Journal of Electronic Imaging 13(3), 492–503 (July 2004).

# Direct and specific least-square fitting of hyperbolæ and ellipses

**Paul O'Leary**
Institute for Automation
Christian Doppler Laboratory for Sensor and Measurement Systems
Peter-Tunner-Strasse 27
Leoben, Austria
E-mail: automation@unileoben.ac.at

**Paul Zsombor-Murray**
McGill University
Center for Intelligent Machines
817 Sherbrooke Street West
Montréal, Québec, Canada

**Abstract.** *A new method based on quadratic constrained least-mean-square fitting to simultaneously determine both the best hyperbolic and elliptical fits to a set of scattered data is presented. Thus a linear solution to the problem of hyperbola-specific fitting is revealed for the first time. Pilu's method to fit an ellipse (with respect to distance) to observed data points is extended to select, without prejudice, both ellipses and hyperbolæ as well as their degenerate forms as indicated by optimality with respect to the algebraic distance. This novel method is numerically efficient and is suitable for fitting to dense datasets with low noise. Furthermore, it is deemed highly suited to initialize a better but more computationally costly least-square minimization of orthogonal distance. Moreover, Grassmannian coordinates of the hyperbolæ are introduced, and it is shown how these apply to fitting a prototypical hyperbola. Two new theorems on fitting hyperbolæ are presented together with rigorous proofs. A new method to determine the spatial uncertainty of the fit from the eigen or singular values is derived and used as an indicator for the quality of fit. All proposed methods are verified using numerical simulation, and working MATLAB® programs for the implementation are made available. Further, an application of the methods to automatic industrial inspection is presented.* © *2004 SPIE and IS&T.*
[DOI: 10.1117/1.1758951]

## 1 Introduction

The fitting of hyperbolæ and ellipses is a far reaching, general pattern recognition and image processing task; for example, in x-ray diffraction imaging, the centers of the atoms lie on hyperbolæ. We present the first known linear solution to hyperbolæ-specific fitting. In the past, nonlinear optimization techniques have been used to solve this problem. Furthermore, the method simultaneously delivers both the best hyperbolic and elliptical solutions from a single reduction of the scatter matrix.

The literature contains much work on fitting of ellipses[1–4] and implicit curves to observed data points. However, hyperbolæ seem to have been neglected. Pilu, Fitzgibbon, and Fisher[5,6] introduced ellipse-specific direct least-square fitting. Later, Halir and Flusser[7] improved the numerical stability and efficiency of this method.

In this research, the work in Refs. 5, 6, and 7 is extended to simultaneously and specifically determine both the best hyperbolic and elliptical fits to scattered data. The focus of this work is on the performance of the hyperbolic fitting, for the following reasons.

1. This is the first known linear solution to fitting a hyperbola to scattered data.
2. The performance of the ellipse fitting is identical to the solution presented by Halir and Flusser.[7] It does not seem necessary to reproduce these results here.

*A-priori* knowledge of the general shape of the object being measured is commonly available in automatic inspection. The Grassmannian coordinates of the hyperbola are introduced here to enable the integration of such *a-priori* knowledge into the fitting procedure. Two new theorems on fitting hyperbolæ are presented, together with rigorous proofs. Furthermore, the proposed methods are verified using both synthetic and experimental datasets.

Automatic industrial inspection applications place additional requirements on the fitting and modeling methods being developed.

1. Usually observation and fitting are unsupervised. The entire procedure must be automated because product processing or some corrective action may have to be taken based on some results of the fit. There is no possibility for an operator to intervene when such conditions occur.
2. The algorithms must be robust and fail-safe when confronted with spurious data. They must verify that

Paper ORNL-014 received Aug. 4, 2003; accepted for publication Mar. 4, 2004.
1017-9909/2004/$15.00 © 2004 SPIE and IS&T.

the expected geometric object reliably describes the data and exhibits numerical stability if the data convey ambiguity.

3. Fitting is carried out under real-time constraints, requiring a tradeoff between speed and accuracy.
4. A dense vector of data points, typically several hundred, is available. The data are only moderately noisy.
5. Image processing accuracy needs to match only that achievable by the subsequent automatic product processing or defect correction system.

To support these requirements, a new and numerically efficient method of calculating the spatial uncertainty of the fit has been developed. Least-square fitting of the algebraic distance is a poor approximation of minimized normal distance in strongly scattered data.[3] This work extends the linear fitting of implicit equations to determine the spatial uncertainty associated with the fit from the eigen or singular values.

## 2 Geometric Background

The following section presents conics in homogeneous form and the necessary properties required to specifically fit hyperbolæ and ellipses.

### 2.1 *Homogeneous Coordinates and the Projective Plane*

The homogeneous coordinates of a point in the 2-D projective plane are,

$$\mathbf{p}=[x \quad y \quad w]^T \equiv [\lambda x \quad \lambda y \quad \lambda w]^T. \tag{1}$$

Now consider the homogeneous equation of a line $\mathbf{l}_e$,

$$\mathbf{l}_e = l_1 x + l_2 y + l_3 w \equiv \lambda l_1 x + \lambda l_2 y + \lambda l_3 w \equiv \lambda [l_1 \quad l_2 \quad l_3][x \quad y \quad w]^T = 0. \tag{2}$$

All points $\mathbf{p}_\infty = [x\ y\ 0]^T$ lie on the line at infinity $\mathbf{l}_\infty = [0\ 0\ 1]$.

The points of intersection, ostensibly at $\infty$, *i.e.*, $w=0$, of a conic and the line at infinity $\mathbf{l}_\infty$, will be used to advantage when selecting the specific type of conic fitting.

### 2.2 *Conics*

Quadratic forms in the plane, *i.e.*, conics,

$$k \triangleq \mathbf{p}^T \mathbf{K} \mathbf{p} = [x \quad y \quad w] \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}, \tag{3}$$

may be defined by the symmetric matrix **K**. The conic has three properties, which are invariant to Euclidean transformations,

$$i_1 = |K|, \quad i_2 = \left| \begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix} \right| = ac - \frac{b^2}{4}, \quad \text{and} \quad i_3 = a - c. \tag{4}$$

These can be understood by considering the intersection of a conic with the line at infinity.

The homogeneous conic equation can be obtained by expanding Eq. (3),

$$k \triangleq ax^2 + bxy + cy^2 + dxw + eyw + fw^2 = 0. \tag{5}$$

Intersection between the conic and the line at infinity is obtained with Eq. (5) and $w=0$ to yield Eq. (6). The closed conics, *i.e.*, ellipses and circles, their special case, have no real points of intersection, whereas hyperbolæ and parabolæ do.

$$k_\infty = ax^2 + bxy + cy^2 = 0. \tag{6}$$

Solving Eq. (6) as $x=x(y)$ produces,

$$x = \left[ \frac{-b \pm (b^2 - 4ac)^{1/2}}{2a} \right] y, \tag{7}$$

which can be rearranged as Eq. (8):

$$\{2ax + [b + (b^2 - 4ac)^{1/2}]y\}\{2ax + [b - (b^2 - 4ac)^{1/2}]y\} = 0 = \mathbf{l}_1 \mathbf{l}_2. \tag{8}$$

The factors represent two lines with the following equations,

$$\mathbf{l}_1 \triangleq 2ax + [b + (b^2 - 4ac)^{1/2}]y = 0 \quad \text{and}$$

$$\mathbf{l}_2 \triangleq 2ax + [b - (b^2 - 4ac)^{1/2}]y = 0. \tag{9}$$

These two lines, which intersect at the origin $\mathbf{o}[001]$, are parallel to the conic asymptotes. The nature of the two lines, and with this the nature of the conic, depend on the value of $b^2 - 4ac$:

$$b^2 - 4ac \times \begin{cases} =0 & \text{real parallel asymptotes} \Rightarrow \text{parabolic conic} \\ >0 & \text{real asymptotes} \Rightarrow \text{hyperbolic conic} \\ <0 & \text{complex asymptotes} \Rightarrow \text{elliptical conic} \end{cases}. \tag{10}$$

This property, one of the invariants under Euclidean transformation, can be used to identify the type of conic from any form of its general equation, *i.e.*, independent of position and orientation.

Later it is seen that it is necessary to determine if the conic is degenerate or not *proper*. This is indicated if the numerical rank $R(\mathbf{K}, \epsilon)$ is deficient.

$$R(\mathbf{K}, \epsilon) = \begin{cases} 3 & \text{indicates a proper conic} \\ 2 & \text{two intersecting lines} \\ 1 & \text{two colinear lines} \end{cases}. \tag{11}$$

A metric or norm must be introduced as a measure of rank. For this purpose, singular-value decomposition is applied to the conic matrix **K**,

$$\mathbf{K} \overset{svd}{\mapsto} [\mathbf{U},\mathbf{S},\mathbf{V}], \tag{12}$$

where, **S** is the diagonal matrix containing the singular values. These correspond to the two-norm distances of the column vectors of **V** from the null space of **K**. The matrix **S** is normalized on the largest singular value **S**[1,1], thus

$$\mathbf{\Lambda} = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & \dfrac{s[n,n]}{s[1,1]} \end{bmatrix} = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & \lambda[n,n] \end{bmatrix}. \tag{13}$$

The numerical rank is defined as $r_\epsilon$ such that,

$$\lambda_1 > \lambda_2 \cdots > \lambda_{r_\epsilon} > \epsilon > \cdots \lambda_n, \tag{14}$$

whereby $\epsilon$ is a small predefined number related to the numerical accuracy of the computational system being used.

## 3 Simultaneous Fitting of Hyperbolæ and Ellipses

The equation of the conic, dehomogenized with $w=1$, can be formed as the product of a row and column vector.

$$k = [x^2 \quad xy \quad y^2 \quad x \quad y \quad 1][a \quad b \quad c \quad d \quad e \quad f]^T. \tag{15}$$

The vector of quadratic point forms or *design vector* $\mathbf{d} = [x^2\ xy\ y^2\ x\ y\ 1]$ describes the fundamental structure of the curve being considered, in this case a conic, and the coefficient vector $\mathbf{z} = [a\ b\ c\ d\ e\ f]^T$ determines the conic type, *e.g.*, hyperbole, *etc.* The design **d** and coefficient **z** vectors correspond to the dual Grassmannian and Grassmannian coordinates of the geometric objects, respectively.

Consider that, in general, points $\mathbf{p}_i = [x_i, y_i, 1]$ lie near but not on the curve being fitted. Therefore, such point coordinates do not exactly satisfy the conic equation but rather yield a (small) residual $r_i$ called the algebraic distance.

$$r_i \triangleq [x_i^2 \quad x_i y_i \quad y_i^2 \quad x_i \quad y_i \quad 1][a \quad b \quad c \quad d \quad e \quad f]^T. \tag{16}$$

Note, however, this is *not* the normal distance of the point to the curve.

Given $n$ points,

$$\begin{bmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^2 & x_n y_n & y_n^2 & x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \begin{bmatrix} r_i \\ \vdots \\ r_n \end{bmatrix} = \mathbf{Dz} = \mathbf{r}. \tag{17}$$

**D** is called the design matrix and **r** is the vector of residuals.

The aim of least-square fitting is to determine the values of the coefficients, *i.e.*, the coefficient vector, which minimizes the sum of the square of the errors. This vector is written

$$\hat{\mathbf{z}} = \arg\min_{\mathbf{z}} \left\{ \sum_{i=1}^{n} F(\mathbf{z}, \mathbf{p}_i) \right\}. \tag{18}$$

Bookstein[2] in a seminal work showed that the least-square fitting problem under a constraint could be solved using generalized eigenvectors. Pilu and Fitzgibbon took advantage of this method to solve ellipse-specific direct least-square fitting by solving the generalized eigenvector problem,

$$\mathbf{D}^T\mathbf{Dz} = \mathbf{Sz} = \lambda\mathbf{Cz}, \tag{19}$$

subject to the constraint,

$$b^2 - 4ac = \mathbf{z}^T\mathbf{Cz} = \mathbf{z}^T \begin{bmatrix} 0 & 0 & -2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{z} < 0, \tag{20}$$

thus admitting a unique elliptical solution. This constraint is clearly equivalent to selecting the case of an ellipse in Eq. (10). An improved numerical calculation method based on block decomposition of the constraint and scatter matrix, together with a more stable procedure to select the correct solution, were presented by Halir and Flusser,[7] where there were no changes or extensions to the principal nature of the solution.

### 3.1 *Extension to Hyperbolæ*

The above method actually provides three valid solutions; this has been overlooked in the past. As a first step. rigorous proof is presented to show that the eigenvector problem yields one elliptical and two hyperbolic solutions to the conic fitting problem. Further, there is a symmetry between only one of the hyperbolic solutions and the elliptical solution. This enables the unique identification of the correct solutions. The following two lemmas are required for this proof,

*Lemma 1.* *The sign of the generalized eigenvalues of*

$$\mathbf{D}^T\mathbf{Dz} = \mathbf{Sz} = \lambda\mathbf{Cz}, \tag{21}$$

*are the same as those of the matrix* **C**, *up to a permutation of the indices.*

*Lemma 2.* *If* $(\lambda_i, z_i)$ *is a solution of the eigen system,*

$$\mathbf{D}^T\mathbf{Dz} = \mathbf{Sz} = \lambda\mathbf{Cz}, \tag{22}$$

*then:* $sign(\lambda_i) = sign(\mathbf{z}_i^T\ \mathbf{C}\ \mathbf{z})$.

See Golub and Van Loan[8] for proofs of these lemmas. Now we formulate our postulate more formally.

*Theorem 1.* *The solution of the conic fitting problem defined by the generalized eigen system,*

$$\mathbf{D}^T\mathbf{D}\mathbf{z}=\mathbf{S}\mathbf{z}=\lambda\mathbf{C}\mathbf{z}, \tag{23}$$

*subject to the constraint matrix* $\mathbf{C}$ *defining the constraint* $b^2-4ac<0$, *delivers three nontrivial solutions: one and only one elliptical and two hyperbolic solutions.*

*Proof.* Since the nonzero eigenvalues of $\mathbf{C}$ are $\lambda_{\mathbf{C}}=\{-2,1,2\}$, from lemma 1, there is one and only one negative eigenvalue ${}_n\lambda_i<0$, and two positive eigenvalues ${}_p\lambda_i>0$, associated with the solutions $\mathbf{z}_i$. Since the equations are homogeneous, both the positive and negative eigenvalues correspond to eigenvectors, which are valid solutions. Then according to lemma 2, there are two cases to be considered.

**Negative eigenvalue** ${}_n\lambda_i<0$: the constraint $\mathbf{z}_i^T\mathbf{C}\mathbf{z}=b^2-4ac$ is negative, and $z_i$ is a set of coefficients representing an ellipse.

**Positive eigenvalues** ${}_p\lambda_i>0$: the constraint $\mathbf{z}_i^T\mathbf{C}\mathbf{z}=b^2-4ac$ is positive, and $z_i$ is a set of coefficients representing a hyperbola; there are two such solutions.

It is clear that there are two hyperbolic solutions, since the constraint only defines that two pairs of real asymptotes must exist. The solution can never be a parabola, since this would require $b^2-4ac=0$.

There are three eigenvectors, one obtained with the negative eigenvalue, and two with the positive ones. However, Halir and Flusser pointed out that datasets with very low noise may have eigenvalues approaching zero. Small numerical perturbations thus lead to sign ambiguity. Such eigenvalues cannot be reliably used to assign solutions. A numerically more stable approach is to evaluate the following constraint for each of the three solutions,

$$\kappa_i=b_i^2-4a_ic_i, \quad i=1,2,3, \tag{24}$$

this yields three values, each of which corresponds to one of the eigenvalues of the constraint matrix. Note: there are two symmetric eigenvalues of the constraint matrix $\{-2,2\}$; these correspond to the optimal elliptical and hyperbolic solutions, respectively. The third eigenvalue $\lambda=1$ also represents a hyperbola. However, it does not minimize the error as desired, but maximizes it.

The symmetry of the hyperbolic and elliptical solutions can be used to identify the two optimal solutions. With strongly scattered data, the solutions are not perfectly symmetric.

*Corollary 1.* *The eigenvector* $\mathbf{z}_i$ *associated with the negative constraint coefficient* $\kappa_i$ *is the best elliptical fit, while the eigenvector* $\mathbf{z}_i$ *associated with the positive constraint coefficient* $\kappa_i$ *with the value nearest to that of the elliptical solution is the best hyperbolic fit.*

This corollary takes advantage of the symmetric nature of the fitting problem, *i.e.*, fitting a hyperbola and an ellipse is effectively the same problem with only a change in sign of an eigenvalue.

In the case of the hyperbola, the condition number of the conic matrix is then used to determine if the hyperbola degenerates into the two lines of its asymptotes, indicating that a rectangular cross section with sharp, not rounded, corners has been identified.

### 3.2 *Numerical Implementation*

Prior to starting the fit procedure, a translation is applied to place the centroid of the data at the origin, then isotropic scaling is made to ensure that the root mean square distance of the points to the origin is $\sqrt{2}$. The matrix corresponding to this similarity transform is:

$$\mathbf{S}=\begin{bmatrix} s & 0 & -s\overline{x} \\ 0 & s & -s\overline{y} \\ 0 & 0 & 1 \end{bmatrix}, \tag{25}$$

where,

$$s=\frac{n\sqrt{2}}{\sum_{i=1}^{n}[(x_i-\overline{x})^2+(y_i-\overline{y})]^{1/2}}. \tag{26}$$

After fitting has been completed, the conic, represented by its coefficient matrix, is transformed back to the original data position and scaling,

$$\mathbf{K}^\star=\mathbf{S}^T\mathbf{K}\mathbf{S}. \tag{27}$$

This improves the numerical stability of the fit result.[9,10]

The design matrix is partitioned into its quadratic and linear submatrices,

$$\mathbf{D}_1=\begin{bmatrix} x_1^2 & x_1y_1 & y_1^2 \\ \vdots & \vdots & \vdots \\ x_n^2 & x_ny_n & y_n^2 \end{bmatrix} \quad \text{and} \quad \mathbf{D}_2=\begin{bmatrix} x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{bmatrix}. \tag{28}$$

Three scatter matrices are defined,

$$\mathbf{S}_1=\mathbf{D}_1^T\mathbf{D}_1, \quad \mathbf{S}_2=\mathbf{D}_1^T\mathbf{D}_2, \quad \text{and} \quad \mathbf{S}_3=\mathbf{D}_2^T\mathbf{D}_2. \tag{29}$$

The constraint matrix is reduced to,

$$\mathbf{C}_1=\begin{bmatrix} 0 & 0 & -2 \\ 0 & 1 & 0 \\ -2 & 0 & 0 \end{bmatrix}, \tag{30}$$

and the coefficient matrix is partitioned into,

$$\alpha_1=[a \quad b \quad c]^T \quad \text{and} \quad \alpha_2=[d \quad e \quad f]^T. \tag{31}$$

The fitting problem is thus carried out in five well-defined steps.

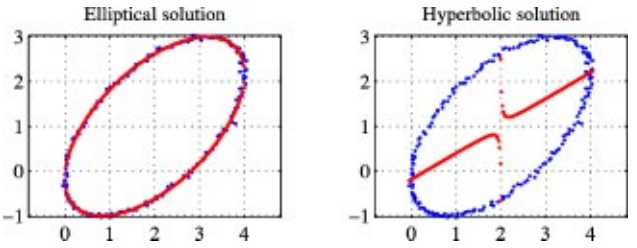


**Fig. 1** Test results when fitting to a noisy dataset, which clearly represent an ellipse.

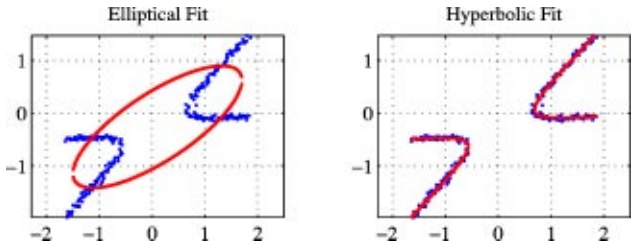


**Fig. 3** Test results when fitting to noisy hyperbolic data.

1. Determine the numerical rank $r_\epsilon$ of $\mathbf{S}_3$ and test if the data presented to the fitting routine are degenerate and correspond to a straight line. If the numerical rank $r_\epsilon = 2$, the fit is finished. (Halir did not consider the case of unsupervised inspection where the possibility of degenerate data had to be taken into account.) Otherwise, continue with step 2.
2. Solve the eigenvector problem,

$$\mathbf{C}_1^{-1}(\mathbf{S}_1 - \mathbf{S}_2\mathbf{S}_3^{-1}\mathbf{S}_2^T)\alpha_1 = \lambda\alpha_1, \tag{32}$$

   to determine $\alpha_1$. It should be noted that this merely requires the reduction of a $3\times 3$ matrix.
3. Calculate the constraint coefficients,

$$\kappa_i = \alpha[2,i]^2 - 4\alpha[1,i]\alpha[3,i]. \tag{33}$$

   Select the elliptical solution, *i.e.*, $\kappa_e = \kappa_i | \kappa_i < 0$ and then find the symmetric hyperbolic solution

$$\kappa_h = \kappa_i | \min\{\kappa_i + \kappa_e\} \text{ and } \kappa_i > 0. \tag{34}$$

4. Determine $\alpha_2$ for each solution by back-substituting $\alpha_1$ into,

$$\alpha_2 = -\mathbf{S}_3^{-1}\mathbf{S}_2^T\alpha_1. \tag{35}$$

5. Concatenate $\alpha_1$ and $\alpha_2$ for each solution,

$$\alpha = \begin{bmatrix}\alpha_1 \\ \alpha_2\end{bmatrix}. \tag{36}$$

### 3.3 *Numerical Tests*

The innovation here is the simultaneous fitting of hyperbolæ and ellipses. (Note that Pilu and Halir have given extensive tests of their algorithm for the fitting of ellipses to scattered data. Our work does not pretend to improve the performance with respect to ellipses, namely, we *use* Pilu's method. Consequently, comparing results would be pointless.) The numerical tests focus on verifying this property. The two solutions found by the algorithm are presented for the following test cases:

1. data that clearly represent an ellipse (Fig. 1)
2. data for an elliptical arc (Fig. 2)
3. data that clearly represent a hyperbola (Fig. 3)
4. data that represent a degenerate hyperbola (Fig. 4).

These tests were considered comprehensive enough to demonstrate the performance of the simultaneous fitting. All tests were performed with 200 data points having 3% noise.

## 4 Grassmannian Coordinates for Hyperbolæ

This section presents a new method to fit hyperbolæ where *a-priori* knowledge is available concerning the relative orientation of asymptotes. Such *a-priori* knowledge is common in many applications, *e.g.*, the cross section of a steel billet is rectangular.

Line coordinates of an axis of symmetry $\mathbf{l}$ of the hyperbola are written as follows,

$$\mathbf{l} \triangleq [L_1 \quad L_2 \quad L_3]. \tag{37}$$

Asymptotes $\mathbf{a}_p, \mathbf{a}_n \equiv \mathbf{a}_{\pm\phi}$ are lines on the center of symmetry of a hyperbola and displaced on either side of an axis of symmetry by an angle $\phi$. Therefore, asymptote line coordinates can be obtained by multiplying $\mathbf{l}$ by the rotation matrix,

$$\mathbf{R}(\phi) = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{38}$$

Abbreviations $cp \equiv \cos\phi$ and $sp \equiv \sin\phi$ are applied next,

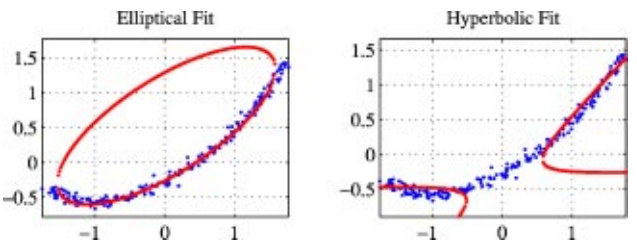


**Fig. 2** Test results when fitting to noisy data of an elliptical arc.

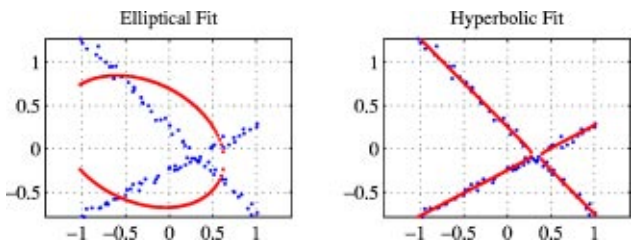


**Fig. 4** Test results when fitting to noisy degenerate hyperbolic data.

$$\mathbf{a}_{\pm\phi}=\mathbf{lR}(\pm\phi), \tag{39}$$

evaluating,

$$\mathbf{a}_p=[cpL_1+spL_2 \quad -spL_1+cpL_2 \quad L_3] \quad \text{and}$$

$$\mathbf{a}_n=[cpL_1-sp \quad L_2 spL_1+cpL_2 \quad L_3]. \tag{40}$$

The products of these asymptotes produce the conic $q$,

$$q=[cpL_1+spL_2 \quad -spL_1+cpL_2 \quad L_3]\begin{bmatrix} x \\ y \\ w \end{bmatrix} \times[cpL_1-spL_2 \quad spL_1+cpL_2 \quad L_3]\begin{bmatrix} x \\ y \\ w \end{bmatrix}. \tag{41}$$

Expanding Eq. (41), setting $w=0$, and collecting common coefficients yield the Grassmannian[11] and dual Grassmannian coordinates of the hyperbola, *i.e.*, the quadratic point forms and the conic coefficients,

$$q_\infty=[cp^2x^2-sp^2y^2 \quad -sp^2x^2+cp^2y^2 \quad 2xy] \times[L_1^2 \quad L_2^2 \quad L_1L_2]^T. \tag{42}$$

Consequently, the design matrix of the hyperbola $\mathbf{D}_h$ with known angle $2\phi$ between the asymptotes is,

$$\mathbf{D}_h=\begin{bmatrix} cp^2x_1^2-sp^2y_1^2 & -sp^2x_1^2+cp^2y_1^2 & 2x_1y_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ cp^2x_n^2-sp^2y_n^2 & -sp^2x_n^2+cp^2y_n^2 & 2x_ny_n & x_n & y_n & 1 \end{bmatrix}. \tag{43}$$

A special case occurs when the asymptotes are orthogonal to each other, *i.e.*, the model being fitted is a rectangular hyperbola, where $cp=sp$. The design matrix reduces to

$$\mathbf{D}_h=\begin{bmatrix} y_1^2-x_1^2 & x_1y_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ y_n^2-x_n^2 & x_ny_n & x_n & y_n & 1 \end{bmatrix}. \tag{44}$$

The same result can be attained another way. Consider two orthogonal direction cosines, *i.e.*, orthogonal lines on the origin:

$$\mathbf{a}\triangleq[a_1 \quad a_2] \quad \text{and} \quad \mathbf{a}^\perp\triangleq[-a_2 \quad a_1]. \tag{45}$$

The line product conic is shown in Eq. (46),

$$-a_1a_2x^2+(a_2^2-a_1^2)xy+a_1a_2y^2. \tag{46}$$

This corresponds to the constraint $C_1+C_3=0$ on the conic coefficients $\mathbf{C}$.

*Theorem 2.* *The conics with real and orthogonal asymptotes are defined on the dual Grassmannian coordinates*:

$$\mathbf{G}^D\triangleq[y^2-x^2 \quad xy \quad x \quad y \quad 1]^T. \tag{47}$$

*Proof.* The Grassmannian coordinates, *i.e.*, the coeficients are defined as,

$$\mathbf{g}\triangleq[g_1 \quad g_2 \quad g_3 \quad g_4 \quad g_5]. \tag{48}$$

The conic equation is,

$$c=\mathbf{G}^D\mathbf{g}=-g_1x^2+g_2xy+g_1y^2+g_3xw+g_4yw+g_5w^2. \tag{49}$$

Evaluating at infinity, *i.e.*, $w\mapsto 0$,

$$c_\infty=\mathbf{G}^D\mathbf{g}=-g_1x^2+g_2xy+g_1y^2. \tag{50}$$

The planar line coordinate of the asymptotes, *i.e.*, the roots of Eq. (50) are:

$${}_1\mathbf{b}:=[{}_1b_1 \quad {}_1b_2 \quad {}_1b_3]=\left[\frac{-g_2-(g_2^2+4g_1^2)^{1/2}}{2g_1}, \quad 1 \quad 0\right], \tag{51}$$

$${}_2\mathbf{b}:=[{}_2b_1 \quad {}_2b_2 \quad {}_2b_3]=\left[\frac{-g_2+(g_2^2+4g_1^2)^{1/2}}{2g_1}, \quad 1 \quad 0\right]. \tag{52}$$

The discriminants are nonnegative regardless of $g_1$ and $g_2$. Consequently, the asymptotes of the conic are also real. For the asymptotes to be orthogonal, it is only necessary that the inner product, taking these line coordinates as vectors, ${}_1\mathbf{b}\cdot{}_2\mathbf{b}$ vanishes. It is obvious from the right-hand sides of Eqs. (51) and (52) that this is indeed the case.

*Corollary 2.* *Reduction of the design matrix*,

$$\mathbf{D}_h=\begin{bmatrix} y_1^2-x_1^2 & x_1y_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ y_n^2-x_n^2 & x_ny_n & x_n & y_n & 1 \end{bmatrix}, \tag{53}$$

*will yield a conic with real and orthogonal aysmptotes*.

Furthermore, the matrix that must be reduced is now $n\times 5$ rather than $n\times 6$, as in the general case. This makes the method numerically more efficient than fitting a general hyperbola.

### 4.1 *Numerical Tests*

Two numerical tests for the fitting of hyperbolæ with orthogonal asymptotes are presented (Fig. 5): 1. a dataset that fulfills the condition with respect to *a-priori* knowledge, *i.e.*, that the asymptotes produced by the dataset must be truly orthogonal, and 2. a dataset where this condition is not fulfilled.

The second test demonstrates that the proposed method forces the asymptotes to be orthogonal irrespective of the nature of the dataset.

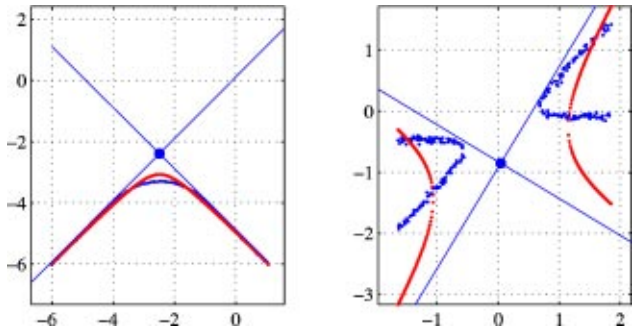

**Fig. 5** Left: Test fit to a dataset where the condition with respect to *a-priori* knowledge is fulfilled, *i.e.*, the asymptotes of the dataset are truly orthogonal. Right: Test fit on a dataset that do not have orthogonal asymptotes. This is obviously a poor fit; however, it demonstrates that the method enforces the orthogonality condition irrespective of the nature of the dataset.

The results of test 1 show that this fitting procedure tends to weight the data toward the ends of the asymptotes more strongly than the data points near the apex. This is seen in more detail in the next section where the confidence interval is analyzed.

## 5 Confidence Interval

Unsupervised inspection and fitting of geometric models place demanding requirements on the quality and certainty of fit. The industrial inspection tasks driving this work have real-time constraints, the 2500 billet cross section profiles, which must be acquired each second, require computationally simple and efficient methods. A new and numerically efficient way to determine the spatial uncertainty of the fitis therefore presented. It should be noted that the spatial uncertainty is nonuniform along the curve. This is due to the difference between the algebraic and normal Euclidean distance from a point to a curve.[3]

### 5.1 *Nature of the Metric and Numerical Considerations*

The proposed metric is the algebraic distance. The relationship between algebraic distance and normal distance depends on the object being considered and the attitude of error vectors, which are a measure of distance, relative to the object. Some insight into the metric can be gained by examining planar curves and the offset surfaces corresponding to the squared distance $d^2$ from the curves. The residual resulting from the implicit equation is not a symmetric function. This can be clearly seen in the case of a parabola, shown in Fig. 6. Points with equal normal distance to the parabola produce a different residual, depending on their position. Consider the parabola,

$$x - y^2 = 0. \tag{54}$$

The residual in this case is a linear function in $x$ and a quadratic function in $y$. As can be seen in Fig. 6, the resulting error surface deviates strongly from $d^2$. Consequently, minimizing the squared algebraic distance leads to a poor fit. This is the case for all curves and surfaces that are highly asymmetric in powers and/or coefficients of $x,y,(z)$. In applications where unsupervised fitting is being applied, the form of the squared residual surface should be evaluated prior to selecting the fitting method. A nonlinear minimization of $d^2$ is to be preferred when:

1. the data points contain a high degree of noise
2. the data points are distributed unevenly along the object, or
3. when the squared residual surface is strongly asymmetric.

The numerical stability of the fitting procedure is considerably improved by applying a similarity transformation to the data prior to fitting.

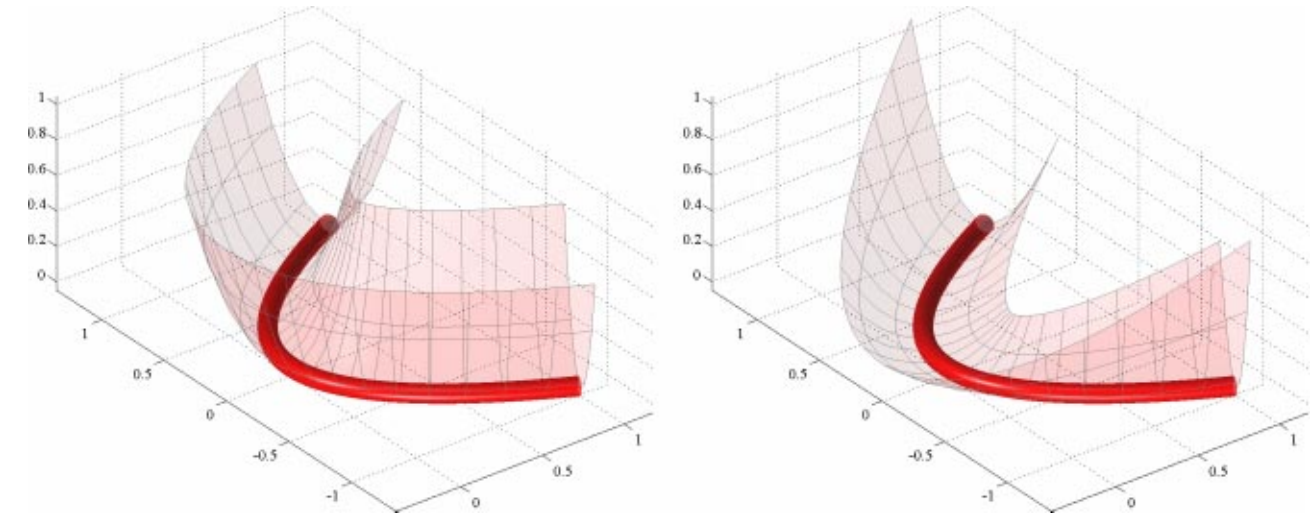

**Fig. 6** Left: The surface corresponding to $d^2$ around a parabola. Right: The square of the residual $r_s^2$ resulting from the implicit Eq. (54) for the parabola.

1. Translate the data so that it is centered around the origin, *i.e.*, zero mean $x$, $y$, and $z$.
2. Scale the data so that the maximum distance of a point to the origin is $\sqrt{2}$. This is particularly important for higher-order objects.

### 5.2 *Singular Value Decomposition*

The notation used for singular value decomposition in this work is $\mathbf{A}\overset{svd}{\mapsto}[\mathbf{U},\mathbf{S},\mathbf{V}]$. If $\mathbf{D}$ is a real $m\times n$ matrix, then the orthogonal matrices,

$$\mathbf{U}=[\mathbf{u}_1,\ldots,\mathbf{u}_m]\in\mathbb{R}^{m\times n}:\|\mathbf{U}\|_2=\mathbf{I} \text{ and}$$

$$\mathbf{V}=[\mathbf{v}_1,\ldots,\mathbf{v}_n]\in\mathbb{R}^{n\times n}:\|\mathbf{V}\|_2=\mathbf{I}, \tag{55}$$

exist, such that,

$$\mathbf{U}^T\mathbf{D}\mathbf{V}=\mathbf{S}=\text{diag}\{\sigma_1,\ldots,\sigma_p\}\in\mathbb{R}^{m\times n} \text{ for } p=\min\{m,n\}, \tag{56}$$

where $\sigma_1\geqslant\sigma_2\geqslant\ldots\sigma_p\geqslant 0$. The $\sigma_i$ are the singular values of $\mathbf{A}$, and $\mathbf{u}_i$ and $\mathbf{v}_i$ are the $i$'th left and right singular column vectors. In this geometric case, $\mathbf{v}_i$ form the orthogonal basis vectors of $\mathbf{D}$, and $\mathbf{u}_i$ are the normalized orthogonal projections of $\mathbf{D}$ onto $\mathbf{u}_i$. The singular value $\sigma_i$ is the two-norm distance of the vector $\mathbf{v}_i$ from the null space of $\mathbf{D}$. Further,

$$\sigma_p=\min_{\mathbf{v}_p\neq 0}\frac{\|\mathbf{D}\mathbf{v}_p\|_2}{\|\mathbf{v}_p\|_2}. \tag{57}$$

The solution being sought corresponds to $\mathbf{c}=\mathbf{v}_p$.

The matrix $\mathbf{D}$ may have a higher-than-unity degree of rank deficiency, depending on the dataset and the nature of the dual Grassmannian coordinates of the geometric object. The numerical rank is defined as follows:

$$r_\epsilon=\text{rank}\{\mathbf{A},\epsilon\}, \tag{58}$$

where, $\sigma_1\geqslant\ldots\sigma_{r_\epsilon}\geqslant\epsilon\geqslant\sigma_{r_\epsilon+1}\geqslant\sigma_p$. There can be multiple column vectors $\mathbf{v}_i:i>r_\epsilon$. The concept of numerical rank requires that all linear combinations of the column vectors of $\mathbf{V}^\perp$ be considered as possible solutions. Consequently, the null$\{\mathbf{D},\epsilon\}$ is the set of all solutions to fitting the geometric object to the data. For example, if $\mathbf{D}$ is rank deficient order 2, then the solution is a pencil of hyperplanes in the dual Grassmannian space. This corresponds to a straight line in the Grassmannian space; interpolation between two points.

### 5.3 *Confidence Interval Associated with the Algebraic Distance*

In automated quality control applications, it is important to have an estimate for the confidence interval[12] of the measurement result. Data points are corrupted by noise, so there is a distribution of fit residuals. There are two different confidence intervals that need to be considered.

1. Given the geometric model that has been found, how large is the region in which a data point may validly lie within a given range of statistical confidence?
2. Given the data and the geometric model being used, what is the tolerance on coefficient values?

The singular value $\sigma_p$ is the two-norm distance of the vector $\mathbf{c}=\mathbf{v}_p$ from the null space of the design matrix $\mathbf{D}$, *i.e.*,

$$\sigma_p=\|\mathbf{D}\mathbf{c}\|_2=\|\mathbf{r}\|_2=\left(\sum_{i=1}^{m}r_i^2\right)^{1/2}. \tag{59}$$

Consequently, the standard deviation of the residual $\delta_r$ is,

$$\delta_r=\frac{\sigma_p}{\sqrt{n}}, \tag{60}$$

where $n$ is the number of points in the sample dataset. The mean residual $\bar{r}$ is zero; further, it is assumed the residual has a normal distribution. (In critical applications, the residual should be tested to determine if it fulfills the requirement of having a Gaussian (normal) distribution.) In this case, the equation for the conic can be rewritten as,

$$c_1x^2+c_2y^2+c_3xy+c_4x+c_5y+c_6=\text{N}(0,\delta_r). \tag{61}$$

That is, the data point residuals are not zero but normally distributed with zero mean and standard deviation $\delta_r$. In engineering applications, it is common to define a confidence interval as some real multiple $f$ of the standard deviation, *e.g.*, within a range of, say, $2.5\sigma$. Applying this to Eq. (61) and rearranging to implicit form gives the following results.

1. The expected (*i.e.*, the mean) result is,

   $$c_1x^2+c_2y^2+c_3xy+c_4x+c_5y+c_6=0. \tag{62}$$

   The values of the coefficients $\mathbf{c}$ found are used for all calculations with respect to the geometric object.
2. The confidence interval in which a point lies is given by

   $$c_1x^2+c_2y^2+c_3xy+c_4x+c_5y+c_6\pm f\sigma_p/\sqrt{n}=0. \tag{63}$$

   The value for $f$ is chosen to achieve the desired probability for the confidence interval. Changing the value of $c_6$ corresponds to a parallel shift of the hyperplane in the dual Grassmann space. This is consistent with the observation that the singular value $\sigma_p$ is the mean square component of the data orthogonal to the dual of the vector $\mathbf{v}_p$. An example of fitting an ellipse to data with the resulting confidence interval can be seen in Fig. 7.

## 6 Application to Automatic Inspection

The original motivation for this work arose from a problem encountered in the automatic inspection of steel billets and the necessity to autonomously discriminate rectangular from elliptical cross sections. The first task is to decide

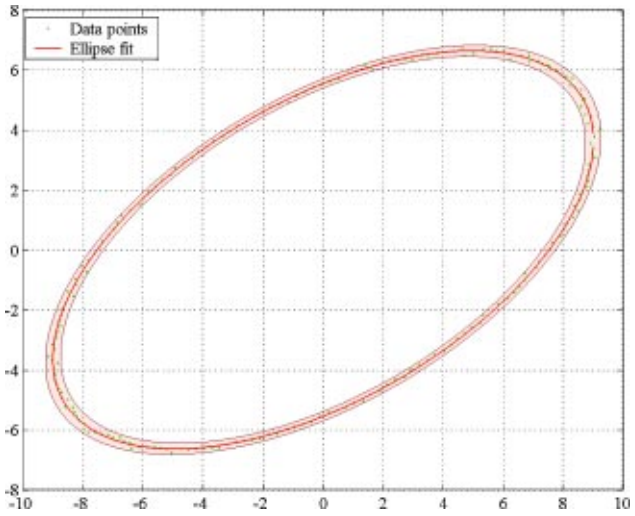


**Fig. 7** Fitting a conic section to noisy data. The confidence interval is for the range $\pm 3\,\delta_r$ and corresponds to a contour on the surface of the residuals.

whether the cross section of a red hot steel billet, emerging from a rolling mill, is circular or rectangular, with sharp or rounded corners. This is achieved by simultaneously fitting an ellipse and a hyperbola to the acquired dataset representing points reflected from the billet surface and captured in a camera image of a plane of laser light intersecting the billet.

The next task is to accurately estimate invariant properties of the cross section, *i.e.*, center point and radius of the circular sections, or the orientation of the rectangular sides and the point of intersection. These properties are used to register sections acquired in a sequence of images taken along the length of the advancing billet. The assembly of registered images is then filtered, thereby enhancing any existing surface defects to allow their machine recognition and, if possible, subsequent correction, *e.g.*, by abrasive grinding.

In the case of steel billet inspection and many other engineering applications, the geometry of the objects being encountered is known, *e.g.*, adjacent faces of the rectangular and rounded-rectangular billets are mutually orthogonal. This *a-priori* knowledge is used to improve the quality of the results by fitting a hyperbola with orthogonal asymptotes. The center point of the hyperbola and the orientation of the asymptotes can be determined directly from the conic matrix.

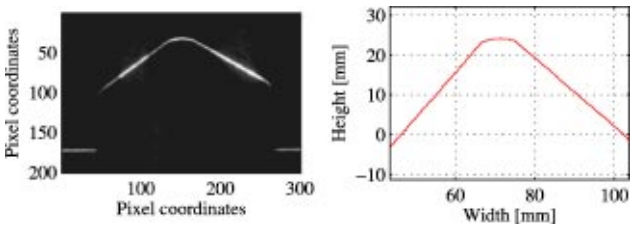


**Fig. 8** Example of an original image from the laser profiler and the relevant portion of the rectified profile data.

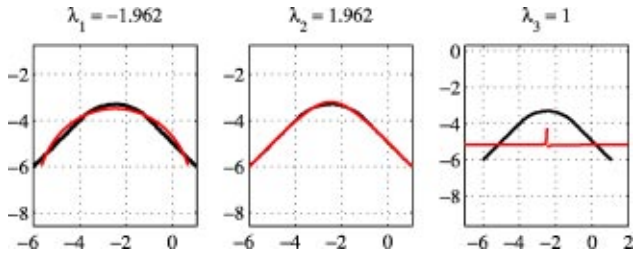


**Fig. 9** The three fit solutions delivered by extending the method of Pilu. Original data are black and the fit results are red.

The experimental data are taken from a laser profiler measuring the cross section of steel billets. An example of the original image and the rectified profile, which is delivered by the profiler, can be seen in Fig. 8.

The measurement system is confronted with billets having three possible cross section types.

**Circular**: can clearly be modeled by an ellipse.

**Rectangular**: can be modeled as a degenerate hyperbola with orthogonal asymptotes.

**Rounded rectangular**: these can only be approximated by a conic. However, the center point of the conic is well suited to determine the translation required to register multiple sections.

The first step in registering the individual sections is to perform a fit according to the constrained eigenvector problem. The results of this fit for the sample profile are shown in Fig. 9. The important feature is that all three solutions are produced simultaneously with a single reduction of the scatter matrix.

These results are used to determine if the cross section is best modeled as an ellipse or a hyperbola. In the hyperbolic case, a second fit with orthogonal asymptotes is performed. A comparison of this fit and the hyperbola from the eigenvector solution is shown in Fig. 10.

Note the difference in the angles between the asymptotes and the coordinates for the center point. These differences are important when rectifying individual sections to form a complete surface model.

A billet is inspected by acquiring sequential sections; however, these are corrupted by motion and vibration of the billet during transportation. The rectangular hyperbolic

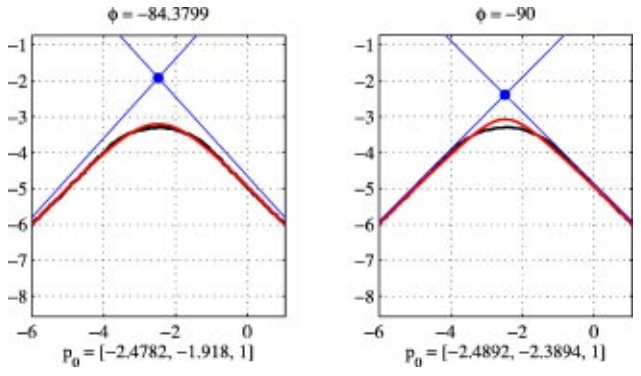


**Fig. 10** Left: The fit delivered by the modified Pilu method. Right: Result from reduction on the dual Grassmannian coordinates for a hyperbola with orthogonal asymptotes.

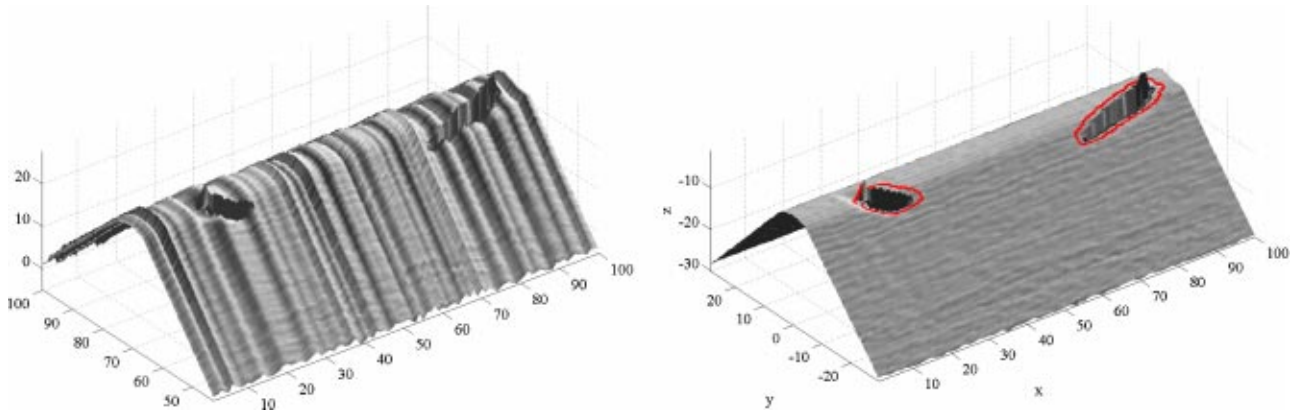

**Fig. 11** Left: Original data as acquired; the uneven surface results from vibration of the billet during transportation. Right: Registered and rectified surface with localized defects. The rectangular hyperbolic model has been used to achieve this result.

model is now used to register the sections to a common center point and orientation (see Fig. 11).

It is interesting to investigate the properties of the confidence interval associated with the algebraic distance to hyperbola (see Fig. 12). The points near the apex have a lower contribution to the fit that points near the asymptotes.

## 7 Conclusions

New linear numerical methods for the fitting of specific conics to data are presented. The methods include solutions for:

3. direct and simultaneous least-squares fitting of both ellipses and hyperbolæ, achieved by using constrained generalized eingenvalue decomposition
4. direct least-square fitting of rectangular hyperbolæ using Grassmannian reduction
5. determination of the spatial confidence interval associated with least-square algebraic fitting.

This is the first linear-least square hyperbola-specific fitting method. Two new theorems on fitting hyperbolæ are presented, together with rigorous proofs. All proposed methods are verified using synthetic and real datasets. Implementations of these methods in MATLAB m-code are made available. An application in automatic industrial inspection is presented.

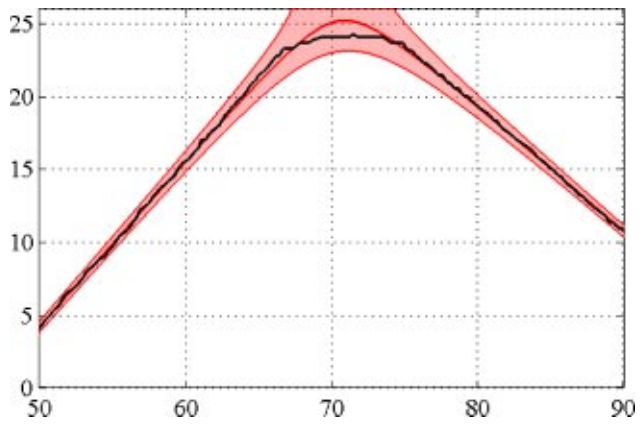

**Fig. 12** Orthogonal fit with associated confidence interval.

Furthermore, this article contributes to industrial applications by describing in some detail the development and validation of effective, real-time algorithms to autonomously inspect bar stock and steel billets of circular and rectangular cross sections. This is done by applying elementary planar projective geometry of points, lines, and conics. Exciting avenues and opportunities for research in unsupervised industrial inspection, based on fast, certain, and accurate artificial decision making, are opened. Consider these three, possibly trite, adjectives in the context of unsupervised inspection.

*Fast.* When transforming dual elements, *i.e.*, lines, as opposed to points, in the plane one needs only to premultiply the homogeneous line coordinate vector by the dual displacement matrix.[13] This matrix is simply the adjoint of the primal one for points. Lines are thus converted with one multiplication rather than two if a point pair is used. Similarly, point-form and line-form conics are tranformed by transforming their coefficient matrix [see Eq. (5)] and adjoint of the coefficient matrix, respectively.

*Certain.* A feature may appear as the image of a hyperbola. To ensure that this, rather than another conic, will emerge from the fit, one begins with a linear (algebraic distance) fit on quadratic forms peculiar to hyperbolae. This is outlined in Sec. 4. The principle is applicable to other curves, a subject for further investigation.

*Accurate.* What if the cross section is known to be a *rectangle*? Then the image is best approximated by a pair of perpendicular lines or by a rectangular hyperbola. To ensure this, one uses the combined quadratic form $[x^2-y^2, xy, x, y, 1]$. This removes a column from the Grassmannian matrix to more simply produce more accurate results. We call this technique ''prejudicial perception.'' Naming things allows us to more effectively recognize, remember, and apply them. The element of research offered here is the challenge to generalize this concept for wider application in design and analysis.

## 8 Appendix: MATLAB Code

The MATLAB m-code presented has been optimized with respect to the clarity of the procedure being implemented, at the sacrifice of numerical efficiency. Further, the similarity transform **S**, *i.e.*, making the data mean free and scaling before fitting, is not shown in the code for reasons of clarity.

### 8.1 *Simultaneously Fit Ellipse and Hyperbola*

```
function [ellipse, hyperbola]
=fitEllipsAndHyperbola(x, y);
%
% it is assumed that x and y are column
vectors
%
x2=x.^2;
y2=y.^2;
xy=x.*y;
%
% Set up the design and scatter matrices
%
D1=[x2,xy,y2];
D2=[x,y,ones(size(x))];
%
S1=D1'*D1;
S2=D1'*D2;
S3=D2'*D2;
%
% test the rank of S3
%
[Us3, Ss3, Vs3]=svd(S3);
condNrs=diag(Ss3)/Ss3(1,1);
%
%epsilon=1e-10;
epsilon=eps;
if condNrs(3)<epsilon
   warning('S3 is degenerate');
   return;
end;
%
% define the constraint matrix and its in-
verse
%
C=[0,0,-2;
   0, 1, 0;
   -2,0,0];
Ci=inv(C);
%
% Setup and solve the generalized eigen-
vector problem
%
T=-inv(S3)*S2';
S=Ci*(S1+S2*T);
%
[evec,eval]=eig(S);
%
% evaluate and sort resulting constraint
values
%
cond=evec(2,:).^2-4*(evec(1,:).*evec(3,:));
[condVals index]=sort(cond);
%
% Pick up the elliptical solution
%
eValE=condVals(1);
alpha1=evec(:,index(1));
alpha2=T*alpha1;
ellipse=[alpha1;alpha2];
%
% Pick up the hyperbolic solution
%
possibleHs=condVals(2:3)+condVals(1);
[minDiffVal,minDiffAt]=min(abs
(possibleHs));
eValH=possibleHs(minDiffAt);
alpha1=evec(:,index(minDiffAt+1));
alpha2=T*alpha1;
hyperbola=[alpha1;alpha2];
```

### 8.2 *Fit Orthogonal Hyperbola*

```
function
[orthConic,fitStd]=fitOrtHyperbola(x,y);
%
% (c) Paul O'Leary 2003
% It is assumed that x and y are column
vectors of the same length
%
x2=x.^2;
y2=y.^2;
xy=x.*y;
%
% Setup the design matrix
%
D=[y2-x2,xy,x,y,ones(size(x))];
%
% Perform singular value decomposition
%
[U,S,V]=svd(D,0);
%
% Select the best solution
%
result=V(:,end)';
%
% Convert it to standard form for a conic
with all 6 coefficients
% NOTE: result(1) is repeated
%
orthConic=[-result(1),result(2),result
(1),result(3:end)];
%
% Determine the standard deviation of the
residual.
%
fitStd=S(end,end)/sqrt(length(x));
```

*Acknowledgments*

We would like to thank Manfred Husty for his patience and perseverance in introducing us to methods of projective geometry, Antony Gfrerrer for insights into the nature of conic matrices and helpful comments on this piece of work,

and Norbert Koller for the acquisition of experimental data and observations during testing of the methods in an industrial application.

## References

1. S. Kaveti, T. Khawang, and H. Wang, ''Second-order implicit polynomials for segmentation of range images,'' *Pattern Recogn.* **29**, 937–949 (1996).
2. F. Bookstein, ''Fitting conic sections to scattered data,'' *Comput. Graph. Image Process.* **9**, 56–71 (1987).
3. P. Sampson, ''Fitting conic sections to 'very scattered' data: An interative refinement of the bookstein algorithm,'' *Comput. Graph. Image Process.* **18**, 97–108 (1982).
4. G. Taubin, ''Estimation of planar curves, surfaces and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **13**(11), 1115–1138 (1991).
5. M. Pilu, A. Fitzgibbon, and R. Fisher, ''Ellipse-specific direct least-square fitting,'' *IEEE ICIP* (1996).
6. A. Fitzgibbon, M. Pilu, and R. Fischer, ''Direct least square fitting of ellipses,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **21**(5), 476–480 (1999).
7. R. Halir and J. Flusser, ''Numerically stable direct least squares fitting of ellipses,'' Technical Report http://citeseer.nj.nec.com/350661.html, Dept. Software Eng., Charles Univ., Czech Republic (2000).
8. G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., John Hopkins Univ. Press, Baltimore, MD (1996).
9. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge Univ. Press, Cambridge, MA (2000).
10. W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley, ''Revisiting hartleys normalized eight-point algorithm,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(9), 1172–1177 (2003).
11. P. J. Zsombor-Murray and M. J. D. Hayes, ''Grassmannian reduction of quadratic forms,'' in *17th Cdn. Congress of Applied Mechanics (CANCAM 99)*, pp. 313–314, Hamilton (1999).
12. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Fannery, *Numerical Recipes in C*, 2nd ed., Cambridge Univ. Press, Cambridge, MA (1995).
13. P. J. Zsombor-Murray, ''Planar point and line transformation,'' private communication (2002).