A Novel Curvature Estimator for Digital Curves and Images

Oliver Fleischmann[1], Lennart Wietzke[2], and Gerald Sommer[1]

[1] Cognitive Systems Group, Department of Computer Science
Kiel University, Germany
[2] Raytrix GmbH, Kiel, Germany

Abstract. We propose a novel curvature estimation algorithm which is capable of estimating the curvature of digital curves and two-dimensional curved image structures. The algorithm is based on the conformal projection of the curve or image signal to the two-sphere. Due to the geometric structure of the embedded signal the curvature may be estimated in terms of first order partial derivatives in $\mathbb{R}^3$. This structure allows us to obtain the curvature by just convolving the projected signal with the appropriate kernels. We show that the method performs an implicit plane fitting by convolving the projected signals with the derivative kernels. Since the algorithm is based on convolutions its implementation is straightforward for digital curves as well as images. We compare the proposed method with differential geometric curvature estimators. It turns out that the novel estimator is as accurate as the standard differential geometric methods in synthetic as well as real and noise perturbed environments.

1 Introduction

The estimation of curvature in two-dimensional images is of interest for several image processing tasks. It is used as a feature indicating corners or straight line segments in the case of digital curves [1], in active contour models [2] or for interpolation [3]. In the case of digital plane curves the most common ways to estimate the curvature rely on the estimation of the tangent angle derivative [4], or the estimation of the osculating circle touching the curve [5]. Most of these methods require the segmentation of the curve into digital straight line segments as a preprocessing step [6,7]. Furthermore the application of the these algorithms is limited to digital curves. If it is desired to estimate the curvature of yet unknown curves in images, e.g. isophotes in grayscale images, one has to use the classical definition of curvature known from differential geometry.

We present a curvature estimator based on the idea introduced in [8] which is based on simple convolutions and may be applied to digital curves as well as images.

M. Goesele et al. (Eds.): DAGM 2010, LNCS 6376, pp. 442–451, 2010.

2 Spherical Signal Embedding

We assume a circular signal model in the Euclidean plane with center $\mathbf{m} = r(\cos\theta_{\mathbf{m}}, \sin\theta_{\mathbf{m}})^T \in \mathbb{R}^2$

$$f_{\mathbf{m}}(\mathbf{u}) = \tilde{f}(||\mathbf{u} - \mathbf{m}||) \tag{1}$$

where $\mathbf{u} \in \mathbb{R}^2, \tilde{f} \in L^2(\mathbb{R})$ and $||\cdot||$ denotes the Euclidean norm in $\mathbb{R}^2$. The isophotes $\gamma_{\mathbf{m}}(\mathbf{u})$ of f through $\mathbf{u}$ are circles around $\mathbf{m}$ with radius $\parallel \mathbf{u} - \mathbf{m} \parallel$ given by

$$\gamma_{\mathbf{m}}(\mathbf{u}) = \{\mathbf{v} \in \mathbb{R}^2 : ||\mathbf{v} - \mathbf{m}|| = ||\mathbf{u} - \mathbf{m}||\}. \tag{2}$$

We will estimate the curvature of an isophote $\gamma_{\mathbf{m}}(\mathbf{u})$ by estimating the radius $r_{\mathbf{m}}(\mathbf{u})$ of the osculating circle touching the isophote $\gamma_{\mathbf{m}}(\mathbf{u})$. Since the isophotes are circles, the radius of the osculating circle at every $\mathbf{v} \in \gamma_{\mathbf{m}}(\mathbf{u})$ is the radius of the isophote itself. The curvature of $\gamma_{\mathbf{m}}(\mathbf{u})$ at $\mathbf{u}$ is then obtained as the inverse of the radius $r_{\mathbf{m}}(\mathbf{u})$ as $\kappa_{\mathbf{m}}(\mathbf{u}) = \frac{1}{r_{\mathbf{m}}(\mathbf{u})} = \frac{1}{||\mathbf{u}-\mathbf{m}||}$ [9]. We seek for a fast and exact method to obtain the radius $r_{\mathbf{m}}(\mathbf{u})$ just from the two-dimensional signal $f_{\mathbf{m}}$. The signal $f_{\mathbf{m}}$ is projected to the sphere $\mathbb{S}^2$ with center $(0, 0, \frac{1}{2})^T \in \mathbb{R}^3$ and radius $\frac{1}{2}$ via the inverse stereographic projection

$$\mathcal{S}^{-1}(\mathbf{u}) = \frac{1}{1 + \sum_{i=1}^{2} u_i^2}(u_1, u_2, u_1^2 + u_2^2)^T. \tag{3}$$

It is well known, that the stereographic projection is a conformal mapping and maps circles in the Euclidean plane to circles on $\mathbb{S}^2$ (see e.g. [10]). Hence the circular signal $f_{\mathbf{m}}$ is mapped to a circular signal on $\mathbb{S}^2$

$$g_{\mathbf{m}}(\mathbf{x}) = \begin{cases} f_{\mathbf{m}}(\mathcal{S}(\mathbf{x})) & \text{if } \mathbf{x} \in \mathbb{S}^2 \subset \mathbb{R}^3 \\ 0 & \text{else} \end{cases} \tag{4}$$

where $\mathcal{S}$ denotes the stereographic projection from the sphere $\mathbb{S}^2$ to $\mathbb{R}^2$.

To illustrate the geometric idea of our method (compare also Figure 1) we fix a single isophote $\gamma_{\mathbf{m}}(\mathbf{u})$ and choose a coordinate system in such a way, that $\mathbf{u}$ is the origin and coincides with the southpole $(0, 0, 0)^T$ of the sphere $\mathbb{S}^2$. Furthermore, we project $\gamma_{\mathbf{m}}(\mathbf{u})$ to $\mathbb{S}^2$ as $\mathcal{S}^{-1}(\gamma_{\mathbf{m}}(\mathbf{u}))$. Since $\gamma_{\mathbf{m}}(\mathbf{u})$ is a circle through $\mathbf{u}$ we know, that $\mathcal{S}^{-1}(\gamma_{\mathbf{m}}(\mathbf{u}))$ is a circle on $\mathbb{S}^2$ passing through the southpole of $\mathbb{S}^2$. Let $\mathbf{v} = (0, 0, 1)^T$ denote the northpole of $\mathbb{S}^2$ and define $\mathbf{w} = \mathcal{S}^{-1}(2\mathbf{m})$. $\mathcal{S}^{-1}(\gamma_{\mathbf{m}}(\mathbf{u}))$ is completely characterized by the intersection of a plane $\mathbf{P}_{\mathbf{m}}$ and $\mathbb{S}^2$ with normal vector $\mathbf{n}_{\mathbf{m}} = (\sin\varphi_{\mathbf{m}} \cos\theta_{\mathbf{m}}, \sin\varphi_{\mathbf{m}} \sin\theta_{\mathbf{m}}, \cos\varphi_{\mathbf{m}})^T$ and distance $d_{\mathbf{m}}$ from the origin. Since $\mathcal{S}^{-1}(\gamma_{\mathbf{m}}(\mathbf{u}))$ passes through $(0, 0, 0)^T$ it follows that $(0, 0, 0)^T \in \mathbf{P}_m$ such that $d_{\mathbf{m}} = 0$. Now we claim that

$$\frac{\mathbf{v} - \mathbf{w}}{||\mathbf{v} - \mathbf{w}||} = \mathbf{n}_{\mathbf{m}}. \tag{5}$$

It is sufficient to show that $\mathbf{w}-\mathbf{v}$ is perpendicular to $\mathbf{u}-\mathbf{w}$ such that $\langle \mathbf{w}-\mathbf{v}, \mathbf{u}-\mathbf{w} \rangle = 0$, where $\langle \cdot, \cdot \rangle$ denotes the inner product in $\mathbb{R}^3$. Since $\mathbf{v} = (0, 0, 1)^T, \mathbf{u} = (0, 0, 0)$

and $\mathbf{w} = \frac{1}{2}(\sin\varphi\,\cos\theta, \sin\varphi\,\sin\theta, \cos\varphi + 1)^T$ equation (5) follows immediately. Further by considering the triangle $\mathbf{u}, 2\mathbf{m}, \mathbf{v}$ it follows that

$$\tan\varphi_{\mathbf{m}}(\mathbf{u}) = 2||\mathbf{m}|| = 2r_{\mathbf{m}}(\mathbf{u}) \Rightarrow \kappa_{\mathbf{m}}(\mathbf{u}) = \frac{1}{r_{\mathbf{m}}(\mathbf{u})} = \frac{2}{\tan\varphi_{\mathbf{m}}(\mathbf{u})}. \tag{6}$$

We conclude that the radius of the osculating circle of $\gamma_{\mathbf{u}}(\mathbf{u})$ can be obtained from the angle $\varphi_{\mathbf{m}}$ which we can obtain from the normal vector of $\mathbf{P}_{\mathbf{m}}$. The problem of estimating the radius of the osculating circle is therefore equivalent to the problem of estimating the normal vector $\mathbf{n}_{\mathbf{m}}$ characterizing the plane which intersects $\mathbb{S}^2$ resulting in $\gamma_{\mathbf{m}}(\mathbf{u})$.

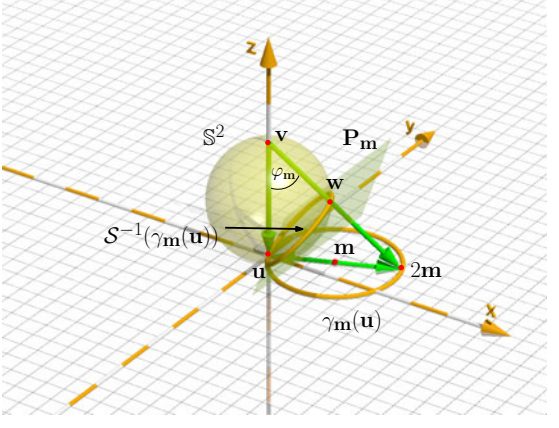

Fig. 1. Illustration of an isophote $\gamma_{\mathbf{m}}(\mathbf{u})$ and its projection $\mathcal{S}^{-1}(\gamma_{\mathbf{m}}(\mathbf{u}))$ to $\mathbb{S}^2$. The radius $||\mathbf{m}||$ can be calculated from angle $\varphi_{\mathbf{m}}$ as $\tan\varphi_{\mathbf{m}} = 2||\mathbf{m}||$. The vector $\frac{\mathbf{w}-\mathbf{v}}{||\mathbf{w}-\mathbf{v}||}$ is equal to the normal vector of the plane $\mathbf{P}_m$. Therefore $\varphi_{\mathbf{m}}$ can be obtained from the normal vector of $\mathbf{P}_m$.

To address the reformulated problem we introduce the Radon transform and its inversion in $\mathbb{R}^3$

$$\mathcal{R}[g](t,\xi) = \int g(\mathbf{x})\delta(t - \langle\xi,\mathbf{x}\rangle)d\mathbf{x} \tag{7}$$

$$\mathcal{R}^{-1}[\mathcal{R}[g]](\mathbf{x}) = \int_{|\xi|=1} h(\langle\mathbf{x},\xi\rangle,\xi)d\xi \quad \text{with} \quad h(t,\xi) = -\frac{1}{8\pi^2}\frac{\partial^2}{\partial t^2}\mathcal{R}[g](t,\xi) \tag{8}$$

with $g \in L^2(\mathbb{R}^3), \mathbf{x} \in \mathbb{R}^3$ and $\xi \in \mathbb{R}^3, |\xi| = 1$. The Radon transform integrates a function g over all two-dimensional hyperplanes in $\mathbb{R}^3$. The hyperplanes are characterized by their normal vectors with orientation ξ and a distance from the origin $t = \langle\xi,\mathbf{x}\rangle$. For a fixed hyperplane orientation ξ, $\mathcal{R}[g](t,\xi)$ describes a one

dimensional function in the Radon domain depending on the distance t of the planes from the origin. We refer to these one-dimensional functions as slices.

Consider the projected circular signal $g_{\mathbf{m}}(\mathbf{x})$ at $\mathbf{x} \in \mathbb{R}^3$. If you imagine the integration over all two-dimensional planes in $\mathbb{R}^3$, than there exists exactly one plane $\mathbf{P}_{\mathbf{m}}$, such that the whole projected circle is located in that plane (compare also Figure 1). The angles $(\theta_{\mathbf{m}}, \varphi_{\mathbf{m}})$ of the normal vector characterizing this plane describe the one-dimensional slice in the Radon domain we are interested in. Therefore we seek for a method to obtain the angles describing exactly this slice in a closed form expression. We use the relationship between partial derivatives in the spatial- and the Radon domain given by (see e.g. [11])

$$\mathcal{R}[\frac{\partial}{\partial x_i} g_{\mathbf{m}}](t, \xi) = \xi_i \frac{\partial}{\partial t} \mathcal{R}[g_{\mathbf{m}}](t, \xi) \tag{9}$$

which states that the Radon transform of every partial derivative with respect to the x_i-axis in the spatial domain is just the partial derivative of the Radon transform of g along each one-dimensional slice in the Radon domain multiplied by the slice angle. Due to our assumed signal model there exists only one slice which is non-constant along t. The slice is exactly the slice described by $(\theta_{\mathbf{m}}, \varphi_{\mathbf{m}})$. If we now apply the inverse Radon transform to Eq. (9) we are able to move the angular components of the normal vector out of the inversion integral such that

$$\frac{\partial}{\partial x_i} g_{\mathbf{m}}(\mathcal{S}^{-1}(\mathbf{u})) = n_{\mathbf{m},i} \mathcal{R}^{-1}[\frac{\partial}{\partial t} \mathcal{R}[g_{\mathbf{m}}]](\mathcal{S}^{-1}(\mathbf{u})). \tag{10}$$

We gain access to the angles of the normal vector

$$\mathbf{n}_{\mathbf{m}} = (n_{\mathbf{m},1}, n_{\mathbf{m},2}, n_{\mathbf{m},3})^T = (\sin\varphi_{\mathbf{m}} \cos\theta_{\mathbf{m}}, \sin\varphi_{\mathbf{m}} \sin\theta_{\mathbf{m}}, \cos\theta_{\mathbf{m}})^T \tag{11}$$

by taking the partial derivatives in $\mathbb{R}^3$ without ever actually performing a Radon- or inverse Radon transform. The angles at $\mathbf{u} \in \mathbb{R}^2$ are obtained as

$$\theta_{\mathbf{m}}(\mathbf{u}) = \arctan \frac{n_{\mathbf{m},2} \mathcal{R}^{-1}[\frac{\partial}{\partial t} \mathcal{R}[g]](\mathcal{S}^{-1}(\mathbf{u}))}{n_{\mathbf{m},1} \mathcal{R}^{-1}[\frac{\partial}{\partial t} \mathcal{R}[g]](\mathcal{S}^{-1}(\mathbf{u}))} = \arctan \frac{\frac{\partial}{\partial x_2} g(\mathcal{S}^{-1}(\mathbf{u}))}{\frac{\partial}{\partial x_1} g(\mathcal{S}^{-1}(\mathbf{u}))} \tag{12}$$

$$\varphi_{\mathbf{m}}(\mathbf{u}) = \arctan \frac{\sqrt{(n_{\mathbf{m},1} \mathcal{R}^{-1}[\frac{\partial}{\partial t} \mathcal{R}[g]](\mathcal{S}^{-1}(\mathbf{u})))^2 + (n_{\mathbf{m},2} \mathcal{R}^{-1}[\frac{\partial}{\partial t} \mathcal{R}[g]](\mathcal{S}^{-1}(\mathbf{u})))^2}}{n_{\mathbf{m},3} \mathcal{R}^{-1}[\frac{\partial}{\partial t} \mathcal{R}[g]](\mathcal{S}^{-1}(\mathbf{u}))} \tag{13}$$

$$= \arctan \frac{\sqrt{(\frac{\partial}{\partial x_1} g(\mathcal{S}^{-1}(\mathbf{u})))^2 + (\frac{\partial}{\partial x_2} g(\mathcal{S}^{-1}(\mathbf{u})))^2}}{\frac{\partial}{\partial x_3} g(\mathcal{S}^{-1}(\mathbf{u}))} \tag{14}$$

where $\theta_{\mathbf{m}}$ is the orientation angle of the center $\mathbf{m}$ in the Euclidean plane and $\varphi_{\mathbf{m}}$ is the angle describing the radius of the osculating circle as stated in Eq. (6). Note that the above formulas are only valid at the origin. Nonetheless this

is no restriction, since we may always treat a point of a plane curve as the origin of a chosen coordinate system, such that the curvature evaluation yields a valid curvature value. For a better understanding of the method introduced above we draw the following analogy: Imagine you want to estimate the curvature of a sampled curve in the Euclidean plane. Then you can estimate the curvature at every sampled point by estimating the osculating circle for a neigborhood of that point. This osculating circle can be estimated by fitting a circle through the points of the neighborhood. The circle fitting is a minimization problem which is, if the geometric distances of the points with respect to the unknown circle are minimized, nonlinear in its nature. By projecting the points on $\mathbb{S}^2$, the nonlinear regression problem turns into a linear one. The circle can now be estimated by fitting a plane through these points. Since the plane has to pass through the origin, the fitting further simplifies. Referring to [12], the minimizing normal vector describing the regression plane is the eigenvector of the moment matrix, obtained from the sample points, corresponding to the smallest eigenvalue. If you further describe this least squares regression as a locally weighted least-squares regression using the Gaussian derivative kernels in $\mathbb{R}^3$, you arrive at the method we introduced by using the relationship of the first order derivatives and the Radon transform in $\mathbb{R}^3$.

2.1 Scale Space Embedding

In general images we want to be able to estimate the curvature of image structures with respect to a certain scale. The embedding of the conformal method in a scale space concept is straightforward. Obvious choices are the Gaussian or the Poisson scale space. Since the curvature is calculated either in terms of partial derivatives, as in this paper, or Riesz transforms as in [8], one may precalculate the convolution kernels resulting in Gaussian derivative kernels or conjugate Poisson kernels (see [13] and [14] for further informations on both scale spaces). The curvature then reads

$$\kappa_s(\mathbf{x}) = 2\frac{G_3^s(\mathbf{x})}{\sqrt{G_1^s(\mathbf{x})^2 + G_2^s(\mathbf{x})^2}} \tag{15}$$

$$G_i^s(\mathbf{x}) = (g * K_i^s)(\mathbf{x}) = \int_{\mathbb{R}^2} g(\mathbf{y}) K_i^s(\mathcal{S}^{-1}(\mathbf{x} - \mathbf{y}))d\mathbf{y} \tag{16}$$

where $\mathbf{x} = \mathcal{S}^{-1}(\mathbf{u}), \mathbf{u} \in \mathbb{R}^2$ and $K_i^s(\mathbf{x})$ either denotes the Gaussian derivative or the conjugate Poisson kernel with respect to x_i in $\mathbb{R}^3$. Further it is possible to implement the method using kernels with bandpass characteristic such as the Difference-Of-Gaussian (DOG) or Difference-Of-Poisson (DOP) kernels, depending on the chosen scale space, as

$$G_i^{s_1,s_2}(\mathbf{x}) = (g * K_i^{s_1})(\mathbf{x}) - (g * K_i^{s_2})(\mathbf{x}). \tag{17}$$

3 Experiments

3.1 Plane Curves

We first evaluate the proposed method for plane curves. Let $\mathbf{l}(t) = (x(t), y(t)), t \in [a, b]$ be a part of a parameterized plane curve. Then we sample $\mathbf{l}$ as $\mathbf{l}_i = (x(i\,d), y(i\,d)), i \in \{1, \ldots, N\}, d = \frac{|a-b|}{N}, N \in \mathbb{N}$. The estimation of the curvature relies on the choice of a scale described by the neighborhood or window size $W \in \mathbb{N}$ with respect to the current point of interest. For each point $\mathbf{l}_i$ we first shift the neighborhood $NB_W(\mathbf{l}_i) = \{\mathbf{l}_{i-W}, \ldots, \mathbf{l}_i, \ldots, \mathbf{l}_{i+W}\}$ to the origin and project it to the sphere $\mathbb{S}^2$ such that $NB'_W(\mathbf{l}_i) = \{\mathcal{S}^{-1}(\mathbf{l}_{i-W} - \mathbf{l}_i), \ldots, (0,0,0), \ldots, \mathcal{S}^{-1}(\mathbf{l}_{i+W} - \mathbf{l}_i)\}$. The curvature is then obtained as

$$\kappa_W(\mathbf{l}_i) = 2\frac{\sum_{j=-W}^{W} K_3^W(\mathcal{S}^{-1}(\mathbf{l}_{i+j} - \mathbf{l}_i)))}{\sqrt{(\sum_{j=-W}^{W} K_1^W(\mathcal{S}^{-1}(\mathbf{l}_{i+j} - \mathbf{l}_i)))^2 + (\sum_{j=-W}^{W} K_2^W(\mathcal{S}^{-1}(\mathbf{l}_{i+j} - \mathbf{l}_i)))^2}} \tag{18}$$

where K_i^W denotes either the Gaussian or the Poisson kernel in the upper half space $\mathbb{R}^3_+$ with respect the scale W. We compare our method to curvature estimation obtained by a circle fitting through the points of the neighborhood $NB'_W(\mathbf{l}_i)$. To fit a circle through these points we use two different distance functions which are minimized, an algebraic distance according to [15] and a geometric distance according to [16]. The distance functions are minimized by solving the corresponding least-squares problems. As a result we obtain the center and the radius r of the fitted circle. The radius serves as a curvature measure due to the already mentioned relation $\kappa = \frac{1}{r}$. Figure 2 shows the comparison of our method with the algebraic and geometric fitting method for three test curves with and without noise. We measured the absolute average error over all curve points for different window sizes as

$$E_W(\mathbf{l}) = \sum_{i=1}^{|\mathbf{l}|} |\kappa(\mathbf{l}_i) - \tilde{\kappa}_W(\mathbf{l}_i)| \tag{19}$$

where $\kappa(\mathbf{l}_i)$ denotes the ground truth curvature and $\tilde{\kappa}_W(\mathbf{l}_i)$ the estimated curvature of the curve at $\mathbf{l}_i$. It turns out that our novel method converges to the true radius of curvature in the case of the assumed signal model, a circle. Compared to the two methods based on circle fittings, it is robust against noise resulting in accurate curvature estimations.

3.2 Digital Images

The method introduced above has the advantage, that it is not limited to digital curves but can also be applied to images, where the curves are not known in advance, e.g. for isophotes in grayscale images. In these cases the curvature is often supposed to serve as a feature indicating corners or straight line segments in the

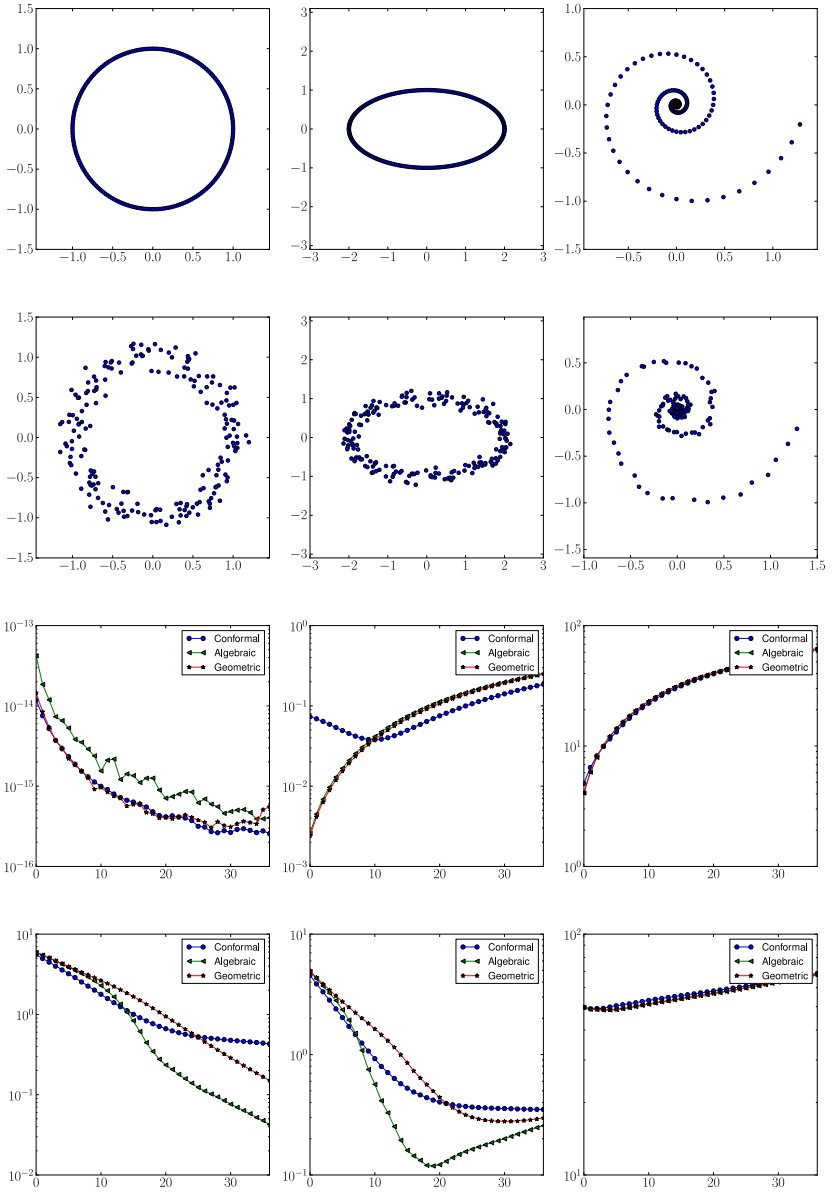

Fig. 2. First row: Test curves sampled at 200 points. Second row: Gaussian white noise perturbed test curves, $\sigma = 0.1$. Third and fourth row: Average absolute curvature errors E_W over all curve points depending on the window size W (abscissa) without and with noise (see also Equ. (19)).

case of high or low curvature. The standard method to obtain the isophote curvature in digital images uses first and second order derivatives. To be comparable to our method, which we introduced in a scale space embedding, we introduce the classical derivative based method to calculate the isophote curvature as [17]

$$\kappa = \frac{2\, G_{1,0}^{s_1,s_2} G_{0,1}^{s_1,s_2} G_{1,1}^{s_1,s_2} - (G_{1,0}^{s_1,s_2})^2 G_{0,2}^{s_1,s_2} - (G_{0,1}^{s_1,s_2})^2 G_{2,0}^{s_1,s_2}}{((G_{1,0}^{s_1,s_2})^2 + (G_{0,1}^{s_1,s_2})^2)^{3/2}} \tag{20}$$

where $G_{i,j}^{s_1,s_2}$ denotes the convolution with the i-th and j-th order derivatives of Difference-Of-Gaussian kernels with scales s_1, s_2 along the x_1 and x_2 axis. To study the accuracy of our estimator we apply it to the artificial signal $f(\mathbf{x}) = \sqrt{x_1^2 + x_2^2}$. Since the test signal depends linearly on the distance from the origin, its ground truth isophote curvature reads $\kappa(\mathbf{x}) = (x_1^2 + x_2^2)^{(-1/2)}$. Figure 3 shows the test signal with and without noise and the average absolute error of both methods for different convolution mask sizes. With increasing convolution mask size the accuracy of our proposed estimator increases. Further it performs better than the derivative based method on the noise perturbed signal even for small convolution mask sizes. Another important aspect of the isophote curvature information is the ability to obtain the ridge curves of an image [17]. The ridge curves are the isophotes for which the gradient vanishes such that the curvature obtained by Equ. (20) is degenerate. Due to their invariance properties concerning translation, rotation and monotonic intensity changes, ridges serve as

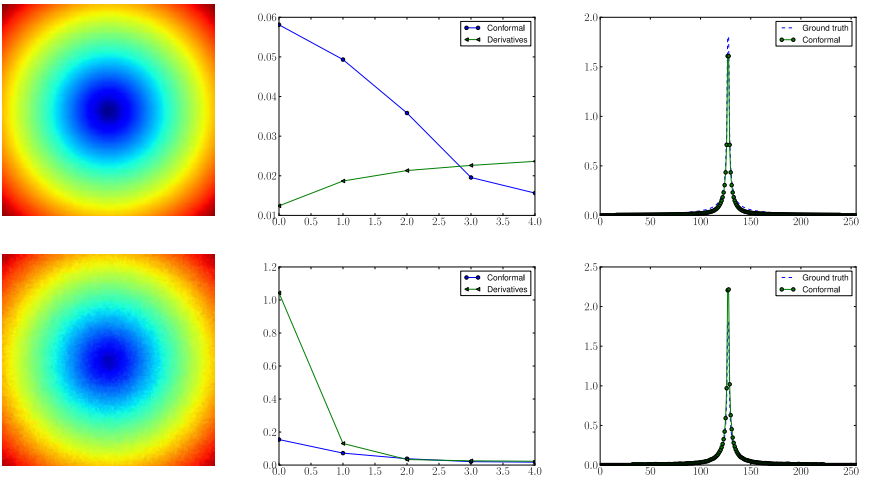

Fig. 3. Left column, top: Ground truth image $f(\mathbf{x}) = \sqrt{x_1^2 + x_2^2}$. Left column, bottom: Ground truth image perturbed with Gaussian white noise, $\sigma = 0.1$. Middle column: Average absolute curvature errors over the whole images for scales $(s_1, s_2) = (2^x, 2^{x+1})$ calculated for the conformal and the derivative based method, where x is the abscissa. Right column: Estimated curvature of the ground truth image for the slice $f(x_1, 0)$ and scales $(s_1, s_2) = (8, 16)$ of the conformal method.

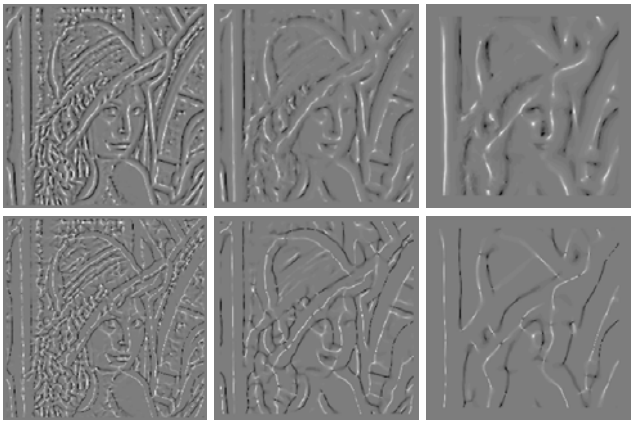

Fig. 4. Top row: Isophote curvatures calculated with the proposed method in Eq. (15) at scales $(s_1, s_2) = (2, 4), (4, 8), (8, 16)$. Bottom row: Isophote curvatures calculated with the classical method in Eq. (20) at scales $(s_1, s_2) = (2, 4), (4, 8), (8, 16)$.

a useful feature, especially if their evolution is considered across multiple scales. Figure 4 shows the ridge curves (degeneracies of the isophote curvature) of the Lenna test-image obtained by the proposed method using Eq. (15) and the curvature obtained according to the classical Eq. (20) using Difference-of-Gaussian convolutions kernels across different scales.

4 Conclusion

We have presented a novel curvature estimator suitable for digital curves and grayscale images. We were able to extract the curvature information by simple linear shift-invariant filters. These filters are applied by convolutions, resulting in a simple and short algorithm. In terms of accuracy our estimator performs at least as accurate as the classical methods. More important is the nature of the construction. Using tools from classical signal processing we transformed the nonlinear problem to a linear one and implicitely developed a method for solving the problem of fitting a circle to a set of points which is classically solved using a least-squares approach. We noticed that our method is equivalent to a corresponding locally weighted least-squares problem. This insight serves as a motivation to construct explicitely the mappings to transform nonlinear image processing problem into linear ones in an appropriate feature space, which was the inverse stereographic projection in this case. The spherical embedding was the first step of investigating certain signal structures in higher dimensional spaces by suitable embeddings.

References

1. Hermann, S., Klette, R.: Global Curvature Estimation for Corner Detection. Technical report, The University of Auckland, New Zealand (2005)
2. Williams, D.J., Shah, M.: A Fast Algorithm for Active Contours and Curvature Estimation. CVGIP: Image Underst. 55(1), 14–26 (1992)
3. Morse, B., Schwartzwald, D.: Isophote-based Interpolation. In: International Conference on Image Processing, vol. 3, p. 227 (1998)
4. Lachaud, J.O., Vialard, A., de Vieilleville, F.: Fast, Accurate and Convergent Tangent Estimation on Digital Contours. Image Vision Comput. 25(10), 1572–1587 (2007)
5. Coeurjolly, D., Miguet, S., Tougne, L.: Discrete Curvature Based on Osculating Circle Estimation. In: Arcelli, C., Cordella, L.P., Sanniti di Baja, G. (eds.) IWVF 2001. LNCS, vol. 2059, pp. 303–312. Springer, Heidelberg (2001)
6. Hermann, S., Klette, R.: Multigrid Analysis of Curvature Estimators. In: Proc. Image Vision Computing, New Zealand, pp. 108–112 (2003)
7. Klette, R., Rosenfeld, A.: Digital Geometry: Geometric Methods for Digital Picture Analysis. Morgan Kaufmann, San Francisco (2004)
8. Wietzke, L., Fleischmann, O., Sommer, G.: 2D Image Analysis by Generalized Hilbert Transforms in Conformal Space. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 638–649. Springer, Heidelberg (2008)
9. do Carmo, M.P.: Differential Geometry of Curves and Surfaces. Prentice-Hall, Englewood Cliffs (1976)
10. Needham, T.: Visual Complex Analysis. Oxford University Press, USA (1999)
11. Zayed, A.: Handbook of Function and Generalized Function Transformations. CRC, Boca Raton (1996)
12. Kanatani, K.: Statistical Optimization for Geometric Computation: Theory and Practice. Elsevier Science Inc., New York (1996)
13. Lindeberg, T.: Scale-space Theory in Computer Vision. Springer, Heidelberg (1993)
14. Felsberg, M., Sommer, G.: The Monogenic Scale-space: A Unifying Approach to Phase-based Image Processing in Scale-space. Journal of Mathematical Imaging and vision 21(1), 5–26 (2004)
15. Gander, W., Golub, G.H., Strebel, R.: Least-Squares Fitting of Circles and Ellipses. BIT (4), 558–578 (1994)
16. Coope, I.D.: Circle Fitting by Linear and Nonlinear Least Squares. Journal of Optimization Theory and Applications 76(2), 381–388 (1993)
17. Romeny, B.M.: Geometry-Driven Diffusion in Computer Vision. Kluwer Academic Publishers, Norwell (1994)