ORIGINAL ARTICLE

A method for local approximation of a planar deformation field

Marcin Ligas[1*], Marek Banaś[2] and Anna Szafarczyk[3]

[1]Department of Integrated Geodesy and Cartography, Faculty of Mining Surveying and Environmental Engineering, AGH University of Science and Technology, 30 A. Mickiewicza Avenue, 30-059, Cracow, Poland
[2]Institute of Technical Engineering, Bronisław Markiewicz State Higher School of Technology and Economics in Jarosław, 16 Czarnieckiego Street, 37-500, Jarosław, Poland
[3]Department of Mine Areas Protection, Geoinformatics and Mine Surveying, Faculty of Mining Surveying and Environmental Engineering, AGH University of Science and Technology, 30 A. Mickiewicza Avenue, 30-059, Cracow, Poland

[*]ligas@agh.edu.pl

Abstract

We present a method of approximation of a deformation field based on the local affine transformations constructed based on *n* nearest neighbors with respect to points of adopted grid. The local affine transformations are weighted by means of inverse distance squared between each grid point and observed points (nearest neighbors). This work uses a deformation gradient, although it is possible to use a displacement gradient instead – the two approaches are equivalent. To decompose the deformation gradient into components related to rigid motions (rotations, translations are excluded from the deformation gradient through differentiation process) and deformations, we used a polar decomposition and decomposition into a sum of symmetric and an anti-symmetric matrices (tensors). We discuss the results from both decompositions. Calibration of a local affine transformations model (i.e., number of nearest neighbors) is performed on observed points and is carried out in a cross-validation procedure. Verification of the method was conducted on simulated data-grids subjected to known (functionally generated) deformations, hence, known in every point of a study area.

Key words: deformation analysis, polar decomposition, affine transformation, rotation matrix, stretch tensor

1 Introduction

Deformation analysis aims at inferring on geometrical changes of objects under study. The object, in fact, may be of material character, for example, deformation of the Earth's crust, and also may have an abstract character, for example, deformation of the gravity field (Dermanis and Livieratos, 1983). Both cases are treated by the same tools originating from, for example, differential geometry, theory of elasticity (also continuum mechanics), estimation theory, and tensor calculus. The geometrical changes are understood as rigid motions and distortions of the object's shape. Deformation analysis has a very broad spectrum of applications in seemingly not related fields but aiming at as accurate as possible description of object's behavior in different time instances (epochs). It is used, amongst others, in biomechanics and medicine-related applications (Tanaka et al., 2012; Bayly et al., 2005), in surveying engineering (Caspary et al., 1990), in Earth's crust movement monitoring (Altiner, 2013; Dermanis and Kotsakis, 2006), and in landslides monitoring (Szafarczyk and Gawalkiewicz, 2016) to mention only a few.

The search for simple and universal methods (and also software development) for the determination of a deformation field is still an up-to-date research topic Goudarzi et al. (2015). In this article, we present a simple method to approximate a deformation field based on the tools that surveyors and geodesists

are familiar with. It uses a distance-weighted affine transformation model performed in a local manner and calibrated in a cross-validation (CV) procedure. The method uses a rigorous approach to recover a rotation and a stretch tensor from a deformation gradient. This approach relies on the use of polar decomposition of a deformation gradient rather than the use of simplified decomposition into symmetric and skew-symmetric parts. However, the two are discussed. The use of the latter mentioned decomposition was justified when the computer processing power was limited or it may be accepted when a priori knowledge of small rotation angles is present. The method provides all estimated quantities (strains and rotation) with their standard errors obtained via variance–covariance propagation law.

The article is organized as follows. The first introductory section defines quantities that are used throughout the article, that is, a deformation gradient (or displacement gradient), the polar decomposition of a matrix (used to separate rigid rotations from deformations), and two strain tensors. The second section shows the mechanics of the method. It presents the affine transformation model in its explicit (geometric parameters equal to rotation and strain tensor entries) and implicit (transformation parameters) forms. It also demonstrates the use of polar decomposition to recover geometric parameters from transformation parameters. In addition, a formula based on the variance–covariance propagation law for error analysis of the geometric parameters of the transformation and a CV technique to calibrate the local affine transformations are presented. The next section is a part of numerical example. Data are simulated to have the full control over the method. The data generation process is explained, and necessary formulas are provided. The subsequent section shows the results of the method and its reliability in comparison to theoretical values of generated deformations. Finally, conclusions are drawn.

2 Deformation Gradient, Polar decomposition, and Strain tensors

Deformation gradient (**F**) or/and displacement gradient (**G**) are basic tools while analyzing deformations (in theory of elasticity and continuum mechanics). If we introduce two configurations of points (homological) representing any object and we denote a vector of coordinates of any point in the first one as $\mathbf{x} = [x\, y]^T$ (initial or reference) and in the second one as $\mathbf{X} = [X\, Y]^T$ (current or deformed), then we may try to define a relation between **x** and **X** of the form $\mathbf{X} = f(\mathbf{x})$. The deformation gradient is defined as the derivative of each component of the deformed **X** vector with respect to each component of the reference **x** vector. Hence, the deformation gradient (here in 2D) reads as follows:

$$\mathbf{F} = \frac{\partial \mathbf{X}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial X}{\partial x} & \frac{\partial X}{\partial y} \\ \frac{\partial Y}{\partial x} & \frac{\partial Y}{\partial y} \end{bmatrix}. \tag{1}$$

On the other hand, if we introduce a displacement vector $\mathbf{u} = \mathbf{X} - \mathbf{x}$, then the deformation gradient may be expressed as

$$\mathbf{F} = \frac{\partial (\mathbf{x} + \mathbf{u})}{\partial \mathbf{x}} = \mathbf{I} + \frac{\partial \mathbf{u}}{\partial \mathbf{x}} = \mathbf{I} + \mathbf{G}, \tag{2}$$

where $\mathbf{G} = \frac{\partial \mathbf{u}}{\partial \mathbf{x}}$ is a displacement gradient. The deformation gradient (or displacement gradient) carries total information about local rotation and deformations of an object.

To recover deformations and rotations from the deformation gradient, a polar decomposition is used. The polar decomposition of a matrix factorizes the matrix into a product of an orthogonal matrix **R** and a symmetric positive (semi) definite

matrix **U** (a stretch tensor), that is,

$$\mathbf{F} = \mathbf{RU}, \tag{3}$$

where **R** must obey det(**R**) = 1 to be a rotation matrix. The polar decomposition may be obtained, for example, by means of singular value decomposition (SVD) or spectral decomposition. Generally, in order to transform a stretch tensor **U** (factor of the polar decomposition) into a strain tensor, one subtracts the identity matrix from it (or takes its logarithm, i.e., log **U**, but this strain measure will not be used in this work) for the review of these mentioned and other strain measures, for example, see Chaves (2013). In this work, we use only two strain measures (tensors):

i. Biot's strain tensor (a straightforward measure of strain but not commonly used):

$$\mathbf{E^B} = \begin{bmatrix} E_{11}^B & E_{12}^B \\ E_{21}^B & E_{22}^B \end{bmatrix} = (\mathbf{U} - \mathbf{I}) = \left[\mathbf{I} + 2 \left(\mathbf{G} + \mathbf{G}^T \right) + \mathbf{G}^T \mathbf{G} \right]^{\frac{1}{2}} - \mathbf{I}, \tag{4}$$

ii. Small (engineering/infinitesimal) strain tensor (common but limited to small rotations):

$$\varepsilon = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} \\ \varepsilon_{21} & \varepsilon_{22} \end{bmatrix} = \frac{1}{2} \left(\mathbf{F} + \mathbf{F}^T \right) - \mathbf{I} = \frac{1}{2} \left(\mathbf{G} + \mathbf{G}^T \right). \tag{5}$$

Using polar decomposition with $\mathbf{R} = \mathbf{I}$ (no rotation case), that is, $\mathbf{F} = \mathbf{RU} = \mathbf{U}$, it is easy to show that the small strain tensor is identical to Biot's tensor:

$$\varepsilon = \frac{1}{2} \left(\mathbf{F} + \mathbf{F}^T \right) - \mathbf{I} = \frac{1}{2} \left(\mathbf{RU} + \mathbf{U}^T \mathbf{R}^T \right) - \mathbf{I} = \mathbf{U} - \mathbf{I} \tag{6}$$

For negligible rotations, not the strains themselves, instead of the polar decomposition, one may use a simple decomposition of a displacement gradient into a symmetric matrix (eq. (5)) and a skew-symmetric one, that is,

$$\mathbf{G} = \frac{1}{2} \left(\mathbf{G} + \mathbf{G}^T \right) + \frac{1}{2} \left(\mathbf{G} - \mathbf{G}^T \right) = \mathbf{G}^s + \mathbf{G}^a =$$
$$= \left[\frac{1}{2} \left(\mathbf{F} + \mathbf{F}^T \right) - \mathbf{I} \right] + \frac{1}{2} \left(\mathbf{F} - \mathbf{F}^T \right), \tag{7}$$

where $\mathbf{G}^s$ (a symmetric matrix) stores information on expansion, contraction, and shearing deformations at a given point and $\mathbf{G}^a$ (a skew-symmetric matrix) carrying information on rotations (infinitesimal ones) (Berber et al., 2012).

3 Mechanics of the method

3.1 Affine transformation

To locally relate coordinates of an undeformed (reference, initial) configuration (or configuration obtained from any previous measurement), denoted in lowercase, with the coordinates of deformed configuration (current or final), denoted in uppercase, we use affine transformation model, known also as 6-parameter transformation. The affine transformation is a simple and flexible model (deforms shapes but parallel lines are preserved) often used in surveying and mapping practice. For a single point, we construct two equations of the following form (e.g., Osada and Sergieieva (2010)):

$$X = t_X + ax + by = t_X + (s_x \cos \varphi - s_{xy} \sin \varphi) x$$
$$+ (s_{xy} \cos \varphi - s_y \sin \varphi) y, \tag{8a}$$

$$Y = t_Y + cx + dy = t_Y + (s_x \sin \varphi + s_{xy} \cos \varphi) x$$
$$+ (s_{xy} \sin \varphi + s_y \cos \varphi) y, \tag{8b}$$

or in a matrix–vector form:

$$\mathbf{X} = \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} t_X \\ t_Y \end{bmatrix} + \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} =$$
$$= \begin{bmatrix} t_X \\ t_Y \end{bmatrix} + \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} s_x & s_{xy} \\ s_{xy} & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{t} + \mathbf{RUx}, \quad (9)$$

where t_X, t_Y, s_x, and s_y are translations and scale factors (stretches) along the x- and y- axes, respectively; s_{xy} is the shear; and φ is the rotation angle while transformation parameters a, b, c, and d are functions of previously introduced geometric parameters. To solve for unknown parameters, we need at least three homological points in no adjustment case or more to adjust the results and obtain some accuracy measures. Equations (8a), (8b) or (9) reveal the relation among transformation parameters a, b, c, and d and geometric parameters s_x, s_y s_{xy}, and φ. The considered model is nonlinear with respect to geometric parameters (scale factors, shear, and rotation), and it can be solved through linearization and iteration to explicitly obtain geometric parameters (rather uncomfortable solution). On the other hand, this model is linear with respect to transformation parameters and may easily be solved in the linear least squares framework (this option was chosen in this work), but this will require a variance–covariance propagation from transformation parameters to geometric parameters (shown hereafter). The local affine transformations are weighted, in this work, by means of inverse distance squared between each grid point (new point) and observed points (nearest neighbors). It is easy to notice that the deformation gradient and its polar decomposition in the affine transformation case read (compare to the right hand side of eq. (9)):

$$\mathbf{F} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \mathbf{RU} = \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} s_x & s_{xy} \\ s_{xy} & s_y \end{bmatrix} \quad (10)$$

3.2 Polar decomposition (once more)

Polar decomposition of a matrix is very often presented and computed via abovementioned SVD. Using SVD to extract a rotation matrix and a stretch matrix (stretch tensor) from the

deformation gradient, we have the following relation:

$$\mathbf{F} = \mathbf{W\Sigma V}^T = \mathbf{RU}, \quad (11)$$

where $\mathbf{W}$ and $\mathbf{V}$ are orthogonal matrices and $\mathbf{\Sigma}$ is a diagonal matrix with non-negative values on the main diagonal. By introducing the product $\mathbf{V}^T\mathbf{V}$ before $\mathbf{\Sigma}$ in (11), one obtains

$$\mathbf{F} = \mathbf{WV}^T\mathbf{V\Sigma V}^T = \mathbf{RU}. \quad (12)$$

Hence, the factors of the polar decomposition read as

$$\mathbf{R} = \mathbf{WV}^T \quad \text{and} \quad \mathbf{U} = \mathbf{V\Sigma V}^T. \quad (13)$$

A firm review of polar decomposition algorithms (iterative and exact) and its properties may be found in, for example, Gander (1990), Markley and Mortari (1999), Higham (1986), Shoemake and Duff (1992).

3.3 Variance and covariance propagation among transformation parameters and geometric parameters

To equip the geometric parameters (obtained with the use of polar decomposition) with standard errors, the variance–covariance propagation law was used. According to equation (8a), (8b) or (9), transformation parameters may explicitly be expressed as functions of geometric parameters, that is,

$$t_X = t_X, \quad (14a)$$
$$t_Y = t_Y, \quad (14b)$$
$$a = s_x \cos\varphi - s_{xy} \sin\varphi, \quad (14c)$$
$$b = s_{xy} \cos\varphi - s_y \sin\varphi, \quad (14d)$$
$$c = s_x \sin\varphi + s_{xy} \cos\varphi, \quad (14e)$$
$$d = s_{xy} \sin\varphi + s_y \cos\varphi. \quad (14f)$$

Jacobi matrix reads as (the order of differentiation is t_X, t_Y, s_x, s_y, s_{xy}, φ)

$$\mathbf{J} = \begin{bmatrix} \nabla t_X \\ \nabla t_Y \\ \nabla a \\ \nabla b \\ \nabla c \\ \nabla d \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \cos\varphi & 0 & -\sin\varphi & -s_{xy}\cos\varphi - s_x\sin\varphi \\ 0 & 0 & 0 & -\sin\varphi & \cos\varphi & -s_y\cos\varphi - s_{xy}\sin\varphi \\ 0 & 0 & \sin\varphi & 0 & \cos\varphi & s_x\cos\varphi - s_{xy}\sin\varphi \\ 0 & 0 & 0 & \cos\varphi & \sin\varphi & s_{xy}\cos\varphi - s_y\sin\varphi \end{bmatrix}. \quad (15)$$

If we denote a vector of transformation parameters as

$$\hat{\mathbf{p}}^T = \begin{bmatrix} t_X & t_Y & a & b & c & d \end{bmatrix} \quad (16)$$

and a vector of geometric parameters as

$$\hat{\mathbf{p}}_g^T = \begin{bmatrix} t_X & t_Y & s_x & s_y & s_{xy} & \varphi \end{bmatrix}, \quad (17)$$

then the variance–covariance propagation law gives

$$\mathrm{Cov}\left(\hat{\mathbf{p}}_g\right) = \mathbf{J}^{-1}\mathrm{Cov}\left(\hat{\mathbf{p}}\right)\mathbf{J}^{T^{-1}} \quad (18)$$

and standard errors are the square roots of entries on the main diagonal of the matrix (18).

3.4 Calibration of the model

Calibration of the model means here selection of optimal number of nearest neighbors (or possibly other factors influencing the model) for local affine transformations. The model is calibrated on known points (observations) with an assumption that the results of calibration should be valid for all points within a study area. As a criterion we use, a CV score that is equal to

$$\mathrm{CV}(nn) = \sum_{i=1}^{N}\left(X_i - \hat{X}_i(nn)\right)^2 + \sum_{i=1}^{N}\left(Y_i - \hat{Y}_i(nn)\right)^2, \quad (19)$$

where $\hat{X}_i(nn)$ and $\hat{Y}_i(nn)$ are fitted (adjusted) values of X_i and Y_i with the observations for point i omitted from the calibration process, nn means the number of nearest neighbors, and

(a) the original (undeformed) grid **(b)** central swirl deformation **(c)** non-central swirl deformation
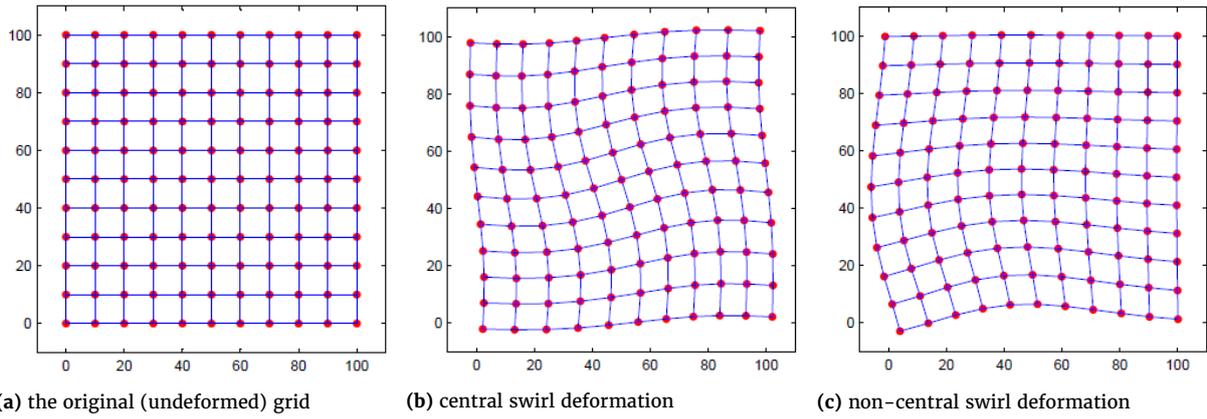
Figure 1. The effect of swirl–like deformation model

N denotes the number of known points. The lowest value of CV score indicates possibly the best model among the tested (optimal in CV-sense number of nearest neighbors or other parameters defining the model).

4 Data

In order to have full control over the operation of the method, the deformed grids (a current configuration) on the basis of undeformed ones (a reference configuration) have been simulated in a functional manner. They were generated according to the swirl–like (Fig. 1), barrel (Fig. 2), and pincushion (Fig. 3) deformation models. Thus generated grids acted as observed points (with known coordinates in current and reference configurations). The grids were generated in a regular (equally spaced points) and irregular (percentage of randomly deleted points from the regular grid) manner (irregular ones are not presented in the figures). Only the kinds (shape) of deformation used in this study are demonstrated in the included figures.

4.1 Swirl–like deformation

Swirl–like deformation is introduced by the following formulas:

$$X = (x - x_0) \cos\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right) - (y - y_0)\sin\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right) + x_0,$$
(20a)

$$Y = (x - x_0)\sin\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right) + (y - y_0)\cos\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right) + y_0,$$
(20b)

where $r = \sqrt{(x - x_0)^2 + (y - y_0)^2}$ is a distance of a point from the center of a deformation, α is a rotation angle, b is a control parameter, and x_0 and y_0 are coordinates of the center of a deformation. A graphical representation of this deformation is presented in Fig. 1

Components of the deformation gradient, given in (1), in this case read as

$$\frac{\partial X}{\partial x} = \cos\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right) + \frac{2e^{-\left(\frac{r}{b}\right)^2}\alpha\,(x - x_0)}{b^2}$$
$$\left[\sin\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right)(x - x_0) + \cos\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right)(y - y_0)\right],$$
(21a)

$$\frac{\partial X}{\partial y} = -\sin\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right) + \frac{2e^{-\left(\frac{r}{b}\right)^2}\alpha\,(y - y_0)}{b^2}$$
$$\left[\sin\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right)(x - x_0) + \cos\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right)(y - y_0)\right],$$
(21b)

$$\frac{\partial Y}{\partial x} = \sin\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right) - \frac{2e^{-\left(\frac{r}{b}\right)^2}\alpha\,(x - x_0)}{b^2}$$
$$\left[\cos\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right)(x - x_0) - \sin\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right)(y - y_0)\right],$$
(21c)

$$\frac{\partial Y}{\partial y} = \cos\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right) - \frac{2e^{-\left(\frac{r}{b}\right)^2}\alpha\,(y - y_0)}{b^2}$$
$$\left[\cos\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right)(x - x_0) - \sin\left(e^{-\left(\frac{r}{b}\right)^2}\alpha\right)(y - y_0)\right].$$
(21d)

4.2 Barrel and Pincushion deformations (distortion)

$$X = (x - x_0)\left(1 + c_1 r^2 + c_2 r^4\right) + x_0,$$
(22a)

$$Y = (y - y_0)\left(1 + c_1 r^2 + c_2 r^4\right) + y_0,$$
(22b)

where $r = \sqrt{(x - x_0)^2 + (y - y_0)^2}$ is a distance of a point from the center of a deformation, c_1 and c_2 are constants (c_1, $c_2 < 0$ for barrel distortion; c_1, $c_2 > 0$ for pincushion distortion), and x_0 and y_0 are the coordinates of the center of a deformation. A graphical representation of barrel and pincushion deformations is presented in Fig. 2 (a barrel deformation) and in Fig. 3 (a pincushion deformation). Components of a deformation gra-
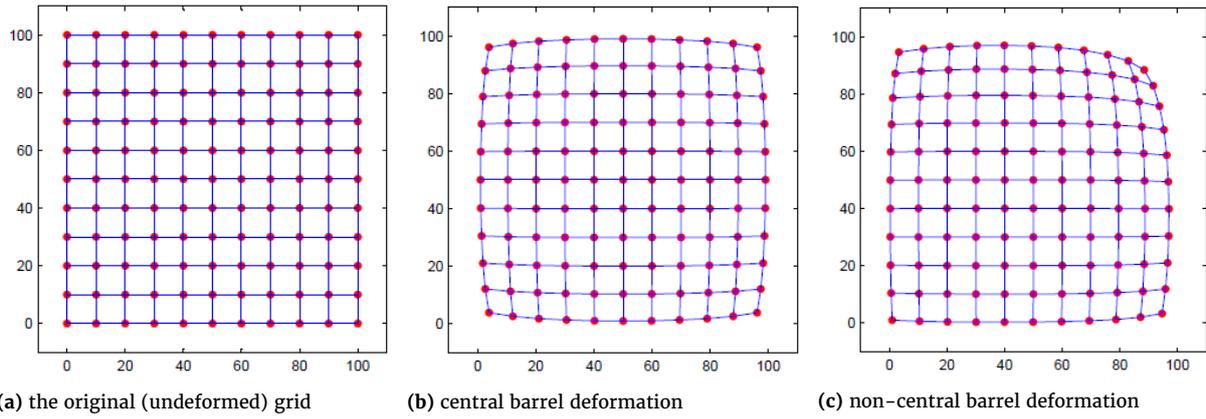
(a) the original (undeformed) grid **(b)** central barrel deformation **(c)** non-central barrel deformation

Figure 2. The effect of barrel deformation model

(a) the original (undeformed) grid **(b)** central pincushion deformation **(c)** non-central pincushion deformation
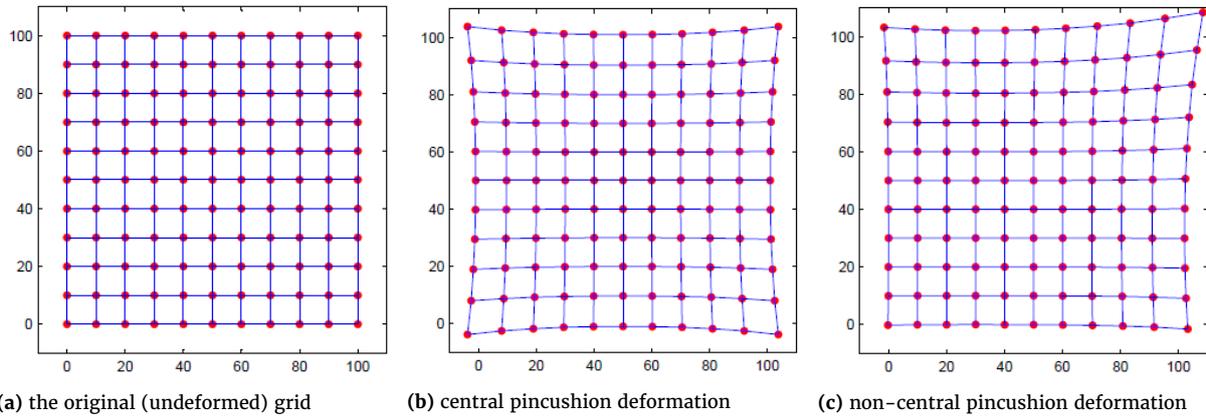
Figure 3. The effect of pincusion deformation model

dient given in (1) read as

$$\frac{\partial X}{\partial x} = c_1 r^2 + 2(x - x_0)^2 \left(c_1 + 2c_2 r^2\right) + c_2 r^4 + 1, \tag{23a}$$

$$\frac{\partial X}{\partial y} = 2(y - y_0)(x - x_0)\left(c_1 + 2c_2 r^2\right), \tag{23b}$$

$$\frac{\partial Y}{\partial x} = 2(y - y_0)(x - x_0)\left(c_1 + 2c_2 r^2\right), \tag{23c}$$

$$\frac{\partial Y}{\partial y} = c_1 r^2 + 2(y - y_0)^2 \left(c_1 + 2c_2 r^2\right) + c_2 r^4 + 1. \tag{23d}$$

5 Results

In this section, some exemplary results from simulation study of the method are presented. Tables 1–3 present selected results for above-described deformation models. As measures of deformation structure recovery, we used minimum value, 1st quartile, median, 3rd quartile and maximum value, mean error (ME as a measure of over- or underestimation), mean absolute error (MAE), and a correlation coefficient between true and estimated strain components. ME and MAE are based on differences between corresponding true and estimated values of strain tensor components. In all presented cases, strain components were determined at points being incenters of Delone triangles covering the simulated study areas. Five nearest neighbors were used to generate locally weighted affine transformation models giving the opportunity of providing the results with error analysis. A spatial domain of 500 m × 500 m was covered by regular and irregular grids of points. The regular grids consisted of 10 points in each row and in each column (100 points). In order to keep the same number of points for irregular grids, a densified set of points (20 in each row and column) was generated and then 75% of points were randomly removed. Simulated deformations resulted in various (from low to high) magnitudes of displacements of grid points.

As first, the results from a swirl-like deformation with significantly varying displacements will be presented (extreme case). The results from both regular and irregular grids will be presented in a single table; in every cell, the first row refers to the regular case and the second row refers to the irregular case (this also concerns remaining deformation models).

The results contained in Table 1 concern data generated according to a central swirl-like deformation with parameters: $x_0 = \text{mean}(x)$, $y_0 = \text{mean}(y)$, $\alpha = \pi/50$, and $b = 100$. These parameters resulted in displacement of points varying from -2.4318 to 2.4318 m (with arbitrary intermediate values) with respect to the reference (undeformed) configuration. Comparison of statistical structure of true and estimated strain components (for both grids) measured by minimum value, maximum value, and quartiles reveal moderate but still very informative correspondence between the two. Values of MEs, on the other hand, reveal negligible overestimation (underestimation) in this case. Correlation coefficient for the regular grid close to unity confirms a linear relationship (correspondence) of true and estimated strain tensor components. Also MAEs prove satisfactory accuracy obtained from the method in this case. It is worth mentioning that absolute differences between true and estimated strain components do not exceed (in majority of points – 90%) values of corresponding standard errors offered by the method (eq. 18). For irregular case, a decrease in structure recovery is visible. MEs reveal stronger underestimation (overestimation) of strain tensor components; MAEs

Table 1. Comparison of statistical measures for true and estimated strain components for regular grid (first rows) and irregular grid (second rows) in a central swirl-like deformation model

	E_{11}^B		E_{22}^B		$E_{12}^B = E_{21}^B$	
	True	Estimated	True	Estimated	True	Estimated
Minimum	−0.01979	−0.01712	−0.02272	−0.01787	−0.02262	−0.02093
	−0.02251	−0.02352	−0.02288	−0.02230	−0.02288	−0.02330
1st quartile	−0.00208	−0.00171	−0.00245	−0.00208	−0.00189	−0.00224
	−0.00209	−0.00326	−0.00283	−0.00359	−0.00202	−0.00196
Median	0.00001	−0.00002	−0.00001	−0.00005	0.00000	0.00001
	0.00000	−0.00018	0.00003	−0.00018	0.00000	0.00000
3rd quartile	0.00246	0.00194	0.00227	0.00209	0.00188	0.00226
	0.00288	0.00324	0.00210	0.00273	0.00334	0.00371
Maximum	0.02325	0.01808	0.02025	0.01713	0.02238	0.02075
	0.02261	0.02553	0.02303	0.02532	0.02285	0.02042
ME	0.00003		0.00002		−0.00001	
	0.00010		0.00014		−0.00024	
MAE	0.00110		0.00108		0.00122	
	0.00285		0.00261		0.00218	
Correlation coefficient	0.98327		0.98265		0.96365	
	0.82977		0.86766		0.91781	

Table 2. Comparison of statistical measures for true and estimated strain components for regular grid (first rows) and irregular grid (second rows) in a non-central barrel deformation model

	E_{11}^B		E_{22}^B		$E_{12}^B = E_{21}^B$	
	True	Estimated	True	Estimated	True	Estimated
Minimum	−0.00114	−0.00100	−0.00114	−0.00106	−0.00076	−0.00071
	−0.00091	−0.00092	−0.00089	−0.00090	−0.00055	−0.00062
1st quartile	−0.00021	−0.00023	−0.00021	−0.00022	−0.00007	−0.00007
	−0.00021	−0.00024	−0.00019	−0.00021	−0.00008	−0.00009
Median	−0.00005	−0.00006	−0.00005	−0.00006	0.00000	0.00000
	−0.00008	−0.00011	−0.00006	−0.00007	−0.00001	−0.00002
3rd quartile	−0.00001	−0.00001	−0.00001	−0.00001	0.00000	0.00000
	−0.00001	−0.00001	−0.00001	−0.00001	0.00000	0.00000
Maximum	0.00000	0.00000	0.00000	0.00000	0.00009	0.00009
	0.00000	0.00000	0.00000	0.00000	0.00011	0.00010
ME	0.00000		0.00000		0.00000	
	0.00002		0.00002		0.00001	
MAE	0.00001		0.00001		0.00000	
	0.00003		0.00002		0.00001	
Correlation coefficient	0.99427		0.99418		0.99925	
	0.98163		0.98620		0.98415	

are roughly 2.5 times larger than those in the previous case. Correlation coefficient, although visibly lower, is still close to unity and confirms a linear relationship (correspondence) of true and estimated strain tensor components. This accuracy degradation is justified by large deformations (generated on purpose) manifesting themselves by points' displacements and irregularity of sampling.

Table 2 contains the results for a non-central barrel deformation with parameters $x_0 = 0.45 \cdot \mathrm{mean}(x)$, $y_0 = 0.45 \cdot \mathrm{mean}(y)$, $c_1 = -5 \cdot 10^{-15}$, $c_2 = -5 \cdot 10^{-15}$. These parameters resulted in displacement of points varying from −0.1747 to 0.0149 m (balanced in comparison to the previous deformation model) with respect to the reference (undeformed) configuration.

The estimated quantities match theoretical ones accurately. Values of MEs reveal insignificant overestimation in this case. Correlation coefficients close to unity confirm a linear correspondence of true and estimated strain tensor components. MAEs prove more than satisfactory accuracy obtained from the method used in this case. The results listed in Table 3 concern data generated according to a non-central pincushion deformation with parameters $x_0 = 0.75 \cdot \mathrm{mean}(x)$, $y_0 = 0.75 \cdot \mathrm{mean}(y)$,

$c_1 = 5 \cdot 10^{-15}$, $c_2 = 5 \cdot 10^{-15}$. These parameters resulted in displacement of points varying from −0.0165 to 0.0596 (mild) with respect to the reference (undeformed) configuration.

Also in this mild-deformation case, one observes very good strain tensor structure recovery with regard to its theoretical counterparts. The results are so satisfactory that they do not require further comment.

As mentioned in the introductory section, the use of small strain tensor (eqs. (5) and (7)) in deformation analysis is justifiable when there is no rotation or it is negligible. The last two simulations are very good examples in this respect because these deformation models do not contain the rotation component at all. Hence, in these cases, there is no difference between strain tensor components obtained by means of eqs. (4) and (5). On the other hand, for a swirl deformation model that consists of variable rotations, the difference between the two strain tensors becomes visible, although the presented example uses rather small values of rotation angles. Maximum difference between the small strain tensor and Biot's strain tensor reaches the value of 1.8 mm/m (for 3.4° rotation angle), and also significant differences are observed for angles down to ap-

Table 3. Comparison of statistical measures for true and estimated strain components for regular grid (first rows) and irregular grid (second rows) in a non-central pincushion deformation model

	E_{11}^B		E_{22}^B		$E_{12}^B = E_{21}^B$	
	True	Estimated	True	Estimated	True	Estimated
Minimum	0.00000	0.00000	0.00000	0.00000	−0.00010	−0.00010
	0.00000	0.00000	0.00000	0.00000	−0.00010	−0.00008
1st quartile	0.00000	0.00001	0.00000	0.00001	−0.00001	−0.00001
	0.00000	0.00000	0.00000	0.00000	−0.00001	−0.00001
Median	0.00002	0.00002	0.00002	0.00002	0.00000	0.00000
	0.00002	0.00002	0.00002	0.00003	0.00000	0.00000
3rd quartile	0.00007	0.00008	0.00007	0.00008	0.00001	0.00001
	0.00007	0.00007	0.00007	0.00007	0.00001	0.00001
Maximum	0.00046	0.00040	0.00046	0.00042	0.00031	0.00029
	0.00046	0.00042	0.00044	0.00046	0.00030	0.00029
ME	0.00000		0.00000		0.00000	
	0.00000		0.00001		0.00000	
MAE	0.00001		0.00001		0.00000	
	0.00001		0.00001		0.00001	
Correlation coefficient	0.99093		0.99057		0.99876	
	0.98281		0.97792		0.98595	

proximately $1.5°$. Comparing the maximum difference between the two tensors with the overall accuracy of the method measured by MAE, one notices that the magnitude of this difference exceeds the MAE for the regular grid and makes approximately 65% of MAE for irregular grid (for particular values, see Table 1). Hence, when a rotation component is not negligible, the use of small strain tensor is a misuse and may lead to unreliable results. Generally, with readily available modern computer power at hand, there is no need to use simplified solutions; hence, the use of polar decomposition is recommended. The use of this decomposition is not computer intensive because we use only 2nd-rank tensors represented by matrices ($2×2$ for 2D and $3×3$ for 3D).

6 Conclusions

In the article, we have presented a simple method that approximates a planar deformation field, that is, strain tensor and rotation components. The method is based on a distance-weighted affine transformation model performed for each generated grid point based on the nearest observations covering the study area. In order to recover a stretch (and later a strain) tensor and a rotation from local affine transformation parameters, we used a standard tool used in continuum mechanics, that is, the polar decomposition performed with the use of another matrix decomposition, that is, singular value decomposition. The latter decomposition is easily accessible in many software packages, for example, MATLAB, Mathematica, Mathcad, R, and others. For comparison purposes, we also used a simplified decomposition, that is, decomposition into symmetric (strain) and antisymmetric (infinitesimal rotation) parts. The comparison demonstrates that when a rotation component is not negligible, the use of small strain tensor may lead to unreliable results. To perform accuracy analysis, we also derived a formula for a covariance matrix for stretch (also strain) and rotation tensors components. The derived formula is based on a variance–covariance propagation law between affine transformation coefficients and geometric parameters. Many numerical (simulation) tests (among them, those presented in the content of the article) proved the potential and usability of this simple method in deformation-analysis-related problems. The method may be extended to a 3-dimensional case what is in fact in progress. In the simulation study, we omit-

ted all issues concerned with eigen (spectral) analysis of strain tensors because it would not bring any additional information to the presented method in the context of this article.

Acknowledgements

References

Altiner, Y. (2013). *Analytical surface deformation theory: for detection of the Earth's crust movements*. Springer, Berlin, Heidelberg, doi:10.1007/978-3-662-03935-9.

Bayly, P. V., Cohen, T., Leister, E., Ajo, D., Leuthardt, E., and Genin, G. (2005). Deformation of the human brain induced by mild acceleration. *Journal of Neurotrauma*, 22(8):845–856, doi:10.1089/neu.2005.22.845.

Berber, M., Kutoglu, H., Dare, P., and Vaníček, P. (2012). Combining surface deformation parameters referred to different terrestrial coordinate systems. *Survey Review*, 44(324):23–29, doi:10.1179/1752270611Y.0000000005.

Caspary, W., Haen, W., and Borutta, H. (1990). Deformation analysis by statistical methods. *Technometrics*, 32(1):49–57.

Chaves, E. W. V. (2013). *Notes on continuum mechanics*. Springer, Dordrecht, doi:10.1007/978-94-007-5986-2.

Dermanis, A. and Kotsakis, C. (2006). Estimating crustal deformation parameters from geodetic data: Review of existing methodologies, open problems and new challenges. In Sansò, F. and Gil, A. J., editors, *Geodetic deformation monitoring: from geophysical to engineering roles*, pages 7–18. Springer, Heidelberg, doi:10.1007/978-3-540-38596-7_2.

Dermanis, A. and Livieratos, E. (1983). Applications of deformation analysis in geodesy and geodynamics. *Reviews of Geophysics*, 21(1):41–50, doi:10.1029/RG021i001p00041.

Gander, W. (1990). Algorithms for the polar decomposition. *SIAM Journal on Scientific and Statistical Computing*, 11(6):1102–1115, doi:10.1137/0911062.

Goudarzi, M. A., Cocard, M., and Santerre, R. (2015). Geostrain: An open source software for calculating crustal strain rates. *Computers & Geosciences*, 82:1–12,

doi:10.1016/j.cageo.2015.05.007.

Higham, N. J. (1986). Computing the polar decomposition—with applications. *SIAM Journal on Scientific and Statistical Computing*, 7(4):1160–1174, doi:10.1137/0907079.

Markley, F. L. and Mortari, D. (1999). How to estimate attitude from vector observations. *AIAA/AAS Paper*, pages 99–427.

Osada, E. and Sergieieva, K. (2010). O badaniu zniekształceń modeli transformacji map na podstawie elipsy Tissota – długości, pola lub kąty. *Magazyn geoinformacyjny Geodeta*, (1):176.

Shoemake, K. and Duff, T. (1992). Matrix animation and polar decomposition. In *Proceedings of the conference on Graphics interface*, volume 92, pages 258–264.

Szafarczyk, A. and Gawalkiewicz, R. (2016). Case study of the tensor analysis of ground deformations evaluated from geodetic measurements in a landslide area. *Acta Geodynamica et Geomaterialia*, 13(2):213–222, doi:10.13168/AGG.2015.0003.

Tanaka, M., Wada, S., and Nakamura, M. (2012). *Computational biomechanics: theoretical background and biological/biomedical problems*, volume 3. Springer, Tokyo, doi:10.1007/978-4-431-54073-1.