



Article General Total Least Squares Theory for Geodetic Coordinate Transformations

Yuxin Qin¹, Xing Fang^{1,*}, Wenxian Zeng¹ and Bin Wang²

- ¹ School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China; 2017301610077@whu.edu.cn (Y.Q.); wxzeng@sgg.whu.edu.cn (W.Z.)
- ² College of Geomatics Science and Technology, Nanjing Tech University, Nanjing 211816, China; binwangsgg@njtech.edu.cn
- * Correspondence: xfang@sgg.whu.edu.cn

Received: 16 March 2020; Accepted: 7 April 2020; Published: 9 April 2020



Abstract: Datum transformations are a fundamental issue in geodesy, Global Positioning System (GPS) science and technology, geographical information science (GIS), and other research fields. In this study, we establish a general total least squares (TLS) theory which allows the errors-in-variables model with different constraints to formulate all transformation models, including affine, orthogonal, similarity, and rigid transformations. Through the adaptation of the transformation models to the constrained TLS problem, the nonlinear constrained normal equation is analytically derived, and the transformation parameters can be iteratively estimated by fixed-point formulas. We also provide the statistical characteristics of the parameter estimator and the unit of precision of the control points. Two examples are given, as well as an analysis of the results on how the estimated quantities vary when the number of constraints becomes larger.

Keywords: total least squares; Gauss–Newton algorithm; errors-in-variables; affine/orthogonal/ similarity/rigid transformations; constraints; general algorithm

1. Introduction

Transformations are a frequently encountered procedure in geodesy, Global Positioning System (GPS) science and technology, geographical information science (GIS), and other scientific fields. For example, (1) a 3D similarity transformation is usually applied to transform GPS- (World Geodetic System 84) WGS84-based coordinates to those in a local coordinate system using a bunch of common points with coordinate values in both systems. (2) In GIS, digital data produced by tracing old paper maps over a digitizing tablet need to be converted from the tablet's non-georeferenced plane data into georeferenced plane data that can be georegistered with other digital data layers. (3) For the purpose of monitoring a whole dam, the combining of multiple point clouds from different laser stations is needed by transformations. This process is called registration of the scans/images in photogrammetry and remote sensing. (4) In computer vision, mapping a single shape obtained from one sensor to a single shape obtained from another by computing an appropriate transformation between them makes the image 3D visible.

A common approach to solving the transformation problem is the least squares (LS) adjustment of the transformation parameters from a redundant set of nonlinear forms of the Gauss–Markov (GM) model. However, the GM model assumes that only the coordinates of the target points are random in the observation vector within the GM model. It is obvious that uncertainties from the source coordinate system are missing in the coefficient matrix within the GM model [1,2].

Taking the uncertainties in the coefficient matrix into account, the symmetric transformation problems are referred to as the errors-in-variables (EIV) model, and the LS estimation within the EIV

model is called total least squares (TLS) estimation [3–8]. There are several classes of methods to obtain TLS solutions: 1) Procrustes analysis (e.g., [9,10]); 2) quaternions (e.g., [5,11,12]); (3) the Lagrange approach (e.g., [8,13–16]); and 4) unconstrained optimization (e.g., [3,17,18]). From the numerical point of view, although the analytical method is available for certain special covariance matrices [19], transformation problems are solved by iterative methods in general, for example, the Gauss–Helmert model, the iterative GM model, or the sequential quadratic program (SQP).

Although the aforementioned transformation problems have been successfully addressed, they all focus on the similarity transformations. When we incorporate constraints into the EIV model, all kinds of transformations, including affine, orthogonal, and rigid types, can be formulated. [17] proposed SQP to provide the constrained TLS solution for certain types of transformations. [13] implemented variance component estimation for the EIV model with constraints. However, no one has solved the constrained TLS problem using a Gauss–Newton (GN)-type solver, which is much easier than SQP, and the statistical characteristics of the parameter estimates are straightforwardly available.

In this paper, we propose a constrained TLS algorithm by iteratively using the constrained TLS normal equation. The algorithm is suitable for solving all kinds of transformation problems, either in 2D or in 3D. Furthermore, the statistical characteristics of the parameter estimates are explicitly given by the inverted constrained normal matrix.

The remainder of this paper is organized as follows: First, we give the mathematical formulation of the constrained TLS problem and its relation to the transformation models in Section 2. Second, we solve the TLS problem algebraically using Lagrange multipliers in Section 3. Third, in Section 4, we design the unconstrained and constrained TLS algorithm for all kinds of transformations. Two examples are given in Section 5 to demonstrate the performance of the proposed robust methods in 2D and 3D. Finally, we draw conclusions in Section 6.

2. Adaptation of the Transformation Models to the Constrained/Unconstrained TLS Problem

In the following, we introduce the constrained TLS problem and the transformation models in 2D and 3D, and finally, all types of the coordinate transformations which adapt to the constrained/unconstrained EIV model.

2.1. The Constrained TLS Problem

Let us start with the functional part of the errors-in-variables model with constraints:

$$\mathbf{y}_{n \times 1} - \mathbf{e}_{\mathbf{y}} = \left(\mathbf{A} - \mathbf{E}_{\mathbf{A}}_{n \times u} \right)_{u \times 1} \boldsymbol{\xi}$$
subject to $\mathbf{c}_{s \times 1} (\boldsymbol{\xi}) = 0$
(1)

In the above equation, **y** and **e**_y are the observation vector and its associated random error vector, respectively. The coefficient matrix **A** is random or partly random, and the matrices $\mathbf{E}_{\mathbf{A}}$ are the associated random error matrices. Vector $\boldsymbol{\xi}$ is the unknown parameter vector. The constraints $\mathbf{c}_{\mathbf{A}}(\boldsymbol{\xi}) = 0$ are available.

The stochastic part of the EIV model to describe the statistical properties of all random errors is as follows:

$$\mathbf{e} := \begin{bmatrix} \operatorname{vec}(\mathbf{E}_{\mathbf{A}}) \\ \mathbf{e}_{\mathbf{y}} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_{\mathbf{A}} \\ \operatorname{un\times 1} \\ \mathbf{e}_{\mathbf{y}} \end{bmatrix} \sim \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \sigma_0^2 \mathbf{Q} \right).$$
(2)

The complete error vector **e** is defined by a vectorization operator which reshapes the matrix $\mathbf{E}_{\mathbf{A}}$ to the long vector $\mathbf{e}_{\mathbf{A}}$ by column order. The symbol σ_0^2 represents the unknown unit variance. Matrix **Q** is the non-negative definite cofactor matrix of the error vector **e**.

2.2. Adaptation of 2D Transformations to the Functional Model of the TLS Problem

The 2D affine transformation with six unknown parameters for a single point is introduced as:

$$\begin{bmatrix} \mathbf{x}_{t'} & \mathbf{y}_{t} \end{bmatrix}^{T} \approx \begin{bmatrix} m_{1} \cos \alpha & -m_{2} \sin(\alpha + \varepsilon) \\ m_{1} \sin \alpha & m_{2} \cos(\alpha + \varepsilon) \end{bmatrix} \begin{bmatrix} \mathbf{x}_{s'} & \mathbf{y}_{s} \end{bmatrix}^{T} + \begin{bmatrix} \Delta \mathbf{x} & \Delta y \end{bmatrix}^{T}.$$
 (3)

 $\begin{bmatrix} x_t, y_t \end{bmatrix}$ are the coordinates of the target system, while $\begin{bmatrix} x_s, y_s \end{bmatrix}$ are the coordinates of the source system. m_1 and m_2 are the scale factors, $\alpha + \varepsilon$ and α are the rotation angles, and $\begin{bmatrix} \Delta x & \Delta y \end{bmatrix}$ are the translations. The sign \approx is used as we omit the random errors in the equation for the sake of simple formulation.

When all six parameters are replaced by $\boldsymbol{\xi} = \left[vec^T (\boldsymbol{\Xi}_{2 \times 2}), \begin{bmatrix} \boldsymbol{\xi}_{\Delta x} & \boldsymbol{\xi}_{\Delta y} \end{bmatrix} \right]^T$, the above equation is rewritten as:

$$\begin{bmatrix} \mathbf{x}_{t}, & \mathbf{y}_{t} \end{bmatrix}^{T} \approx \Xi_{2\times 2} \begin{bmatrix} \mathbf{x}_{s}, & \mathbf{y}_{s} \end{bmatrix}^{T} + \begin{bmatrix} \boldsymbol{\xi}_{\Delta x} & \boldsymbol{\xi}_{\Delta y} \end{bmatrix}^{T}$$
(4)

with transformation matrix $\Xi_{2\times 2} = \begin{bmatrix} \xi_{11} & \xi_{12} \\ \xi_{21} & \xi_{22} \end{bmatrix}$. vec^T is the row vectorization operator according to the row order, i.e., the nth row stacks after the (n-1)th row.

Considering all control points, the unconstrained EIV model of the affine transformation in 2D can be given by:

$$\begin{bmatrix} \mathbf{x}_t^T & \mathbf{y}_t^T \end{bmatrix}^T \approx \begin{bmatrix} \mathbf{I}_2 \otimes \begin{bmatrix} \mathbf{x}_s & \mathbf{y}_s \end{bmatrix}, \mathbf{I}_2 \otimes 1 \end{bmatrix} \boldsymbol{\xi}.$$
 (5)

Equation (5) explicitly defines the matrix \mathbf{A} and the vector \mathbf{y} in Eq (1) for all other transformations in 2D.

The operator \otimes denotes the Kronecker Product. $\begin{bmatrix} \mathbf{x}_s & \mathbf{y}_s \end{bmatrix}$ and $\begin{bmatrix} \mathbf{x}_t & \mathbf{y}_t \end{bmatrix}$ are the source x and y coordinate vectors and the target x and y coordinate vectors, respectively, for all control points. The vector 1 denotes the vector of ones with length the control point number.

The orthogonal, similarity, and rigid transformation models for the single point:

$$\begin{bmatrix} \mathbf{x}_{t'} & \mathbf{y}_{t} \end{bmatrix}^{T} \approx \begin{bmatrix} m_{1} \cos \alpha & -m_{2} \sin \alpha \\ m_{1} \sin \alpha & m_{2} \cos \alpha \end{bmatrix} \begin{bmatrix} \mathbf{x}_{s'} & \mathbf{y}_{s} \end{bmatrix}^{T} + \begin{bmatrix} \Delta \mathbf{x} & \Delta \mathbf{y} \end{bmatrix}^{T},$$
(6)

$$\begin{bmatrix} \mathbf{x}_{t}, & \mathbf{y}_{t} \end{bmatrix}^{T} \approx \begin{bmatrix} m\cos\alpha & -m\sin\alpha\\ m\sin\alpha & m\cos\alpha \end{bmatrix} \begin{bmatrix} \mathbf{x}_{s}, & \mathbf{y}_{s} \end{bmatrix}^{T} + \begin{bmatrix} \Delta \mathbf{x} & \Delta \mathbf{y} \end{bmatrix}^{T},$$
(7)

$$\begin{bmatrix} \mathbf{x}_{t'} & \mathbf{y}_{t} \end{bmatrix}^{T} \approx \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} \mathbf{x}_{s'} & \mathbf{y}_{s} \end{bmatrix}^{T} + \begin{bmatrix} \Delta \mathbf{x} & \Delta \mathbf{y} \end{bmatrix}^{T},$$
(8)

are simplified versions of Equation (3). Moreover, it is obvious that Equations (6)–(8) can be reformulated using Equation (3) with three different type constraints, respectively:

$$\xi_{11}\xi_{12} + \xi_{21}\xi_{22} = 0, \tag{9}$$

$$\xi_{11}^2 + \xi_{12}^2 - \xi_{21}^2 - \xi_{22}^2 = 0, \ \xi_{11}\xi_{21} + \xi_{12}\xi_{22} = 0,$$
(10)

$$\xi_{11}^2 + \xi_{12}^2 - \xi_{21}^2 - \xi_{22}^2 = 0, \ \xi_{11}\xi_{21} + \xi_{12}\xi_{22} = 0, \ \xi_{11}^2 + \xi_{12}^2 - 1 = 0.$$
(11)

Equation (5) explicitly defines the matrix \mathbf{A} and the vector \mathbf{y} in Equation (1) for all other transformations in 2D.

When the original unknown parameters are fewer in the transformation models, the number of constraints increases.

2.3. Adaptation of 3D Transformations to the Functional Model of the TLS Problem

In full analogy with the 2D affine transformation model (5), the 3D affine transformation model can be explicitly extended as follows:

$$\begin{bmatrix} \mathbf{x}_t^T & \mathbf{y}_t^T & \mathbf{z}_t^T \end{bmatrix}^T \approx \begin{bmatrix} \mathbf{I}_3 \otimes \begin{bmatrix} \mathbf{x}_s^T & \mathbf{y}_s^T & \mathbf{z}_s^T \end{bmatrix}, \mathbf{I}_3 \otimes 1 \end{bmatrix} \boldsymbol{\xi},$$
(12)

with $\boldsymbol{\xi} = \begin{bmatrix} vec^T (\boldsymbol{\Xi}_{3 \times 3}), \begin{bmatrix} \Delta x & \Delta y & \Delta z \end{bmatrix} \end{bmatrix}^T$.

Equation (12) explicitly defines the matrix \mathbf{A} and the vector \mathbf{y} in equation (1) for all other transformations in 3D.

When the other three kinds of transformations are considered, the three, five, and six constraints:

$$\xi_{11}\xi_{21} + \xi_{12}\xi_{22} + \xi_{13}\xi_{23} = 0, \ \xi_{11}\xi_{31} + \xi_{12}\xi_{32} + \xi_{13}\xi_{33} = 0, \ \xi_{31}\xi_{21} + \xi_{32}\xi_{22} + \xi_{33}\xi_{23} = 0$$
(13)

$$\begin{aligned} \xi_{11}^2 + \xi_{12}^2 + \xi_{13}^2 - \xi_{31}^2 - \xi_{32}^2 - \xi_{33}^2 &= 0, \ \xi_{21}^2 + \xi_{22}^2 + \xi_{23}^2 - \xi_{31}^2 - \xi_{32}^2 - \xi_{33}^2 &= 0 \\ \xi_{11}\xi_{21} + \xi_{12}\xi_{22} + \xi_{13}\xi_{23} &= 0, \ \xi_{11}\xi_{31} + \xi_{12}\xi_{32} + \xi_{13}\xi_{33} &= 0, \ \xi_{31}\xi_{21} + \xi_{32}\xi_{22} + \xi_{33}\xi_{23} &= 0 \end{aligned}$$
(14)

$$\begin{aligned} \boldsymbol{\xi}_{11}^{2} + \boldsymbol{\xi}_{12}^{2} + \boldsymbol{\xi}_{13}^{2} - \boldsymbol{\xi}_{31}^{2} - \boldsymbol{\xi}_{32}^{2} - \boldsymbol{\xi}_{33}^{2} &= 0, \ \boldsymbol{\xi}_{21}^{2} + \boldsymbol{\xi}_{22}^{2} + \boldsymbol{\xi}_{23}^{2} - \boldsymbol{\xi}_{31}^{2} - \boldsymbol{\xi}_{32}^{2} - \boldsymbol{\xi}_{33}^{2} &= 0 \\ \boldsymbol{\xi}_{11}^{2} + \boldsymbol{\xi}_{12}^{2} + \boldsymbol{\xi}_{13}^{2} - 1 &= 0, \ \boldsymbol{\xi}_{11} \boldsymbol{\xi}_{21} + \boldsymbol{\xi}_{12} \boldsymbol{\xi}_{22} + \boldsymbol{\xi}_{13} \boldsymbol{\xi}_{23} &= 0 \\ \boldsymbol{\xi}_{11} \boldsymbol{\xi}_{31} + \boldsymbol{\xi}_{12} \boldsymbol{\xi}_{32} + \boldsymbol{\xi}_{13} \boldsymbol{\xi}_{33} &= 0, \ \boldsymbol{\xi}_{31} \boldsymbol{\xi}_{21} + \boldsymbol{\xi}_{32} \boldsymbol{\xi}_{22} + \boldsymbol{\xi}_{33} \boldsymbol{\xi}_{23} &= 0 \end{aligned}$$
(15)

can be correspondingly incorporated for orthogonal, similarity, and rigid transformations, respectively.

2.4. Adaptation of the Transformation Models to the Stochastic Model of the TLS Problem

In Sections 2.2 and 2.3, we showed that the four kinds of transformation models can be formulated using a constrained or unconstrained EIV model. In order to adapt the stochastic part of the EIV model, we propagate the covariance matrix of the observed coordinates in both the source and target systems to the covariance matrix of the EIV model for all 2D cases:

$$D(\mathbf{e}) = \sigma_0^2 \mathbf{Q} = \begin{bmatrix} \mathbf{F}_{2D} & 0\\ 0 & \mathbf{I} \end{bmatrix} D \left(\begin{bmatrix} \mathbf{x}_s^T & \mathbf{y}_s^T & \mathbf{x}_t^T & \mathbf{y}_t^T \end{bmatrix}^T \right) \begin{bmatrix} \mathbf{F}_{2D} & 0\\ 0 & \mathbf{I} \end{bmatrix}^T$$
(16)

where D denotes the dispersion operator, and the Jacobian matrix for 2D is:

$$\mathbf{F}_{2D} = \frac{\partial \textit{vec}(\mathbf{A})}{\partial \begin{bmatrix} \mathbf{x}_{s}^{T} & \mathbf{y}_{s}^{T} \end{bmatrix}} = \frac{\partial \textit{vec}(\begin{bmatrix} \mathbf{I}_{2} \otimes \begin{bmatrix} \mathbf{x}_{s} & \mathbf{y}_{s} \end{bmatrix}, \mathbf{I}_{2} \otimes 1])}{\partial \begin{bmatrix} \mathbf{x}_{s}^{T} & \mathbf{y}_{s}^{T} \end{bmatrix}}$$

Similarly, we give the stochastic model of the EIV model for all kinds of 3D transformations:

$$D(\mathbf{e}) = \sigma_0^2 \mathbf{Q} = \begin{bmatrix} \mathbf{F}_{3D} & 0\\ 0 & \mathbf{I} \end{bmatrix} D \left(\begin{bmatrix} \mathbf{x}_s^T & \mathbf{y}_s^T & \mathbf{z}_s^T & \mathbf{x}_t^T & \mathbf{y}_t^T & \mathbf{z}_t^T \end{bmatrix}^T \right) \begin{bmatrix} \mathbf{F}_{3D} & 0\\ 0 & \mathbf{I} \end{bmatrix}^T$$
(17)

where D denotes the dispersion operator, and the Jacobian matrix for 3D is:

$$\mathbf{F}_{3D} = \frac{\partial vec(\mathbf{A})}{\partial \left[\begin{array}{cc} \mathbf{x}_{s}^{T} & \mathbf{y}_{s}^{T} & \mathbf{z}_{s}^{T} \end{array} \right]} = \frac{\partial vec(\left[\mathbf{I}_{3} \otimes \left[\begin{array}{cc} \mathbf{x}_{s} & \mathbf{y}_{s} & \mathbf{z}_{s} \end{array} \right], \mathbf{I}_{3} \otimes 1 \right])}{\partial \left[\begin{array}{cc} \mathbf{x}_{s}^{T} & \mathbf{y}_{s}^{T} & \mathbf{z}_{s}^{T} \end{array} \right]}.$$
(18)

3. A Fixed-Point Solution to the Constrained TLS Problem

In order to solve the constrained TLS optimization problem:

min
$$\mathbf{e}^T \mathbf{Q}^{-1} \mathbf{e}$$

subject to $\mathbf{y} - \mathbf{A}\boldsymbol{\xi} + \mathbf{E}_{\mathbf{A}}\boldsymbol{\xi} - \mathbf{e}_{\mathbf{y}} = 0$ (19)
and $\mathbf{c}(\boldsymbol{\xi}) = 0$,

the traditional Lagrange approach [20] is applied as follows:

$$\Phi(\mathbf{e}, \boldsymbol{\lambda}, \boldsymbol{\xi}, \boldsymbol{\mu}) = \mathbf{e}^{T} \mathbf{Q}^{-1} \mathbf{e} + 2\boldsymbol{\lambda}^{T} (\mathbf{y} - \mathbf{A}\boldsymbol{\xi} + \mathbf{E}_{\mathbf{A}}\boldsymbol{\xi} - \mathbf{e}_{\mathbf{y}}) + 2\boldsymbol{\mu}^{T} \mathbf{c}(\boldsymbol{\xi})$$

= $\mathbf{e}^{T} \mathbf{Q}^{-1} \mathbf{e} + 2\boldsymbol{\lambda}^{T} (\mathbf{y} - \mathbf{A}\boldsymbol{\xi} + \mathbf{B}\mathbf{e}) + 2\boldsymbol{\mu}^{T} \mathbf{c}(\boldsymbol{\xi}),$ (20)

where λ and μ are the vectors of the Lagrange multipliers associated with the functional part of the EIV model and the constraints, respectively. The matrix $\mathbf{B}_{n \times n(u+1)} = [\boldsymbol{\xi}^T \otimes \mathbf{I}_n, -\mathbf{I}_n]$ is expressed via Kronecker product operator.

By calculating the first derivative of Equation (21), the necessary condition of the objective function (20) is given as follows:

$$\frac{1}{2} \frac{\partial \Phi}{\partial \xi} \Big|_{\hat{\xi}, \tilde{\mathbf{e}}, \hat{\lambda}, \hat{\mu}} = -\mathbf{A}^T \hat{\boldsymbol{\lambda}} + \tilde{\mathbf{E}}_{\mathbf{A}}^T \hat{\boldsymbol{\lambda}} + \hat{\mathbf{C}}^T \hat{\boldsymbol{\mu}} = 0$$
(21)

$$\frac{1}{2} \frac{\partial \Phi}{\partial \mathbf{e}} \Big|_{\hat{\mathbf{\xi}}, \tilde{\mathbf{e}}, \hat{\boldsymbol{\lambda}}}^{*} = \mathbf{Q}^{-1} \tilde{\mathbf{e}} + \tilde{\mathbf{B}}^{T} \hat{\boldsymbol{\lambda}} = 0$$
(22)

$$\frac{1}{2} \frac{\partial \Phi}{\partial \lambda} \Big|_{\hat{\boldsymbol{\xi}}, \tilde{\mathbf{e}}} = \mathbf{y} - \mathbf{A} \hat{\boldsymbol{\xi}} + \hat{\mathbf{B}} \hat{\mathbf{e}} = 0$$
(23)

$$\frac{1}{2} \frac{\partial \Phi}{\partial \mu} \Big|_{\hat{\xi}} = \mathbf{c} \left(\hat{\xi} \right) = 0 \tag{24}$$

with $\mathbf{C} = \mathbf{C}(\boldsymbol{\xi}) := \partial \mathbf{c}(\boldsymbol{\xi}) / \partial \boldsymbol{\xi}^T$.

From Equation (23), we know that

$$\tilde{\mathbf{e}} = -\mathbf{Q}\tilde{\mathbf{B}}^{T}\boldsymbol{\lambda}.$$
(25)

By inserting Equation (26) into Equation (24), the vector of the Lagrange multipliers is:

$$\hat{\boldsymbol{\lambda}} = \left(\hat{\boldsymbol{B}}\boldsymbol{Q}\hat{\boldsymbol{B}}^{T}\right)^{-1} \left(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{\xi}}\right).$$
(26)

Combining Equations (23) and (27), the error vector can be predicted:

$$\tilde{\mathbf{e}} = -\mathbf{Q}\hat{\mathbf{B}}^{T} \left(\hat{\mathbf{B}} \mathbf{Q} \hat{\mathbf{B}}^{T} \right)^{-1} \left(\mathbf{y} - \mathbf{A} \hat{\boldsymbol{\xi}} \right),$$
(27)

which also implies that the error matrix \tilde{E}_A is predicted.

Equations (22) and (25) are immediately reformulated by:

$$\begin{pmatrix} \tilde{\mathbf{E}}_{\mathbf{A}} - \mathbf{A} \end{pmatrix}^{T} \begin{pmatrix} \mathbf{\hat{B}} \mathbf{Q} \mathbf{\hat{B}}^{T} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{y} - \mathbf{A} \mathbf{\hat{\xi}} \end{pmatrix} + \mathbf{\hat{C}}^{T} \mathbf{\hat{\mu}} = 0$$

$$\mathbf{c} \begin{pmatrix} \mathbf{\hat{\xi}} \end{pmatrix} = 0$$
(28)

which is equivalent to:

$$\begin{pmatrix} \mathbf{A} - \tilde{\mathbf{E}}_{\mathbf{A}} \end{pmatrix}^{T} \begin{pmatrix} \mathbf{\hat{B}}_{\mathbf{Q}} \mathbf{\hat{B}}^{T} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{A} - \tilde{\mathbf{E}}_{\mathbf{A}} \end{pmatrix} \mathbf{\hat{\xi}} + \mathbf{\hat{C}}^{T} \mathbf{\hat{\mu}} = \begin{pmatrix} \mathbf{A} - \tilde{\mathbf{E}}_{\mathbf{A}} \end{pmatrix}^{T} \begin{pmatrix} \mathbf{\hat{B}}_{\mathbf{Q}} \mathbf{\hat{B}}^{T} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{y} - \tilde{\mathbf{E}}_{\mathbf{A}} \mathbf{\hat{\xi}} \end{pmatrix}$$

$$\mathbf{c} \begin{pmatrix} \mathbf{\hat{\xi}} \end{pmatrix} = 0.$$

$$(29)$$

In order to simplify the above equation,

$$\hat{\mathbf{N}} := \left(\mathbf{A} - \tilde{\mathbf{E}}_{\mathbf{A}}\right)^{T} \left(\hat{\mathbf{B}}_{\mathbf{Q}} \hat{\mathbf{B}}^{T}\right)^{-1} \left(\mathbf{A} - \tilde{\mathbf{E}}_{\mathbf{A}}\right)$$
$$\hat{\mathbf{n}} := \left(\mathbf{A} - \tilde{\mathbf{E}}_{\mathbf{A}}\right)^{T} \left(\hat{\mathbf{B}}_{\mathbf{Q}} \hat{\mathbf{B}}^{T}\right)^{-1} \left(\mathbf{y} - \tilde{\mathbf{E}}_{\mathbf{A}} \hat{\boldsymbol{\xi}}\right)$$
(30)

are defined.

Now, we establish the constrained nonlinear normal equation of the constrained TLS problem:

$$\begin{bmatrix} \hat{\mathbf{N}} & \hat{\mathbf{C}}^{T} \\ \hat{\mathbf{N}} & \hat{\mathbf{C}} \\ \hat{\mathbf{C}} & 0 \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\xi}} \\ \hat{\boldsymbol{\mu}} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{n}} \\ \hat{\mathbf{n}} \\ \hat{\mathbf{C}}\boldsymbol{\xi} - \mathbf{c} \begin{pmatrix} \hat{\boldsymbol{\xi}} \\ \hat{\boldsymbol{\xi}} \end{pmatrix} \end{bmatrix}$$
(31)

and the corresponding solution is:

$$\begin{bmatrix} \hat{\boldsymbol{\xi}} \\ \hat{\boldsymbol{\mu}} \\ \hat{\boldsymbol{\mu}} \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{n}} & \hat{\boldsymbol{n}}^{T} \\ \hat{\boldsymbol{n}} & \boldsymbol{C} \\ \hat{\boldsymbol{c}} & \boldsymbol{0} \end{bmatrix}^{-1} \begin{bmatrix} \hat{\boldsymbol{n}} \\ \hat{\boldsymbol{n}} \\ \hat{\boldsymbol{C}}\boldsymbol{\xi} - \boldsymbol{c} \begin{pmatrix} \hat{\boldsymbol{\xi}} \\ \hat{\boldsymbol{\xi}} \end{pmatrix} \end{bmatrix}.$$
(32)

When no constraints are available (e.g., for affine transformation), the estimator of the unknown parameter is

$$\hat{\mathbf{N}}\boldsymbol{\xi}_{u} = \hat{\mathbf{n}}.$$
(33)

The estimator of the unknown parameter can be expressed alternatively as:

$$\hat{\boldsymbol{\xi}} = \mathbf{f}\left(\hat{\boldsymbol{\xi}}\right) = \hat{\boldsymbol{\xi}}_{u} - \hat{\mathbf{C}}^{T}\left(\hat{\mathbf{C}}\hat{\mathbf{N}}^{-1}\hat{\mathbf{C}}^{T}\right)^{-1}\left(\hat{\mathbf{C}}\hat{\boldsymbol{\xi}}_{u} - \hat{\mathbf{C}}\hat{\boldsymbol{\xi}} + \mathbf{c}\left(\hat{\boldsymbol{\xi}}\right)\right)$$
(34)

where $\hat{\boldsymbol{\xi}} = \mathbf{f}(\hat{\boldsymbol{\xi}})$ denotes that the solution is of fixed-point type and can therefore be solved iteratively.

Furthermore, the statistical properties of the estimator can be also approximately given by:

$$\mathbf{Q}_{\hat{\boldsymbol{\xi}}\hat{\boldsymbol{\xi}}} = \hat{\mathbf{N}}^{-1} - \hat{\mathbf{N}}^{-1}\hat{\mathbf{C}}^{T} \left(\hat{\mathbf{C}}\hat{\mathbf{N}}^{-1}\hat{\mathbf{C}}^{T}\right)^{-1}\hat{\mathbf{C}}\hat{\mathbf{N}}^{-1},$$
(35)

$$\hat{\sigma}_0^2 = \frac{\left(\mathbf{y} - \mathbf{A}\hat{\boldsymbol{\xi}}\right)^T \left(\hat{\mathbf{B}} \mathbf{Q} \hat{\mathbf{B}}^T\right)^{-1} \left(\mathbf{y} - \mathbf{A}\hat{\boldsymbol{\xi}}\right)}{n - u + s},$$
(36)

$$D(\hat{\boldsymbol{\xi}}) = \hat{\sigma}_0^2 \mathbf{Q}_{\hat{\boldsymbol{\xi}}\hat{\boldsymbol{\xi}}}.$$
(37)

4. Algorithm Design

In this part, we summarize the developed formulas to present the general algorithm for all kinds of 3D coordinate transformations. The coordinates of the source system $\begin{bmatrix} \mathbf{x}_s & \mathbf{y}_s & \mathbf{z}_s \end{bmatrix}$, the coordinates

Through Equation (12), the coefficient matrix A and the observation vector y can be established as:

$$\mathbf{y} = \begin{bmatrix} \mathbf{x}_t^T & \mathbf{y}_t^T & \mathbf{z}_t^T \end{bmatrix}^T \mathbf{A} = \begin{bmatrix} \mathbf{I}_3 \otimes \begin{bmatrix} \mathbf{x}_s^T & \mathbf{y}_s^T & \mathbf{z}_s^T \end{bmatrix}, \mathbf{I}_3 \otimes 1 \end{bmatrix}$$
(38)

According to Equation (18), we propagate the cofactor matrix \mathbf{Q} of the matrix $vec(\begin{bmatrix} \mathbf{A} & \mathbf{y} \end{bmatrix})$ by the dispersion matrix of the target and source coordinates. The a priori dispersion matrix of the target and source coordinates is usually given by the instrument precision or the precision determined by adjustment in the previous period.

The transformation kinds define the constraints. For affine transformation, no constraints are available. For orthogonal, similarity, and rigid transformations, the constraints are defined according to (13), (14), and (15), respectively.

After the abovementioned preprocessing, we can present the input for the computation: **Input (the available conditions):**

- ✓ Coefficient matrix **A** with dimensions $n \times u$.
- ✓ Observation vector **y** with dimensions $n \times 1$.
- ✓ Cofactor matrix **Q** with dimensions $(u + 1)n \times (u + 1)n$.
- ✓ If constraints exist, multiple quadratic constraints $c(\xi) = 0$ are given.

From the given input, we calculate the initial value of the parameter vector using $\hat{\boldsymbol{\xi}}^{0} = (\mathbf{A}^{T}\mathbf{A})^{-1}\mathbf{A}^{T}\mathbf{y}$. As soon as the initial values for the parameter vector are obtained, **the iterative procedure** starts with the following steps:

- > Create the matrix $\mathbf{\hat{B}}_{n \times n(u+1)} = \begin{bmatrix} \mathbf{\hat{\xi}}^T \otimes \mathbf{I}_{n, -\mathbf{I}_n} \end{bmatrix}$.
- > Calculate the error vector $\tilde{\mathbf{e}}$ according to Equation (28) and reshape the error matrix $\mathbf{E}_{\mathbf{A}}$.

> Create the Jacobian matrix
$$\hat{\mathbf{C}} = \partial \mathbf{c} \left(\hat{\boldsymbol{\xi}}\right) / \partial \hat{\boldsymbol{\xi}}^{T}$$
, if constraints exist.

- > Create the normal matrix and the normal vector.
 - If constraints exist, the normal matrix and the normal vector are given by the left-hand side and the right-hand side of Equation (32), respectively.
 - If no constraints exist, the normal matrix and the normal vector are given by the left-hand side and the right-hand side of Equation (34), respectively.
- Calculate the estimates of the unknown parameter.
 - If constraints exist, we calculate ξ according to Equation (33) or (35).
 - If no constraints exist, we calculate ξ according to Equation (34).

When $\left\| \begin{matrix} \lambda^{i} & \lambda^{i-1} \\ \xi & -\xi \end{matrix} \right\| \le \varepsilon$ (a suitably small positive value), the iteration terminates. Then, we provide

the precision evaluation, that is, we 1) calculate the cofactor matrix of $\hat{\xi}$ by Equation (36); 2) calculate the posterior unit variance $\hat{\sigma}_0$ by Equation (37); and 3) calculate the variance–covariance matrix of $\hat{\xi}$ by Equation (38).

Finally, we present the estimate of the parameter vector $\hat{\boldsymbol{\xi}} := \hat{\boldsymbol{\xi}}^{t}$ and the corresponding variance–covariance matrix as **the final output**. In order to illustrate the entire computation process, we give the following flow chart of the algorithm for all kinds of transformations in Figure 1.



Figure 1. A flow chart of the constrained/unconstrained total least squares algorithm for all kinds of coordinate transformations.

5. Numerical Examples

In this part, two numerical examples are provided to verify the proposed constrained TLS algorithm. In the first example, data from [21] presents a typical photogrammetry problem which consists of the 2-D coordinates that were measured and respectively calibrated of four side fiducial marks in a photograph, as seen in Table 1. Here, the coordinates in the target system x_t , y_t ("Calibrated Coordinates") and the coordinates in the source system x_s , y_s ("Measured Coordinates") are presented in Figure 2 to clarify their positions, and we regard them as equally weighted uncorrelated observations. The similarity transformation for a dataset was adjusted for the EIV model before by [22] and by [6] in two different parameterizations, i.e., either with or without shift/translation parameters, all in 2D. They both need complicated preprocessing, i.e., either the reduction of parameters or the selection of independent random elements within the coefficient matrix. Furthermore, these authors only implemented the similarity transformation.



Table 1. The common points of the 2D transformation.

Figure 2. The positions of the source and target coordinates in the 2D transformations.

By applying Algorithm 1 to the data set of Table 1, the estimated 2D transformation parameters for all kinds of transformations and the TLS objective function were determined and are presented in Table 2. The four parameters are completely different in the transformation matrix $\Xi_{2\times2}$ in the affine and orthogonal transformations, whereas the four parameters are pairwise different in the similarity and rigid transformations. The results successfully correspond with the transformation models, as the affine and orthogonal transformations have different scale factors. The estimated values of the objective functions become larger and larger from the affine transformation to the rigid transformation (see the last row of Table 2), since more and more constraints are considered. The standard deviation of the parameter estimates is given in Table 3. The table reveals that the precision of the parameter estimates becomes higher with more constraints. However, in the last row of Table 3, the transformation models expressed by larger numbers of constraints do not provide larger estimates of the unit standard deviation, which depends on both the redundant number and the objective function values.

	Parameter				
	Affine	Orthogonal	Similarity	Rigid	
ξ ₁₁	0.99902905	0.99902817	0.99900748	0.99915487	
ξ ₁₂	0.04111867	0.04109721	0.04109806	0.04110413	
$\hat{\boldsymbol{\xi}}_{21}$	-0.04107747	-0.04109892	-0.04109806	-0.04110413	
$\hat{\boldsymbol{\xi}}_{22}^{}$	0.99898590	0.99898678	0.99900748	0.99915487	
$\Delta \hat{\mathbf{x}}$	-141.26879	-141.26546	-141.26279	-141.28363	
$\Delta \hat{y}$	-143.93120	-143.92843	-143.93164	-143.95288	
Objective	0.00061868	0.00063141	0.00064325	0.00124379	

Table 2. Estimates of the 2D transformation parameters.

Table 3. Precision	n evaluation	of the 2D	transformation	parameters
--------------------	--------------	-----------	----------------	------------

	Standard Deviation					
	Affine	Orthogonal	Similarity	Rigid		
ξ̂ ₁₁	1.4969×10^{-4}	1.2342×10^{-4}	7.6328×10^{-5}	3.9027×10^{-6}		
ξ ₁₂	1.4974×10^{-4}	8.7393×10^{-5}	7.6328×10^{-5}	9.4866×10^{-5}		
ξ ₂₁	1.4969×10^{-4}	8.7397×10^{-5}	7.6328×10^{-5}	$9.4866 imes 10^{-5}$		
ξ ₂₂	1.4974×10^{-4}	1.2346×10^{-4}	7.6328×10^{-5}	3.9027×10^{-6}		
Δx̂	3.2661×10^{-2}	2.3286×10^{-2}	1.7817×10^{-2}	1.7641×10^{-2}		
Δŷ	3.2661×10^{-2}	2.4474×10^{-2}	1.7817×10^{-2}	$1.7445 imes 10^{-2}$		
$\hat{\sigma}_0$	0.017588	0.014508	0.012681	0.015772		

In the second example, a dataset for 3D transformations, shown in Table 4, was given by the Coordinate Systems Analysis Team (CSAT) at the National Geospatial-Intelligence Agency [9]. CSAT preserves and releases a set of datum transformation parameter estimates for different countries via the CSAT website. Felus and Burtch provided the adjustment result for similarity transformation based on Procrustes analysis on the assumption that the covariance matrix must be expressed by the Kronecker product. Here, we implemented our proposed algorithm for all kinds of transformations by applying the dataset in Table 4. It is important to note that our algorithm can adjust the dataset with an arbitrary covariance matrix, which is beyond the inherent assumption in Procrustes analysis.

Table 4. The common points of the 3D transformation.

_							
	Point No.	x _s	y _s	zs	x _t	y _t	z _t
	80,601	5,234,251.25	905,003.2011	3,518,869.674	5,233,991.482	905,003.106	3,519,305.459
	32,127	5,218,851.932	919,148.9749	3,537,928.348	5,218,595.021	919,152.324	3,538,363.627
	80,600	5,220,818.669	772,128.3613	3,569,828.606	5,220,565.466	772,130.563	3,570,253.01
	32,136	5,148,067.252	803,912.306	3,668,491.426	5,147,806.722	803,921.322	3,668,928.371
	80,598	5,081,676.23	771,786.8122	3,765,023.787	5,081,410.788	771,799.426	3765460.689
	80,597	5,022,479.06	955 <i>,</i> 283.5487	3,801,754.143	5,022,218.176	955,297.254	3,802,185.975

With the dataset in Table 4, four types of 3D transformation models were adjusted for the six control points. The estimation results for the affine, orthogonal, similarity, and rigid transformations are presented in Table 5, Table 6, Table 7, and Table 8, respectively. The results indicate that the translation parameters differ significantly among the distinct transformation models, whereas the estimated elements within the transformation matrix change slightly, which matches the precision

analysis in Table 9. The other analysis results are almost the same as the 2D results in that the values of the objective function become larger with more constraints, the precision is higher with more constraints, and the variation of the estimates $\hat{\sigma}_0$ has no rule.

Shift Parameters		The	Transformation N	latrix
Δâ	4274.5307	0.999438051	-0.000101814	-0.000425541
$\Delta \hat{y}$	-5094.8874	0.000622535	1.000112976	0.000493015
$\Delta \hat{z}$	-17,013.5695	0.002199299	0.000407742	1.001581580
Objective Function			58.5720	

Table 5. Estimates of the 3D affine transformation parameters.

Table 6.	Estimates of the	3D orthogonal	transformation	parameters.
		0		1

Shift Parameters		The T	Transformation M	latrix
Δâ	-1956.3996	1.000224798	0.000041651	0.000137955
$\Delta \hat{y}$	168.5691	-0.000041663	0.999993142	0.000016147
$\Delta \hat{z}$	1495.9485	-0.000137998	-0.000016154	0.999907421
Objective Function			85.6586	

Table 7. Estimates of the 3D similarity transformation parameters.

Shift Parameters		The T	ransformation N	latrix
Δâ	-293.3670	1.000010668	0.000021228	-0.000010763
$\Delta \hat{y}$	40.7974	-0.000021228	1.000010668	0.000018196
$\Delta \hat{z}$	354.7273	0.000010763	-0.000018196	1.000010668
Objective Function			115.2651	

Table 8. Estimates of the 3D rigid transformation parameters.

Shift Parameters		The 🛛	Fransformation N	/latri x
Δâ	-238.3801	1.000000000	0.000021228	-0.000010763
$\Delta \hat{y}$	49.9133	-0.000021228	1.000000000	0.000018196
$\Delta \hat{z}$	393.5986	0.000010763	-0.000018196	1.000000000
Objective Function			123.4189	

 Table 9. Precision evaluation of the 3D transformation parameters.

	Standard Deviation						
	Affine	Orthogonal	Similarity	Rigid			
ξ ₁₁	1.5382×10^{-3}	1.3051×10^{-4}	1.2094×10^{-5}	4.7351×10^{-10}			
$\hat{\boldsymbol{\xi}}_{12}$	2.7840×10^{-4}	2.3529×10^{-5}	2.1435×10^{-5}	2.1236×10^{-5}			
$\hat{\boldsymbol{\xi}}_{13}$	1.1019×10^{-3}	9.3074×10^{-5}	1.3800×10^{-5}	1.3672×10^{-5}			
$\hat{\boldsymbol{\xi}}_{21}$	1.5387×10^{-3}	2.3536×10^{-5}	2.1436×10^{-5}	2.1236×10^{-5}			
$\hat{\boldsymbol{\xi}}_{22}$	2.7849×10^{-4}	2.4293×10^{-5}	1.2094×10^{-5}	3.6525×10^{-10}			
$\hat{\boldsymbol{\xi}}_{23}$	1.1022×10^{-3}	1.6879×10^{-5}	1.7551×10^{-5}	1.7388×10^{-5}			
$\hat{\boldsymbol{\xi}}_{31}$	1.5399×10^{-3}	9.3131×10^{-5}	1.3800×10^{-5}	1.3672×10^{-5}			
$\hat{\boldsymbol{\xi}}_{32}$	2.7869×10^{-4}	1.6878×10^{-5}	1.7551×10^{-5}	1.7388×10^{-5}			
$\hat{\boldsymbol{\xi}}_{33}$	1.1031×10^{-3}	6.8095×10^{-5}	1.2094×10^{-5}	3.4510×10^{-10}			
Δâ	1.2180×10^4	1.0182×10^3	82.2330	53.1347			
Δŷ	1.2184×10^4	1.6745×10^2	1.5756×10^{2}	1.5576×10^{2}			
Δź	1.2193×10^4	7.2344×10^{2}	85.3863	72.4568			
$\hat{\sigma}_0$	3.1244	3.0851	3.2371	3.2070			

6. Conclusions and Outlook

In this contribution, we presented transformation models in the context of the TLS method, developed the corresponding algorithm based on constrained nonlinear normal equations, and provided a statistical assessment of the TLS adjustment results, including the cofactor matrix of the parameter estimator and the a posteriori variance factor.

In the adaptation of the transformation problems to the EIV model, we explicitly expressed the functional and stochastic models by the source and target coordinates and emphasized the differences between the transformations distinguished only by the quadratic constraints. The adaptations to 2D and 3D are quite similar. In particular, the structure of the matrix *A* and the vector **y** need to be enlarged by the z coordinates in the assigned place. For the adaptation, it is important to note that the number of constraints equals the number of the transformation matrix minus the number of independent parameter numbers. The number of constraints fixes the degree of freedom in the model.

After formulating the transformation models using an unconstrained or constrained EIV model, Lagrange multipliers were applied to provide the first-order necessary conditions of the TLS optimization. After some rearrangements, the constrained nonlinear normal equations were established, based on which the Newton-type iterative procedure could be implemented. The further advantage of the formulation of the constrained nonlinear normal equations is that one can explicitly compute the cofactor matrix and the variance factor, unlike with other existing methods, e.g., the sequential quadratic program.

We applied the proposed algorithm to selected examples so as to present and explain the adjustment results of all transformations with regard to the objective function value and statistical characteristics. We showed that with more available constraints, the objective function values are larger and the cofactors of the parameter estimates are smaller. The numerical results correspond to the theoretical inference.

This algorithm is not only valid for the case of many geodetic datum conversion problems, but also for other applications (photogrammetry, GIS, etc.) where the scale changes may be different or be fixed to one, which will justify the use of suitable constraints within the EIV model.

Furthermore, we hope that the discussed model and the developed algorithm contribute to convincing many—not only geodetic—researchers that the benefits arising from the use of orthogonal regression analysis outweigh the additional effort. From the methodological point of view, our TLS estimation can be generalized to any M- or L-type estimation, which will be promising in robustifying data processing for large data sets such as point clouds.

Author Contributions: Y.Q.: Methodology and Edit; X.F.: Supervision and Idea; W.Z.: Methodology and Funding support; B.W.: Coding. All authors have read and agreed to the published version of the manuscript.

Funding: We greatly thank three anonymous reviewers for their valuable comments. This research was supported by the National Natural Science Foundation of China (41774009, 41674002) and the Natural Science Foundation of Jiangsu Province (BK20180720).

Conflicts of Interest: The authors declare no conflicts of interest

References

- 1. Fang, X. Weighted Total Least Squares Solutions for Applications in Geodesy. Ph.D. Thesis, Leibniz University Hannover, Hannover, Germany, 2011.
- Fang, X. Weighted Total Least Squares: Necessary and sufficient conditions, fixed and random parameters. J. Geod. 2013, 87, 733–749. [CrossRef]
- 3. Fang, X. On non-combinatorial weighted Total Least Squares with inequality constraints. *J. Geod.* **2014**, *88*, 805–816. [CrossRef]
- 4. Golub, G.; Van Loan, C. An analysis of the Total least –squares problem. *Siam J. Numer. Anal.* **1980**, 17, 883–893. [CrossRef]
- 5. Mercan, H.; Akyilmaz, O.; Aydin, C. Solution of the weighted symmetric similarity transformations based on quaternions. *J. Geod.* **2018**, *2*, 1–18. [CrossRef]

- 6. Neitzel, F. Generalization of total least-squares on example of unweighted and weighted 2D similarity transformation. *J. Geod.* **2010**, *84*, 751–762. [CrossRef]
- Schaffrin, B.; Wieser, A. On weighted total least-squares adjustment for linear regression. J. Geod. 2008, 82, 415–421. [CrossRef]
- 8. Wang, B.; Liu, C.; Fang, X.; Chen, W.J. A universally efficient algorithm and precision assessment for seamless 3D similarity transformation. *Meas. Sci. Technol.* **2020**. [CrossRef]
- 9. Felus, Y.A.; Burtch, R.C. On symmetrical three-dimensional datum conversion. *GPS Solut.* **2009**, *13*, 65. [CrossRef]
- 10. Grafarend, E.; Awange, J.L. Nonlinear analysis of the three-dimensional datum transformation [conformal group C7(3)]. *J. Geod.* **2003**, *77*, 66–76. [CrossRef]
- 11. Shen, Y.Z.; Chen, Y.; Zheng, D.H. A quanternion-based geodetic datum transformation algorithm. *J. Geod.* **2006**, *80*, 233–239. [CrossRef]
- 12. Zeng, H.; Fang, X.; Chang, G.; Yang, R. A dual quaternion algorithm of the Helmert transformation problem. *Earth Planets Space* **2018**, *70*, 26. [CrossRef]
- 13. Amiri-Simkooei, A. Parameter estimation in 3D affine and similarity transformation: Implementation of variance component estimation. *J. Geod.* **2018**, *92*, 1285–1297. [CrossRef]
- 14. Chang, G. On least-squares solution to 3D similarity transformation problem under Gauss–Helmert model. *J. Geod.* **2015**, *89*, 573–576. [CrossRef]
- 15. Li, B.; Shen, Y.; Zhang, X.; Li, C.; Lou, L. Seamless multivariate affine error-in-variables transformation and its application to map rectification. *Int. J. Geogr. Inf. Sci.* **2013**, *27*, 1572–1592. [CrossRef]
- 16. Wang, B.; Yu, J.; Liu, C.; Li, M.F.; Zhu, B. Data snooping algorithm for universal 3D similarity transformation based on generalized EIV model. *Measurement* **2018**, *119*, 56–62. [CrossRef]
- 17. Fang, X. Weighted total least-squares with constraints: A universal formula for geodetic symmetrical transformations. *J. Geod.* **2015**, *89*, 459–469. [CrossRef]
- 18. Kanatani, K.; Niitsuma, H. Optimal computation of 3-D similarity: Gauss–Newton vs. Gauss–Helmert. *Comput. Stat. Data Anal.* 2012, *56*, 4470–4483. [CrossRef]
- Akyilmaz, O. Total Least Squares Solution of Coordinate Transformation. Surv. Rev. 2007, 39, 68–80. [CrossRef]
- 20. Koch, K.R. Parameter Estimation and Hypothesis Testing in Linear Models, 2nd ed.; Springer: Berlin/Heidelberg, Germany; New York, NY, USA, 1999.
- 21. Mikhail, E.M.; Bethel, J.S.; McGlone, C.J. *Introduction to Modern Photogrammetry*; Wiley: New York, NY, USA; Chichester, UK, 2001.
- 22. Felus, Y.A.; Schaffrin, B. Performing similarity transformations using the Errors-in-Variables Model. In Proceedings of the ASPRS Annual Conference, Baltimore, MD, USA, 7–11 March 2005.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).