Distance, curvature

Hugues Talbot and Clara Jaquet

December 13, 2016

1 Normals and curvature

There is a deep link between the evolution of the normal vector along a curve and the curvature itself.

Let C be a planar curve. To define the curvature at a point, we can consider the case of a straight line. We can admit that in this case the curvature is zero along the line. For a portion of a circle, the curvature is defined to be inversely proportional to the radius of the circle:

$$\kappa = \frac{1}{R} \tag{1}$$

More precisely, for any curve, we can locally approximate a sufficiently regular curve ($\mathcal{C}^2$ is enough) at a point by a circle that best approximates it locally (see Fig. 1). This circle is tangent to C and is called an *osculating circle*. The radius of this circle defines the curvature.

Another way to define the curvature is to consider a point moving at a constant speed along the curve C. The variation of the tangent vector along the curve defines the curvature. This is equivalent to specifying the acceleration of the point.

$$\kappa = \frac{d\mathbf{T}}{ds}, \tag{2}$$

where s is a parametrisation of the curve. Both definition of the curvature are in fact equivalent (see Fig. 2).

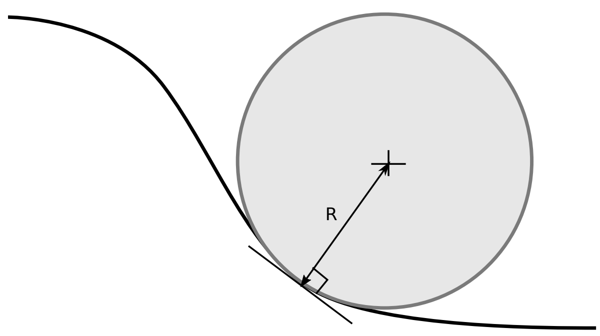

Figure 1: The notion of an osculating circle (best local approximation) defines the curvature as the inverse of the radius of the circle.

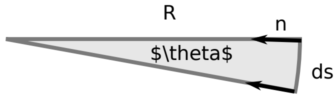

Figure 2: We can write $\sin d\theta \approx d\theta = \frac{ds}{R}$. As ds tends to zero, we have $R = \frac{d\theta}{ds} = \frac{d\mathbf{T}}{ds}$.

Parametrisation

Let $\gamma(t)$ be a parametrisation of the curve C, i.e.

$$\gamma(t) = (x(t), y(t)) \tag{3}$$

This defines the position of a point on the curve over time. We assume an injective parametrisation, i.e. such that the speed $\gamma'(t)$ is never zero. This means

$$\forall t, \|\gamma'(t)\|^2 = x'(t)^2 + y'(t)^2 > 0 \tag{4}$$

In this case, we can re-paramametrise the curve with curvilinear abcissa s in such a way that the speed is constant and equal to one.

$$\forall s, \gamma'(s)^2 = x'(s)^2 + y'(s)^2 = 1 \tag{5}$$

In this parametrisation, γ' is the unit tangent velocity vector $\mathbf{T}$. If $\mathbf{N}$ is the unit normal vector to the curve, we have

$$\mathbf{T}'(s) = \kappa(s)\mathbf{N}(s) \tag{6}$$

We note that instead of deriving the unit tangent vector, we can also consider deriving the unit normal vector. This is because $\mathbf{N}$ is $\mathbf{T}$ rotated by $\frac{\pi}{2}$, i.e. $\mathbf{N}(x, y) = (-y'(s), x'(s))$. This yields

$$\mathbf{N}'(s) = \kappa(s)\mathbf{T}(s) \tag{7}$$

We will make use of that fact in the next section.

Level sets

In imaging it can be difficult to represent a parametric curve because of discretization effects. It is common to represent it by a *level set* (see Fig. 3).

Let $\phi(x, y)$ be a $\mathcal{C}^2$ function in a domain Ω. We define the curve Γ as the zero-level-set of this function

$$\Gamma = \{(x, y), \phi(x, y) = 0\} \tag{8}$$

Γ is a set and no longer a parametrized curve, however we can manipulate it by working on the underlying ϕ function. This is the main idea behind the level-set method [**?**].

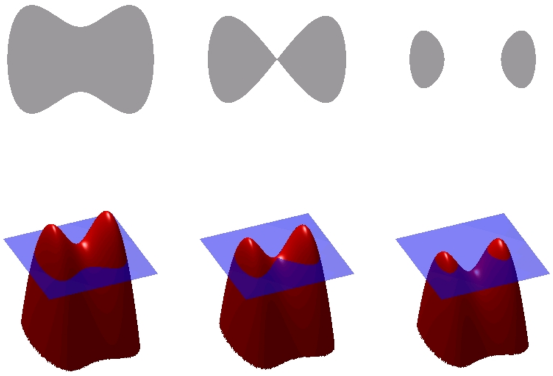

Figure 3: Representing a curve by a level set.

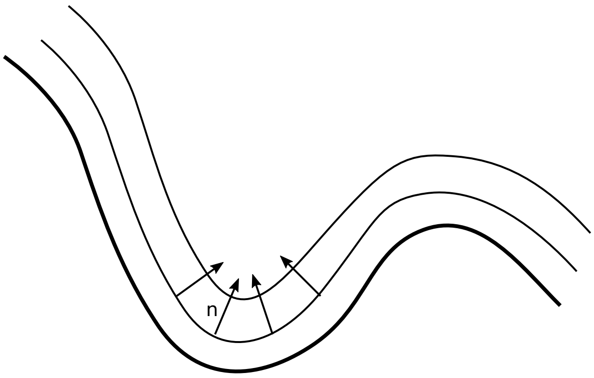

Figure 4: The faster the normal evolves along a curve, the higher the curvature.

Level sets and curvature

Curvature is easy to define in the level set case. For any point (x, y) in Ω, $\nabla\phi(x, y)$ is the gradient at (x, y). If we consider the level-set at $(x, y, \phi(x, y))$, i.e. the curve that passes through (x, y) at level $\phi(x, y)$, then $\nabla\phi(x, y)$ is the normal vector to this curve at (x, y). The unit normal is given by

$$\mathbf{n}(x, y) = \frac{\nabla\phi}{|\nabla\phi|}(x, y) \tag{9}$$

The curvature is given by the derivative of this expression. However this is a multidimensional derivative. Since $\nabla\phi$ is a vector, we must use the divergence operator $\nabla.\mathbf{F} \equiv \sum_i^d \frac{\partial\mathbf{F}}{\partial x_i}$:

$$\kappa = \nabla.\frac{\nabla\phi}{|\nabla\phi|} \tag{10}$$

This is in particular true for the computation of the curvature of Γ.

2 Application to our problem

Here we assume a non-convex territory with two nested surfaces as borders. Every point p of the territory is associated with a value $\omega(p)$ so that $\omega(p) = 0$ if p belongs to the inner surface, and $\omega(p) = 1$ on the outer surface. Numerically, ω can for example be computed from solving the electrostatic Poisson equation, i.e. a random walker. Our formulations and illustrations are in 2D but carry over to 3D without significant changes.

Let $[p_1 p_2]$ be the segment defined by the points $\{p_1, p_2\}$. We want to define how to sample this segment, so that we can conclude with high confidence whether or not the whole segment is located inside a non-convex territory. The number of samples $(k+1)$ shall be optimized in order to avoid too many tests (minimize computation time) but also to guarantee the result within a reasonable tolerance (maximize test accuracy).

2.1 First approach

We can estimate k, the number of samples along the segment, from:

$$k \propto \frac{|p_2 - p_1|}{R} \tag{11}$$

R is an approximation of the local radius of curvature along the segment by estimating the divergence at each point p_1 and p_2, called κ_1 and κ_2 respectively :

$$R = \max(|\frac{1}{\kappa_1}|, \frac{1}{|\kappa_2|}) \tag{12}$$

The figure 5 illustrates the relationship between $D = |p_2 - p_1|$, $R_1 = \frac{1}{|\kappa_1|}$, $R_2 = \frac{1}{|\kappa_2|}$ and k.

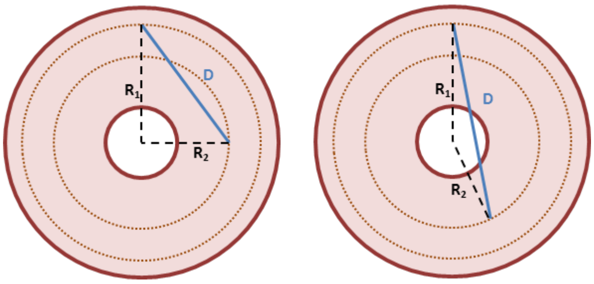

Figure 5: When D becomes larger relatively to $\max(|R_1, R_2|)$, the value of k is also higher to detect if the segment crosses the perfusion territory.

The divergence of a a continuously differentiable vector field $\mathbf{w} = (w_x, w_y)$ is equal to the scalar value function:

$$\mathrm{div}(\mathbf{w}) = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}) \cdot (w_x, w_y) \tag{13}$$

$$\mathrm{div}(\mathbf{w}) = (w_x(x,y) - w_x(x-1,y)) + (w_y(x,y) - w_y(x,y-1)) \,. \tag{14}$$

Since we are interested in estimating a curvature, the field $\mathbf{w}$ is the normalized gradient of ω.

$$w_x = \frac{\nabla_x \omega}{\|\nabla \omega\|} = \frac{\omega(x+1,y) - \omega(x,y)}{\sqrt{(\nabla_x \omega)^2 + (\nabla_y \omega)^2}} \tag{15}$$

and

$$w_y = \frac{\nabla_y \omega}{\|\nabla \omega\|} = \frac{\omega(x, y+1) - \omega(x, y)}{\sqrt{(\nabla_x \omega)^2 + (\nabla_y \omega)^2}} \tag{16}$$

This approach does not take into account the position of p_1 and p_2 within the territory, and so it is easy to find counter-examples where this approach fails.

2.2 Second approach

Here we consider an iterative process to take into account the curvature along a segment.

2.2.1 Interpolating the point along the segment line that crosses the concavity

We calculate the distances λ_1, λ_2 that solves:

$$\omega(p_1) + \lambda_1 \langle \nabla \omega(p_1), \overrightarrow{p_1 p_2} \rangle = 0, \tag{17}$$

and symmetrically for λ_2.

Then we select:

$$\lambda = \min(\lambda_1, \lambda_2) \tag{18}$$

to obtain :

$$k = \left\lceil \frac{1}{\lambda} \right\rceil, \tag{19}$$

where $\lceil . \rceil$ denotes the ceiling operator, i.e. the function that maps a real number to the smallest integer that is larger or equal to it.

If $\lambda < 0$ or $\lambda > 1$, it means the line $(p_1 p_2)$ crosses the concavity outside of the segment $[p_1 p_2]$. We could consider that there is actually no need of refining the test along the segment. But we have to keep in mind that this interpretation comes only from a linear local approximation. In Fig. 6, we show a counter-example, where this approach fails.

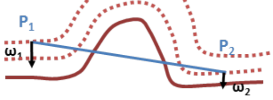

Figure 6: Specific situation where the local gradient doesn't help detecting the concavity.

Also, this method would fail for any case where the dot product $\langle \nabla \omega(p_1), \overrightarrow{p_1 p_2} \rangle \simeq 0$, and in some specific situations such as the figure 2.2.1. Hence we need to define a global boundary:

$$k \geq \frac{||\overrightarrow{p_1 p_2}||}{R_{\max}} \tag{20}$$

with $R_{\max}$ the maximal curvature radius of both inner and outer surfaces.

Note: if the segmentation is noisy, we might actually measure non-physical very high curvatures, i.e. R_{max} close to zero, hence it is necessary to consider a tolerance t so that

$$R^*_{max} = R_{max} - t \tag{21}$$

The semi-cord c defined by R^*_{max} and t is calculated as:

$$c = \sqrt{R^2_{max} - R^{*2}_{max}} \tag{22}$$

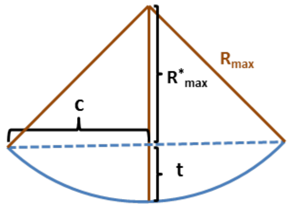

Figure 7: The maximally inscribed radius and tolerance t define a semi-cord c.

That can be simplified as:

$$c = \sqrt{R_{max}^2 - (R_{max} - t)^2} \tag{23}$$

The global boundary follows this tolerance:

$$k \geq \frac{||\overrightarrow{p_1 p_2}||}{c} \tag{24}$$

We take the largest k between maximal curvature and gradient method. This provides a robust sampling definition considering both local and global informations, see figure 2.2.1.

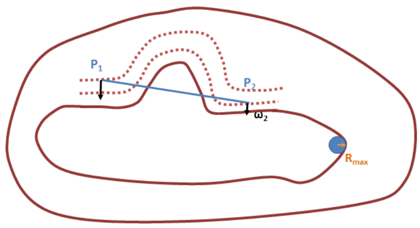

Figure 8: In this situation, considering the global maximum curvature will allow us to detect the concavity between p_1 and p_2.

2.2.2 Estimating a sampling distance from the projection of the border

Given p_1 and p_2, we want to find length M_1 in Fig. 9 from local information.

First we denote p_0 the intersection of the steepest descent from p_1 with the 0-surface, the surface where $\omega = 0$. since $\nabla\omega(p_1)$ and $\overrightarrow{p_0 p_1}$ are co-linear, to first order we can write

$$\nabla\omega(p_1) \approx \frac{\omega(p_0) - \omega(p_1)}{\overrightarrow{p_1 p_0}} \tag{25}$$

Point p_0 is actually unknown, but $\omega(p_0) = 0$, so we have:

$$\overrightarrow{p_1 p_0} \approx \frac{-\omega(p_1)}{\nabla\omega(p_1)} \tag{26}$$

Then we can project this vector $\overrightarrow{p_1 p_0}$ on the segment $gvecp_1 p_2$:

$$M_1 = \frac{\langle \overrightarrow{p_1 p_0}, \overrightarrow{p_1 p_2} \rangle}{||\overrightarrow{p_1 p_2}||} \tag{27}$$

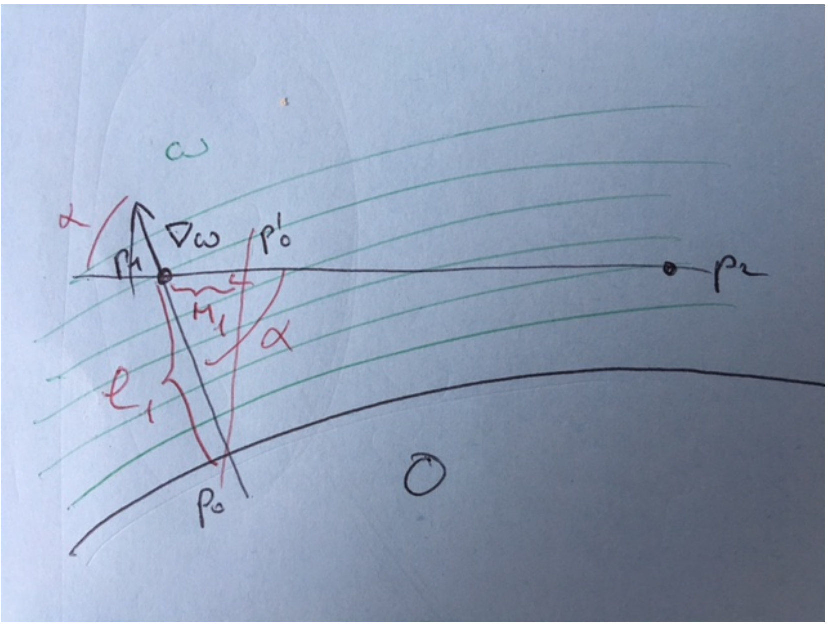

Figure 9: We want to measure length M_1 as a way to sample the segment $[p_1 p_2]$.

Note: the research of $\overrightarrow{p_1 p_0}$ is the Newton method for finding a root, so this estimate can be iteratively refined if need be, although in this case it is not probably very useful.

Symmetrically we can find M_2, by first denoting p_0 the intersection of the steepest descent from p_2 with the 0-surface, and minding the vector reversal:

$$M_2 = \frac{\langle \overrightarrow{p_2 p_0}, \overrightarrow{p_2 p_1} \rangle}{||\overrightarrow{p_2 p_1}||}. \tag{28}$$

From this we take the smallest projection:

$$d_{\text{proj}} = \min(M_1, M_2) \tag{29}$$

This distance d_{proj} is a geometrically meaningful way to sample the segment, so n can be defined as:

$$k = \left\lceil \frac{||\overrightarrow{p_1 p_2}||}{d_{\text{proj}}} \right\rceil \tag{30}$$

These two last methods are expected to be more accurate than the one defined in section 2.1, because we consider the potential along the segment as well as local and global curvatures and not solely at the tips. It will be interesting to compare these methods.