the Nyquist rate dictates the scale lower bound to be $\sigma_{min} = \alpha/\pi$.

5.2 Contour and Tangent Computations

Theoretically, we can use Eq. (6) to compute the tangent θ. However, as indicated earlier, we can compute image contours in the Fourier domain by treating θ as a quantized parameter. This greatly increases the efficiency of computation at the price of being less precise in estimating θ. On the other hand, after potential contours are located, we need only compute the geometric properties at these contour points and, because of the sinusoidal property indicated by Eq. (10), we can estimate θ with great precision.

6 Conclusions

We showed in this paper that various local geometric invariants of image contours can be computed directly and reliably from the raw image. This approach not only eliminates the drawbacks of error propagation and amplification in conventional approaches but also improves the precision of the results drastically. All of the computations are local without sacrificing the resolution and without resorting to global processes such as energy minimization, which are computationally expensive.

Being able to accurately and reliably compute higher order differential invariants such as derivative of curvature allows us to explore the connection between perception and these geometric invariants (e.g., curve partition and perception [4]). This also makes 2-D visual processes such as perceptual organization more meaningful.

References

[1] J. G. Daugman. Uncertainty relation for resolution in space, spatial frequency. and orientation optimized by two-dimensional visual cortical filters. *J. Opt. Soc. Am.*, 2(7):1160–1169, 1985.

[2] M. P. do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, Englewood Cliffs, NJ, 1976.

[3] L. M. Florack, B. M. H. Romeny, J. Koenderink, and M. A. Viergever. Scale and the differential structure of images. *Image and Vision Computing*, 1992.

[4] D. Hoffman and W. Richards. Parts of recognition. *Cognition*, 18:65–96, 1985.

[5] D. H. Hubel and T. N. Wiesel. Sequence regularity and geomety of orientation columns in the monkey striate cortex. *Journal of Comparative Neurology*, 158(3):267–294, 1974.

[6] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: active contour models. In *Proc. 1st Int. Conf. on Computer Vision*, pages 259–268, 1987.

[7] J. J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.

[8] J. J. Koenderink and W. Richards. Two-dimensional curvature operators. *J. Opt. Soc. Amer. A*, 5(7), 1988.

[9] J. J. Koenderink and A. J. van Doorn. Receptive field families. *Biological Cybernetics*, 63:291–297, 1990.

[10] P. Parent and S. Zucker. Trace inference, curvature consistency, and curve detection. *IEEE Trans. Patt. Anal. Machine Intell.*, 11(8):823–839, August 1989.

[11] A. Witkin. Scale-space filtering. In *Proceedings of the Eighth IJCAI*, pages 1019–1022, 1983.