Improved Curvature and Anisotropy Estimation for Curved Line Bundles

Piet W. Verbeek, Lucas J. van Vliet and Joost van de Weijer

Pattern Recognition Group of the Faculty of Applied Physics
Delft University of Technology
Lorentzweg 1, 2628CJ Delft, The Netherlands

email: P.W.Verbeek@ph.tn.tudelft.nl
http://www.ph.tn.tudelft.nl

Abstract

The gradient-square tensor describes the orientation dependence of the squared directional derivative in images. The ratio of eigenvalues is a measure of local anisotropy. For an area showing shift invariance along some orientation (think of a piece of straight rail track) one of the tensor eigenvalues is zero. In practical situations (think of a piece of curved rail track) rotation invariance (perhaps around a remote center) occurs more often than shift invariance. Then curvature contributes to the smallest eigenvalue. In order to avoid this we deform a local area in such a way that the rotational symmetry becomes a translational one. Next the gradient square tensor, defined on the transformed area, is expressed in derivatives of the original area. A curvature corrected anisotropy measure is defined. The correction turns out to be simple and straightforward. An average-curvature estimate for the area results as a valuable byproduct.

1. Introduction

Local structures in 2D and 3D images can be directionally smooth i.e. approximately translation invariant. One way to establish this is the analysis of the average gradient-square tensor (Kass and Witkin, 1987; Haglund, 1992). In its traditional form the tensor describes the quality of translation invariance. There may be two reasons for a bad quality. One is a genuine lack of smoothness, the other is a lack of straightness e.g. if the structure shows rotation invariance. The method proposed here eliminates the curvature contribution and handles local rotation invariance on equal footing to local translation invariance.

In section 2 the traditional approach is discussed, in section 3 a non-linear local coordinate transformation is introduced which straightens curved structures, in section 4 the traditional approach is applied to a tentatively straightened structure, the transformation is optimized to maximize translation invariance in the transformed space. Section 5 deals with the actual implementation and section 6 gives some experimental results.

2. The average gradient-square tensor in Cartesian coordinates

Consider a point (x,y) in an image $f(x,y)$. The component of the gradient (f_x,f_y) along a direction ϕ is (derivative notation by indexes)

$$\begin{aligned} C(\phi,x,y) &= (f_x,f_y)\bullet(\cos\phi,\sin\phi)^T \\ &= f_x\cos\phi + f_y\sin\phi \end{aligned} \tag{1}$$

its square is

$$\begin{aligned} C^2(\phi,x,y) &= (\cos\phi,\sin\phi)\bullet G'\bullet(\cos\phi,\sin\phi)^T \\ &= \tfrac{1}{2}(f_x^2+f_y^2)+\tfrac{1}{2}(f_x^2-f_y^2)\cos 2\phi + f_x f_y\sin 2\phi \\ &= \tfrac{1}{2}g'^2+\tfrac{1}{2}g'^2\cos 2(\phi-\gamma') \end{aligned} \tag{2}$$

where G' is the gradient square tensor

$$G'_{ij} = f_i f_j,\quad i,j=x,y,\ g'^2 = f_x^2+f_y^2$$

and $\gamma' = \arcsin\left(f_y/g'\right) = \arccos\left(f_x/g'\right)$ is the gradient direction. The eigenvalues of G' are $\lambda_1 = g'^2$ and $\lambda_2 = 0$ corresponding to gradient square along the high gradient direction and the low gradient direction respectively.

Now consider a local area $\{x+\xi, y+\eta\}$ around (x,y). Let $U\{.\}$ indicate some weighted averaging over this area. Then we have the average squared gradient component along direction ϕ

$$\begin{aligned} &U\{C^2(\phi,x+\xi,y+\eta)\} = (\cos\phi,\sin\phi)\bullet G\bullet(\cos\phi,\sin\phi)^T \\ &= \tfrac{1}{2}U\{f_x^2+f_y^2\} \\ &\qquad +\tfrac{1}{2}U\{f_x^2-f_y^2\}\cos 2\phi + \tfrac{1}{2}U\{2f_x f_y\}\sin 2\phi \\ &= \tfrac{1}{2}g^2+\tfrac{1}{2}d^2\cos 2(\phi-\gamma) \end{aligned} \tag{3}$$

where G is the average gradient square tensor

$$G_{ij} = U\{f_i f_j\},\quad i,j=x,y$$

$$d^4 = U^2\{f_x^2-f_y^2\}+U^2\{2f_x f_y\}$$

Proceedings of the 14th International Conference on Pattern Recognition, ICPR'98,
Brisbane (Australia), 16-20 August 1998, IEEE Computer Society Press, Vol. I, pp. 528-533.

$$g^2 = U\{f_x^2 + f_y^2\}$$

$$2\gamma = \arcsin\left(\frac{U\{2 f_x f_y\}}{d^2}\right) = \arccos\left(\frac{U\{f_x^2 - f_y^2\}}{d^2}\right)$$

and 2γ is twice the average gradient orientation γ. For later use we introduce local coordinates *v,w* along average gradient (high gradient) and average contour (low gradient) orientation

$$v = \xi \cos\gamma + \eta \sin\gamma \qquad w = -\xi \sin\gamma + \eta\cos\gamma \qquad (4)$$

As γ is only defined modulo π we choose it so that the positive *v*-axis is roughly along the gradient directions in the area (this choice will prove irrelevant for the end result). The eigenvalues of *G* are

$$\begin{aligned} \lambda_1 &= \tfrac{1}{2} g^2 + \tfrac{1}{2} d^2 = U\left\{f_v^2\right\} \\ \lambda_2 &= \tfrac{1}{2} g^2 - \tfrac{1}{2} d^2 = U\left\{f_w^2\right\} \end{aligned} \qquad (5)$$

corresponding to average gradient-square along the high gradient orientation (eigenvector along *v*-axis) and the low gradient orientation (eigenvector along *w*-axis) respectively. For translation invariant structures (with straight parallel isophotes) we still have one zero eigenvalue. For a *y*-translation invariant structure, $f_y = 0$, λ_1=G_{xx} and λ_2=G_{yy}=G_{xy}=0. For an area without preferred direction (noise) we have

$d^2 = U\{f_x^2 - f_y^2\} = U\{2 f_x f_y\} = 0$ and $\lambda_1 = \lambda_2 = \tfrac{1}{2} g^2$.

As a measure of anisotropy, or quality of translation invariance, we use

$$A = \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2} = \frac{d^2}{g^2} \qquad (6)$$

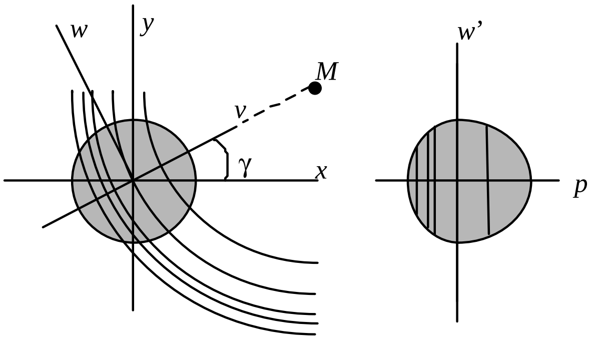

Figure 1: The transform to parabolic coordinates *pw'* straightens circular line pieces.

3. Parabolic coordinates

We already introduced local coordinates *v,w* along average gradient (high gradient) and average contour (low gradient) orientation, now we shall amend these to accommodate curved structures (with approximately circular concentric isophotes). A circle with radius *R–p*, with center *M* on the *v*-axis, and passing through the point *v=p, w*=0 within the area is given by $(v-R)^2 + w^2 = (R-p)^2$. If *R* is large compared to the area size, the circle can be approximated by the parabola $v - \tfrac{1}{2}\kappa w^2 = p$, κ=1/*R*. Thus circles around *M* in *vw*-space are transformed into straight lines in *pw'*-space by the transformation $p = v - \tfrac{1}{2}\kappa w^2$, $w' = w$. Consequently rotation invariance around *M* in *vw*-space is transformed into translation invariance along *w* in *pw'*-space. We propose to use the quality of translation invariance in *pw'*-space as quality of rotation invariance in *vw*- and *xy*-space. Usually the radius *R*=1/κ of (approximate) rotation symmetry is not known for any given area. Our method fits κ to the data.

4. The average gradient-square tensor in parabolic coordinates (*pw'*-space)

Applying the traditional average gradient-square tensor method to *pw'*-space for arbitrary κ to a rotationally symmetric area with center on the *v*-axis but not necessarily at *v*=1/κ we get the tensor

$$\begin{bmatrix} U\left\{f_p^2\right\} & U\left\{f_p f_{w'}\right\} \\ U\left\{f_p f_{w'}\right\} & U\left\{f_{w'}^2\right\} \end{bmatrix} = \begin{bmatrix} U\left\{f_v^2\right\} & 0 \\ 0 & Q + U\left\{f_w^2\right\} - K \end{bmatrix} \quad (7)$$

with the abbreviations

$$\begin{aligned} Q &\equiv (\kappa - \kappa_0)^2 U\left\{w^2 f_v^2\right\} \\ \kappa_0 &\equiv -U\left\{w f_v f_w\right\} \big/ U\left\{w^2 f_v^2\right\} \\ K &\equiv U^2\left\{w f_v f_w\right\} \big/ U\left\{w^2 f_v^2\right\} \end{aligned}$$

Here we have used that for symmetry reasons

$$U\left\{f_p f_{w'}\right\} = \kappa U\left\{w f_v^2\right\} + U\left\{f_v f_w\right\} = 0 \qquad (8)$$

and we have inserted the derivatives

$$\begin{aligned} f_p &= v_p f_v + w_p f_w = f_v \\ f_{w'} &= v_{w'} f_v + w_{w'} f_w = \kappa w f_v + f_w \end{aligned} \qquad (9)$$

derived from the inverse transform $v = p + \tfrac{1}{2}\kappa w'^2$ and $w = w'$. The eigenvalues are

$$\begin{aligned} \lambda_1 &= U\left\{f_v^2\right\} \\ \lambda_2 &= (\kappa - \kappa_0)^2 U\left\{w^2 f_v^2\right\} + U\left\{f_w^2\right\} - K \end{aligned} \qquad (10)$$

corresponding to gradient square along the high gradient direction and the low gradient direction respectively. We see that the low-gradient-square eigenvalue λ_2 can be minimized by choosing κ=κ_0. Therefore we interprete κ_0

as the actual curvature. Choosing a different κ leads to a too high gradient-square estimate in the low gradient direction. This is what happens in the traditional analysis, where κ=0 and $\lambda_2 = U\left\{f_w^2\right\}$. Fitting κ=κ_0 the tensor gets the "curvature corrected" eigenvalues

$$\lambda_1 = U\left\{f_v^2\right\} \text{ and } \lambda_2 = U\left\{f_w^2\right\} - K$$

which leads to the new anisotropy measure

$$A = \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2} = \frac{U\left\{f_v^2\right\} - U\left\{f_w^2\right\} + K}{U\left\{f_v^2\right\} + U\left\{f_w^2\right\} - K} \qquad (11)$$

Note that this measure only depends on the orientation and not on the direction of the *v*-axis. The expression for κ_0 depends on the direction and not only on the orientation of the *v*-axis, hence as a byproduct we have the estimator of *absolute* curvature

$$\left|\hat{\kappa}\right| = \frac{\left|U\left\{w f_v f_w\right\}\right|}{U\left\{w^2 f_v^2\right\}} \qquad (12)$$

Compared to the traditional anisotropy A=d^2/g^2 we now have

$$A = \frac{d^2 + K}{g^2 - K} \qquad (13)$$

Expressions for $U\{w^2 f_v^2\}$ and $U^2\{w f_v f_w\}$ in terms of ξ, η, f_x and f_y are given in Appendix A.

5. Implementation

Using the expansion of $U\{w^2 f_v^2\}$ and $U^2\{w f_v f_w\}$ in terms of ξ, η, f_x and f_y from Appendix A we implemented the average gradient square tensor in parabolic coordinates. The derivatives f_x and f_y are implemented as regularized derivative filters

$$f_x \equiv f(x,y) \otimes \frac{\partial g\left(x,y;\sigma_g\right)}{\partial x} \xleftrightarrow{\mathcal{F}} j\omega_x\, \tilde{f}\left(\omega_x,\omega_y\right) \tilde{g}\left(\omega_x,\omega_y;\sigma_g\right) \qquad (14)$$

with $\tilde{f}$ the Fourier transform of *f* and $g(x,y;\sigma_g)$ a Gaussian regularization function of scale σ_g

$$g(x,y;\sigma) = \frac{1}{2\pi\sigma^2} e^{-\left(x^2+y^2\right)/2\sigma^2} \xleftrightarrow{\mathcal{F}} \tilde{g}\left(\omega_x,\omega_y;\sigma\right) = e^{-\frac{1}{2}\left(\omega_x^2+\omega_y^2\right)\sigma^2} \qquad (15)$$

The terms $U\left\{\xi_a \xi_b f_c f_d\right\}$ with a,b,c,d = x,y (from appendix A) can be implemented in the Fourier domain. At every position (x,y) we compute a weighted inner product between $\xi_a \xi_b$ and $f_c f_d$. This is equivalent to a convolution of $f_c f_d$ with a filter $U\{\xi_a \xi_b\}$. These convolution filters can easily be implemented in the Fourier domain

$$U\left\{\xi^p \eta^q f_x^r f_y^s\right\} = u\left(p,q,\sigma_t\right) \otimes f_x^r f_y^s \xleftrightarrow{\mathcal{F}} \tilde{u}\left(p,q,\sigma_t\right) \mathsf{F}\left\{f_x^r f_y^s\right\} \qquad (16)$$

with $\mathcal{F}\{f_x^r f_y^s\}$ the Fourier transform of $f_x^r f_y^s$. For the weighting function $U\{\}$ we choose a Gaussian window of scale σ_t. The Fourier transform of the convolution filter *u*() is

$$u\left(p,q,\sigma_t\right) \equiv \xi^p \eta^q\, g\left(\xi,\eta;\sigma_t\right) \xleftrightarrow{\mathcal{F}} \tilde{u}\left(p,q,\sigma_t\right) \equiv j^{p+q} \frac{\partial^{p+q} \tilde{g}\left(\omega_\xi,\omega_\eta;\sigma_t\right)}{\partial\omega_\xi^p\, \partial\omega_\eta^q} \qquad (17)$$

Note that the average gradient square tensor first computes the gradient at a (fine) scale σ_g. The tensor elements are then averaged over a larger scale σ_t.

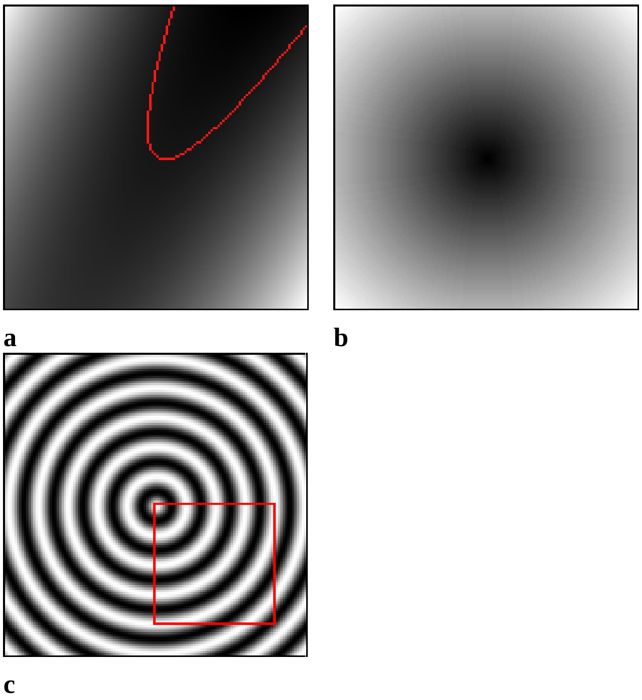

Figure 2: a) $f_1(x,y)$ with a = 0.1, b = 1 and φ = 0.5. The isophote through the center of the image – coordinate (0,0) – is indicated. **b)** $f_2(x,y)$. **c)** $f_3(x,y)$ with a = 2 and n = N(0,0), i.e., noise free. The results inside the square will be displayed in subsequent figures.

6. Experiments

The experiments serve various goals. First to test the theory and its implementation. For this goal we have constructed several test images (c.f. Figure 2). Image $f_1(x,y)$ is a slightly tilted valley with parabolic isophotes and symmetry axis along the φ-direction

$$f_1(x,y) = f_1'(x',y') = ax'^2 + by' \tag{18}$$

with $x' = x\cos\varphi + y\sin\varphi$ and $y' = -x\sin\varphi + y\cos\varphi$. To test the accuracy of the calculations we compute the absolute curvature at position $(x,y)=(0,0)$ in image $f_1(x,y)$ for random orientations φ for values of a and b with $a<<b$. Test image $f_2(x,y)$ is a cone

$$f_2(x,y) = \sqrt{x^2+y^2} \tag{19}$$

and is used to test the applicability of the method for circular symmetric patters of a continuous varying curvature. We can measure the difference in anisotropy with and without curvature correction. We can test the accuracy of the curvature estimation with increasing curvature inside a constant area size.

Test image $f_3(x,y)$ is a bull's eye of concentric rings

$$f_3(x,y) = \cos\left(\sqrt{x^2+y^2}\Big/a\right) + n \qquad n = N\left(0,\sigma^2\right) \tag{20}$$

and allows us to test the performance of the method in data distorted by noise. Traditional (non-averaged) curvature estimation (van Vliet and Verbeek, 1993) using eq. (21) works well on sloped regions (edges), but fails on ridges and in valley's, especially in the presence of noise.

$$\kappa = \frac{-f_{ww}}{f_v} = -\frac{f_{xx}f_y^2 - 2f_{xy}f_xf_y + f_{yy}f_x^2}{\left(f_x^2+f_y^2\right)^{3/2}} \tag{21}$$

Finally, the practical applicability of the extended gradient square tensor is demonstrated on fingerprint images. An example is displayed below. The highly curved region is an important clue and needs to be extracted from noisy data.

7. Results

The results on test image f_1 show that the method is correct. The estimated curvature at position (0,0) is equal to the curvature of the test image at position (0,0), i.e., $2a$. The relative difference of 10^{-7} is caused by round-off errors in the floating point notation. We have tested the method using σ_g = 1.0 and σ_t = 2.0. The test image has a fixed b (b = 1), a random orientation of the symmetry axis φ, and values for a in the range from 0.01 to 0.1.

The results obtained on the cone (test image f_2) are shown in Figure 3. This figure shows that for small isophote radii the smoothness measure (anisotropy) improves significantly when using parabolic coordinates instead of Cartesian coordinates.

The improvements in measuring the local anisotropy and the average curvature are displayed in Figure 4. It is clear that the traditional curvature estimation method works on the edges, but fails on the ridges and in the valleys of the low-frequency images. The traditional curvature estimation method fails completely when applied to high-frequency line bundles which are completely suppressed by the Gaussian regularization. A smaller regularization kernel yields a result dominated by noise (not shown in the figures). The average curvature estimation performs excellent on both low-frequency and high-frequency line bundles. The improvement in local anisotropy is directly related to the performance of the average curvature.

Figure 5 shows the average curvature of a fingerprint image. The two peaks in the curvature image correspond to important minutia for fingerprint recognition (Maio et al., 1997; Jain et al., 1997).

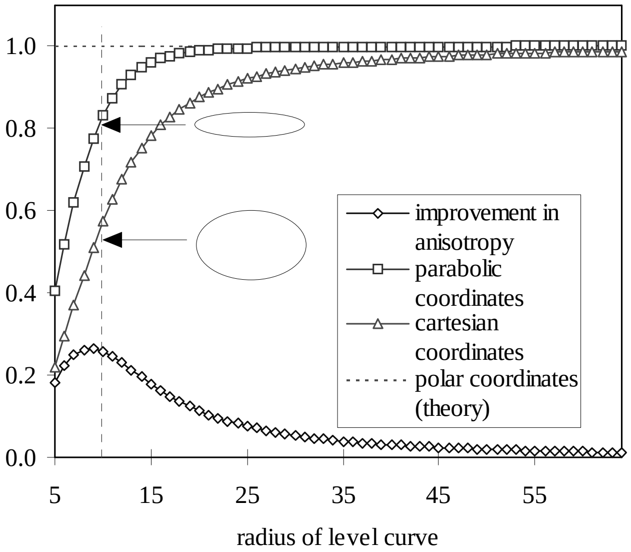

Figure 3: Anisotropy measures obtained on a cone (test image f_2) as a function of curve radius (distance to the center of the cone). The method uses scales σ_g = 1.0 and σ_t = 5.0. The anisotropy of the orientation tensor in Cartesian and parabolic coordinates are measured. Their difference is called improvement in anisotropy. The improvement is largest at R=9. The anisotropy in polar coordinates is per definition equal to one.

8. Conclusions

The orientation tensor according to Haglung yields the local orientation and the local anisotropy in Cartesian coordinates. We extended the orientation tensor to local parabolic coordinates with the symmetry axis parallel to the local orientation. The curvature of the parabolic coordinate system is adaptive such that it maximizes the anisotropy measure.

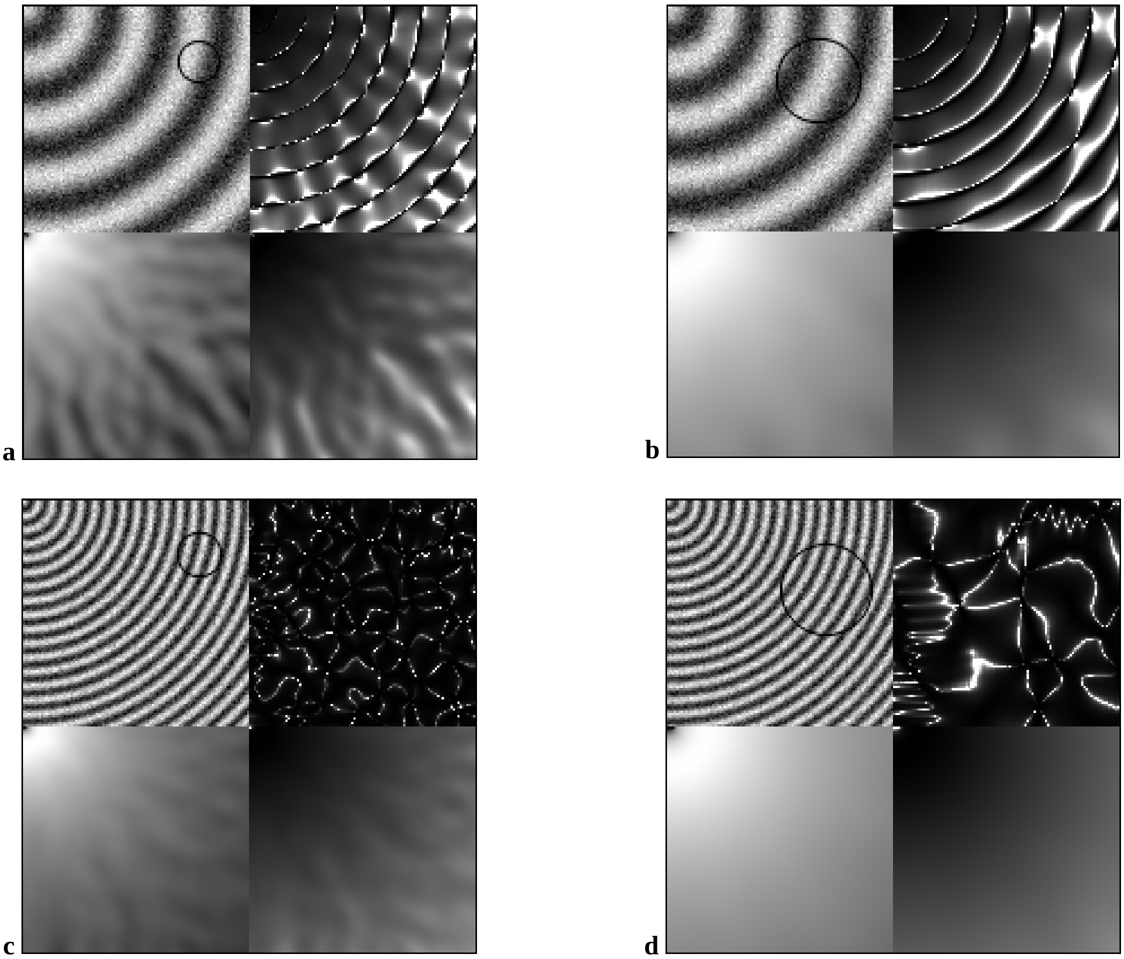

Figure 4: Test results obtained on noisy curved line bundles of different spatial frequency and fixed SNR. (signal amplitude 1, noise $N(0,0.04)$, SNR=10 $\log_{10}(2^2/0.04)$ = 20 dB. Each image contain four parts: ***top-left*** is the input image with the filter support superimposed ($4\sigma_t$ diameter); ***bottom-left*** shows the improvement in local anisotropy (difference between local anisotropy in parabolic and Cartesian coordinates) after logarithmic stretching of the grey-scale; ***top-right*** shows the traditional curvature measured at the same scale as the tensor smoothing; ***bottom-right*** shows the average curvature measured by the orientation tensor in parabolic coordinates.

a) low frequency image ($\sigma_g = 1$, $\sigma_t = 5$); **b)** low frequency image ($\sigma_g = 1$, $\sigma_t = 10$);

c) high frequency image ($\sigma_g = 1$, $\sigma_t = 5$); **d)** high frequency image ($\sigma_g = 1$, $\sigma_t = 10$).

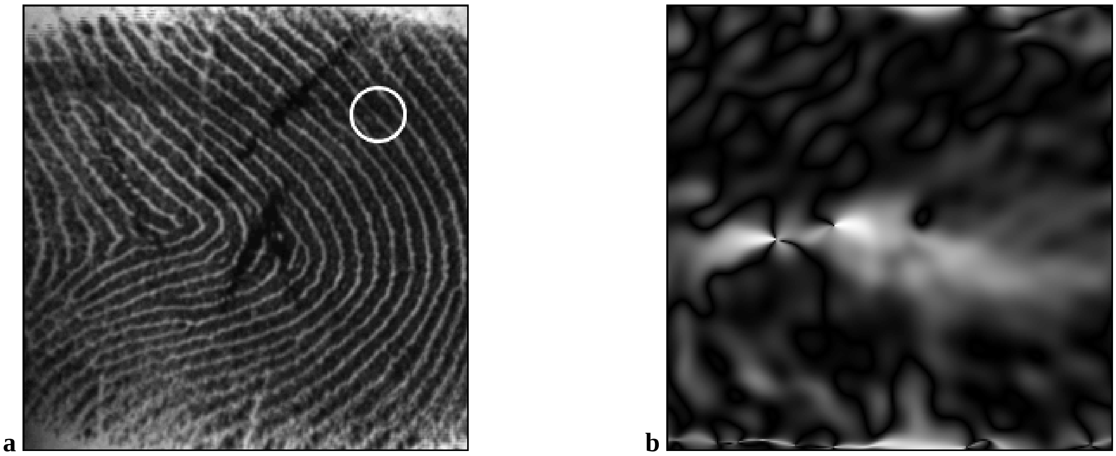

Figure 5: Average curvature of a fingerprint image ($4\sigma_t$ diameter) ($\sigma_g = 1.0.$ $\sigma_t = 7.5$).

This eliminates the contribution of curvature to directional non-smoothness and yields our improved anisotropy measure. As a valuable byproduct we obtain a robust estimator for average curvature. This curvature estimator performs excellent on curved line bundles (of low signal-to-noise ratio) where the traditional curvature estimator, based on a mixture of regularized first and second derivatives, fails.

The method can be extended to higher dimensions. For stacked isophote planes in 3-D, with main non-smoothness in a single direction the correction is still feasible. For isophote bundles in 3-D, with a single smooth direction the method is clear but the burden of computation may be impractical.

9. Acknowledgments

This work was partially supported by the Royal Netherlands Academy of Arts and Sciences (KNAW) and the Rolling Grants program of the Foundation for Fundamental Research in Matter (FOM).

10. References

[1] Kass and Witkin, Analyzing oriented patterns, Computer Vision Graphics and Image Processing, vol. 37, pp. 362-385, 1987.

[2] L. Haglund, Adaptive multi-dimensional filtering, Ph.D. thesis, Linköping, 1992.

[3] L.J. van Vliet and P.W. Verbeek, Curvature and bending energy in 2-D and 3-D digitized images, 8th Scandinavian Conference on Image Analysis, Tromsø (Norway), pp. 1403-1410, 1993.

[4] D. Maio and D. Maltoni, Direct gray-scale minutiae detection in fingerprints, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 1, pp. 27-40, 1997.

[5] A. Jain, L. Hong and R. Bolle, On-line fingerprint verification, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 4, pp. 302-314, 1997.

11. Appendix A

Define normalized eigenvectors

$$n_v = (x_v, y_v) = (\cos\gamma, \sin\gamma)$$

$$n_w = (x_w, y_w) = (-\sin\gamma, \cos\gamma) \tag{22}$$

with

$$\sin 2\gamma = U\{2 f_x f_y\}/d^2 \text{ and } \cos 2\gamma = U\{f_x^2 - f_y^2\}/d^2 \tag{23}$$

then, writing $\xi_x \equiv \xi, \quad \xi_y \equiv \eta$

$$\begin{aligned} v &= \xi_x x_v + \xi_y y_v \quad & f_v &= f_x x_v + f_y y_v \\ w &= \xi_x x_w + \xi_y y_w & f_w &= f_x x_w + f_y y_w \end{aligned} \tag{24}$$

we get the 9 terms of $U\{w^2 f_v^2\}$

$$\begin{aligned} U\{w^2 f_v^2\} &= U\left\{\left(\xi_x x_w + \xi_y y_w\right)^2 \left(f_x x_v + f_y y_v\right)^2\right\} \\ &= \sum_{a,b,c,d=x,y} a_w b_w c_v d_v U\{\xi_a \xi_b f_c f_d\} \end{aligned} \tag{25}$$

and the 6 terms of $U\{w f_v f_w\}$

$$U\{w f_v f_w\} = \sum_{a,b,c=x,y} a_w b_v c_w U\{\xi_a f_b f_c\} \tag{26}$$

Using the definitions for v, w, f_v, and f_w we get

$$\begin{aligned} U\{w^2 f_v^2\} &= U\left\{(-\xi \sin\gamma + \eta \cos\gamma)^2 \left(f_x \cos\gamma + f_y \sin\gamma\right)^2\right\} \\ &= A_{00} + (A_{01} - A_{10}) \cos 2\gamma + (A_{02} - A_{20}) \sin 2\gamma \\ &\quad -A_{11} \cos^2 2\gamma - (A_{12} + A_{21}) \sin 2\gamma \cos 2\gamma \\ &\quad -A_{22} \sin^2 2\gamma \end{aligned} \tag{27}$$

with

$$A_{00} = \tfrac{1}{4} U\left\{\left(\xi^2 + \eta^2\right)\left(f_x^2 + f_y^2\right)\right\} \quad A_{11} = \tfrac{1}{4} U\left\{\left(\xi^2 - \eta^2\right)\left(f_x^2 - f_y^2\right)\right\}$$

$$A_{22} = \tfrac{1}{4} U\left\{2\xi\eta 2 f_x f_y\right\}$$

$$A_{01} = \tfrac{1}{4} U\left\{\left(\xi^2 + \eta^2\right)\left(f_x^2 - f_y^2\right)\right\} \quad A_{10} = \tfrac{1}{4} U\left\{\left(\xi^2 - \eta^2\right)\left(f_x^2 + f_y^2\right)\right\}$$

$$A_{02} = \tfrac{1}{4} U\left\{\left(\xi^2 + \eta^2\right) 2 f_x f_y\right\} \qquad A_{20} = \tfrac{1}{4} U\left\{2\xi\eta\left(f_x^2 + f_y^2\right)\right\}$$

$$A_{12} = \tfrac{1}{4} U\left\{\left(\xi^2 - \eta^2\right) 2 f_x f_y\right\} \qquad A_{21} = \tfrac{1}{4} U\left\{2\xi\eta\left(f_x^2 - f_y^2\right)\right\}$$

and

$$\begin{aligned} U^2\{w f_v f_w\} &= U^2 \left\{\begin{matrix} (-\xi \sin\gamma + \eta \cos\gamma) \\ \left(f_x \cos\gamma + f_y \sin\gamma\right) \\ \left(-f_x \sin\gamma + f_y \cos\gamma\right) \end{matrix}\right\} \\ &= \begin{pmatrix} A_{41}^2 + A_{51}^2 + \left(A_{51}^2 - A_{41}^2\right) \cos 2\gamma \\ -2 A_{41} A_{51} \sin 2\gamma \end{pmatrix} \sin^2 2\gamma \\ &\quad + \begin{pmatrix} A_{42}^2 + A_{52}^2 + \left(A_{52}^2 - A_{42}^2\right) \cos 2\gamma \\ -2 A_{42} A_{52} \sin 2\gamma \end{pmatrix} \cos^2 2\gamma \\ &\quad +2 \begin{pmatrix} -A_{41} A_{42} - A_{51} A_{52} \\ +(-A_{51} A_{52} + A_{41} A_{42}) \cos 2\gamma \\ +(A_{41} A_{52} + A_{51} A_{42}) \sin 2\gamma \end{pmatrix} \sin 2\gamma \cos 2\gamma \end{aligned} \tag{28}$$

with

$$A_{41} = \tfrac{\sqrt{2}}{8} U\left\{\xi\left(f_x^2 - f_y^2\right)\right\} \quad A_{51} = \tfrac{\sqrt{2}}{8} U\left\{\eta\left(f_x^2 - f_y^2\right)\right\}$$

$$A_{42} = \tfrac{\sqrt{2}}{8} U\left\{\xi 2 f_x f_y\right\} \qquad A_{52} = \tfrac{\sqrt{2}}{8} U\left\{\eta 2 f_x f_y\right\}$$