Scale Space Edge Curvature Estimation and Its Application to Straight Lines Detection

Ekaterina V. Semeikina, Dmitry V. Yurin

Laboratory of Mathematical Methods of Image Processing
Faculty of Computational Mathematics and Cybernetic
Moscow Lomonosov State University, Moscow, Russia
esemeikina@graphics.cs.msu.ru, yurin@cs.msu.su

Abstract

The differential geometry based formula for edge curvature estimation at a fixed scale is derived in the paper. It is shown that having lines of various curvatures in the same image, one is not able to estimate them correctly using a fixed scale. So we propose scale space algorithm which allows to choose proper scale adaptively in every point of interest. Relationship between curvature and scale required to estimation was derived analytically, constant was found and the relationship was proved via numerical modeling. The scale space algorithm for edge curvature estimation can be used in particularly for straight line detection. It allows to select only points of interest (points with low curvature). This speeds up detection algorithms and decreases probability of missing lines for complex scenes.

Keywords: *curvature estimation, edge detection, scale space, differential geometry, straight line detection, circle detection.*

1. INTRODUCTION

The idea of scale-space was suggested by Witkin [1] and Koenderink [2] and then was developed by Lindeberg [3]. Scale space approach is used in the theory of differential invariants [8] which have been successfully applied to feature detection, describing of detected features [4,5,6] and edge detection [7]. Informativeness of such differential invariants as gradient absolute value and Laplacian has been proved by effectiveness of their application to edge and feature detection tasks. However, informativeness of high order invariants is unclear from the tasks of matching or learning [12]. In this paper it is first time suggested to use 4-th order invariant – edge curvature – for measurements instead of characterization. It is shown in the paper that edge curvature calculated in scale space is stable enough to classify edge points.

Typical straight line detection scheme consists of two steps: edge detection and straight segment finding. The last step are generally realized via Hough transform (HT) [10,11] or some randomized algorithm. If we remove after the first step all points where edges have high curvature a lot of curve lines will be removed. Thus maxima in accumulator array will be sharper for HT and probability of successful choice will be larger for randomized algorithm.

2. CURVATURE ESTIMATION AT A FIXED SCALE

We will deal with images smoothed via convolution with Gaussian kernel, i.e. we use scale-space $L(x, y, t)$ parameterized with $t = \sigma^2$, where σ is the Gaussian's standard deviation.

Let us consider edge point P on the plane (x, y). We will use so called *gauge coordinates* [8] (v, w) in the point P, where v-axis is tangent to the curve in the point P and w-axis is normal to the curve. Origin of coordinates is in the point P, so $v = 0$, $w = 0$. Curvature in the point (0,0) is $\kappa = w'' = d^2 w(v) / dv^2$.

Let us consider a curve equation $f(x, y) = const$ and obtain the curvature formula for it in a manner similar to [8]. After taking first and second derivatives with respect to v we obtain:

$$f_v + f_w w' = 0 \qquad \textbf{(1)}$$

and $$f_{vv} + f_{vw} w' + f_{wv} w' + f_{ww} (w')^2 + f_w w'' = 0 \qquad \textbf{(2)}$$

As v-axis is tangent to curve in point P, $f_v(0,0) = 0$ and (1) results in $w' = 0$. Thus (2) can be reduced to:

$$f_{vv}(0,0) + f_w(0,0) w''(0) = 0 \,, \qquad \textbf{(3)}$$

$$\kappa = w''(0) = -\, f_{vv}(0,0) / f_w(0,0) \qquad \textbf{(4)}$$

Expression (4) is the equation for curvature of the line defined by the implicit equation $f(x, y) = const$.

In the gauge coordinates edge can be described as points where absolute value of intensity derivative in the w-axis direction $\partial L / \partial w$ have a maxima, i.e. $L_{ww} = 0$.

For edge defined by $f(x, y) = L_{ww}$, the expression $f_v(0,0) = 0$ is satisfied and hence (4) can be used to obtain curvature equation:

$$\kappa_{edge} = w''(0) = -L_{wwvv}(0,0) / L_{www}(0,0) \qquad \textbf{(5)}$$

Equation (5) in coordinates (x, y) can be rewritten explicitly:

$$\kappa_{edge} = -A/B\,, \quad \text{where}$$

$$\begin{aligned} A = {} & \left(L_{xxxx} + L_{yyyy}\right) L_x^2 L_y^2 + L_{xxyy}\left(L_x^4 + L_y^4 - 4 L_x^2 L_y^2\right) + \\ & + 2 L_x L_y \left(L_y^2 - L_x^2\right)\left(L_{xxxy} - L_{xyyy}\right) \end{aligned} \qquad \textbf{(6)}$$

$$B = \sqrt{L_x^2 + L_y^2}\left(L_{xxx} L_x^3 + L_{yyy} L_y^3 + 3 L_x L_y \left(L_{xxy} L_x + L_{xyy} L_y\right)\right)$$

Figure 1 demonstrates curvature estimation with various values of σ for image 1a of size 512×512 containing the circles with radii 10, 30, 50, 100, 150, 200 pixels. It is obvious from (6) that curvature estimation can be carried out only near edges where gradient absolute value is not near to zero. Other regions are marked with white in Figure 1c-e.

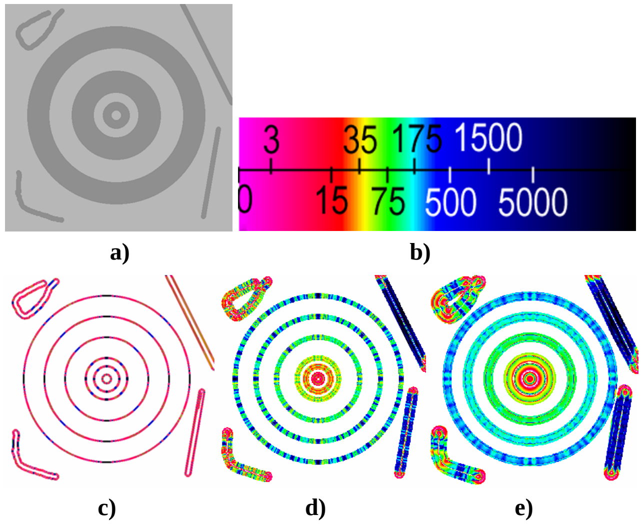


Figure 1: a) the original image, b) correspondence between color and curvature radius $R = 1/\kappa$ in pixels, c-e) curvature estimation using edge curvature formula (6): c) $\sigma = 1$, d) $\sigma = 3.17$, e) $\sigma = 6.35$.

Nonconstant curvature along circle is caused by pixel structure of the image. It is seen that this radial effect decreases with increasing scale σ. Derivatives in (6) are computed via convolutions with derivatives of Gaussian function, i.e. via integration in neighbourhood with radius of order of σ. So, an acceptable accuracy of curvature estimation can be reached if the filled area in Figure 2 contains large enough number of pixels:

$$\Delta \cdot \sigma > const .$$

It can be easily seen from Figure 2 that

$$\Delta = R - R \cdot \cos(\arcsin(\sigma/R)) =$$
$$= R \cdot \left(1 - \sqrt{1 - \sin^2(\arcsin(\sigma/R))}\right) =$$
$$= R \cdot \left(1 - \sqrt{1 - (\sigma/R)^2}\right) \approx 1/2 \cdot R(\sigma/R)^2 = \sigma^2/(2R)$$

So, substituting this result into previous formula we obtain

$$\sigma^3 / R > c \text{, where } c \text{ is constant} \qquad (7)$$

When such $\sigma = \sigma_{req}$ is reached we can see the uniformly colored circles in Figure 1d (red circle $R=10$, $\sigma=3.17$) and 1e (green circle $R=100$, $\sigma=6.35$). While $\sigma < \sigma_{req}$, curvature estimation demonstrated radial structure caused by pixel structure of a curve.

In Figure 1e the typical ring-like artifacts of various colors appeared near circles of small radii. These artifacts are caused by influence of neighbour edges which are closer to each other than distance proportional to σ. In such cases integrals value are result of some “edge mixture” and curvature estimation is bad.

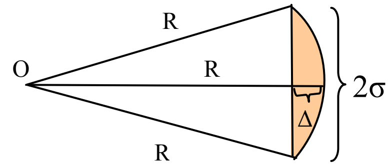


Figure 2: Circle arc.

Any edge in real image is likely to be situated near other edges. Since distance to the closest edge is unknown we use the smallest size of sigma which satisfies (7) to decrease possibility of neighbour edges influence:

$$\sigma_{req}(R) = \sqrt[3]{cR} \qquad (8)$$

So, having edges with different curvatures in the same image it is impossible estimate their curvature using the same σ. It is the reason to develop a scale-space algorithm proposed in section 5.

3. PRECISION OF CURVATURE ESTIMATION AND DETERMINATION OF THE PROPER SCALE

Estimation (8) is confirmed by the experiments with model images each of which contains a single circle of the known radius $R_0 \in [10, 900]$. For these images we calculate curvature in edge points using the same σ for the whole image. Then we build histogram of the obtained curvature values.

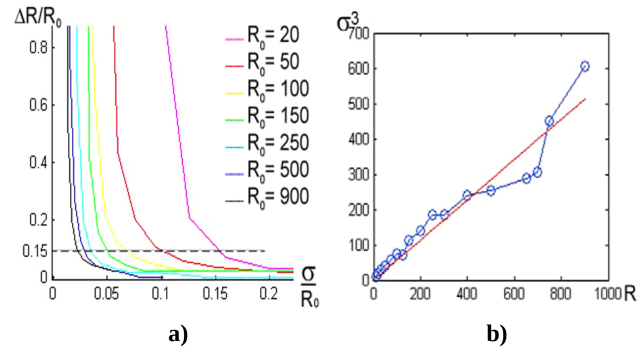


Figure 3: a) the dependence of curvature estimation error $\Delta R/R_0$ and σ/R_0. ΔR is histogram dispersion, R_0 is known radius, b) – minimal value of σ^3 allowing estimation with precision 15%. – approximation of the experimental line.

Curvatures in the points of ideal circle are the same, but we deal with circle at a finite pixel grid and there is radial effect. For curvature estimation with $\sigma < \sigma_{req}(R)$ there is spread of obtained curvature values and its histogram has width $\Delta R = \left\langle (R - \langle R \rangle)^2 \right\rangle^{1/2}$ other than 1. So we need to analyze histogram width (see Figure 3a). Fixing the desired precision ($\Delta R/R_0 = 15\%$ was used) we can determine $\sigma_{req}(R)$ for which chosen precision reaches. Inclination angle of approximation line in Figure 3b gives the coefficient for (7), (8) $c = 1.4$.

4. SCALE SPACE CURVATURE ESTIMATION

In this section we propose an algorithm for scale space curvature estimation. In our tests scale-space was built using σ grid: $\sigma_0 = 1.0$, $\sigma_{i+1} = s \cdot \sigma_i$, $s = \sqrt[3]{2}$, and the constants $\alpha = \beta = 0.2$ were used.

Algorithm 1.

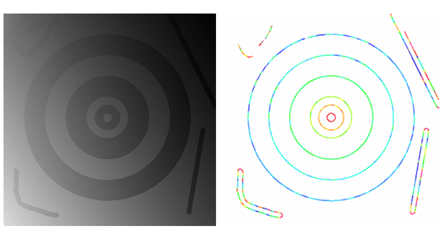
for each scale level σ_i :

for each pixel where curvature has not been calculated yet:

1. Get curvature value κ_i using σ_i from (6).
2. Calculate average $\langle \kappa_i \rangle$ and d mean deviation of κ_i in the edge-neighbourhood of the current point
3. Calculate $R_i := 1/\langle |\kappa_i| \rangle$ and $\sigma_{req}(R_i)$ using (8)
4. **If** $\left(\sigma_{\text{req}}(R_i) \le \sigma_i \text{ or } d \le \alpha|\kappa_i| \text{ or } |\kappa_i - \kappa_{i-1}| \le \beta|\kappa_i|\right)$ **then** $\langle |\kappa_i| \rangle$ is correct, do not calculate value in this point with further scales.

Edge-neighbourhood means only edge points which are situated within neighbourhood of the current point. Following the logic of scale space approach we choose neighbourhood radius dependent on scale and equal to $\max(2, \sigma_i/2)$.

On step 5 of the algorithm we check stability and correctness of the curvature κ_i calculated using σ_i. If curvature deviation within edge-neighbourhood is high: $d > \alpha|\kappa_i|$ then it is cased by noise and radial effect and greater scale should be chosen to correct estimation. Also if curvature value changes sharply from previous scale to the current: $|\kappa_i - \kappa_{i-1}| > \beta|\kappa_i|$, then current scale σ_i is not large enough.



a) b)

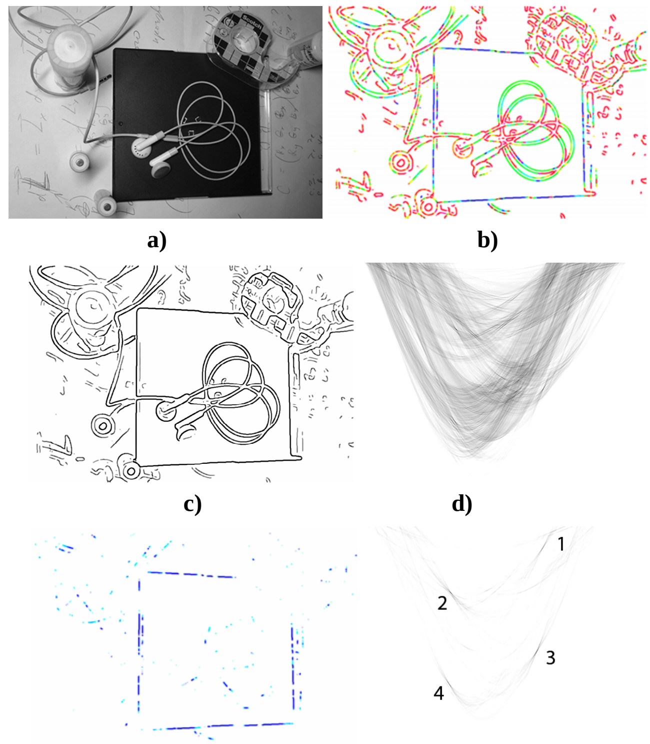
Figure 4: b) Scale space curvature estimation for image 4a. Curvature is shown only in edge points (see 1b for color notation).

It can be seen from Figure 4 that the proposed multiscale algorithm allows correct curvature estimation for images containing lines with various curvatures. Application of statistical differencing [9, p. 288] to images before curvature estimation allows to enhance image quality and to obtain more precise estimation.

5. CURVATURE FILTERING FOR LINE DETECTION

Straight line detection is an example of the area where the proposed curvature algorithm can be applied. A common scene usually contains not only straight lines and selection of points with low curvature allows straight line detector to work only with points of interest and to exclude from consideration impossible line positions (some kind of noise). We demonstrate this benefit on example of Hough transform [10].

For each image from the test base (containing synthetic and real images) we have compared HT sinogram [11] (accumulator array) mapped from all edge points and sinogram mapped from points with low curvature. Typical examples of such sinograms are presented in Figures 5d, f.



a) b) c) d) e) f)

Figure 5: b) curvature estimation for image 6a, c) all edges, d) sinogram built for all edge points, e) edge points with high curvature radius $R > 150$, f) sinogram built for 6e).

In order to compare this sinograms we introduce measure of contrast of sinogram local maxima $T = I_V / I$, where I is intensity in local maxima, I_V is average intensity in a ring neighbourhood of maxima. This comparison has shown that contrast of sinogram maxima corresponding to real straight lines increased significantly (see Table 1), while sinogram maxima due to false line detections were suppressed.

Table 1: Measure of contrast in sinogram local maximum $T = I_V / I$

	Sinogram built for:		
	gradient absolute value	all edges	**filtered edges**
Point 1	0.5	0.32	0.034
Point 2	0.4	0.13	0.03
Point 3	0.37	0.17	0.029
Point 4	0.52	0.18	0.022
Average over test base	**0.4**	**0.15**	**0.03**

Some examples of high curvature edges suppression are shown in the Figure 6.

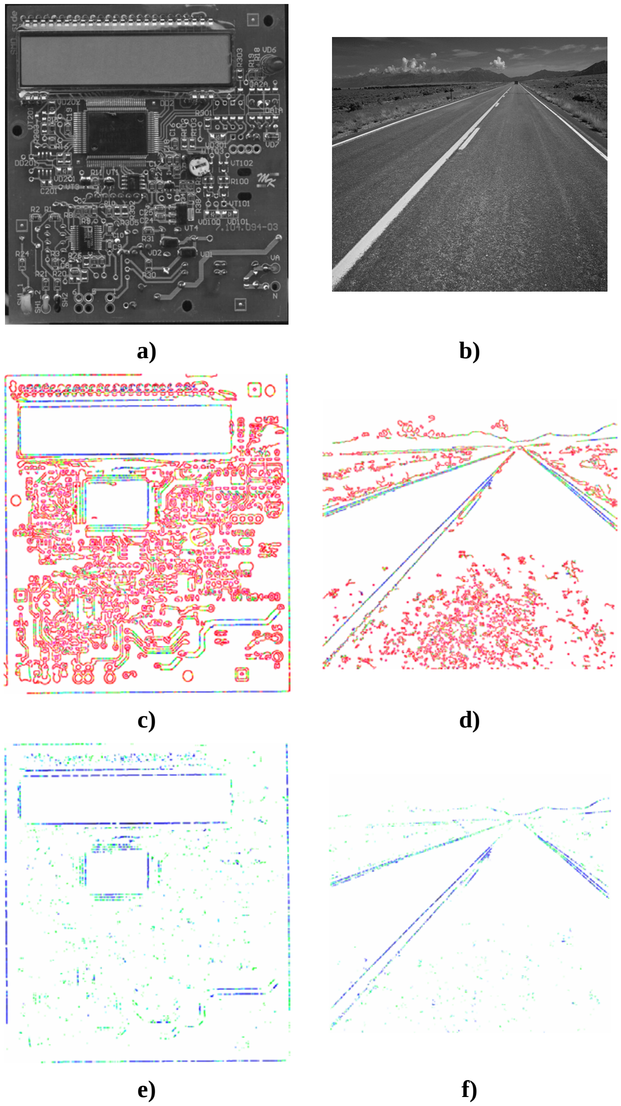


Figure 6: c), d) edges curvature estimation, e), f) edges with low curvature. Colors correspond to curvatures according to Figure 1b

6. CONCLUSION

The differential geometry based formula (6) for edge curvature at a fixed scale has been derived in the paper. It has been shown that curvature estimation at a fixed scale suffers of lack of accuracy so a scale space algorithm for edge curvature estimation has been proposed.

We have analyzed curvature estimation accuracy in dependence on used scale and have obtained a requirement for minimal scale selection; the coefficient in this formula has been obtained with numerical modeling.

The tests performed on synthetic and natural images show that the proposed algorithm allows correct curvature estimation for most of edge points even for images where lines curvature changes in range from small circles to straight lines.

An application of curvature analysis for straight line detection has been suggested. It has been shown that suppression of edge points with high curvature results in increasing sinogram quality and better contrast of its pikes correspondent to straight lines.

7. ACKNOWLEDGMENTS

The research is done under support of RFBR, grants 09-07-92000-HHC_a, 09-01-92470-MHKC_a and federal target program “Scientific and scientific-pedagogical personnel of innovate Russia in 2009-2013”.

8. REFERENCES

[1] A.P.Witkin. *Scale-Space Filtering*. 8th International Joint Conference on Artificial Intelligence, 1019 - 1022, August 1983.

[2] J. J. Koenderink, A. J. van Doorn. *The Structure of Images.* Biological Cybernetics, 50: 363 - 370, 1984.

[3] T. Lindeberg. *Scale–Space Theory in Computer Vision*. Kluwer Academic Publishers, Dordrecht, 1994.

[4] H. Bay, T. Tuytelaars, L. V. Gool. *SURF: Speeded Up Robust Features.* 9^{th} ECCV, 3951(1): 404 - 417, 2006.

[5] D G. Lowe. *Object Recognition from Local Scale-Invariant Features*. ICCV, 1150 - 1157, September 1999.

[6] E. Tola, V. Lepetit, P. Fua. *A Fast Local Descriptor for Dense Matching*. IEEE CVPR, 1 - 8, 2008.

[7] T. Lindeberg. *Edge Detection and Ridge Detection with Automatic Scale Selection*. CVPR, 30(2): 117 - 156, 1998.

[8] L. Florack, B. M. H. Romeny, J. J. Koenderink, M. A. Viergever. *Scale and the Differential Structure of Images.* Image and Vision Computing, 10: 376 - 388, 1992.

[9] W. K. Pratt. *Digital Image Processing: PIKS Scientific inside. – 4^{th} ed*. Wiley-Interscience, 2007.

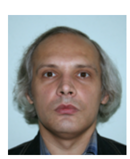
[10] R. Duda, P. Hart. *Use of the Hough Transformation to Detect Lines and Curves in Pictures*. Comm. ACM 15: 1 – 15, 1972.

[11] C.G.Ho, R.C.D.Young, C.D.Bradfield, C.R.Chatwin. *A Fast Hough Transform for the Parameterization of Straight Lines using Fourier Methods.* Real-Time Imaging, 6(2): 113-127, 2000

[12] C. Schmid, R. Mohr. *Local grayvalue invariants for image retrieval*. PAMI, 19(5):530–534, 1997.

About the authors

Ekaterina V. Semeikina is PhD student at Laboratory of Mathematical Methods of Image Processing, Faculty of Computational Mathematics and Cybernetics, Moscow Lomonosov State University http://imaging.cs.msu.su. Her contact email is esemeikina@graphics.cs.msu.ru



Dmitry V. Yurin, PhD, is a senior scientist at Laboratory of Mathematical Methods of Image Processing, Faculty of Computational Mathematics and Cybernetics, Moscow Lomonosov State University http://imaging.cs.msu.su. His contact email is yurin@cs.msu.su.