Numerical reconstruction of a circle from its photographical image at subpixel precision *

Stanislav Šťastník and Jiří Vala,

Brno University of Technology, Faculty of Civil Engineering

602 00 Brno, Veveří 95, Czech Republic

e-mail: Stastnik.S@fce.vutbr.cz, Vala.J@fce.vutbr.cz

Abstract

The paper gives an overview of methods of the analysis of gray-scaled photographical images for circular domains, motivated by the need of sufficiently precise and cheap monitoring of displacements of parts of building constructions in time. The overview of methods applied in the literature is followed by the geometrical analysis of the projection of a planar circle to plane of the snapshot. Several algorithms, based on the numerical analysis of partial differential equations, adopted to this problem, are presented together with the analysis of their properties.

1 Introduction

The analysis of gray-scaled photographical images for circular domains is a serious problem in image processing, namely in applications in astronomy, physics, biology, quality control and metrology, etc. A photographical image of a circle, located in a real plane, as a result of central projection to an other real plane, may be a real two-dimensional quadric (ellipse, hyperbola or parabola, in degenerated cases alternatively some linear set), for good-arranged experiment typically an ellipse, in general with the centre that does not coincide with the image of the original centre of a circle.

Such preliminary sketched problem can be studied using the general theory of pattern recognition. Since we are sure (unlike military applications where nobody can know a priori whether the objects on a real-time image correspond to a geese gaggle or to enemy's bombs) that we are reconstructing a circle, we are allowed to switch to special methods and algorithms for ellipse detection, ignoring other shapes. The number of references only partially documents

*This research was partially supported by the Grant Agency of the Czech Republic, Reg. No. 103/05/0292.

a great number of activities in this research field in the last two decades. Thus it is useful to remind at least their principal directions and approaches. In our rough classification the criterion of mathematical basis will be dominant. However, a lot of other criteria is possible: some methods prefer robustness, other methods computational efficiency, including either low CPU time (which is important namely in real-time applications) or small memory requirements or make use of possibilities of parallel computing, etc.; the result is every time some compromise.

The discussed methods can be divided into two big groups:

1) optimization,
2) voting/clustering.

The characteristic property of methods from the group 1) is that some kind of minimization in some norm or real couples, typically expressing the distance between the points on a boundary curve (that should be an ellipse here) and the measured and pre-processed data; in general a resulting ellipse does contain no given data exactly. Such methods can be relatively simple only if the image is a circle (which is given by 2 parameters only), too. Since the complete description an ellipse needs 5 parameters, in all other cases they are rather complicated; moreover, their quality depends strongly on the good detection of a boundary curve from the noised two-dimensional finite map of gray-scaled pixels (alternatively transformed from the red/green/blue color representation). This gives serious arguments for the development of methods of the group 2), based on some advanced suboptimal algorithms of choice of "correct data" to determine an ellipse, usually in less expensive way. In some more details, the group 1) contains namely

a) least squares fitting,
b) moment of inertia optimization,
c) genetic algorithms,

whereas the group 2) incorporates

d) Huge transform,
e) random sample consensus,
f) algorithms based of fuzzy logic,
g) algorithms based on competitive learning.

Most methods from the literature belonging to the group 1) can be classified as a). Nevertheless, the formal algebraic fitting from the seemingly linear least squares method (LSM), based on the determination of 5 real parameters A, B, C, D, E in the general equation of a plane quadric

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + 1 = 0$$

from a lot of pre-processed n discrete couples (x_i, y_i), $i \in \{1, \ldots, n\}$, of Cartesian coordinates (x, y) in the two-dimensional Euclidean space, is rarely applied because of the difficulties with the realistic error evaluation; nearly all algorithms, as [35] or [13], come from deeper geometrical considerations. The theory of non-linear least squares fitting of an ellipse is summarized in [1]: the cost function is the sum the second powers of distances (Euclidean norms of differences) between all couples (x_i, y_i) and corresponding couples $(\widehat{x}_i, \widehat{y}_i)$, obtained in each iteration step of the generalized Newton method for 5 unknown parameters by orthogonal projection of (x_i, y_i) onto an ellipse. The unpleasant difficulties come already in the setting of calculation of (x_i, y_i): for an arbitrary couple (x_i, y_i) there are 4 possibilities how to construct $(\widehat{x}_i, \widehat{y}_i)$; this corresponds with the formulation of certain algebraic equation of order 4, solvable exactly by the general Cardan formulae (which would lead to a very complicated algorithm – cf. [27], p. 69), by the auxiliary Newton iteration for such algebraic equation (which is unstable by experience here) or for 2 algebraic equations of order 2 (with better stability results). Let us notice that the arguments from the classical projective geometry, presented in [21], p. 55, justify that much better results cannot be expected: all 4 alternatives of choice $(\widehat{x}_i, \widehat{y}_i)$ characterize the intersections of the so-called Apollonius hyperbola with both axes parallel to corresponding axes of an ellipse, thus in generalized polar coordinates with a planar angle φ, $0 \leq \varphi < 2\pi$, the evaluation of $(\widehat{x}_i, \widehat{y}_i)$ lead to the calculation of corresponding φ from an algebraic equation of order 4 again. However, the (not quite simple) constructive geometrical algorithm by [21], pp. 45 and 55, is available.

The above sketched difficulties motivates the derivation of still other method, based of some kind of optimization; some of them cannot be interpreted as certain version of LSM only. This is true e. g. for [6] with its optimization of certain moment of inertia, mentioned as b) here, and also for the application of the theory of genetic algorithms, mentioned as c), derived in [34].

In the group 2) the most frequent approaches belong to d). The theory of the Huge transform (HT) is more than 40 years old and dates back to the US Patent [19]. As a major advantage of HT its insensitivity to imperfect data is appreciated. However, for detecting elliptic shape HT needs 5 real parameters again: usually the center (ξ, η) (2 real numbers), the semiaxes $(\bar{\xi}, \bar{\eta})$ (2 positive numbers) and the orientation angle ϑ, $0 \leq \vartheta < 2\pi$. Various versions, extensions and improvements of HT differs in concrete implementations of its 3 principal steps:

- applying some smoothing mask, usually based on some Gaussian-like convolutions as on semi-Gaussian ones in sense of [38] or on Gaussian convolutions (including derivatives) combined with Hermite integration by [28],
- calculating the edge strength (if possible, with subpixel precision), alternatively also the direction (or convexity, curvature, etc.) at each pixel,
- locating the final edge, based on the voting for parameter values in accumulator arrays and on their best estimation.

From the extensive literature we can mention only several items. Among the variants of HT, the so-called fast Hough transform (FHT), designed in [24], became popular for its computational efficiency. In [35] an ellipse is constructed from 5 points, using certain symmetry in subimages.

In [18] an ellipse is generated from its center, obtained by the intersection of curves from two arrays of midpoints of the pairs of edge points in the same horizontal and vertical position. The way how to obtain edge tangent information is described in [2]; nevertheless, up to now, some authors avoid such tricks completely, as in practice the setting of tangents may be imperfect due to the noise working conditions. The contribution of [23] is in an original efficient setting of symmetrical axes of an ellipse from its symmetric contours. In [33] the detection of the major axis in the outer calculation cycle is preferred, the minor axis is then estimated in the inner cycle (without any tangent information). Further algorithms refer to some properties of an ellipse, known from the descriptive geometry, namely that the line going through the intersection of 2 tangents constructed in 2 points of an ellipse and through the central point on the line connecting such two points contains the centre of an ellipse; certain variant of FHT, called (by the authors) the fast ellipse Hough transform (FEHT), making use of such considerations, is developed in [16]. Following FEHT, in [38] the original robust and fast software code for real-time application, denoted as the real-time ellipse Huge transform (RTEHT), has been presented.

Regardless to the quantity and quality of algorithms d), their certain alternatives e) should not be quite neglected. The idea of random simple consensus (RANSAC) with its special voting scheme was originally formulated in [12]. In [7] RANSAC was improved by some acceleration techniques (and presented as K-RANSAC). The contemporary RANSAC-like methods can be represented by the edge-projected integration of cues (EPIC), described in details in [30]. Only a minority of methods from the group 2) cannot be classified neither as HT-like nor as RANSAC-like approaches. In the case e) they are produced by specialists in fuzzy sets – cf. [9], in the case f) by specialists in neural networks and their applications to competitive learning – cf. [37] (although both authors are active also in HT research).

An important part of the correct ellipse detection, needed in both groups 1) and 2), is the good choice of approximating points from a gray-scaled two-dimensional map, consisting of a finite number of rectangles with constant intensities between black and white. Quick (especially real-time) applications make use of some intuitive searching for maximal derivatives, as the convexity matching scheme in 8 equal planar sections in [37], p. 276, preceded by certain discretized version of Gaussian smoothing. Better approximations may be exploited using the methods of numerical solution of partial differential equations of evolution, namely a nonlinear diffusion equation of Perona-Malik type. Much more information to such methods can be found in [10] and [32]; the complete theory, including existence and uniqueness results, coming from the properties of Rothe sequences, is presented in [20]. From our field of interest namely the development of geodesic active contours, sketched in [32], p. 45 (including an original numerical approximation, based on certain finite difference scheme with harmonic averaging), theoretically analyzed in [4], should be applicable, e. g. as an efficient pre-processing to a).

2 Least squares technique in central projection of a circle

The following analysis is motivated by the fact that, unlike most cited references, we are sure that our ellipse is an image of a circle; The main aim is to find the position of its center from such image with subpixel precision; the speed of calculation (as in real-time applications) is not the decisive criterion. We intend to derive a method based on least squares technique, but without tough repeated calculations of distances between detected points and some (quasi)ideal ellipse.

Let us now study the reconstruction of the position of a real circle from its photographical image in the Euclidean space with the Cartesian coordinates (x, y, z). We shall start with the formulation of 3 parametric equations of a circle ω with a center $S = (x_0, y_0, z_0)$ and a radius r in certain plane ϖ

$$a(x - x_0) + b(y - y_0) + c(z - z_0) = 0\,, \tag{1}$$

defined (up to a multiplicative factor) by its normal (a, b, c); a, b and c are real parameters, at least one must be non-zero. Without loss of generality we are allowed to consider the projection from the centre $P = (0, 0, -\zeta)$ (a non-zero real parameter ζ is set by the camera), onto the plane $\widetilde{\varpi}$, characterized by $z = 0$; the image of ω is an ellipse $\widetilde{\omega}$. We do not admit P contained in ϖ, thus we must suppose by (1)

$$ax_0 + by_0 + c(z_0 + \zeta) \neq 0\,.$$

In such geometrical configuration ω lies in the intersection of a sphere

$$x = x_0 + r \sin\psi \cos\varphi\,, \qquad y = y_0 + r \sin\psi \sin\varphi\,, \qquad z = z_0 + r \cos\psi$$

with variable angles $0 \le \varphi < 2\pi$ and $0 \le \psi \le \pi$ and of a plane defined by (1). Let us remark that such parameterization is not suitable for $c = 0$; however, this case with ϖ perpendicular to $\widetilde{\varpi}$ is of low practical importance (we strive to preserve rather small angle between ϖ and $\widetilde{\varpi}$), thus the derivation of analogous equations with another parameterization can be left to the inquisitive reader. For simplicity (to avoid other possible degenerations with our camera close to ω) we shall also assume $z_0 + \zeta > r$. Thus we have

$$ar \sin\psi \cos\varphi + br \sin\psi \sin\varphi + cr \cos\psi = 0$$

and consequently

$$\sin\psi = \frac{c}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2}}\,, \qquad \cos\psi = -\frac{a\cos\psi + b\sin\psi}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2}}\,.$$

The resulting equations of ω are

$$\begin{aligned} x &= x_0 + \frac{rc\cos\varphi}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2}}\,, \\ y &= y_0 + \frac{rc\sin\varphi}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2}}\,, \\ z &= z_0 - \frac{r(a\cos\varphi + b\sin\varphi)}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2}}\,. \end{aligned} \tag{2}$$

Without loss of generality we are allowed to consider the projection from the centre $P = (0, 0, -\zeta)$ (the positive parameter ζ is set by the camera), onto the plane $\widetilde{\varpi}$, characterized by $z = 0$; the image of ω is an ellipse $\widetilde{\omega}$; Using (2), we can see that the equations of all surface lines of a cone containing ω and going through S and through some point of ω are

$$\begin{aligned} x &= \left(x_0 + \frac{rc\cos\varphi}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2}} \right) t\,, \\ y &= \left(y_0 + \frac{rc\sin\varphi}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2}} \right) t\,, \\ z &= -\zeta + \left(z_0 + \zeta - \frac{r(a\cos\varphi + b\sin\varphi)}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2}} \right) t \end{aligned}$$

for an arbitrary real parameter t. Their intersection with the projection plane $z = 0$ corresponds to

$$t = \zeta \left(z_0 + \zeta - \frac{r(a\cos\varphi + b\sin\varphi)}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2}} \right)^{-1}.$$

This generates the equations of $\widetilde{\omega}$

$$\begin{aligned} x &= \frac{\zeta\left(x_0\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2} + rc\cos\varphi \right)}{(z_0 + \zeta)\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2} - r(a\cos\varphi + b\sin\varphi)}\,, \\ y &= \frac{\zeta\left(y_0\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2} + rc\sin\varphi \right)}{(z_0 + \zeta)\sqrt{(a\cos\varphi + b\sin\varphi)^2 + c^2} - r(a\cos\varphi + b\sin\varphi)}\,, \\ z &= 0\,; \end{aligned} \tag{3}$$

φ is still a variable angle. The coordinates of the image $\widetilde{S} = (\widetilde{x}_0, \widetilde{y}_0, 0)$ of S are

$$\widetilde{x}_0 = \frac{\zeta x_0}{z_0 + \zeta}\,, \qquad \widetilde{y}_0 = \frac{\zeta y_0}{z_0 + \zeta}\,;$$

this can be verified easily from (3) for $r \to 0$.

Let us notice some special cases. We have excluded $\widetilde{\varpi}$ perpendicular to ϖ, corresponding to $c = 0$. If, in particular, $\widetilde{\varpi}$ is parallel to ϖ then $a = b = 0$ and (3) degenerate to

$$x = \frac{\zeta(x_0 + r\cos\varphi)}{z_0 + \zeta}\,, \qquad y = \frac{\zeta(y_0 + r\sin\varphi)}{z_0 + \zeta}\,, \qquad z = 0\,.$$

It is evident that $\widetilde{\omega}$ here is a circle again (not a general ellipse); however, this cannot be arranged usually in practice. The (perhaps more realistic) choice $a = 0 \neq b$ or $b = 0 \neq a$ does not bring substantial simplifications; the corresponding equations can be easily rewritten from (3).

Let us also remark that the relations between ω and $\widetilde{\omega}$ can be alternatively studied by means of the classical descriptive geometry, applying the theory of central projection and fotogrammetry. Much more information can be found in [22], p. 91, but the discussed constructive approaches (as searching for intersections of couples of tangents to an ellipse $\widetilde{\omega}$ and solving a corresponding inverse problem) do not allow a sufficiently simple and transparent algebraic description, needed in our considerations; Nevertheless, such approaches could be helpful to construct (using a ruler and a drawing compass only) the first rough approximation of the location of ω from $\widetilde{\omega}$ if no better information is available for the subsequent algebraic improvement.

Let us come back to (3) with $c = 1$ (if $c \neq 0$, this can be assumed with no loss of generality); let us introduce the brief notation

$$\rho = \frac{r}{z_0 + \zeta}\,, \qquad \Phi(a, b, \varphi) = \frac{1}{\sqrt{(a\cos\varphi + b\sin\varphi)^2 + 1}}\,.$$

Then (3) obtain the seemingly simple form

$$x = \frac{\widetilde{x}_0 + \rho\Phi\cos\varphi}{1 - \rho\Phi(a\cos\varphi + b\sin\varphi)}\,, \qquad y = \frac{\widetilde{y}_0 + \rho\Phi\sin\varphi}{1 - \rho\Phi(a\cos\varphi + b\sin\varphi)} \tag{4}$$

(the third equation is not needed). Let us believe that we know r (diameter of ω) and ζ (camera characteristic) exactly. For some fixed parameters a, b (global position of ϖ), $\widetilde{x}_0$, $\widetilde{y}_0$ and ρ (location of ω in ϖ, or $\widetilde{\omega}$ in $\widetilde{\varpi}$, respectively) we need to determine φ from x and/or y. We shall assume that we have (as the result of some pre-processing procedure) a finite number n couples of inexact coordinates (x_i, y_i), $i \in \{1, \ldots, n\}$, of points of $\widetilde{\omega}$. Then for any such i the best way for setting corresponding angle φ_i should be to minimize the function of one variable

$$f(\varphi_i) = \frac{1}{2}\left(\gamma_{xi} - x_i\right)^2 + \frac{1}{2}\left(\gamma_{yi} - y_i\right)^2$$

where

$$\gamma_{xi} = \frac{\psi_{xi}}{\kappa_i}\,, \qquad \gamma_{yi} = \frac{\psi_{xi}}{\kappa_i}\,,$$

using the brief notation

$$\psi_{xi} = \tilde{x}_0 + \rho\Phi_i \cos\varphi_i\,, \qquad \psi_{yi} = \tilde{y}_0 + \rho\Phi_i \sin\varphi_i\,,$$

$$\kappa_i = 1 - \rho\Phi_i(a\cos\varphi_i + b\sin\varphi_i)$$

with Φ_i substituting $\Phi(a,b,\varphi_i)$. This can be done numerically, using the standard Newton method: if the prime symbol denotes the the derivative by φ_i then we receive the iterative algorithm

$$\varphi_i \leftarrow \varphi_i - f'(\varphi_i)/f''(\varphi_i) \tag{5}$$

where

$$\begin{aligned} f'(\varphi_i) &= (\gamma_{xi} - x_i)\gamma'_{xi} + (\gamma_{yi} - y_i)\gamma'_{yi}\,,\\ f''(\varphi_i) &= (\gamma_{xi} - x_i)\gamma''_{xi} + (\gamma_{yi} - y_i)\gamma''_{yi} + \gamma'^2_{xi} + \gamma'^2_{yi}\,; \end{aligned}$$

the (rather long) evaluation formulae for the first and second order derivatives of γ_{xi} and γ_{yi} can be generated e. g. with help of the MATLAB toolbox "symbolic" directly to the MATLAB program code with the following result (presented here after small formal modifications):

$$\begin{aligned} \gamma'_{xi} &= \frac{\psi'_{xi}\kappa_i - \psi_{xi}\kappa'_i}{\kappa_i^2}\,,\\ \gamma'_{yi} &= \frac{\psi'_{yi}\kappa_i - \psi_{yi}\kappa'_i}{\kappa_i^2}\,,\\ \gamma''_{xi} &= \frac{\psi''_{xi}\kappa_i^2 + \psi_{xi}(2\kappa'^2_i - \kappa_i\kappa''_i) + 2\psi_{xi}\kappa'^2_i}{\kappa_i^4}\,,\\ \gamma''_{yi} &= \frac{\psi''_{yi}\kappa_i^2 + \psi_{yi}(2\kappa'^2_i - \kappa_i\kappa''_i) + 2\psi_{yi}\kappa'^2_i}{\kappa_i^4}\,; \end{aligned}$$

here the first and second derivatives of Φ_i

$$\begin{aligned} \Phi'_i &= \Phi_i^3(a\cos\varphi_i + b\sin\varphi_i)(a\sin\varphi_i - b\cos\varphi_i)\,,\\ \Phi''_i &= \Phi_i^3\Big((a\cos\varphi_i + b\sin\varphi_i)^2 - (a\sin\varphi_i - b\cos\varphi_i)^2\Big)\\ &\quad - 3\Phi_i^5(a\cos\varphi_i + b\sin\varphi_i)^2(a\sin\varphi_i + b\cos\varphi_i)^2 \end{aligned}$$

and of other auxiliary functions

$$\begin{aligned} \psi'_{xi} &= \rho(-\Phi_i\sin\varphi_i + \Phi'_i\cos\varphi_i)\,,\\ \psi'_{yi} &= \rho(\Phi_i\cos\varphi_i + \Phi'_i\sin\varphi_i)\,,\\ \kappa'_i &= -\rho\left(\Phi'(a\cos\varphi + b\sin\varphi) + \Phi_i(-a\sin\varphi + b\cos\varphi)\right)\,,\\ \psi''_{xi} &= \rho(-\Phi_i\cos\varphi_i - 2\Phi'_i\sin\varphi_i + \Phi''_i\cos\varphi_i)\,,\\ \psi''_{yi} &= \rho(-\Phi_i\sin\varphi_i + 2\Phi'_i\cos\varphi_i + \Phi''_i\sin\varphi_i)\,,\\ \kappa''_i &= -\rho\left(\Phi''_i(-a\cos\varphi + b\sin\varphi) + 2\Phi'_i(-a\sin\varphi + b\cos\varphi) - \Phi_i(a\cos\varphi_i + b\sin\varphi_i)\right) \end{aligned}$$

are needed. The algorithm (5) must be applied n-times (for all pre-processed points), but all calculations can be parallelized (which can be also MATLAB-supported).

Unfortunately, it is not realistic to know 5 parameters a, b, $\widetilde{x}_0$, $\widetilde{y}_0$ and ρ in advance. Thus we have to minimize the function of 5 variables

$$F(a,b,\widetilde{x}_0,\widetilde{y}_0,\rho) = \frac{1}{2}\sum_{i=1}^{n}(\gamma_{xi}-x_i)^2 + \frac{1}{2}\sum_{i=1}^{n}(\gamma_{yi}-y_i)^2$$

where γ_{xi} and γ_{yi} (introduced above) are considered as functions of a, b, $\widetilde{x}_0$, $\widetilde{y}_0$ and ρ now. We can apply the Newton iterative algorithm again (for a function of 5 variables, but only once) in the form

$$\begin{bmatrix} a \\ b \\ \widetilde{x}_0 \\ \widetilde{y}_0 \\ \rho \end{bmatrix} \leftarrow \begin{bmatrix} a \\ b \\ \widetilde{x}_0 \\ \widetilde{y}_0 \\ \rho \end{bmatrix} - M^{-1} \begin{bmatrix} \partial F/\partial a \\ \partial F/\partial b \\ \partial F/\partial \widetilde{x}_0 \\ \partial F/\partial \widetilde{y}_0 \\ \partial F/\partial \rho \end{bmatrix} \tag{6}$$

with

$$M = \begin{bmatrix} \partial^2 F/\partial a^2 & \partial^2 F/\partial a\partial b & \partial^2 F/\partial a\partial \widetilde{x}_0 & \partial^2 F/\partial a\partial \widetilde{y}_0 & \partial^2 F/\partial a\partial \rho \\ \partial^2 F/\partial a\partial b & \partial^2 F/\partial b^2 & \partial^2 F/\partial b\partial \widetilde{x}_0 & \partial^2 F/\partial b\partial \widetilde{y}_0 & \partial^2 F/\partial b\partial \rho \\ \partial^2 F/\partial a\partial \widetilde{x}_0 & \partial^2 F/\partial b\partial \widetilde{x}_0 & \partial^2 F/\partial \widetilde{x}_0^2 & \partial^2 F/\partial \widetilde{x}_0\partial \widetilde{y}_0 & \partial^2 F/\partial \widetilde{x}_0\partial \rho \\ \partial^2 F/\partial a\partial \widetilde{y}_0 & \partial^2 F/\partial b\partial \widetilde{y}_0 & \partial^2 F/\partial \widetilde{x}_0\partial \widetilde{y}_0 & \partial^2 F/\partial \widetilde{y}_0^2 & \partial^2 F/\partial \widetilde{y}_0\partial \rho \\ \partial^2 F/\partial a\partial \rho & \partial^2 F/\partial b\partial \rho & \partial^2 F/\partial \widetilde{x}_0\partial \rho & \partial^2 F/\partial \widetilde{y}_0\partial \rho & \partial^2 F/\partial \rho^2 \end{bmatrix}$$

(F and its partial derivatives depend still on 5 variables, but this is not emphasized explicitly here). To get some partial derivatives of F is simple, namely those by $\widetilde{x}_0$ and $\widetilde{y}_0$; even $\partial^2 F/\partial \widetilde{x}_0 \partial \widetilde{y}_0 = 0$. Their complete set can be evaluated using the scheme (similar to that in the minimization of f)

$$\begin{aligned} \partial F/\partial u &= \sum_{i=1}^{n}(\gamma_{xi}-x_i)\partial\gamma_{xi}/\partial u + \sum_{i=1}^{n}(\gamma_{yi}-y_i)\partial\gamma_{yi}/\partial u\,, \\ \partial^2 F/\partial u\partial v &= \sum_{i=1}^{n}(\gamma_{xi}-x_i)\partial^2\gamma_{xi}/\partial u\partial v + \sum_{i=1}^{n}(\gamma_{yi}-y_i)\partial^2\gamma_{yi}/\partial u\partial v \\ &\quad + \sum_{i=1}^{n}\partial\gamma_{xi}/\partial u\,\partial\gamma_{xi}/\partial v + \sum_{i=1}^{n}\partial\gamma_{yi}/\partial u\,\partial\gamma_{yi}/\partial v \end{aligned}$$

where $u, v \in \{a, b, \widetilde{x}_0, \widetilde{y}_0, \rho\}$ and the formal differentiation of γ_{xi} and γ_{yi} is allowed to be MATLAB-supported again. We obtain

$$\begin{aligned} \partial\gamma_{xi}/\partial u &= \frac{\partial\psi_{xi}/\partial u\,\kappa_i - \psi_{xi}\partial\kappa_i/\partial u}{\kappa_i^2}\,, \\ \partial\gamma_{yi}/\partial u &= \frac{\partial\psi_{yi}/\partial u\,\kappa_i - \psi_{yi}\partial\kappa_i/\partial u}{\kappa_i^2}\,, \end{aligned}$$

$$\begin{aligned}
\partial^2\gamma_{xi}/\partial u\partial v = {} & \frac{\partial^2\psi_{xi}/\partial u\partial v\,\kappa_i^2 + \psi_{xi}(2\partial\kappa_i/\partial u\,\partial\kappa_i/\partial v - \kappa_i\partial^2\kappa_i/\partial u\partial v)}{\kappa_i^4} \\
& + \frac{2\psi_{yi}(\partial\kappa_i/\partial uv)^2}{\kappa_i^4}\,, \\
\partial^2\gamma_{yi}/\partial u\partial v = {} & \frac{\partial^2\psi_{yi}/\partial u\partial v\,\kappa_i^2 + \psi_{yi}(2\partial\kappa_i/\partial u\,\partial\kappa_i/\partial v - \kappa_i\partial^2\kappa_i/\partial u\partial v)}{\kappa_i^4} \\
& + \frac{2\psi_{yi}(\partial\kappa_i/\partial uv)^2}{\kappa_i^4}
\end{aligned}$$

with the first derivatives

$$\begin{array}{ll}
\partial\psi_{xi}/\partial a \;= \rho\partial\Phi_i/\partial a\cos\varphi_i\,, & \partial\psi_{yi}/\partial a \;= \rho\partial\Phi_i/\partial a\sin\varphi_i\,, \\
\partial\psi_{xi}/\partial\widetilde{y}_0 = \partial\psi_{xi}/\partial\widetilde{x}_0 - 1 = 0\,, & \partial\psi_{yi}/\partial\widetilde{x}_0 = \partial\psi_{yi}/\partial\widetilde{y}_0 - 1 = 0\,, \\
\partial\psi_{xi}/\partial\rho \;= \Phi_i\cos\varphi_i\,, & \partial\psi_{yi}/\partial\rho \;= \Phi_i\sin\varphi_i\,,
\end{array}$$

$$\begin{aligned}
\partial\kappa_i/\partial a &= -\rho\left(\partial\Phi_i/\partial a(a\cos\varphi_i + b\sin\varphi_i) - \Phi_i\cos\varphi_i\right)\,, \\
\partial\kappa_i/\partial b &= -\rho\left(\partial\Phi_i/\partial b(a\cos\varphi_i + b\sin\varphi_i) - \Phi_i\sin\varphi_i\right)\,, \\
\partial\kappa_i/\partial\rho &= \Phi_i(a\cos\varphi_i + b\sin\varphi_i)\,, \qquad \partial\kappa_i/\partial\widetilde{x}_0 = \partial\kappa_i/\partial\widetilde{y}_0 = 0\,, \\
\partial\Phi_i/\partial a &= -\Phi_i^3\cos\varphi_i\,, \qquad \partial\Phi_i/\partial b = -\Phi_i^3\sin\varphi_i
\end{aligned}$$

and with the non-zero second ones for arbitrary $u, v \in \{a, b\}$

$$\begin{array}{ll}
\partial^2\psi_{xi}/\partial u\partial v = \rho\partial^2\Phi_i/\partial u\partial v\cos\varphi_i\,, & \partial^2\psi_{yi}/\partial u\partial v = \rho\partial^2\Phi_i/\partial u\partial v\sin\varphi_i\,, \\
\partial^2\psi_{xi}/\partial u\partial\rho = \partial\Phi_i/\partial\rho\cos\varphi_i\,, & \partial^2\psi_{yi}/\partial u\partial\rho = \partial\Phi_i/\partial\rho\sin\varphi_i\,,
\end{array}$$

$$\begin{aligned}
\partial^2\kappa_i/\partial a^2 &= -\rho\partial^2\Phi_i/\partial a^2(a\cos\varphi_i + b\sin\varphi_i) - 2\rho\partial\Phi_i/\partial a\cos\varphi_i\,, \\
\partial^2\kappa_i/\partial b^2 &= -\rho\partial^2\Phi_i/\partial a\partial b(a\cos\varphi_i + b\sin\varphi_i) - 2\rho\partial\Phi_i/\partial b\sin\varphi_i\,, \\
\partial^2\kappa_i/\partial a\partial b &= -\rho\partial^2\Phi_i/\partial a\partial b(a\cos\varphi_i + b\sin\varphi_i) \\
&\quad -\rho\left(\partial\Phi_i/\partial a\sin\varphi_i + \partial\Phi_i/\partial b\cos\varphi_i\right)\,, \\
\partial^2\kappa_i/\partial a\partial\rho &= \partial\Phi_i/\partial a(a\cos\varphi_i + b\sin\varphi_i) + \Phi\cos\varphi_i\,, \\
\partial^2\kappa_i/\partial b\partial\rho &= \partial\Phi_i/\partial b(a\cos\varphi_i + b\sin\varphi_i) + \Phi\sin\varphi_i\,,
\end{aligned}$$

$$\partial^2\Phi_i/\partial a^2 = 3\Phi_i^5\cos^2\varphi_i\,, \quad \partial^2\Phi_i/\partial b^2 = 3\Phi_i^5\sin^2\varphi_i\,, \quad \partial^2\Phi_i/\partial a\partial b = \frac{3}{2}\Phi_i^5\sin(2\varphi_i)\,.$$

In practical calculation the evaluation of M^{-1} in (6) can be avoided e. g. by the Gauss elimination scheme.

In most technical applications with repeated reconstructions of ω (a series of images with time-dependent positions of ω is available) the algorithm (6) can be simplified dramatically. Namely if we are sure that ω is still moving in ϖ then a and b can be set in the first calculation and in subsequent calculations F depends only on 3 variables. The importance of the quality of the first estimate of all 5 parameters may be more clear from an intuitive geometrical consideration (which could be verified by proper analytical computations or using constructive arguments from [22], p. 99): a cone given by P and $\widetilde{\omega}$ has 2 systems of circular cuts by parallel planes (that coincide only for a rotational cone), thus 4 circles with the given radius r belong to such system, among them 2 violating the orientation of the projection from P, but remaining 2 to be distinguished.

3 Data smoothing and contour detection

The previous section needs n data couples (x_i, y_i), thus the quality of the result is conditioned by the quality of their recognition from the finite gray-scale map, produced by our camera. Because of the presence of noise some smoothing is needed, then the points approximating an ellipse can be detected. We shall see that we can apply certain least squares access again.

Let $w(x, y)$ be a smooth map of gray intensities (its smoothness can be analyzed more precisely in terms of Lebesgue, Sobolev and Bochner spaces of integrable functions – see [32], p. 46), received from our gray-scale data $g(x, y)$, assigned to particular rectangular pixels; all such pixels create a rectangular domain Ω. Its subpixel precision can be based on the following considerations. By [32], p. 45, $w(x, y)$ is analyzed numerically as a solution $w(x, y, \tau)$ of one partial differential equation of evolution in time $\tau > 0$

$$\dot{w} = |\nabla w| \operatorname{div}\left(g \frac{\nabla w}{|\nabla w|}\right) \tag{7}$$

with $\dot{w}$ defined as $\partial w/\partial\tau$, although its transparent derivation from some integral version of the least squares access, similar to that from classical diffusion problems in computer vision, sketched in [32], p. 41, is not available. The evolution is starting at zero time $t = 0$ from some prescribed initial estimate $w(x, y, 0)$; then $w(x, y)$ is taken as such "stationary status" $w(x, y, \tau)$ when all substantial changes of w in τ vanish. This approach coincides with the study of the parabolic problem of mean curvature motion by [10], p. 17; some its modifications and generalizations are mentioned in [5], p. 267.

An unpleasant imperfection of (7) is that $g(x, y)$ is a simple function, in practice discontinuous on most pixel edges. Thus it is needed, following [32], p. 45, to substitute such g by certain function $g(|\nabla S_\varepsilon * g|)$ (independent of w); here $*$ denotes the convolution and S_ε is some smoothing function with a positive parameter ε, set by experience. In [32], p. 35, the Gaussian smoothing $S_\varepsilon(x, y) = \delta_\varepsilon(\sqrt{x^2 + y^2})$ with

$$\delta_\varepsilon(s) = \frac{1}{2\pi\varepsilon^2} \exp\left(-\frac{s^2}{2\varepsilon^2}\right) \tag{8}$$

is recommended and the "diffusivity" function g is defined as

$$g(s) = 1 - \exp\left(-\frac{\upsilon\lambda^4}{s^4}\right)$$

for $s > 0$, otherwise (for $s = 0$) to $g(s) = 1$; λ is some positive contrast characteristic (e. g. in [32], p. 46, $\varepsilon = 1$ and $\lambda = 5$). The remaining positive characteristic ξ must be received from some additional requirement, e. g. that the flux $sg(s)$ is increasing for $s < \lambda$ and decreasing for $s > \lambda$; the elementary differential calculus gives then the condition $\exp(\upsilon) = 1 + 8\upsilon$, satisfied for $\upsilon \approx 3.31487736178606$.

The algorithm of [32] searches for the best contours of certain domain Θ (approximating $\widetilde{\omega}$ in our notation) in Ω from its exterior, consisting of some "nearly-black" or "nearly-white" noise of some average intensity c_2 between 0 and 1; moreover, a sufficient number of coordinate couples (x_i, y_i) of $\partial\Theta$ must be found a posteriori. However, we know that in our problem also the interior of Θ should consist of some similar noise, only of another intensity c_1 between 0 and 1. To incorporate this information, let us follow the main ideas of [5]. The crucial problem is always to localize $\partial\Theta$; for a sufficient number of its points all algorithms of the previous section are applicable.

Let μ be the standard 2-dimensional Lebesgue measure on Ω, thus $\mathrm{d}\mu = \mathrm{d}x\,\mathrm{d}y$. Let λ and $\widehat{\lambda}$ be 2 prescribed positive constants. Let us try to minimize the real functional

$$\begin{aligned} G(c_1, c_2, \phi) = {} & \lambda \int_\Omega \delta(\phi)|\nabla\phi|\,\mathrm{d}\mu \\ & + \widehat{\lambda} \int_\Omega (g - c_1)^2 H(\phi)\,\mathrm{d}\mu + \widehat{\lambda} \int_\Omega (g - c_2)^2 (1 - H(\phi))\,\mathrm{d}\mu \end{aligned} \tag{9}$$

where H denotes the Heaviside function and δ the Dirac measure (which is the derivative of H is sense of distributions); the third variable $\phi(x, y)$ is certain "level-set" function with zero values on $\partial\Theta$, positive values on Θ, otherwise with negative values on Ω, Lebesgue integrable including its gradient on Ω. Evidently the problem of localization of Θ coincides with the analysis of zero points of ϕ. For a fixed ϕ clearly G can be identified with a real function of 2 variables only, whose differentiation by c_1 and c_2 gives

$$\begin{aligned} \partial G/\partial c_1(c_1, c_2) &= 2\widehat{\lambda} \int_\Omega (c_1 - g) H(\phi)\,\mathrm{d}\mu\,, \\ \partial G/\partial c_2(c_1, c_2) &= 2\widehat{\lambda} \int_\Omega (c_2 - g)(1 - H(\phi))\,\mathrm{d}\mu \end{aligned}$$

and its minimum is attached for

$$\begin{aligned} c_1 &= \left(\int_\Omega H(\phi)\,\mathrm{d}\mu \right)^{-1} \int_\Omega g H(\phi)\,\mathrm{d}\mu\,, \\ c_2 &= \left(\int_\Omega (1 - H(\phi))\,\mathrm{d}\mu \right)^{-1} \int_\Omega g(1 - H(\phi))\,\mathrm{d}\mu\,. \end{aligned} \tag{10}$$

This result has a simple geometric interpretation: c_1 and c_2 are the averages of g on Θ and $\Omega \setminus \Theta \setminus \partial\Theta$. Nevertheless, in general we must respect that both c_1 and c_2 depend on ϕ; later in practical calculations some formula for numerical integration (e. g. the rectangular rule) cannot be avoided.

The differentiation of G with respect to ϕ is more delicate For both fixed c_1 and c_2 it is a real functional on a set of all admissible ϕ, clearly with $\delta(\phi)$ unchanged. Therefore its Gâteaux differential (in sense of [14], p. 89) with any variation $\widetilde{\phi}$, Lebesgue integrable and bounded including its gradient on Ω, is

$$\begin{aligned} \mathrm{D}G(\phi,\widetilde{\phi}) &= \lambda \frac{\mathrm{d}}{\mathrm{d}t}\left(\int_\Omega \delta(\phi)\sqrt{(\nabla\phi + t\nabla\widetilde{\phi})\cdot(\nabla\phi + t\nabla\widetilde{\phi})}\,\mathrm{d}\mu\right)_{t=0} \\ &+ \widehat{\lambda}\frac{\mathrm{d}}{\mathrm{d}t}\left(\int_\Omega (g-c_1)^2 H(\phi + t\widetilde{\phi})\,\mathrm{d}\mu\right)_{t=0} \\ &+ \widehat{\lambda}\frac{\mathrm{d}}{\mathrm{d}t}\left(\int_\Omega (g-c_2)^2 (1 - H(\phi + t\widetilde{\phi}))\,\mathrm{d}\mu\right)_{t=0} \\ &= \lambda \int_\Omega \delta(\phi)\frac{\nabla\widetilde{\phi}\cdot\nabla\phi}{|\nabla\phi|}\,\mathrm{d}\mu \\ &+ \widehat{\lambda}\int_\Omega \widetilde{\phi}\,\delta(\phi)(g-c_1)^2\,\mathrm{d}\mu - \widehat{\lambda}\int_\Omega \widetilde{\phi}\,\delta(\phi)(g-c_2)^2\,\mathrm{d}\mu\,. \end{aligned}$$

Let us assume that ϕ satisfies the boundary condition

$$\delta(\phi)\frac{\nabla\phi\cdot\nu}{|\nabla\phi|} = 0 \qquad (11)$$

on $\partial\Omega$; here ν is the unit (e. g. exterior) normal to $\partial\Omega$. Using the Green-Ostrogradskiĭ theorem, we obtain the corresponding Euler-Lagrange differential equation

$$\delta(\phi)\left(\lambda\,\mathrm{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - \widehat{\lambda}(g-c_1)^2 + \widehat{\lambda}(g-c_2)^2\right) = 0 \qquad (12)$$

on Ω. For numerical calculation it is necessary to replace δ by some "regularized Dirac function" δ_ε, e. g. by that from (8) or by that recommended in [5], p. 270,

$$\delta_\varepsilon(s) = \frac{\varepsilon}{\pi(\varepsilon^2 + s^2)}\,. \qquad (13)$$

The corresponding "regularized Heaviside function" H_ε, substituting H, follows here easily by the integration with respect to s in the form

$$H_\varepsilon(s) = \frac{1}{2}\left(1 + \frac{2}{\pi}\arctan\left(\frac{s}{\varepsilon}\right)\right)\;;$$

in case of (8) not in a simple analytical form. The existence and convergence results for the minimization of G, consequently also for the solution of (12) and its modifications, follow from the theory of Mumford-Shah segmentation problems, presented in [8]. An alternative proof is sketched in [5], p. 269; it is based on the equivalent form of (9)

$$G(\chi) = \lambda\int_\Omega |\nabla\chi|\,\mathrm{d}\mu + \widehat{\lambda}\int_\Omega (g - c_1(\chi))^2\,\mathrm{d}\mu - \widehat{\lambda}\int_\Omega (g - c_2(\chi))^2\,\mathrm{d}\mu$$

with an arbitrary characteristic function $\chi(x,y)$ of Θ (i. e. a function with values 0 or 1 almost everywhere with respect to μ, in our notation identical with $H(\phi(x,y)))$ of a two-dimensional set with finite perimeter and on some classical arguments (as lower-semicontinuity and its consequences, cf. [14], p. 96) of calculus of variations.

To solve (12) directly, even for fixed c_1 and c_2 and a regularized δ, is not quite easy. However, its reformulation

$$\phi = \phi + \tau\delta_\varepsilon(\phi)\left(\lambda\,\mathrm{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - \widehat{\lambda}(g-c_1)^2 + \widehat{\lambda}(g-c_2)^2\right)$$

for some positive τ motivates the construction of ϕ as a limit of a sequence of iterations $\phi_0, \phi_1, \phi_2, \ldots$ by the formula

$$\phi_{k+1} = \phi_k + \tau\delta_\varepsilon(\phi_k)\left(\lambda\,\mathrm{div}\left(\frac{\nabla\phi_{k+1}}{|\nabla\phi_k|}\right) - (g-c_{1k})^2 + (g-c_{2k})^2\right) \tag{14}$$

with $k \in \{1,2,\ldots\}$ for some initial estimate ϕ_0; the indices k in c_1 and c_2 cannot be removed because we know that c_1 and c_2 depend on ϕ_k. Let us remark that in the nomenclature of evolution equations this is the Euler semi-implicit scheme for an initial problem, corresponding to the "time-continuous" equation

$$\dot{\phi} = \delta_\varepsilon(\phi)\left(\lambda\,\mathrm{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - (g-c_1)^2 + (g-c_2)^2\right)\;;$$

thus we have obtained certain kind of nonlinear diffusion with expected final stationary status again.

After the standard finite difference discretization (14) generates a system of linear algebraic equations with a sparse system matrix in each iteration step, whose numerical solution is not not expensive. The decomposition of Ω is reasonable to be done exactly into the system of square pixels: if h is the length of their edges, we are able to approximate the first derivatives of ϕ_k for pixel centers with coordinates (x,y), using the central differences as

$$\partial\phi_k/\partial x(x,y) \approx \frac{\phi_k(x+h,y)-\phi_k(x-h,y)}{2h}\,,$$
$$\partial\phi_k/\partial y(x,y) \approx \frac{\phi_k(x,y+h)-\phi_k(x,y-h)}{2h}\,.$$

Later we shall need also to approximate the second derivatives as

$$\partial^2\phi_k/\partial x^2(x,y) \approx \frac{\phi_k(x+h,y)-2\phi_k(x,y)+\phi_k(x-h,y)}{h^2}\,,$$
$$\partial^2\phi_k/\partial y^2(x,y) \approx \frac{\phi_k(x,y+h)-2\phi_k(x,y)+\phi_k(x,y-h)}{h^2}\,,$$

$$\partial^2\phi_k/\partial x\partial y(x,y) \approx \frac{\phi_k(x+h,y+h)-\phi_k(x-h,y+h)}{2h^2} + \frac{\phi_k(x-h,y-h)+\phi_k(x+h,y-h)}{2h^2}\,;$$

much more general formulae can be found in [31], p. 249, where the proper accuracy analysis of various finite difference approximations is done. To force (11), some artificial nodes (outside Ω) are needed: e. g. if a segment of $\partial\Omega$ is characterized by $x = 0$ and arbitrary y (from some real interval, in practice for discrete values of y) then

$$\partial\phi_k/\partial x(0,y) \approx \frac{\phi_k(h/2,y)-\phi_k(-h/2,y)}{h} = 0\,;$$

this is a very special case of the formula from [31], p. 264.

Let us believe that we have found a "stationary" $\phi(x, y)$. To generate all couples (x_i, y_i) from the previous section, we need to calculate the zero points of ϕ. Since the minimization of f is rather difficult, it would be useful to avoid it at all. For some fixed parameters a, b, $\widetilde{x}_0$, $\widetilde{y}_0$ and ρ from the preceding section let us choose $\varphi_i = 2\pi i/n$ for $i \in \{1, \ldots, n\}$ and calculate all $x = \widetilde{x}_i$ and $y = \widetilde{y}_i$ for $\varphi = \varphi_i$ from (4); in this way we get the coordinates of certain points S_i. For such arbitrary i let us consider 2 real functions

$$\begin{aligned}
p(x_i,y_i) &= \phi(x_i,y_i) - \sigma\left((y_i-\widetilde{y}_0)(\widetilde{x}_i-\widetilde{x}_0)-(x_i-\widetilde{x}_0)(\widetilde{y}_i-\widetilde{y}_0)\right)(\widetilde{y}_i-\widetilde{y}_0)\,,\\
q(x_i,y_i) &= \phi(x_i,y_i) + \sigma\left((y_i-\widetilde{y}_0)(\widetilde{x}_i-\widetilde{x}_0)-(x_i-\widetilde{x}_0)(\widetilde{y}_i-\widetilde{y}_0)\right)(\widetilde{x}_i-\widetilde{x}_0)
\end{aligned}$$

where σ is some positive constant; their partial derivatives with respect to x_i are

$$\begin{aligned}
p_x(x_i,y_i) &= \partial\phi/\partial x(x_i,y_i) + \sigma(\widetilde{y}_i-\widetilde{y}_0)^2\,,\\
q_x(x_i,y_i) &= \partial\phi/\partial x(x_i,y_i) - \sigma(\widetilde{x}_i-\widetilde{x}_0)(\widetilde{y}_i-\widetilde{y}_0)
\end{aligned}$$

and with respect to y_i similarly

$$\begin{aligned}
p_y(x_i,y_i) &= \partial\phi/\partial y(x_i,y_i) - \sigma(\widetilde{y}_i-\widetilde{y}_0)(\widetilde{y}_i-\widetilde{y}_0)\,,\\
q_y(x_i,y_i) &= \partial\phi/\partial y(x_i,y_i) + \sigma(\widetilde{x}_i-\widetilde{x}_0)^2\,.
\end{aligned}$$

Let us put $p(x_i, y_i) = q(x_i, y_i) = 0$. Then (independently of σ) the matrix equation

$$\begin{bmatrix} \widetilde{x}_i-\widetilde{x}_0 & \widetilde{y}_i-\widetilde{y}_0 \\ \widetilde{x}_i-\widetilde{x}_0 & \widetilde{y}_0-\widetilde{y}_i \end{bmatrix} \begin{bmatrix} \phi(x_i,y_i) \\ (y_i-\widetilde{y}_0)(\widetilde{x}_i-\widetilde{x}_0)-(x_i-\widetilde{x}_0)(\widetilde{y}_i-\widetilde{y}_0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

is valid. Since both $\widetilde{x}_i = \widetilde{x}_0$ and $\widetilde{y}_i = \widetilde{y}_0$ cannot be true simultaneously, the existence of a unique trivial solution of this system (by the classical Frobenius theorem) yields $\phi(x_i, y_i) = 0$ and $(y_i - \widetilde{y}_0)(\widetilde{x}_i - \widetilde{x}_0) = (x_i - \widetilde{x}_0)(\widetilde{y}_i - \widetilde{y}_0)$. This has a simple geometric interpretation: (x_i, y_i)

determines the point of intersection of $\partial\Theta$ with the line connecting $\widetilde{S}$ and S_i. Consequently the Newton algorithm

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} \leftarrow \begin{bmatrix} x_i \\ y_i \end{bmatrix} - \begin{bmatrix} p_x(x_i,y_i) & p_y(x_i,y_i) \\ q_x(x_i,y_i) & q_y(x_i,y_i) \end{bmatrix}^{-1} \begin{bmatrix} p(x_i,y_i) \\ q(x_i,y_i) \end{bmatrix} \tag{15}$$

is applicable again; the first estimate can be e. g. $(x_i, y_i) \approx (\widetilde{x}_i, \widetilde{y}_i)$. Nevertheless, one difficulty occurs: all values of ϕ have been computed only for discrete nodes, identical with pixel centers, thus its first and second partial derivatives should be (maybe slowly) interpolated or taken from nearest nodes (which could reduce the rate of convergence); in much more general context such problems are studied in [25], p. 117.

Let (10*) refer to the "regularized" version of (10) with H_ε instead of H and also formally with ϕ_k instead of ϕ. Now we are ready to implement the following software algorithm:

I. initialize ϕ_0, set $k \leftarrow 0$,
II. compute $c_1(\phi_k)$ and $c_2(\phi_k)$ from (10*),
III. solve (14) to receive ϕ_{k+1},
IV. check whether ϕ_{k+1} is sufficiently close to ϕ_k; if not, set $k \leftarrow k+1$ and go back to II., else accept ϕ_k as final ϕ,
V. initialize old $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$,
VI. prepare $(x_1, y_1), \dots, (x_n, y_n)$ by (15),
VII. compute new $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$ from (6),
VIII. check whether new and old $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$ are nearly the same; if not, take new values as old ones and go back to VI.

The output should be the final estimates of $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$ at a subpixel quality level, sufficient for the complete reconstruction of ω in ϖ. Both initial settings I. and V. must be done carefully. In case of V. the argumentation is evident from the concluding geometrical consideration in the previous section. In case of I. namely $\phi_0 = 0$ everywhere is prohibited; moreover, the convergence of the algorithm to some local (non-global) minimum of G (which is usually non-convex) can occur, depending both on a "bad" estimate and on a "bad" regularization of type (8) or (13); [5], p. 270, presents even another goniometric regularization with a strong tendency to compute local minimizers.

The above sketched algorithm is able to be modified and generalized in several directions. The idea to normalize $|\nabla\phi_k|$ to 1, discussed in [5], p. 272, comes from [29]; this prevents the level set functions ϕ_k to become to flat which can be also avoided by some optional rescaling or reinitialization, in our algorithm as an auxiliary step between VII. and VIII. In [3] the information on an elliptical shape of $\partial\Theta$ is included in the form of certain penalization functional, added to that analogous to 9. The variational formulations of [11] lead to the study of finite element approximations; moreover, [11] includes the extensive overview of the state of art in the theory of Mumford-Shah functionals, treated as shape optimization problems and solved numerically using level-set techniques.

4 Elliptical shape information in contour detection

The algorithm from the previous section consists evidently from two large blocks: the first one (from step I. to step IV.) searches for $\widetilde{\omega}$, the second one (from step V. to step VIII.) assigns the location of ω to any given $\widetilde{\omega}$. Unlike the second block, the first block ignores the information that ω is an ellipse; thus it is seems to be natural to integrate this information into the first block and simplify the second one. Although this is not easy, we shall show that such access is possible, applying several results from two preceding sections.

Let us start with the basic idea of [3]: to select always such Θ that $\partial\Theta$ is an ellipse. We know that $\widetilde{\omega} \approx \partial\Theta$ is some image of a circle in central projection, thus $\widetilde{\omega}$ must be an ellipse which justifies our assumption. In the simplest (non-realistic) case $\widetilde{\omega}$ could be a unit circle, thus it is reasonable to introduce a function

$$\bar{\phi}(x,y) = x^2 + y^2 - 1\,,$$

negative inside $\widetilde{\Omega}$, positive outside $\widetilde{\Omega}$ and zero-valued on $\widetilde{\Omega}$ and $|\phi(x,y) - \widehat{\phi}(x,y)|^2$ characterizes the imperfectness of the relation $\widetilde{\omega} \approx \partial\Theta$. In the realistic case we must take

$$\bar{\phi}(x,y) = \bar{x}^2 + \bar{y}^2 - 1 \tag{16}$$

with $(\bar{x},\bar{y})$ coming from some affine transformation

$$\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix} = \begin{bmatrix} \xi \\ \eta \end{bmatrix} + \begin{bmatrix} \cos\vartheta & \sin\vartheta \\ -\sin\vartheta & \cos\vartheta \end{bmatrix} \begin{bmatrix} \alpha x \\ \beta y \end{bmatrix};$$

our notation from Introduction is used here, $\alpha = \bar{\xi}^{-1/2}$ and $\beta = \bar{\eta}^{-1/2}$ for brevity,

In [3] only a “rigid transformation” with a priori known $\alpha = \beta$ is studied; in our more general case 5 new parameters α, β, ξ, η and ϑ occur. This is the same number of parameters as that for the reconstruction of ω from $\widetilde{\omega}$, thus it could be seemingly useful to reformulate (16) with $\widetilde{x}_0$, $\widetilde{y}_0$, a, b and ρ instead of α, β, ξ, η and ϑ. Such reformulation would remove the second block of the algorithm at all, but this leads to complicated evaluations, including e. g. many Newton iterations of type (5), avoided even in the previous section. This could be expected also by geometrical considerations: $\widetilde{\omega}$ (in the central projection) is not affine to ω, only collinear. However, the determination of parameters $\widetilde{x}_0$, $\widetilde{y}_0$, a, b and ρ if will be much easier here – theoretically (for the quite exact values of x_i and y_i from (15)) $n = 5$ is always sufficient.

We can easily see that in (16)

$$\bar{x} = \alpha x\cos\vartheta + \beta y\sin\vartheta + \xi\,, \qquad \bar{y} = -\alpha x\sin\vartheta + \beta y\cos\vartheta + \eta\,.$$

We must still consider $\widehat{\phi}$, $\bar{x}$ and $\bar{y}$ as functions of x and y; nevertheless, we shall differentiate them (for any fixed x and y) with respect to α, β, ξ, η and ϑ. We shall need especially the first derivatives of $\bar{x}$ and $\bar{y}$ with respect to ϑ

$$\bar{x}' = -\alpha x \sin\vartheta + \beta y \cos\vartheta\,, \qquad \bar{y}' = -\alpha x \cos\vartheta - \beta y \sin\vartheta$$

and the second ones

$$\bar{x}'' = -\alpha x \cos\vartheta - \beta y \sin\vartheta\,, \qquad \bar{y}'' = \alpha x \sin\vartheta - \beta y \cos\vartheta\,.$$

Then the first derivatives of $\bar{\phi}$ with respect to particular parameters are

$$\begin{aligned}
&\partial\bar{\phi}/\partial\alpha = \bar{x}x\cos\vartheta - \bar{y}x\sin\vartheta\,, \qquad \partial\bar{\phi}/\partial\beta = \bar{x}y\sin\vartheta + \bar{y}y\cos\vartheta\,,\\
&\partial\bar{\phi}/\partial\xi = \bar{x}\,, \qquad \partial\bar{\phi}/\partial\eta = \bar{y}\,, \qquad \partial\bar{\phi}/\partial\vartheta = \bar{x}\bar{x}' + \bar{y}\bar{y}'\,,
\end{aligned}$$

the second derivatives similarly

$$\begin{aligned}
&\partial^2\bar{\phi}/\partial\alpha^2 = x^2\,, \qquad \partial^2\bar{\phi}/\partial\beta^2 = y^2\,, \qquad \partial^2\bar{\phi}/\partial\xi^2 = \partial^2\bar{\phi}/\partial\eta^2 = 1\,,\\
&\partial^2\bar{\phi}/\partial\vartheta^2 = \bar{x}\bar{x}'' + \bar{y}\bar{y}'' + \bar{x}'^2 + \bar{y}'^2\,,\\
&\partial^2\bar{\phi}/\partial\alpha\partial\xi = x\cos\vartheta\,, \qquad \partial^2\bar{\phi}/\partial\alpha\partial\eta = -x\sin\vartheta\,,\\
&\partial^2\bar{\phi}/\partial\beta\partial\xi = y\sin\vartheta\,, \qquad \partial^2\bar{\phi}/\partial\beta\partial\eta = y\cos\vartheta\,,\\
&\partial^2\bar{\phi}/\partial\alpha\partial\beta = \partial^2\bar{\phi}/\partial\xi\partial\eta = 0\,, \qquad \partial^2\bar{\phi}/\partial\xi\partial\vartheta = \bar{x}'\,, \qquad \partial^2\bar{\phi}/\partial\eta\partial\vartheta = \bar{y}'\,,\\
&\partial^2\bar{\phi}/\partial\alpha\partial\vartheta = x((\bar{x}' - \bar{y})\cos\vartheta - (\bar{y}' + \bar{x})\sin\vartheta)\,,\\
&\partial^2\bar{\phi}/\partial\beta\partial\vartheta = y((\bar{y}' + \bar{x})\cos\vartheta + (\bar{x}' - \bar{y})\sin\vartheta)\,.
\end{aligned}$$

Let us introduce one additional functional

$$\bar{G}(\phi,\alpha,\beta,\xi,\eta,\vartheta) = \bar{\lambda}\int_\Omega \bar{\phi}^2\delta(\phi)|\nabla\phi|\,\mathrm{d}\mu \tag{17}$$

with some positive constant $\bar{\lambda}$. Substituting G in (9) by $G+\bar{G}$, we can see that $\bar{G}$ returns certain penalization value, measuring the non-ellipticity of $\partial\Theta$. Keeping α, β, ξ, η and ϑ fixed, we can calculate the Gâteaux differential (in the same way as that of G)

$$\mathrm{D}\bar{G}(\phi,\tilde{\phi}) = \bar{\lambda}\int_\Omega \bar{\phi}^2\delta(\phi)\frac{\nabla\tilde{\phi}\cdot\nabla\phi}{|\nabla\phi|}\,\mathrm{d}\mu\,.$$

Keeping ϕ fixed, we have

$$\begin{aligned}
\partial\bar{G}/\partial u &= 2\bar{\lambda}\int_\Omega \bar{\phi}\,\partial\bar{\phi}/\partial u\,\delta(\phi)|\nabla\phi|\,,\\
\partial^2\bar{G}/\partial u\partial v &= 2\bar{\lambda}\int_\Omega \left(\bar{\phi}\,\partial^2\bar{\phi}/\partial u\partial v + \partial\bar{\phi}/\partial u\,\partial\bar{\phi}/\partial v\right)\delta(\phi)|\nabla\phi|
\end{aligned}$$

for any $u, v \in \{\alpha, \beta, \xi, \eta, \vartheta\}$ Thus, searching for the minimum of $\bar{G}$ (still with fixed ϕ), using some formulae for numerical integration, similarly to (6) we can calculate

$$\begin{bmatrix} \alpha \\ \beta \\ \xi \\ \eta \\ \vartheta \end{bmatrix} \leftarrow \begin{bmatrix} \alpha \\ \beta \\ \xi \\ \eta \\ \vartheta \end{bmatrix} - \bar{M}^{-1} \begin{bmatrix} \partial\bar{G}/\partial\alpha \\ \partial\bar{G}/\partial\beta \\ \partial\bar{G}/\partial\xi \\ \partial\bar{G}/\partial\eta \\ \partial\bar{G}/\partial\vartheta \end{bmatrix} \tag{18}$$

with

$$\bar{M} = \begin{bmatrix} \partial^2\bar{G}/\partial\alpha^2 & \partial^2\bar{G}/\partial\alpha\partial\beta & \partial^2\bar{G}/\partial\alpha\partial\xi & \partial^2\bar{G}/\partial\alpha\partial\eta & \partial^2\bar{G}/\partial\alpha\partial\vartheta \\ \partial^2\bar{G}/\partial\alpha\partial\beta & \partial^2\bar{G}/\partial\beta^2 & \partial^2\bar{G}/\partial\beta\partial\xi & \partial^2\bar{G}/\partial\beta\partial\eta & \partial^2\bar{G}/\partial\beta\partial\vartheta \\ \partial^2\bar{G}/\partial\alpha\partial\xi & \partial^2\bar{G}/\partial\beta\partial\xi & \partial^2\bar{G}/\partial\xi^2 & \partial^2\bar{G}/\partial\xi\partial\eta & \partial^2\bar{G}/\partial\xi\partial\vartheta \\ \partial^2\bar{G}/\partial\alpha\partial\eta & \partial^2\bar{G}/\partial\beta\partial\eta & \partial^2\bar{G}/\partial\xi\partial\eta & \partial^2\bar{G}/\partial\eta^2 & \partial^2\bar{G}/\partial\eta\partial\vartheta \\ \partial^2\bar{G}/\partial\alpha\partial\vartheta & \partial^2\bar{G}/\partial\beta\partial\vartheta & \partial^2\bar{G}/\partial\xi\partial\vartheta & \partial^2\bar{G}/\partial\eta\partial\vartheta & \partial^2\bar{G}/\partial\vartheta^2 \end{bmatrix};$$

all derivatives of $\bar{G}$ here must be expressed using the formulae from those of $\bar{\phi}$ and some numerical integration scheme. Moreover, some ellimination scheme should be prefered to the construction of $\bar{M}^{-1}$ in (18); let us remember the same comment with M^{-1} in (6).

Let us come back to the case with variable ϕ and fixed α, β, ξ, η and ϑ. The minimization of the $G + \bar{G}$ from (9) and (17) leads (in comparison with the previous section) to the following modifications: the left-hand side of (12) gets $\lambda + \bar{\lambda}\bar{\phi}^2$ instead of λ and the right-hand side of (14), after the change referenced as (14*), gets $\lambda + \bar{\lambda}\bar{\phi}_k^2$ instead of λ where $\bar{\phi}_k$ is set by the last update of α, β, ξ, η and ϑ; all remaining relations can be extended in the similar way.

Let (18*) refer to (18) where in the definition of G (similarly to the assignment of (10*) to (10), occuring in the algorithm in the preceding section yet) δ is replaced by δ_ε and ϕ by ϕ_k. Let (16*) refer to (16) with $\bar{\phi}$ replaced by $\bar{\phi}_k$. Our new software algorithm reads:

I. initialize ϕ_0 and also old $\alpha, \beta, \xi, \eta, \vartheta$, set $k \leftarrow 0$,

II. compute $c_1(\phi_k)$ and $c_2(\phi_k)$ from (10*),

II.a find $\bar{\phi}_k$ by (16*),

II.b compute new $\alpha, \beta, \xi, \eta, \vartheta$ from (18*),

II.c check whether new and old $\alpha, \beta, \xi, \eta, \vartheta$ are nearly the same; if not, take new values as old ones and go back to II.a,

III. solve (14*) to receive ϕ_{k+1},

IV. check whether ϕ_{k+1} is sufficiently close to ϕ_k; if not, set $k \leftarrow k+1$ and go back to II., else accept ϕ_k as final ϕ,

V. initialize old $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$,

VI. prepare $(x_1, y_1), \ldots, (x_n, y_n)$ by (15),

VII. compute new $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$ from (6),

VIII. check whether new and old $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$ are nearly the same; if not, take new values as old ones and go back to VI.

This algorithm is improved (in comparison with the previous one) by the nested cycle in II. Its drawback is that especially II.b is rather expensive; this is only partially compensated by the above explained simplification in the second block (from V. to VIII.). However, various programmer tricks may be applied to avoid too time-consuming calculations (with the risk of slower convergence): e. g. $\bar{\phi}_j$ may be considered in iv) instead for $\bar{\phi}_k$ until $k-j$ is smaller than certain prescribed integer.

Following [3], p. 427, let us now notice another trick, seemingly removing the nested cycle at all. Let us consider the symbolic vector operator

$$\bar{\nabla} = (\partial/\partial\alpha, \partial/\partial\beta, \partial/\partial\xi, \partial/\partial\eta, \partial/\partial\vartheta)\ .$$

Then (18) can be interpreted, thanks to the integration over Ω, as searching for the real vector $V = (\alpha, \beta, \xi, \eta, \vartheta)$ from the nonlinear system of 5 algebraic equations

$$\int_\Omega \bar{\nabla}\bar{\phi}\,\bar{\phi}\,\delta_\varepsilon(\phi)|\nabla\phi|\,\mathrm{d}\mu = 0\,. \tag{19}$$

Thus (18) is possible to be replaced by another iteration scheme

$$V_{k+1} = V_k + 2\tau\bar{\lambda}\int_\Omega \bar{\nabla}\bar{\phi}_k\,\bar{\phi}_k\,\delta_\varepsilon(\phi_k)|\nabla\phi_k|\,\mathrm{d}\mu = 0 \tag{20}$$

where V_k, $k \in \{0, 1, 2, \dots\}$, are the approximations of V (which could be identified again as certain "diffusion"). This consideration results in the algorithm where no high accuracy ellipse fitting in every step is needed:

I. initialize ϕ_0 and V_0, set $k \leftarrow 0$,

II. compute $c_1(\phi_k)$ and $c_2(\phi_k)$ from (10*),

II.a find $\bar{\phi}_k$ by (16*),

III. solve (14*) to receive ϕ_{k+1},

III.a solve (20) to receive V_{k+1},

IV. check whether ϕ_{k+1} is sufficiently close to ϕ_k and V_{k+1} is sufficiently close to V_k; if not, set $k \leftarrow k+1$ and go back to II., else accept ϕ_k as final ϕ and V_k as final V

V. initialize old $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$,

VI. prepare $(x_1, y_1), \dots, (x_n, y_n)$ by (15),

VII. compute new $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$ from (6),

VIII. check whether new and old $a, b, \widetilde{x}_0, \widetilde{y}_0, \rho$ are nearly the same; if not, take new values as old ones and go back to VI.

In this algorithm no high-accuracy ellipse fitting in every k-th iteration is necessary, each $\bar{\phi}_k$ by (16) is computed only once, being not predicted and corrected.

Finally let us come back to the idea of total removing of all steps after IV. To express x and y for (16) is not easy – cf. the iteration (5) for discrete points, thus it is better to rewrite (16) indirectly as

$$\bar{\phi}(x,y) = 1 - t^2 , \qquad x = \widetilde{x}_0 + (\gamma_x - \widetilde{x}_0)t , \qquad y = \widetilde{y}_0 + (\gamma_y - \widetilde{y}_0)t \tag{21}$$

for $t \geq 0$ and $0 \leq \varphi < 2\pi$ (the new coordinates t and φ, applied in $\widetilde{\varpi}$, correspond to standard polar coordinates in ϖ) where γ_x and γ_y are 2 (rather complicated) functions of φ, introduced (and also differentiable) in the same way as γ_{xi} and γ_{yi}, hidden in (5); later (16*) will refer to the same with $\bar{\phi}$ replaced $\bar{\phi}_k$. Instead of $\bar{\nabla}$ let us consider the symbolic vector operator

$$\bar{\nabla}_* = (\partial/\partial a, \partial/\partial b, \partial/\partial\widetilde{x}_0, \partial/\partial\widetilde{y}_0, \partial/\partial\rho) \; .$$

Then (18) can be interpreted, thanks to the integration over Ω, as searching for the real vector $U = (a, b, \widetilde{x}_0, \widetilde{y}_0, \rho)$ from (19) with $\bar{\nabla}_*$ substituting $\bar{\nabla}$. Thus (20) can be rewritten with U instead of V (all indices remain unchanged) which will be referenced as (20*). The resulting algorithm seems to be very short:

I. initialize ϕ_0 and U_0, set $k \leftarrow 0$,
II. compute $c_1(\phi_k)$ and $c_2(\phi_k)$ from (10*),
II.a find $\bar{\phi}_k$ by (21),
III. solve (14*) to receive ϕ_{k+1},
III.a solve (20*) to receive U_{k+1},
IV. check whether ϕ_{k+1} is sufficiently close to ϕ_k and U_{k+1} is sufficiently close to U_k; if not, set $k \leftarrow k+1$ and go back to II., else accept ϕ_k as final ϕ and U_k as final U .

Nevertheless, to solve (20*) we have to integrate in fact not over Ω, but over certain transformed domain, due to the transformation from (21); its Jacobi matrix is

$$J(t,\varphi) = \begin{bmatrix} \partial x/\partial\varphi & \partial x/\partial t \\ \partial y/\partial\varphi & \partial y/\partial t \end{bmatrix} = \begin{bmatrix} \gamma'_x & \gamma_x - \widetilde{x}_0 \\ \gamma'_y & \gamma_y - \widetilde{y}_0 \end{bmatrix} ,$$

thus

$$\mathrm{d}\mu = \mathrm{d}x\,\mathrm{d}y = |J(t,\varphi)|\,\mathrm{d}\varphi\,\mathrm{d}t = \left| t\left(\gamma'_x(\gamma_y - \widetilde{y}_0) - \gamma'_y(\gamma_x - \widetilde{x}_0)\right)\right|\,\mathrm{d}\varphi\,\mathrm{d}t$$

which complicates any numerical integration scheme substantially. To illustrate this statement, let us remind that

$$\nabla\phi_k = (\partial\phi_k/\partial\varphi\,\partial\varphi/\partial x + \partial\phi_k/\partial t\,\partial t/\partial x \, , \, \partial\phi_k/\partial\varphi\,\partial\varphi/\partial y + \partial\phi_k/\partial t\,\partial t/\partial y)$$

as a function of φ and t is needed in (20*); this must be done using the formal differentiation of x and y in (21):

$$\begin{bmatrix} \gamma'_x t & 0 & \gamma_x - \widetilde{x}_0 & 0 \\ 0 & \gamma'_x t & 0 & \gamma_x - \widetilde{x}_0 \\ \gamma_y - \widetilde{y}_0 & 0 & \gamma'_y t & 0 \\ 0 & \gamma_y - \widetilde{y}_0 & 0 & \gamma'_y t \end{bmatrix} \begin{bmatrix} \partial\varphi/\partial x \\ \partial\varphi/\partial y \\ \partial t/\partial x \\ \partial t/\partial y \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} .$$

The same fact obstructs to rewrite (10*) and (14*) in ϕ and t easily. Nevertheless, the advantage of such algorithm is that the "fully diffusional approach" may give all resulting parameters U without forcing any a posteriori Newton iterations.

Let us notice some other technical problems. Namely all steps III. (including their modifications) contain "regularized Dirac functions" δ_ε; their numerical treatment for ε close to 0 can be expected to force local mesh refinements in the finite difference approach. Some authors are thus motivated to rewrite equations like (14) into their variational form and analyse them using some multigrid finite element technique; e. g. [11] prefers CFEM (the "composite finite element method") by [17], an other efficient access to such multi-scale problems has been suggested in [15]. For illustration of the variational approach, let us rewrite our crucial equation (14*) in its integral form, respecting boundary conditions of type (11). The result is the "discretized evolution equation"

$$\int_\Omega \widetilde{\phi}\, \frac{\phi_{k+1} - \phi_k}{\tau}\, \mathrm{d}\mu + \int_\Omega \nabla \left(\widetilde{\phi}\, \delta_\varepsilon(\phi_k) \left(\lambda + \bar{\lambda} \bar{\phi}_k^2 \right) \right) \frac{\nabla \phi_{k+1}}{|\nabla \phi_{k+1}|}\, \mathrm{d}\mu$$
$$= \int_\Omega \widetilde{\phi}\, \delta_\varepsilon(\phi_k)(g - c_{2k})^2\, \mathrm{d}\mu - \int_\Omega \widetilde{\phi}\, \delta_\varepsilon(\phi_k)(g - c_{1k})^2\, \mathrm{d}\mu ,$$

satisfied for any admissible test function $\widetilde{\phi}$ (e. g. , in the simplest case, linear on the trianular mesh); in practice the functions $\widetilde{\phi}$ form a basis of some finite-dimensional function space containing ϕ_{k+1} that can be determined by solving a finite system of linear algebraic equations. The macro-scale choice of $\widetilde{\phi}$ can correspond to the size of pixels, the lower-scale one enables us refinements due to δ_ε.

5 A remark to one practical application

As an illustrative example of practical application of the above mentioned theory let us consider the following situation: we need to monitor the displacement of some part of building construction in time, caused by various loads, in time, related to some a priori known reference configuration. Our camera (rather cheap, but with guaranteed properties) obtains and stores sequences of images in particular times. The above analyzed algorithms give us a chance to detect any displacement at rather high accuracy level.

The real experiments with such snapshot sequencies are now a part of research at the Department of Technology of Buliding Materials and Prefabricated Elements of the Faculty of Civil Engineering at the Brno University of Technology. The analysis of results should be prepared for publication in the near future.

References

[1] Ahn S. J., Rauh W., Warnecke H.-J. Least squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola and parabola. *Pattern Recognition* **34**, 2001, 2283-2303.

[2] Augado A. S., Montiel M. E., Nixon M. S. On using directional information for parameter space decomposition in ellipse detection. *Pattern Recognition* **29**, 1996, 369-381.

[3] Bresson X., Vandergheynst P. Thiran J.-P. A priori information in image segmentation: energy functional based on shape statistical model and image information. *Proc. Int. Conf. on Image Processing in Barcelona*, 2003, 425-428.

[4] Casseles V., Kimmel R., Sapiro G. Geodesic active contours. *Int. J. Comput. Vision* **22**, 1997, 61-79.

[5] Chan T. F., Vese L. A. Active contours without edges. *IEEE Trans. Image Processing* **10**, 2001, 266-277.

[6] Chaudhuri B. B., Samanta G. P. Elliptic fit of objects in two and three dimensions by moment of inertia optimization *Pattern Recognition Lett.* **12**, 1991, 1-7.

[7] Cheng Y. C., Lee C. S. A new method for quadratic curve detection using K-RANSAC with acceleration techniques. *Pattern Recognition* **28**, 1995, 663-682.

[8] Dal Maso G., Morel J. M., Solimini S. A variational method in image segmentation: existence and approximation results. *Acta Matematica* **168**, 1992, 89-151.

[9] Dave R. N. Generalized fuzzy c-shells clustering and detection of circular and elliptical boundaries. *Pattern Recognition* **25**, 1992, 713-721.

[10] Diewald U., Preusser T, Rumpf M. Strzodka R. Diffusion models and their accelerate solution in image and surface processing. *Acta Math. Univ. Comenianae* **LXX**, 2000, 15-31.

[11] Droske M., Ring W. A Mumford-Shah level-set approach for geometric image registration. *Preprint*, University in Bonn, 2005.

[12] Fishler M. A., Bolles R. C. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. ACM* **24**, 1981, 381-395.

[13] Fitzgibbon A., Pilu M., Fisher R. B. Direct least square fitting of ellipses. *IEEE Trans. PAMI* **21**, 1999, 477-480.

[14] Gajewski H., Gröger K., Zacharias K. *Nonlinear Operator Equations and Operator Differential Equations*. (In German.) Academie, Berlin 1974.

[15] Glowinski R., He J., Lozinski A., Rappaz J., Wagner J. Finite element approximation of multi-scale elliptic problems using patches of elements. *Num. Math.* **101**, 2005, 663-688.

[16] Guil N., Zapata E. L. Lower-order circle end ellipse Hough transform. *Pattern Recognition* **30**, 1997, 1729-1744.

[17] Hackbush W., Sauter S. Composite finite elements for the approximation of PDEs on domainswith complicated micro-structures. *Num. Math.* **75**, 1997, 447-472.

[18] Ho C., Chan L. A fast ellipse/circle detector using geometric symmetry. *Pattern Recognition* **28**, 1995, 117-124.

[19] Hough P. V. C. Methods and means for recognizing complex patterns. *US Patent 3069654*, 1962.

[20] Kačur J., Mikula K. Slow and fast diffusion effects in image processing. *Comput. Visual Sci.* **3**, 2001, 185-195.

[21] Kadeřávek P. , Klíma J., Kounovský I. *Descriptive Geometry I.* (In Czech.) Czechoslovak Academy of Sciences, Prague 1954.

[22] Klapka J., Piska R., Zezula J. *Descriptive Geometry II.* (In Czech.) Technical University of Dr. E. Beneš, Brno 1951.

[23] Lei Y., Wong C. Ellipse detection based on symmetry. *Pattern Recognition Lett.* **20**, 1999, 41-47.

[24] Li H., Lavin M. A., Master L. Fast Hough transform: a hierarchical approach. *J. Comput. Vision Graphics Image Process.* **36**, 1986, 139-161.

[25] Lukšan L. *Methods with Variable Metric.* (In Czech.) Academia, Prague 1990.

[26] Mc Lauglin R. A. Randomized Hough transform: improved ellipse detection with comparison. *Pattern Recognition Lett.* **19**, 1998, 299-305.

[27] Rektorys K. and all. *Survey of Applied Mathematics.* (In Czech.) SNTL, Prague 1963.

[28] Shen J., Shen W. Image smoothing and adge detection by Hermite itegration. *Pattern Recognition* **28**, 1995, 1159-1166.

[29] Sussman S., Smereka P., Osher S. A level set approach for compressible solutions to incompressible two-phase flow. *J. Comput. Phys.* **119**, 1994, 146-159.

[30] Vincze M. Robust tracking of ellipses at frame rate. *Pattern Recognition* **34**, 2001, 487-498.

[31] Vitásek E. *Fundamentals of the Theory of Numerical Methods for the Analysis of Differential Equations.* (In Czech.) Academia, Prague 1994.

[32] Weickert J. Application of nonlinear diffusion in image processing and computer vision. *Acta Math. Univ. Comenianae* **LXX**, 2000, 33-50.

[33] Xie Y., Ji Q. A new efficient ellipse detection method. *Proc. Int. Conf. on Pattern Recognition in Quebec*, 2002, 957-960.

[34] Yin P.-Y. A new circle/ellipse detector using genetic algorithms. *Proc. IPRR Conf. in Taipei*, 1998, 362-368.

[35] Yin P.-Y., Chen L. New method for elliptic detection by means of symmetry. *Journal of Electronic Imaging* **3**, 1994, 20-29.

[36] Yoo J. H., Seithi I. K. An ellipse detection method from the polar and pole definition of conics. *Pattern Recognition* **26**, 1993, 307-315.

[37] Zhang Y. J., Liu Z.-Q. Self-splitting competitive learning: a new on-line clustering paradigm. *IEEE Trans. Neural Networks* **13**, 2002, 369-380.

[38] Zhang Y. J., Liu Z.-Q. A robust, real-time ellipse detector. *Pattern Recognition* **38**, 2005, 273-287.