Metric Vision Methods for Material and Product Inspection

Doctoral Thesis

Peter Schalk

Institute for Automation
Department Product Engineering
University of Leoben
Leoben, Austria

March, 2007

Supervisors:

o.Univ.-Prof. Dipl.-Ing. Dr.techn. Paul O'Leary
University of Leoben, Austria

Ao.Univ.-Prof. Mag.rer.nat. Dr.techn. Anton Gfrerrer
Graz University of Technology, Austria

I hereby declare that this thesis and the work reported herein was composed and originated entirely by myself, unless stated otherwise.

Leoben, March 2007 Peter Schalk

Acknowledgements

First of all I would like to thank Prof. Paul O'Leary for the inestimable support during the last years, for the possibility to compose this thesis at the Institute for Automation, and for the fruitful discussions regarding the thesis. I appreciate his critical and supporting comments on this work.

I would also like to express my gratitude to Prof. Anton Gfrerrer for co-supervising this work and for the valuable comments contributing to this thesis.

I also appreciate the great support through the team of the Institute for Automation. I am especially grateful to my office colleagues Ewald Fauster, who accompanied me through all my time at the University of Leoben, and Matthew Harker for the pleasurable working atmosphere in our office. I would like to thank Doris Widek, Beate Oswald, Norbert Koller, Ronald Ofner, Gerold Probst, Gerhard Rath, Ingo Reindl, and Gernot Wally for their great support whenever needed.

I am deeply grateful to my parents Agnes and Vinzenz for laying a solid foundation on which I can make my own way. My present and future success is always based on their success. I am thankful for their valuable and enduring support.

Furthermore, I appreciate the real and loyal friendship with my brother Michael. His support both in technical and human aspects is an indescribable present. It is comforting and encouraging to know, that such a exceptional person stands at one's side.

Finally I would like to express my gratitude to Nina, the sunshine of my life. I wish to thank her for her great mental support, for the trust she places in me, and for the countless encouragements she bestows upon me.

Abstract

This thesis deals with topics central to metric vision, i.e. the use of digital image processing techniques to gain quantitative information about position, orientation, dimensions and shape of objects.
Starting from images and a-priori knowledge of the optical arrangements, points of interest are extracted using basic image processing algorithms: Gradient based methods enhance object edges; center of gravity calculation and contour algorithms deliver data points on sub-pixel level. Geometric models such as lines and circles are fitted to these noisy data points. Further processing steps are strongly dependent on the actual measurement task and include e.g. simple distance calculation and sophisticated methods such as curve reconstruction.
An analytical approach to estimate the uncertainty of measurement results by first-order Taylor-series expansion is presented. The outcome of the analytical computation is verified by Monte-Carlo simulation. The error propagation in line and circle fitting algorithms, and in the derivation of results from the fitted parameters is thoroughly analyzed and quantified. An analytical expression of the covariance matrix of the line parameters is derived. The uncertainty associated with fitting circles is statistically analyzed. All investigations consider the correlations between parameters.
A special issue of research is the reconstruction of convex Jordan curves from curvature data. Two algorithms based on differential geometry are designed and tested for specific measurement setups and enable the determination of the shape of a profile from radii of curvature over its circumference.
Two industrial applications, that incorporate the results of the theoretical approaches, are presented: (1) A profile measurement system for rotating objects based on light sectioning and curve reconstruction, and (2) a video extensometer designed for strain measurement of hot glowing material exemplarily demonstrate the applicability of the developed methods.
The profile measurement system is implemented to check the quality of seamless steel pipes. The quality of pipes is determined in terms of straightness and circularity of the outside profile. The eccentricity the pipe exhibits when rotated on rollers is a measure of its straightness. The circularity is determined by reconstructing the cross-sectional profile of the pipe.
The second measurement system, the video extensometer, enables the determination of properties of refractory material at temperatures up to 1400°C. The influence of temperature, camera orientation, and line fitting approaches on the measurement uncertainty is analyzed in detail in order to elaborate potentials to increase the accuracy of the system.

Kurzfassung

Diese Arbeit behandelt zentrale Themen der metrischen Bildverarbeitung, d.h. der Verwendung von Bildverarbeitungstechniken zur Bestimmung quantitativer Informationen über Position, Orientierung, Größe, und Form von Messobjekten.
Ausgehend von Bildern und a priori-Wissen über optische Anordnungen werden Basisalgorithmen zur Extraktion von Bildpunkten verwendet: Gradienten-basierende Methoden dienen der Hervorhebung von Kanten; Intensitätsschwerpunktsberechnungen und Konturalgorithmen liefern Punkte mit Subpixel-Genauigkeit. Geometrische Modelle, wie z.B. Geraden und Kreise, werden an diese verrauschten Punkte angepasst. Weitere Prozessschritte sind von der eigentlichen Messaufgabe abhängig und beinhalten z.B. einfache Abstandsberechnungen bis hin zu komplexen Methoden wie Kurvenrekonstruktion.
Ein analytischer Ansatz zur Abschätzung der Unsicherheit der Messergebnisse durch Taylorreihenentwicklung erster Ordnung wird präsentiert. Ergebnisse dieser Berechnung werden durch Monte-Carlo-Simulationen verifiziert. Die Fehlerfortpflanzung in Geraden- und Kreisanpassungsmethoden, und in der Ableitung von Ergebnissen aus den angepassten Parametern wird untersucht und quantifiziert. Ein analytischer Ausdruck der Kovarianzmatrix der Geradenparameter wird abgeleitet. Die mit der Kreisanpassung verbundene Messunsicherheit wird statistisch analysiert. Alle Untersuchungen berücksichtigen etwaige Korrelationen zwischen den Anpassungsparametern.
Ein spezielles Forschungsthema ist die Rekonstruktion konvexer Jordanscher Kurven aus Krümmungsdaten. Zwei auf Differentialgeometrie basierende Algorithmen werden für spezifische Messaufbauten entwickelt und ermöglichen die Bestimmung der Form eines Profils aus Krümmungsradien, die über den Umfang gemessenen werden.
Zwei industrielle Anwendungen, die die Ergebnisse der theoretischen Ansätze integrieren, werden präsentiert: (1) Ein Profilmesssystem für rotierende Objekte, basierend auf dem Lichtschnittverfahren und der Kurvenrekonstruktion, sowie (2) ein Videoextensometer für die Dehnungsmessung an heißglühendem Material, zeigen exemplarisch die Anwendbarkeit der entwickelten Methoden. Das vorgeschlagene Profilmesssystem wird zur Qualitätsprüfung von Nahtlosrohren implementiert. Die Qualität der Rohre wird durch deren Geradheit und der Kreisförmigkeit des Querschnitts bestimmt. Als Maß für die Geradheit wird die Exzentrizität, die ein auf Rollen rotierendes Rohr aufweist, herangezogen. Die Kreisförmigkeit wird durch die Rekonstruktion des Querschnittsprofils ermittelt.
Das zweite Messsystem, das Videoextensometer, ermöglicht die Bestimmung von Eigenschaften von Feuerfestmaterialien bei Temperaturen von bis zu 1400°C. Der Einfluss der Temperatur, der Kameraorientierung und der gewählten Geradenanpassungsmethode auf die Messunsicherheit wird im Detail analysiert, um Potentiale zur Erhöhung der Messgenauigkeit des Systems zu erarbeiten.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Metric vision comprises the optical measurement of size, shape, position and orientation of geometric objects by means of digital image processing, and furthermore, the determination of relationships between properties of geometric objects.

Actually no measurement is complete without the statement of uncertainty in the measurement result. According to the *ISO Guide to the Expression of Uncertainty in Measurement*[37], the uncertainty in the result is an indication of its quality and it is obligatory to evaluate and express it. Without the statement of their uncertainties, measurement results cannot reasonably be compared with each other or with a reference standard.

No thorough analysis of error propagation and uncertainty is available for the problems encountered in and techniques used for metric vision. Past analyzes are restricted to simple geometric constellations[12]. The anisotropy in the error structure and the correlations between uncertainties of the measured values has been neglected.

There are two principal types of error associated with a metric vision system (see also Figure 1.1):

1. Calibration errors: The error in the calibration information has the same effect on each individual measurement performed with this information. This is a systematic deviation and thus leads to a bias in the measurement result.

2. Errors in a single measurement: These errors originate in the image acquisition process and are propagated through the measurement and analysis chain. They primarily lead to a statistical uncertainty in the measurement result.

This work is focussed on the second type: The propagation of measurement errors. It is assumed, that the error of feature points extracted from the image is known (e.g. from experience or from repeatability measurement) or estimated. The influence of this error on the uncertainty in parameters of objects fitted to the points, and on the uncertainty in measurement results derived from these parameters, is investigated. An analytical and a stochastic approach to estimate the error propagation in an evaluation process are presented.

Lines and circles are the most important objects encountered in metric vision. Thus a thorough analysis of the error propagation in algorithms for fitting lines and circles to sets of noisy data points is performed. The anisotropy in the error structure and correlations between parameters are determined and incorporated in the derivation of measurement results from relationships between these parameters.

A second field of investigations in this thesis is the measurement of cross-sectional profiles. A new approach to determine the profile curve over the complete circumference of rotating objects by local curvature measurement and curve reconstruction is presented. The curvature measurement is performed with a single light sectioning measurement head. Methods of differential geometry are used to reconstruct the curve from the measured radii of curvature.

The complete measurement chain from instrumentation to the final measurement result is elaborated and presented for two specific metric vision applications. Introductorily, a possible abstraction of the measurement chain is presented in the following section.

1.1 Measurement Chain in a Metric Vision System

A possible abstraction of the measurement chain and the error propagation within a metric vision system is shown in Figure 1.1.

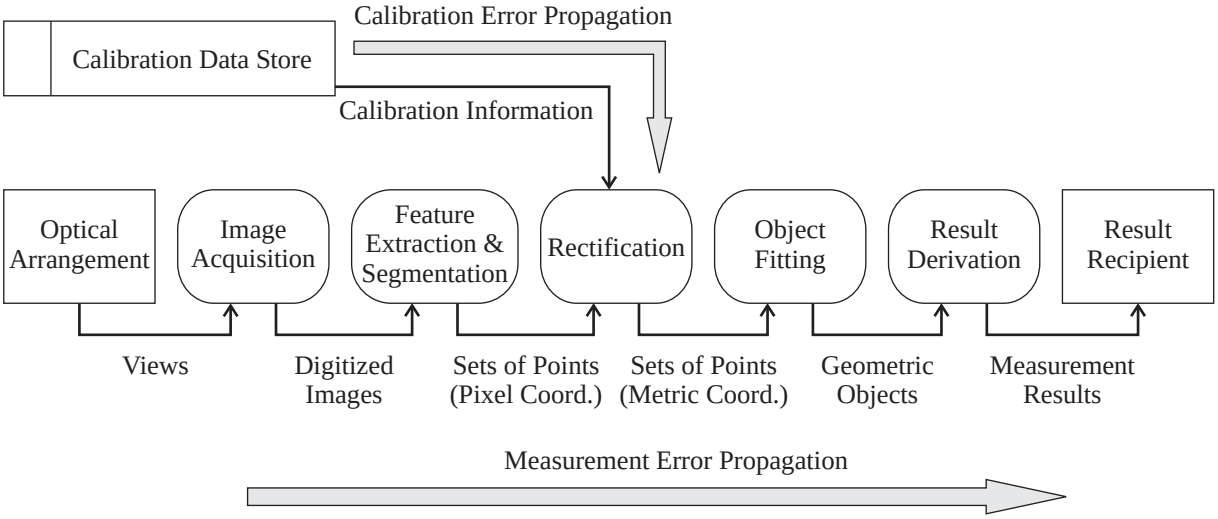

Figure 1.1: Measurement data flow within a metric vision system. The gray-shaded arrows indicate that the measurement and calibration errors are propagated and contribute to the uncertainty of the measurement result

The measurement chain comprises a sequence of processes and is bounded by an input and an output node. These nodes are:

Optical Arrangement: The optical arrangement provides the basis to create an image of the measurement scene. It is determined by the setup of the optical components

(filters, lenses, light sources, etc.) and the cameras with respect to the object, which is to be measured. The optical arrangement provides the views for the image acquisition process.

Result Recipient: The result recipient is the output node of the metric vision system. In case of a stand-alone measurement system, this entity is usually a visualization unit. If the metric vision system is embedded in an automated plant (e.g. a material testing machine, an industrial production plant, etc.) the result recipient provides an interface to the superior control system.

Between the input and output nodes, the measurement evaluation is performed in the following processing steps:

Image Acquisition: The initial process of each measurement is the acquisition of an image (or a sequence of images). The view provided by the optical arrangement is imaged on a camera chip and the light intensity that falls on each pixel is transduced to an electric signal, which is then digitized[30, 36]. The output of the Image Acquisition process is a digitized image. The primary measurement errors originate in this process. These are imaging errors caused by air vibrations (and associated refraction of light; see Subsection 8.4.2 and Sonka *et. al.*[57]) and lens distortion[35]. The errors due to the spatial quantization on the pixel grid of the camera chip and the noise and quantization of the pixel intensity[30] are added. In single-chip color imaging applications the demosaicing of the color array contributes with additional errors of the pixel intensities[31, 48].

Feature Extraction and Segmentation: The next step in metric vision measurements is the extraction of points of interest from the digitized images. These points describe edges of objects, contours of measurement marks or some other features in the image. Usual extraction steps are image enhancement, edge detection[5, 18], thresholding[57], contouring, and center-of-gravity calculation[45]. In the process of segmentation the extracted data points are grouped to sets of points, which correspond to individual features in the image. Segmentation techniques are discussed by Sonka *et. al.*[57]. The output of this processing step are sets of 2D-points in pixel coordinates. The point coordinates are perturbed with noise, since the imaging and acquisition errors are propagated to the extracted points.

Rectification: Utilizing the calibration information (see also Subsection 1.1.1) , the Rectification process transforms measurement data from pixel coordinates to metric coordinates. The calibration information may be a mere scaling factor (see e.g. Subsection 8.3.2), the entries of a homography matrix (i.e. a collineation between an image plane and a metric coordinate plane), the parameters or the lookup-table of a non-linear transformation function, or a combination of several such entries for a multi-stage transformation. In the Rectification process the error associated with the image points and the uncertainty in the calibration information contribute to the uncertainty of the rectified point coordinates.

Object Fitting: The point sets are abstracted by geometric objects such as lines, circles, general conics, polynomial curves, splines, etc. A geometric object is represented by a set of parameters. In the Object Fitting process these parameters are estimated by minimization of an error function, which may either be based on the algebraic expression of the object or on the geometric distances of the points to the object. The uncertainty in the fit results is influenced by both the error of the input points and the applied fitting algorithm.

Result Derivation: Finally the measurement result is computed from interdependencies and relations between fitted objects. Two collections of objects are considered:

1. Spacial collection: The objects are available in a single measurement coordinate frame and results are obtained by applying geometric operation. Examples are positions of intersection points, distances between points or parallel lines, etc.
2. Temporal collection: The input data is collected from a temporal sequence of images. Signal processing techniques can be used to extract the sought measurement result (e.g. filter operations, Fourier transform, etc).

1.1.1 Calibration of the Measurement System

The calibration information is determined in the calibration procedure. During this reference measurement a calibration target with known dimensions is measured with the metric vision system to establish a relationship between a metric world coordinate system and the internal image coordinate systems. The structure of the calibration procedure is similar to the structure of the measurement evaluation as shown in Figure 1.1. However, the dimensions of the calibration target are utilized rather than the yet unknown calibration information, and the output of the procedure is the sought calibration information. The resulting uncertainty in the calibration information is influenced by the uncertainty in the dimensions of the calibration target and the uncertainty associated with the reference measurement.

In case of an *auto-calibrating* measurement system, the calibration procedure is integrated in the measurement chain. The physical structure that serves as the calibration target is visible in the measurement scene and the calibration is performed with the acquired measurement image. Nevertheless, the dimension of the calibration target must be known.

The calibration target and procedure and with it the calibration information permit the traceability of the measurement result obtained with the metric vision system to a measurement standard as defined by the International System of Units (SI units, see [21, 37]).

1.2 Outline of the Thesis

The thesis is organized in two parts. The first part introduces mathematical basics and fundamental investigations of metric vision methods. The second part comprises two application-oriented chapters.

At the beginning of part I, in Chapter 2, a thorough review of common algebraic and geometric routines for fitting lines and circles to scattered data points is given. Furthermore, the estimation of the parameters of a homography from point correspondences is discussed.

Two methods to investigate the propagation of measurement errors in computational processes are presented in Chapter 3. These methods enable the estimation of the error in processing results, given the input data and its associated uncertainty. The first method, Monte-Carlo simulation, is a numerical approach based on simulated measurement data. In addition an analytical method, based on the first-order Taylor-series expansion of the process function, is presented.

The uncertainty of lines fitted to noisy data points and the uncertainty of quantities derived from these lines is discussed in Chapter 4. A general model for the points is established and the covariance matrix of the fitted line parameters is analytically derived. Monte-Carlo simulation is used to selectively prove the analytic results.

In Chapter 5 the error sensitivity of four algorithms to fit circles to noisy data points is investigated. At first the tangent position of a circle, fitted to circular arc data, is introduced as a measure of its position. By means of Monte-Carlo simulation the influence of the noise, the number, and the subtending arc angle of the points on the uncertainty of the parameters of the fitted circle (center point, radius, tangent position) is evaluated.

Reconstructing a planar curve from measured curvature data is discussed in Chapter 6. Thereby two different types of sampling of the curvature data and the respective measurement setups are taken into account. The developed algorithms are the basis for measurement and reconstruction of cross-sectional profiles of rotating objects.

In the first chapter of Part II (Chapter 7) a profile measurement system for rotating material is presented. A light sectioning measurement head is used to repeatedly measure the local curvature. Utilizing the algorithms presented in Chapter 6, the profile curve is reconstructed from the measured radii of curvature. An industrial implementation of the profile measurement system to determine the circularity of seamless steel pipes is presented.

In Chapter 8 a video extensometer application is introduced. The video extensometer is designed to measure the deformation of refractory material under load at temperatures of up to $1400°C$. The temperature-related hardware adaptations and the elaborated measurement chain are presented. Error sources and potentials to decrease the uncertainty of the measurement result are identified.

1.3 Original Work

The original contribution of the work presented in this thesis comprises two domains:

1. measurement uncertainty, and
2. profile measurement.

The contributions concerning the uncertainty in measurements are as follows:

1. A symbolic expression of the uncertainty (expressed as covariance matrix) of a line, which is fitted to a set of noisy data points, is derived. A general model for the point set is established, including the number of points, the spread of the points, the position of the centroid of the points, the orientation of the point set and the uncertainty in the position of an individual point. The derivation yields an expression of each entry of the covariance matrix of the line parameters as a function of the listed parameters of the point set model.

2. The error propagation associated with different circle fitting algorithms is statistically investigated. The influence of the number of input points, the noise in the points, the subtending arc angle (in case of circular arc data), and the applied fitting algorithm on the uncertainty in the fit results are determined.

3. The analysis of circle fitting algorithms reveals unexpected correlations in the fit results (i.e. the center point and the radius), although the input data points are uncorrelated. Being aware of such correlations, they can be used to derive quantities, which exhibit a lower uncertainty than the original fit results: Here the tangent point is introduced as a measure of the position of the circle, which is fitted to circular arc data. It is shown, that the statistical behavior of this point with respect to error distribution and standard deviation is significantly better than for the center point.

4. The uncertainty of quantities, which are derived from relations between fitted objects, is determined. Anisotropic noise distributions and correlations between parameters of the objects are incorporated in the error propagation model.

A new solution to measure cross-sectional profiles of rotating objects is presented. The individual contributions within this approach are:

1. It is shown how differential geometry can be applied to reconstruct the profile from local measurements of the radius of curvature. The parameterization of the measurement signal (i.e. the discrete measurement sample spacing) is compatible with the measurement setup (i.e. the way of driving the rotation of the measurement object).

2. Aperiodic auto-correlation is used to identify the periodicity in the curvature measurement signal corresponding to exactly one revolution of the cross-section. In this manner the segmented data is perfectly cyclic, consequently there is no Gibb's error when performing Fourier analysis. This property is used to enable non-causal filtering in the frequency domain.

3. Curves of constant width are frequently encountered as an unwanted result of a production process for circular shapes (e.g. drilled holes, rolled pipes, etc.). Thus a least-squares fitting procedure is presented to detect their occurrence when characterizing measured profiles.

Part I

Metric Vision Methods

Chapter 2

Fitting of Geometric Primitives and Parameter Estimation

This chapter deals with the topic of fitting objects to scattered (i.e. noisy) 2D-points by least-squares methods based on algebraic and/or geometric error measures. Such a fitting operation is used in the vast majority of image metrology applications (e.g. [13, 19, 56, 60]) to estimate the parameters of a geometric object. An alternative approach for parameter estimation is the Hough transform[30]. The Hough transform can operate directly on the intensity image (thus omitting the extraction of data points) and is more robust against outliers. However, due to the explicit quantization of the parameter space (the Hough space), the computational effort is high. The computational complexity of the Hough transform is in the order of $O(s^{p-1}N)$, whereby s is the number of discrete samples along the Hough dimension, p is the number of parameters and N is either the number of columns/rows (the larger value) of the intensity image or the number of extracted points[51, 62]. In metric vision applications a high resolution of the parameter space would be needed to obtain the maximal precision. The complexity would become intractable for $p > 2$. Considering Singular Value Decomposition[29] to solve a least-squares problem, the computational complexity is approximately $O(Np^2+p^3)$. Least-squares fitting methods are preferred in metric vision applications, due to the avoidance of quantizing the parameter space and thus the lower complexity.

2.1 Normalization of Data

Prior to the fitting procedure the input data is normalized with respect to position and scale. This is motivated by the following facts:

1. In general the result of algebraic fitting procedures is not invariant to similarity transformations of the input data (e.g. shown in [35] for estimating homographies), unless the error measure which is minimized, can be geometrically interpreted.

2. The numerical stability of the fitting procedure is improved by the normalization of

the input data[11].

Thus, the common workflow for fitting routines is:

1. Normalize the input data,
2. Fit the geometric object to normalized data,
3. Denormalize the fitted object (i.e. back-transform to the original input data).

Choinacki *et al.*[11] show, that normalizing the N input data points (given in homogeneous coordinates[25]) $\boldsymbol{p}_i = [x_i, y_i, 1]^T$, $i = \{1 \dots N\}$ such that the points are centered at the origin and the root mean square distance of the mean free points to the origin is equal to $\sqrt{2}$, improves the numerical stability of fitting algorithms. The normalization is a similarity transformation, which can be represented in matrix form as:

$$\mathsf{T}_{norm} = \begin{bmatrix} \frac{1}{s} & 0 & -\frac{\bar{x}}{s} \\ 0 & \frac{1}{s} & -\frac{\bar{y}}{s} \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.1}$$

where

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i, \;\; \bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i, \quad \text{and} \tag{2.2}$$

$$s = \sqrt{\frac{1}{2N}\sum_{i=1}^{N}(x_i - \bar{x})^2 + (y_i - \bar{y})^2} \tag{2.3}$$

Hence, the normalized points $\boldsymbol{p}_{i_{norm}}$ are obtained with:

$$\boldsymbol{p}_{i_{norm}} = \mathsf{T}_{norm}\boldsymbol{p}_i \tag{2.4}$$

The inverse transformation is represented by the matrix

$$\mathsf{T}_{norm}^{-1} = \begin{bmatrix} s & 0 & \bar{x} \\ 0 & s & \bar{y} \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.5}$$

It is assumed, that the input data points are normalized according to the above procedure, when applying the fitting algorithms that are described in the following sections. Nevertheless, for reasons of readability, an extra notation of normalized data is avoided. The operations of back transformation to the original position and scaling (denormalization) are specific to the type of the fitted entity (line, circle, etc.). Hence, these operations are presented at the end of the respective fitting sections.

2.2 Line Fitting

A line in the plane is represented by the homogeneous algebraic equation

$$l_1 x_h + l_2 y_h + l_3 w_h = 0 \tag{2.6}$$

where l_1, l_2, and l_3 denote the line parameters and x_h, y_h and w_h, are the homogeneous coordinates of a point on the line. Setting $\boldsymbol{l} = [l_1,\, l_2,\, l_3]^T$ and $\boldsymbol{p}_h = [x_h,\, y_h,\, w_h]^T$, Equation 2.6 can be rewritten in vector form as:

$$\boldsymbol{p}_h^T \boldsymbol{l} = 0. \tag{2.7}$$

Given a set of $N > 2$ homogenized (such that $w_i = 1$ for all points) 2D-points (not all of them identical) $\boldsymbol{p}_i = [x_i,\, y_i,\, 1]^T$, $i = \{1 \dots N\}$; they are arranged in a design matrix D, such that

$$\mathsf{D} = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_N & y_N & 1 \end{bmatrix}, \tag{2.8}$$

If the points are collinear, the design matrix D is rank deficient with rank 2 and the following equation is satisfied:

$$\mathsf{D}\boldsymbol{l} = \boldsymbol{0}, \tag{2.9}$$

where $\boldsymbol{l}$ is a vector representing the null-space of D and represents the line on which all points lie.

In the case of point coordinates perturbed with noise, Equation 2.9 in general does not hold true. The null-vector on the right hand side is to be replaced by a vector of residual errors $\boldsymbol{e}$:

$$\mathsf{D}\boldsymbol{l} = \boldsymbol{e}, \tag{2.10}$$

It is a common mistake to fit the line by minimizing the norm $\|\boldsymbol{e}\| = \|\mathsf{D}\boldsymbol{l}\|$ (or its square $\boldsymbol{e}^T\boldsymbol{e} = \boldsymbol{l}^T\mathsf{D}^T\mathsf{D}\boldsymbol{l}$) under the constraint $\|\boldsymbol{l}\| = 1$. It can be shown, that the obtained line does not pass through the centroid of the data points, unless the points are collinear. A correct solution is obtained when minimizing

$$\min_{\boldsymbol{l}} \|\mathsf{D}\boldsymbol{l}\| \quad \text{under the constraint} \quad l_1^2 + l_2^2 = 1 \tag{2.11}$$

The method of Lagrange multipliers[29] can be used to include this constraint in the minimization process. Another possibility to implicitly impose the correct constraint to the minimization is shown in the following: The design matrix D is partitioned in two parts. One part contains the data perturbed with noise, and the second statistically invariant

part is the column of ones:

$$\mathsf{D} = \begin{bmatrix}\mathsf{D}_1 & \mathsf{D}_0\end{bmatrix} = \left[\begin{array}{cc|c} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_N & y_N & 1 \end{array}\right]. \tag{2.12}$$

The design matrix is redefined by projecting D_1 onto the orthogonal complement of D_0 (orthogonal residualization):

$$\hat{\mathsf{D}} = \mathsf{D}_1 - \mathsf{D}_0\mathsf{D}_0^+\mathsf{D}_1, \tag{2.13}$$

where D_0^+ is the pseudoinverse of D_0. The pseudoinverse of a real matrix $\mathsf{A} \in \mathbb{R}^{m\times n}$ is computed as:

$$\mathsf{A}^+ = (\mathsf{A}^T\mathsf{A})^{-1}\mathsf{A}^T, \tag{2.14}$$

if $\mathsf{A}^T\mathsf{A}$ is invertible. Otherwise a more general definition of the pseudoinverse is used[22]:

$$\mathsf{A}^+ = \mathsf{V}\Sigma^+\mathsf{U}^T, \tag{2.15}$$

where $\mathsf{U} \in \mathbb{R}^{m\times m}$ and $\mathsf{V} \in \mathbb{R}^{n\times n}$ are orthogonal matrices containing the left and right singular vectors of A and

$$\Sigma^+ = \operatorname{diag}\left(\frac{1}{\sigma_1}, \ldots, \frac{1}{\sigma_r}, 0, \ldots, 0\right) \in \mathbb{R}^{n\times m}, \tag{2.16}$$

where σ_i, $i = \{1 \ldots r\}$, are the singular values of A and r is the rank of A.

The pseudoinverse of the column of ones D_0 yields:

$$\mathsf{D}_0^+ = \frac{1}{N}\mathsf{D}_0^T, \tag{2.17}$$

where N is the length of D_0, i.e. the number of points. Hence equation 2.13 yields:

$$\hat{\mathsf{D}} = \mathsf{D}_1 - \mathsf{D}_0\frac{1}{N}\mathsf{D}_0^T\mathsf{D}_1 \tag{2.18}$$

This is equivalent with subtracting the respective mean value in each column of D and omitting the last column of resulting zeros:

$$\hat{\mathsf{D}} = \begin{bmatrix} x_1 - \bar{x} & y_1 - \bar{y} \\ x_2 - \bar{x} & y_2 - \bar{y} \\ \vdots & \vdots \\ x_N - \bar{x} & y_N - \bar{y} \end{bmatrix}, \tag{2.19}$$

where $\bar{x} = \frac{1}{N}\sum x_i$ and $\bar{y} = \frac{1}{N}\sum y_i$. Due to the prior normalization of the points, they are already mean free, hence in this case: $\hat{\mathsf{D}} = \mathsf{D}_1$. The problem is now formulated as:

$$\min_{\hat{\boldsymbol{l}}} \|\hat{\mathsf{D}}\hat{\boldsymbol{l}}\| \quad \text{subject to} \quad \|\hat{\boldsymbol{l}}\| = 1 \tag{2.20}$$

There are two principle ways to solve the problem: Singular Value Decomposition (SVD) or eigenvector calculation (refer to [29] for both methods). SVD decomposes $\hat{\mathsf{D}}$ in matrices U, S, and V such that:

$$\hat{\mathsf{D}} = \mathsf{U}\Sigma\mathsf{V}^T, \tag{2.21}$$

where U and V are orthogonal matrices and Σ is a diagonal matrix. The entries of Σ are the singular values and the columns of U and V are the left and right singular vectors of $\hat{\mathsf{D}} = \mathsf{D}_1$, respectively. The right singular vector corresponding to the smallest singular value is the sought solution $\hat{\boldsymbol{l}}$.

Equivalently the solution is found by solving the eigenvector problem

$$\mathsf{S}\hat{\boldsymbol{l}} = \lambda\hat{\boldsymbol{l}}, \tag{2.22}$$

where S is defined as the scatter matrix $\mathsf{S} = \hat{\mathsf{D}}^T\hat{\mathsf{D}}$, and selecting the eigenvector $\hat{\boldsymbol{l}}_{\min}$ corresponding to the smallest eigenvalue $\lambda_{\min}$ of S.

Both methods deliver the orientational part $\hat{\boldsymbol{l}} = [l_1, l_2]^T$ of the sought line. The third coefficient l_3 of the line $\boldsymbol{l}$ is found by back substitution:

$$l_3 = -\mathsf{D}_0^+\mathsf{D}_1\hat{\boldsymbol{l}} = -\begin{bmatrix}\bar{x} & \bar{y}\end{bmatrix}\hat{\boldsymbol{l}}. \tag{2.23}$$

Finally the sought line is found by $\boldsymbol{l} = [l_1, l_2, l_3]^T = \left[\hat{\boldsymbol{l}}^T, l_3\right]^T$.

Metric of the solution:

- *Proposition:* Due to the implicit constraint $\|\hat{\boldsymbol{l}}\| = \sqrt{l_1^2 + l_2^2} = 1$ in Equation 2.20, the oriented normal distance d of a point $\boldsymbol{p} = [x, y, 1]^T$ to the line $\boldsymbol{l} = \left[\hat{\boldsymbol{l}}^T, l_3\right]^T$ is given by:

  $$d = \boldsymbol{p}^T\boldsymbol{l}. \tag{2.24}$$

  Proof: Considering a line $\boldsymbol{l} = [l_1, l_2, l_3]^T$ defined by two points $\boldsymbol{p}_1 = [x_1, y_1, 1]^T$ and $\boldsymbol{p}_2 = [x_2, y_2, 1]^T$, and a third point $\boldsymbol{p} = [x, y, 1]^T$ (see Figure 2.1a), the area of the triangle, which is defined by these three points, is[39]:

  $$A_\triangle = \frac{1}{2}\begin{vmatrix} x & y & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{vmatrix} = \frac{1}{2}\boldsymbol{p}^T\boldsymbol{l}, \quad \text{whereby} \quad \boldsymbol{l} = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} = \begin{bmatrix} y_1 - y_2 \\ -(x_1 - x_2) \\ x_1y_2 - y_1x_2 \end{bmatrix}. \tag{2.25}$$

The area can also be computed as

$$A_{\triangle} = \frac{1}{2}bd, \tag{2.26}$$

whereby $b = \sqrt{(y_1 - y_2)^2 + (x_1 - x_2)^2} = \sqrt{l_1^2 + l_2^2}$ is the distance between $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$, and d is the normal distance between the line $\boldsymbol{l}$ and the point $\boldsymbol{p}$ (cp. Figure 2.1a). From the constraint $\sqrt{l_1^2 + l_2^2} = 1$, it follows $b = 1$. Equating 2.25 and 2.26 and solving for d yields:

$$d = \frac{\boldsymbol{p}^T\boldsymbol{l}}{b} = \boldsymbol{p}^T\boldsymbol{l}. \tag{2.27}$$

q.e.d. ■

- Alternatively a geometric approach can be used to show the relation between the error measure and the normal distance between a point and a line (or in general a plane or hyperplane):

  Proposition: Let

  $$\hat{\boldsymbol{l}}^T\boldsymbol{x} + l_{d+1} = 0 \tag{2.28}$$

  be the equation of a hyperplane H in the d-dimensional Euclidean space $\mathbb{E}_d$ and let $\boldsymbol{p}_e = [p_1, \ldots, p_d]^T$ be the position vector of a point P of $\mathbb{E}_d$, then the oriented distance of P and H is given by

  $$\text{dist}(P, H) = \frac{\hat{\boldsymbol{l}}^T\boldsymbol{p}_e + l_{d+1}}{\|\hat{\boldsymbol{l}}\|} \tag{2.29}$$

  Proof: Since $\hat{\boldsymbol{l}}$ is a normal vector of H,

  $$\boldsymbol{x} = \boldsymbol{p}_e - \lambda\frac{\hat{\boldsymbol{l}}}{\|\hat{\boldsymbol{l}}\|} \tag{2.30}$$

  is a parameterization of the line n orthogonal to H through P (see Figure 2.1b). The parameter λ is the oriented distance of the running point X (represented by the positional vector $\boldsymbol{x}$) on n from P. The common point S of n and H is found by substitution of Equation 2.30 into 2.28 and computation of λ:

  $$\lambda_S = \frac{\hat{\boldsymbol{l}}^T\boldsymbol{p}_e + l_{d+1}}{\|\hat{\boldsymbol{l}}\|} = \text{dist}(P, S) = \text{dist}(P, H) \tag{2.31}$$

  q.e.d. ■
  Considering the special case of $\|\hat{\boldsymbol{l}}\| = 1$:

  $$\text{dist}(P, H) = \hat{\boldsymbol{l}}^T\boldsymbol{p}_e + l_{d+1}. \tag{2.32}$$

  Assuming that P is represented as a homogeneous vector $\boldsymbol{p} = [\boldsymbol{p}_e, 1]^T$ and the line (plane or hyperplane) H is denoted as $\boldsymbol{l} = \left[\hat{\boldsymbol{l}}, l_{d+1}\right]^T$, Equations 2.32 and 2.27 are equivalent.

- Thus, the solution found with Equations 2.20 and 2.23 minimizes the normal distances of the given points in a least-squares sense, i.e. the geometric error is minimized and an unbiased maximum-likelihood solution is obtained.

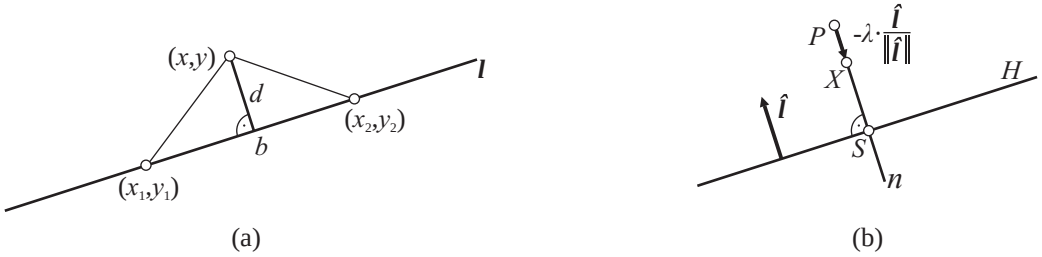

Figure 2.1: Normal distance of a point to a line (plane or hyperplane): Line $\boldsymbol{l}$ is defined by two points (a); general hyperplane H is given by its equation and λ is the oriented normal distance of the running point X on n from P (b).

2.2.1 Generalization to Higher Dimensions

The presented method of fitting lines in the plane can be expanded to fitting planes in space or generally fitting hyperplanes in hyperspace. Generally in an M-dimensional space the design matrix $\hat{\mathsf{D}}$ is expanded to M columns (cf. Equation 2.19)

$$\hat{\mathsf{D}} = \begin{bmatrix} x_{11} - \overline{x_1} & x_{12} - \overline{x_2} & \dots & x_{1M} - \overline{x_M} \\ x_{21} - \overline{x_1} & x_{22} - \overline{x_2} & \dots & x_{2M} - \overline{x_M} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} - \overline{x_1} & x_{N2} - \overline{x_2} & \dots & x_{NM} - \overline{x_M} \end{bmatrix}, \tag{2.33}$$

where $\overline{x_i} = \frac{1}{N}\sum_{j=1}^{N} x_{ji}$ and the solution vector found by SVD is $\hat{\boldsymbol{v}} = [v_1, v_2, \dots v_M]^T$, describing the orientational part of the hyperplane. Accordant to Equation 2.23 back substitution delivers the last coefficient of the vector representing the hyperplane:

$$v_{M+1} = -\begin{bmatrix} \overline{x_1} & \overline{x_2} & \dots & \overline{x_M} \end{bmatrix} \hat{\boldsymbol{v}}, \tag{2.34}$$

and the sought hyperplane is determined by $\boldsymbol{v} = \begin{bmatrix} \hat{\boldsymbol{v}}^T, v_{M+1} \end{bmatrix}^T$.

The general case of fitting hyperplanes in hyperspace exhibits the same metric properties as the special 2D-case of fitting lines in the plane, i.e. the geometric error is minimized and a maximum-likelihood estimation of the hyperplane is obtained.

2.2.2 Fitting Sets of Parallel Lines

Two parallel lines $\boldsymbol{l}_1 = [l_1, l_2, {}_1l_3]^T$ and $\boldsymbol{l}_2 = [l_1, l_2, {}_2l_3]^T$ are represented by the affine algebraic equations:

$$l_1 x + l_2 y + {}_1l_3 = 0 \tag{2.35}$$

$$l_1 x + l_2 y + {}_2l_3 = 0 \tag{2.36}$$

The lines have the common coefficients l_1 and l_2, which represent the orientational part of the lines. The position of each line is determined by the last coefficient ${}_1l_3$ and ${}_2l_3$, respectively. If $l_1^2 + l_2^2 = 1$, ${}_1l_3$ and ${}_2l_3$ is the oriented normal distance of the respective line to the origin.

Given two sets of M and N data points, represented by ${}_1\boldsymbol{p}_i = [{}_1x_i,\ {}_1y_i,\ 1]^T$ and ${}_2\boldsymbol{p}_j = [{}_2x_j,\ {}_2y_j,\ 1]^T$, where $i = \{1 \dots M\}$ and $j = \{1 \dots N\}$, and considering the task of fitting a pair of parallel lines to these two data sets, the design matrix D and the solution vector $\boldsymbol{l}_{12}$ can be defined as[46]:

$$\mathsf{D} = \left[\begin{array}{c|cc} {}_1\mathsf{D}_1 & {}_1\mathsf{D}_0 & \mathsf{0} \\ \hline {}_2\mathsf{D}_1 & \mathsf{0} & {}_2\mathsf{D}_0 \end{array}\right] = \left[\begin{array}{cc|cc} {}_1x_1 & {}_1y_1 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ {}_1x_M & {}_1y_M & 1 & 0 \\ \hline {}_2x_1 & {}_2y_1 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ {}_2x_N & {}_2y_N & 0 & 1 \end{array}\right] \quad \text{and} \quad \boldsymbol{l}_{12} = \left[\begin{array}{c} l_1 \\ l_2 \\ \hline {}_1l_3 \\ {}_2l_3 \end{array}\right] \tag{2.37}$$

Orthogonal projection of ${}_1\mathsf{D}_1$ onto ${}_1\mathsf{D}_0$ and ${}_2\mathsf{D}_1$ onto ${}_2\mathsf{D}_0$ (cp. Equations 2.13 to 2.19) delivers the redefined design matrix $\hat{\mathsf{D}}$, the solution vector $\hat{\boldsymbol{l}}$, and the new problem formulation:

$$\min_{\hat{\boldsymbol{l}}} \|\hat{\mathsf{D}}\hat{\boldsymbol{l}}\| = \left\| \left[\begin{array}{cc} {}_1x_1 - {}_1\bar{x} & {}_1y_1 - {}_1\bar{y} \\ \vdots & \vdots \\ {}_1x_M - {}_1\bar{x} & {}_1y_M - {}_1\bar{y} \\ \hline {}_2x_1 - {}_2\bar{x} & {}_2y_1 - {}_2\bar{y} \\ \vdots & \vdots \\ {}_2x_N - {}_2\bar{x} & {}_2y_N - {}_2\bar{y} \end{array}\right] \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} \right\| \quad \text{subject to} \quad \|\hat{\boldsymbol{l}}\| = 1. \tag{2.38}$$

The solution of the common line coefficients $\hat{\boldsymbol{l}} = [l_1,\ l_2]^T$ is found by applying SVD on $\hat{\mathsf{D}}$ and selecting the right singular vector corresponding to the smallest singular value. The individual third coefficient of each line is determined by back substitution:

$${}_1l_3 = -\begin{bmatrix} {}_1\bar{x} & {}_1\bar{y} \end{bmatrix} \hat{\boldsymbol{l}}, \tag{2.39}$$

$${}_2l_3 = -\begin{bmatrix} {}_2\bar{x} & {}_2\bar{y} \end{bmatrix} \hat{\boldsymbol{l}}. \tag{2.40}$$

This method can also be used to fit a set of $K > 2$ lines by expanding the design matrices D and $\hat{\mathsf{D}}$. Furthermore, accordant to Subsection 2.2.1 the method can be generalized to fitting parallel hyperplanes in hyperspace.

2.2.3 Fitting Sets of Orthogonal Lines

Similar to fitting parallel lines, orthogonal lines can be fitted, by rotating one set of data points by 90°. After orthogonal residualization, the problem formulation has the following

form:

$$\min_{\hat{\boldsymbol{l}}} \|\hat{\mathsf{D}}\hat{\boldsymbol{l}}\| = \left\| \begin{bmatrix} {}_1x_1 - {}_1\bar{x} & {}_1y_1 - {}_1\bar{y} \\ \vdots & \vdots \\ {}_1x_M - {}_1\bar{x} & {}_1y_M - {}_1\bar{y} \\ \hline -({}_2y_1 - {}_2\bar{y}) & {}_2x_1 - {}_2\bar{x} \\ \vdots & \vdots \\ -({}_2y_N - {}_2\bar{y}) & {}_2x_N - {}_2\bar{x} \end{bmatrix} \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} \right\| \quad \text{subject to} \quad \|\hat{\boldsymbol{l}}\| = 1. \tag{2.41}$$

The singular vector corresponding to the smallest singular value of $\hat{\mathsf{D}}$ delivers the solution for $\hat{\boldsymbol{l}}$, which is in this case the direction of the first line $\hat{\boldsymbol{l}}_1 = \hat{\boldsymbol{l}}$. The direction of the second line is perpendicular to the first line, thus: $\hat{\boldsymbol{l}}_2 = [-l_2, l_1]^T$. The respective third line coeffcients are found by:

$${}_1l_3 = -\begin{bmatrix} {}_1\bar{x} & {}_1\bar{y} \end{bmatrix} \hat{\boldsymbol{l}}_1, \tag{2.42}$$

$${}_2l_3 = -\begin{bmatrix} {}_2\bar{x} & {}_2\bar{y} \end{bmatrix} \hat{\boldsymbol{l}}_2. \tag{2.43}$$

2.2.4 Back Transformation of Lines

Due to normalization of the input data points prior to the fitting procedure (see Section 2.1), the fitted lines need to be back transformed to the position of the original points. Using the normalization transformation T_{norm} defined in Equation 2.1, the back transformation of a line vector $\boldsymbol{l}_{norm}$ is performed by:

$$\boldsymbol{l}_{orig} = \mathsf{T}_{norm}^T \boldsymbol{l}_{norm}. \tag{2.44}$$

2.3 Circle Fitting

An algebraic expression of a circle in the plane is given by

$$(x - x_c)^2 + (y - y_c)^2 - r^2 = 0, \tag{2.45}$$

where the coordinates x and y denote a point on the circle, the parameters x_c and y_c are the coordinates of the center of the circle, and the parameter r is the radius of the circle. The equation of a circle can also be written in the form

$$c_1(x^2 + y^2) + c_2x + c_3y + c_4 = 0, \tag{2.46}$$

where the coefficients have to fullfill

$$c_1 \neq 0 \quad \text{and} \tag{2.47}$$

$$\frac{(c_2^2 + c_3^2)}{4c_1^2} - \frac{c_4}{c_1} > 0 \tag{2.48}$$

The relationships between the circle coefficients $\boldsymbol{c} = [c_1, c_2, c_3, c_4]^T$ and the parameters x_c, y_c and r are:

$$\begin{aligned} x_c &= -\frac{c_2}{2c_1} \\ y_c &= -\frac{c_3}{2c_1} \\ r &= \sqrt{x_c^2 + y_c^2 - \frac{c_4}{c_1}} \end{aligned} \tag{2.49}$$

2.3.1 Simple Algebraic Fitting

Given a set of N planar points $\boldsymbol{p}_i = [x_i, y_i]^T$, $i = \{1 \dots N\}$, which are perturbed with noise, the task is to find a circle that approximates the data points in a least-squares sense. According to Equation 2.46 the algebraic distance of a point $\boldsymbol{p}_i$ to the circle is:

$$c_1(x_i^2 + y_i^2) + c_2 x_i + c_3 y_i + c_4 = e_i \tag{2.50}$$

Each point contributes such an equation. The set of equations can be formulated in matrix form as:

$$\begin{bmatrix} x_1^2 + y_1^2 & x_1 & y_1 & 1 \\ x_2^2 + y_2^2 & x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_N^2 + y_N^2 & x_N & y_N & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}, \tag{2.51}$$

or short

$$\mathsf{D}\boldsymbol{c} = \boldsymbol{e}. \tag{2.52}$$

A least-squares algebraic solution is found by minimizing the norm of the residual vector:

$$\min_{\boldsymbol{c}} \|\boldsymbol{e}\| = \|\mathsf{D}\boldsymbol{c}\| \quad \text{subject to} \quad \|\boldsymbol{c}\| = 1 \tag{2.53}$$

The constraint $\|\boldsymbol{c}\| = 1$ is necessary to avoid the trivial solution $\boldsymbol{c} = [0, 0, 0, 0]^T$. The algebraic least squares solution for $\boldsymbol{c}$ is found by applying SVD on the design matrix D and selecting the right singular vector corresponding to the smallest singular value of D (see also Section 2.2).

Although it can be shown, that the constraint $\|\boldsymbol{c}\| = 1$ does not imply a total least squares solution[44], the circle parameters estimated with this fitting method exhibit subjective statistical advantages over solutions of alternative algebraic fitting methods when fitting circles to normalized circular arc data. This will be shown in Chapter 5.

2.3.2 Total Least Squares Fitting

Accordant to the orthogonal residualization in fitting a line (cp. Equation 2.13), Nievergelt [44] proposes a similar strategy for fitting of circles, spheres and hyperspheres. The design matrix D is partitioned in a part containing noisy data and a statistically invariant part, i.e. for circles:

$$\mathsf{D} = \begin{bmatrix} \mathsf{D}_1 & \mathsf{D}_0 \end{bmatrix} = \left[\begin{array}{ccc|c} x_1^2 + y_1^2 & x_1 & y_1 & 1 \\ x_2^2 + y_2^2 & x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_1^N + y_1^N & x_N & y_N & 1 \end{array}\right] \tag{2.54}$$

Then, D_1 is projected onto the orthogonal complement of D_0 (cp. Equations 2.13 to 2.19); this yields a redefined design matrix:

$$\hat{\mathsf{D}} = \begin{bmatrix} x_1^2 + y_1^2 - \overline{x^2 + y^2} & x_1 - \bar{x} & y_1 - \bar{y} \\ x_2^2 + y_2^2 - \overline{x^2 + y^2} & x_2 - \bar{x} & y_2 - \bar{y} \\ \vdots & \vdots & \vdots \\ x_N^2 + y_N^2 - \overline{x^2 + y^2} & x_N - \bar{x} & y_N - \bar{y} \end{bmatrix} \tag{2.55}$$

where $\overline{x^2 + y^2}$ means $\frac{1}{N}\sum(x_i^2 + y_i^2)$. The problem is now formulated as

$$\min_{\hat{\boldsymbol{c}}} \|\hat{\mathsf{D}}\hat{\boldsymbol{c}}\| \quad \text{subject to} \quad \|\hat{\boldsymbol{c}}\| = 1, \tag{2.56}$$

where $\hat{\boldsymbol{c}} = [c_1,\, c_2,\, c_3]^T$ is the vector of the first three circle coefficients. Again, it is found by SVD. The fourth coefficient c_4 is found by back substitution:

$$c_4 = -\begin{bmatrix} \overline{x^2 + y^2} & \bar{x} & \bar{y} \end{bmatrix} \hat{\boldsymbol{c}}. \tag{2.57}$$

2.3.3 Partitioned Total Least Squares

Considering the redefined design matrix $\hat{\mathsf{D}}$ in Equation 2.55, the statistically invariant part, the column of ones, has been removed. However, there is a difference in the type of distribution and variance of the statistical error (noise) between the left column and the other two columns. This is due to the fact, that the data points contribute quadratically to the design of the leftmost column and linearly to the design of the middle and right column. Halíř *et al.*[32] propose the partitioning of the design matrix for fitting ellipses into a quadratic part and a linear part. In [34], this approach was used to fit even higher order curves and deal with each order of data individually in an incrementing manner. The circle-specific solution of this approach is presented here.

The design matrix $\hat{\mathsf{D}}$ is partitioned such that:

$$\hat{\mathsf{D}} = \begin{bmatrix} \hat{\mathsf{D}}_2 & \hat{\mathsf{D}}_1 \end{bmatrix} = \left[\begin{array}{c|cc} x_1^2 + y_1^2 - \overline{x^2 + y^2} & x_1 - \bar{x} & y_1 - \bar{y} \\ x_2^2 + y_2^2 - \overline{x^2 + y^2} & x_2 - \bar{x} & y_2 - \bar{y} \\ \vdots & \vdots & \vdots \\ x_N^2 + y_N^2 - \overline{x^2 + y^2} & x_N - \bar{x} & y_N - \bar{y}, \end{array}\right] \tag{2.58}$$

The projection of $\hat{\mathsf{D}}_2$ onto the orthogonal complement of $\hat{\mathsf{D}}_1$ delivers a matrix with a single column:

$$\hat{\hat{\mathsf{D}}}_2 = \hat{\mathsf{D}}_2 - \hat{\mathsf{D}}_1 \hat{\mathsf{D}}_1^+ \hat{\mathsf{D}}_2 \tag{2.59}$$

The singular vector corresponding to the smallest singular value of such a matrix is always 1 (i.e. a scalar). Thus the first circle coefficient $c_1 \equiv 1$. The coefficients c_2 and c_3 are determined by a first step of back substitution:

$$\begin{bmatrix} c_2 \\ c_3 \end{bmatrix} = -\hat{\mathsf{D}}_1^+ \hat{\mathsf{D}}_2 c_1 \tag{2.60}$$

Finally, the fourth circle coefficient c_4 is found by another back substitution step:

$$c_4 = -\begin{bmatrix} \overline{x^2 + y^2} & 0 & 0 \end{bmatrix} \begin{bmatrix} c_1 & c_2 & c_3 \end{bmatrix}^T = -\overline{x^2 + y^2}. \tag{2.61}$$

Geometric interpretation of the solution

Due to the implicit constraint $c_1 = 1$, the algebraic error measure, that is minimized by this approach can be geometrically interpreted. The error measure is:

$$e_i = x_i^2 + y_i^2 + c_2 x_i + c_3 y_i + c_4 \tag{2.62}$$

Without introducing a disturbing scaling factor, the right hand side of Equation 2.62 can be rewritten as:

$$e_i = (x_i - x_c)^2 + (y_i - y_c)^2 - r^2 = \|\boldsymbol{x}_i - \boldsymbol{x}_c\|^2 - r^2, \tag{2.63}$$

where $x_c = \frac{-c_2}{2}$, $y_c = \frac{-c_3}{2}$, $r = \sqrt{x_c^2 + y_c^2 - c_4}$, $\boldsymbol{x}_i = [x_i,\, y_i]^T$ and $\boldsymbol{x}_c = [x_c,\, y_c]^T$.

According to Figure 2.2a this measure can be interpreted as the square of tangential distance of the point $\boldsymbol{x}_i$ to the circle, in case that the point is outside the circle. If the point is inside the circle, the error measure $-e_i$ (In this case, e_i is negative, thus $-e_i$ is positive and its square root is real.) is the square of the distance of $\boldsymbol{x}_i$ to the circle, measured in the direction orthogonal to the vector $\boldsymbol{x}_i - \boldsymbol{x}_c$ (cp. Figure 2.2b).

The geometric meaning of the error measure indicates that the fitting result of the PTLS algorithm is invariant to Euclidean- or similarity transformations, since Euclidean distances are preserved or uniformly scaled, respectively, under such transformations.

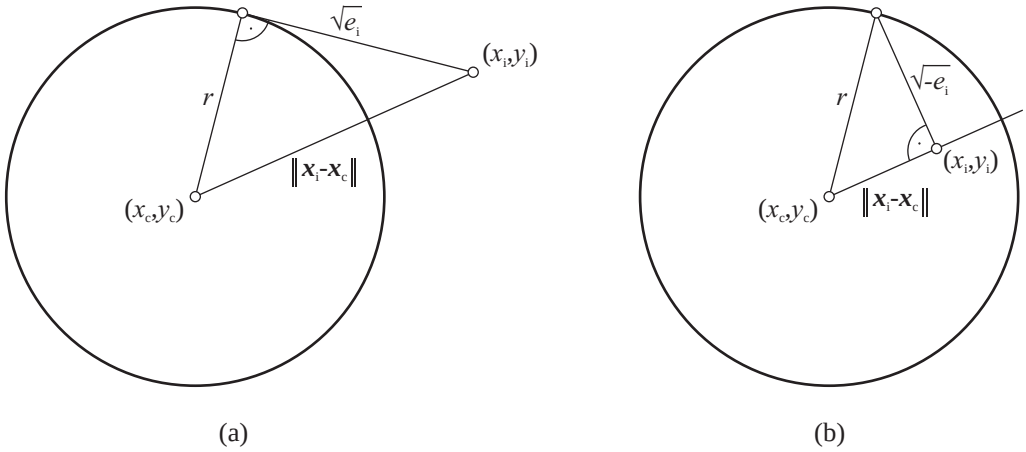

Figure 2.2: Geometric interpretation of the PTLS-error measure e_i with a point outside (a) and inside (b) the circle.

2.3.4 Geometric Circle Fitting using the Gauss-Newton Method

All circle fitting procedures presented in the previous subsections minimize some algebraic error measure. However, to find a maximum-likelihood estimation of a circle, the sum of squares of the geometric normal distances of the points to the circle must be minimized. The normal distance of a point $\boldsymbol{p}_i = [x_i,\, y_i]^T$ to a circle, represented by its center $\boldsymbol{p}_c = [x_c,\, y_c]^T$ and radius r, is given by (see Figure 2.3):

$$e_i = \sqrt{(x_c - x_i)^2 + (y_c - y_i)^2} - r \tag{2.64}$$

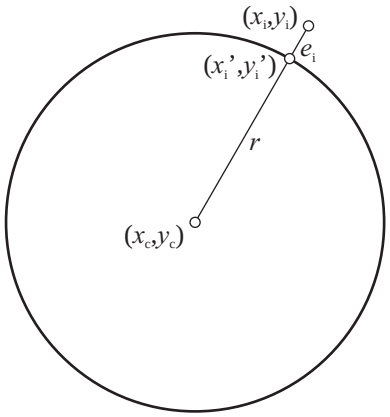

Figure 2.3: Normal distance e_i of a point to a circle

Defining a vector of parameters $\boldsymbol{v} = [x_c,\, y_c,\, r]^T$ and a function vector of geometric distances $\boldsymbol{e}(\boldsymbol{v}) = [e_1(\boldsymbol{v}),\, e_2(\boldsymbol{v}), \ldots e_N(\boldsymbol{v})]^T$ the problem of geometrically fitting a circle to N points in a least-squares sense is formulated as:

$$\min_{\boldsymbol{v}} E(\boldsymbol{v}) = \|\boldsymbol{e}(\boldsymbol{v})\| = \sum_{i=1}^{N} e_i^2(\boldsymbol{v}). \tag{2.65}$$

As the parameters x_c and y_c are not linear in Equation 2.64, there is no direct way to solve 2.65, but several iterative methods are found in literature [2, 3, 49]. The Gauss-Newton algorithm is exemplarily chosen and demonstrated here:

1. Find an initial parameter vector $\boldsymbol{v}^0$. Possible initializations: Either the solution of a direct fitting method is used; or the centroid of the points is selected as the initial center of the circle and the RMS-distance of the points to this center is chosen as the initial radius.

2. Iteratively update the parameter vector by
$$\boldsymbol{v}^{k+1} = \boldsymbol{v}^k + \boldsymbol{\delta}^k, \tag{2.66}$$
where $\boldsymbol{\delta}^k$ is computed by solving the linear system
$$\mathsf{J}_{\boldsymbol{e}}(\boldsymbol{v}^k)^T \mathsf{J}_{\boldsymbol{e}}(\boldsymbol{v}^k)\boldsymbol{\delta}^k = -\mathsf{J}_{\boldsymbol{e}}(\boldsymbol{v}^k)^T \boldsymbol{e}(\boldsymbol{v}^k), \tag{2.67}$$
and $\mathsf{J}_{\boldsymbol{e}}$ denotes the Jacobian matrix of the function vector $\boldsymbol{e}$ with respect to $\boldsymbol{v}$:
$$\mathsf{J}_{\boldsymbol{e}} = \begin{bmatrix} \frac{\partial e_1}{\partial x_c} & \frac{\partial e_1}{\partial y_c} & \frac{\partial e_1}{\partial r} \\ \vdots & \vdots & \vdots \\ \frac{\partial e_N}{\partial x_c} & \frac{\partial e_N}{\partial y_c} & \frac{\partial e_N}{\partial r} \end{bmatrix} = \begin{bmatrix} \frac{x_c - x_1}{\sqrt{(x_c-x_1)^2+(y_c-y_1)^2}} & \frac{y_c - y_1}{\sqrt{(x_c-x_1)^2+(y_c-y_1)^2}} & -1 \\ \vdots & \vdots & \vdots \\ \frac{x_c - x_N}{\sqrt{(x_c-x_N)^2+(y_c-y_N)^2}} & \frac{y_c - y_N}{\sqrt{(x_c-x_N)^2+(y_c-y_N)^2}} & -1 \end{bmatrix} \tag{2.68}$$

3. Stop the iteration when the norm of the update vector $\boldsymbol{\delta}^k$ falls below a defined threshold. Then $\boldsymbol{v}^{k+1}$ is taken as the solution of 2.65.

A MATLAB® implementation of the algorithm can be found in [23].

2.3.5 Back-transformation to Non-normalized Form

Considering the circle coefficients $\boldsymbol{c}_{norm} = [c_1, c_2, c_3, c_4]^T$, obtained by algebraic circle fitting to normalized data $\boldsymbol{p}_{i_{norm}}$, these coefficients need to be back transformed to fit the original data $\boldsymbol{p}_{i_{orig}}$. This is performed by arranging the circle coefficients in the respective conic matrix (see e.g. [35])
$$\mathsf{C}_{norm} = \begin{bmatrix} c_1 & 0 & \frac{c_2}{2} \\ 0 & c_1 & \frac{c_3}{2} \\ \frac{c_2}{2} & \frac{c_3}{2} & c_4 \end{bmatrix}, \tag{2.69}$$
and re-transforming this conic matrix by the transformation rule
$$\mathsf{C}_{orig} = \mathsf{T}_{norm}^T \mathsf{C}_{norm} \mathsf{T}_{norm}. \tag{2.70}$$
The circle coefficients $\boldsymbol{c}_{orig} = [c_{1_o}, c_{2_o}, c_{3_o}, c_{4_o}]^T$ are extracted from the conic matrix C_{orig}.

2.4 Calculating and Estimating a Homography from Point Correspondences

A homography is a linear mapping of points $\boldsymbol{p}' \leftrightarrow \boldsymbol{p}$ between two planes or between points on the same plane. If the points are represented by homogeneous 3-vectors, a homography is written as a non-singular 3×3 matrix:

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} \tag{2.71}$$

or short

$$\boldsymbol{p}' = \mathsf{H}\boldsymbol{p}. \tag{2.72}$$

Homographies may also be referred as collineations, projectivities, or projective transformations. The matrix H has nine entries and it is homogeneous, i.e. it is defined up to a non-zero scaling factor. Hence, H has 8 degrees of freedom. To estimate a homography from point correspondences, at least 4 correspondences, each contributing with 2 restrictions, must be known.

Given two sets of $N \geq 4$ corresponding points $\boldsymbol{p}'_i \leftrightarrow \boldsymbol{p}_i$, $i = \{1 \dots N\}$, the homography can be computed in the following way (referred to as "direct linear transformation (DLT) algorithm"):

Expanding Equation 2.71, and normalizing by the homogeneous component yields two equations for each point correspondence:

$$\frac{x'_i}{w'_i} = \frac{h_{11}x_i + h_{12}y_i + h_{13}w_i}{h_{31}x_i + h_{32}y_i + h_{33}w_i} \tag{2.73}$$

$$\frac{y'_i}{w'_i} = \frac{h_{21}x_i + h_{22}y_i + h_{23}w_i}{h_{31}x_i + h_{32}y_i + h_{33}w_i} \tag{2.74}$$

These equations can be rearranged to:

$$(-h_{11}x_i - h_{12}y_i - h_{13}w_i)w'_i + (h_{31}x_i + h_{32}y_i + h_{33}w_i)x'_i = 0 \tag{2.75}$$

$$(-h_{21}x_i - h_{22}y_i - h_{23}w_i)w'_i + (h_{31}x_i + h_{32}y_i + h_{33}w_i)y'_i = 0 \tag{2.76}$$

Therefore the equations for all point correspondences can be written in matrix form by defining

$$\boldsymbol{h} = [h_{11},\, h_{12},\, h_{13},\, h_{21},\, h_{22},\, h_{23},\, h_{31},\, h_{32},\, h_{33}]^T \tag{2.77}$$

and

$$\mathsf{D} = \begin{bmatrix} -x_1w'_1 & -y_1w'_1 & -w_1w'_1 & 0 & 0 & 0 & x_1x'_1 & y_1x'_1 & w_1x'_1 \\ 0 & 0 & 0 & -x_1w'_1 & -y_1w'_1 & -w_1w'_1 & x_1y'_1 & y_1y'_1 & w_1y'_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -x_Nw'_N & -y_Nw'_N & -w_Nw'_N & 0 & 0 & 0 & x_Nx'_N & y_Nx'_N & w_Nx'_N \\ 0 & 0 & 0 & -x_Nw'_N & -y_Nw'_N & -w_Nw'_N & x_Ny'_N & y_Ny'_N & w_Ny'_N \end{bmatrix} \tag{2.78}$$

as

$$\mathsf{D}\boldsymbol{h} = \mathbf{0}. \tag{2.79}$$

In case of $N = 4$ point correspondences, $\boldsymbol{h}$ is determined exactly as the null-space of D.

If more correspondences are given and the point coordinates are perturbed with noise, the system of equations 2.79 is overdetermined and an algebraic least-square estimation for $\boldsymbol{h}$ might be found by applying SVD on D and selecting the right singular vector corresponding to smallest singular value of D. However, the DLT algorithm does not factor in the error structure in D: Considering that in most cases the w_i and w'_i are ones, D is composed of statistical invariant entries, entries with the uncertainty of individual coordinates and entries with the uncertainty of the product of two coordinates. This leads to systematic errors in the least-squares solution obtained by the DLT algorithm. In [33] this issue is addressed and a more efficient algorithm is presented, that eliminates the estimation bias.

In [35] a total error function (referred to as "reprojection error") and an iterative method are discussed that find a maximum-likelihood estimation of the homography.

2.4.1 Back-transformation to Original Input Data

Back-transformation of the calculated or estimated homography is necessary, if the input data points have been normalized prior to the fitting procedure. Assuming that the normalization of the points $\boldsymbol{p}_i$ and the corresponding points $\boldsymbol{p}'_i$ was performed using the matrices T_{norm} and T'_{norm}, respectively, the back-transformation of the homography is

$$\mathsf{H}_{orig} = \mathsf{T}'^{-1}_{norm}\mathsf{H}_{norm}\mathsf{T}_{norm}. \tag{2.80}$$

Chapter 3

Error Propagation and Measurement Uncertainty

Any measurement data is subject to errors. In general the true value of the error is not known, but it can be estimated. This estimate is called measurement uncertainty. The terminology and definition of the uncertainty in measurement used in this thesis is according to the *ISO Guide to the Expression of Uncertainty in Measurement*[37]. The term "uncertainty" in this context means both the concept of "doubt about the validity of the result of a measurement" and "the specific quantities that provide quantitative measures of the concept". The uncertainty of the measurement result is an indication of its quality and it is obligatory to evaluate and express it, since without the statement of their uncertainties, measurement results cannot reasonably be compared with each other or with a reference standard.

Two principal ways to determine the measurement uncertainty of the final result, obtained with a metric vision system, are considered:

1. The direct way: Perform repeatability measurements on a reference target with known dimension. The dispersion of the results of these measurements is a measure of the precision error of the system; the mean deviation of the measurement results from the known dimension of the target yields the measurement bias. Requirements to the reference target are, that its "pseudo-true" dimension is determined utilizing an independent measurement system, and that the uncertainty associated with the known dimension must be significantly lower than the assumed or required uncertainty of the metric vision system.

2. Estimating the error propagation: Assuming that the uncertainty of the raw input data is known (e.g. the uncertainty of the position of a point in an image), the error propagation in each subsequent processing step is estimated. This estimation can be performed stochastically (Monte Carlo-simulation, see Section 3.2) or analytically (modelling the error propagation, see Section 3.3).

The aims of the direct way are to determine the uncertainty associated with an existing

measurement system, to proof that a measurement system fulfills the requirements on the accuracy of the results, to gain information about the uncertainty in raw measurement data, and to selectively verify the uncertainty determined by estimation.

Procedures to estimate the uncertainties associated with a measurement system are especially needed in the design and development phase. They enable the a priori estimation of the achievable accuracy of the system which is in development. Furthermore, the estimation procedures may yield knowledge about causes of errors and about sub-processes that are sensitive to errors.

This chapter is intended to present two methods to estimate the uncertainty in a processing result. Both methods are based on the tutorial on modelling uncertainty by Clarke[12] and the underlying work of Faugeras[18] and Press *et al.*[49]. The first method, Monte-Carlo simulation, is an approach that works with simulated measurement data. The results of the processing of this data are statistically evaluated to determine the uncertainty of the process output. The second method is based on the first order Taylor-series expansion of the function that describes the processing step and yields an analytic expression of the uncertainty of the result.

3.1 Expression of Uncertainty

In the context of this thesis the quantitative measure used to express the uncertainty in a measurement result is the variance of a measurement result or in case of a vector-valued measurement result the covariance matrix of the measurement values. This is closely related to the term "standard uncertainty" defined in the ISO Guide which expresses the uncertainty of a measurement as a standard deviation. However within the thesis the variance (and covariance matrix) is preferred to the standard deviation due to the more convenient possibility to express correlations between measurement values in the covariance matrix.

The covariance matrix of a random vector $\boldsymbol{X}$ is defined as

$$\Lambda_{\boldsymbol{X}} = \mathrm{E}[(\boldsymbol{X} - \mathrm{E}[\boldsymbol{X}])(\boldsymbol{X} - \mathrm{E}[\boldsymbol{X}])^T], \tag{3.1}$$

where $\mathrm{E}[\boldsymbol{X}]$ denotes the expectation value of $\boldsymbol{X}$. For a large number N of samples $\boldsymbol{x}_i$ drawn from $\boldsymbol{X}$, the expectation value can be approximated by the mean $\boldsymbol{\mu}_{\boldsymbol{x}}$:

$$\mathrm{E}[\boldsymbol{X}] = \boldsymbol{\mu}_{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i \tag{3.2}$$

3.2 Monte-Carlo Simulation

Monte-Carlo simulation is a numerical method to estimate the uncertainty of a processing result $\boldsymbol{y}_0$ when given the input data $\boldsymbol{x}_0$ and the uncertainty $\Lambda_{\boldsymbol{x}}$ associated with it[49].

The idea behind Monte-Carlo simulation is, that if a large number of measurements of the input data was given, a large population of output data could be computed. Its uncertainty could then be estimated from that population.

When applying the method of Monte-Carlo simulation, the population of input data $\boldsymbol{x}_i$ is not actually measured, but synthetically created. Each $\boldsymbol{x}_i$ is obtained by adding a random error to a vector of measured or modelled input data $\boldsymbol{x}_0$. The error structure must be related to the known covariance matrix $\Lambda_{\boldsymbol{x}}$. Each $\boldsymbol{x}_i$ is then used to compute a result $\boldsymbol{y}_i$. Finally the covariance matrix of the result is obtained by

$$\Lambda_{\boldsymbol{y}} = \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{y}_i - \boldsymbol{\mu}_{\boldsymbol{y}})(\boldsymbol{y}_i - \boldsymbol{\mu}_{\boldsymbol{y}})^T, \tag{3.3}$$

where $\boldsymbol{\mu}_{\boldsymbol{y}}$ denotes the mean vector (see Equation 3.2).

If the probability distribution function (PDF) of the error in the input data is known and the error is modelled according to this PDF, it is also possible to estimate the PDF of the error in the processing result $\boldsymbol{y}_0$.

3.3 Analytical Estimation of the Error Propagation

In this section an analytical approach to estimate the covariance matrix of a processing result is presented. Assuming that the input data is represented by the random vector $\boldsymbol{x}$ with mean $\boldsymbol{\mu}_{\boldsymbol{x}}$ and covariance matrix $\Lambda_{\boldsymbol{x}}$ and the process is mathematically expressed as a C^1-continuous function $\boldsymbol{f}$:

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}), \tag{3.4}$$

the covariance matrix $\Lambda_{\boldsymbol{y}}$ of the resulting random vector $\boldsymbol{y}$ can be estimated as shown in the following:

The Taylor-series expansion of $\boldsymbol{f}$ around $\boldsymbol{\mu}_{\boldsymbol{x}}$ yields

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{\mu}_{\boldsymbol{x}}) + \mathsf{J}_{\boldsymbol{f}}(\boldsymbol{\mu}_{\boldsymbol{x}})(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}}) + O(\|(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}})\|^2), \tag{3.5}$$

where $\mathsf{J}_{\boldsymbol{f}}$ denotes the Jacobian of the function $\boldsymbol{f}$ (see Subsection 3.3.1 and following for a detailed explanation). The mean of the output vector is approximated by

$$\boldsymbol{\mu}_{\boldsymbol{y}} \approx \boldsymbol{f}(\boldsymbol{\mu}_{\boldsymbol{x}}). \tag{3.6}$$

This relation is exact if the function $\boldsymbol{f}$ is a linear function of $\boldsymbol{x}$. The covariance matrix of $\boldsymbol{y}$ is then obtained by

$$\Lambda_{\boldsymbol{y}} = \mathrm{E}[(\boldsymbol{y} - \boldsymbol{\mu}_{\boldsymbol{y}})(\boldsymbol{y} - \boldsymbol{\mu}_{\boldsymbol{y}})^T] \tag{3.7}$$

Utilizing the approximation of the mean $\boldsymbol{\mu}_{\boldsymbol{y}}$ (Equation 3.6) and taking the first-order term of the Taylor-series expansion of $\boldsymbol{f}(\boldsymbol{x})$ the covariance of $\boldsymbol{y}$ is approximately given by

$$\begin{aligned}
\Lambda_{\boldsymbol{y}} &\approx \mathrm{E}[(\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{\mu}_{\boldsymbol{x}}))(\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{\mu}_{\boldsymbol{x}}))^T] \\
&\approx \mathrm{E}[(\mathsf{J}_{\boldsymbol{f}}(\boldsymbol{\mu}_{\boldsymbol{x}})(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}}))(\mathsf{J}_{\boldsymbol{f}}(\boldsymbol{\mu}_{\boldsymbol{x}})(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}}))^T] \\
&= \mathrm{E}[\mathsf{J}_{\boldsymbol{f}}(\boldsymbol{\mu}_{\boldsymbol{x}})(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}})(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}})^T \mathsf{J}_{\boldsymbol{f}}(\boldsymbol{\mu}_{\boldsymbol{x}})^T] \\
&= \mathsf{J}_{\boldsymbol{f}}(\boldsymbol{\mu}_{\boldsymbol{x}})\mathrm{E}[(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}})(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}})^T]\mathsf{J}_{\boldsymbol{f}}(\boldsymbol{\mu}_{\boldsymbol{x}})^T \\
&= \mathsf{J}_{\boldsymbol{f}}(\boldsymbol{\mu}_{\boldsymbol{x}})\Lambda_{\boldsymbol{x}}\mathsf{J}_{\boldsymbol{f}}(\boldsymbol{\mu}_{\boldsymbol{x}})^T
\end{aligned} \tag{3.8}$$

This approach to estimate the covariance matrix of a computational result is consistent with the methods to determine the "combined standard uncertainty" as described in the *ISO Guide to the Expression of Uncertainty in Measurement*[37], however outruns their restriction to a scalar output. It is presented in several publications and used for various applications (e.g. [4, 14, 18]). Monte Carlo simulation can be used to confirm the goodness of the approximation.

As can be seen in Equation 3.8, the explicit functional relationship $\boldsymbol{f}$ between $\boldsymbol{x}$ and $\boldsymbol{y}$ is not required to determine the first order approximation of the output covariance matrix; only its Jacobian needs to be known. This enhances the scope of computational processes which this method can be applied to. In the following subsections, the computation of the Jacobian is discussed for explicit and implicit functions and for linear least squares problems.

3.3.1 The Jacobian of Explicit Functions

If the output $\boldsymbol{y} \in \mathbb{R}^n$ is obtained by an explicit vector valued function $\boldsymbol{f}$: $\mathbb{R}^m \mapsto \mathbb{R}^n$ of the input data $\boldsymbol{x} \in \mathbb{R}^m$

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}), \tag{3.9}$$

the Jacobian matrix is obtained by taking the partial derivatives:

$$\mathsf{J}_{\boldsymbol{f}} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_m} \end{bmatrix} \tag{3.10}$$

3.3.2 The Jacobian of Implicit Functions

If the mathematical relation $\boldsymbol{f}$ between the input $\boldsymbol{x}$ and output $\boldsymbol{y}$ is defined as an implicit function

$$\boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{f}(\boldsymbol{x})) = \boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y}) = \mathbf{0}, \tag{3.11}$$

whereby

$$\boldsymbol{\Phi}: \quad \mathbb{R}^m \times \mathbb{R}^n \longrightarrow \mathbb{R}^n \quad \text{and} \tag{3.12}$$

$$\boldsymbol{f}: \quad \mathbb{R}^m \longrightarrow \mathbb{R}^n \tag{3.13}$$

then, according to the *Implicit Functions Theorem*[18], the Jacobian $\mathsf{J}_{\boldsymbol{f}} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}$ exists if

$$\det\left(\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{y}}\right) \neq 0, \tag{3.14}$$

and is found as

$$\mathsf{J}_{\boldsymbol{f}} = -\left(\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{y}}\right)^{-1} \frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{x}} \tag{3.15}$$

3.3.3 Homogeneous Linear Least Squares Problems

A special case of implicit functions are linear least squares problems of the form

$$\min_{\boldsymbol{y}} \|\mathsf{D}\boldsymbol{y}\| \quad \text{subject to} \quad \|\boldsymbol{y}\| = 1. \tag{3.16}$$

In this formulation the input data $\boldsymbol{x}$ is packed into the design matrix D. All algebraic fitting routines (for lines, circles and homographies) presented in Chapter 2 can be ascribed to such a problem formulation.

As shown in Chapter 2, the least squares solution to Equation 3.16 is either obtained by Singular Value Decomposition (SVD) of D or by solving the eigenvalue problem

$$\mathsf{S}\boldsymbol{y} = \lambda \boldsymbol{y}, \tag{3.17}$$

whereby the scatter matrix S is defined as $\mathsf{S} = \mathsf{D}^T\mathsf{D}$, and selecting the eigenvector $\boldsymbol{y}_{\min}$ corresponding to the smallest eigenvalue $\lambda_{\min}$ of S as the solution to 3.16.

Both procedures (SVD and eigenvector computation) deliver the same solution, however the latter way is used to derive an approximation of the Jacobian of this minimization process.

Using Equation 3.17 and assuming that the eigenvalue $\lambda_{\min} \approx 0$, $\boldsymbol{y}$ is implicitly defined as

$$\boldsymbol{\Phi} = \mathsf{S}\boldsymbol{y} = \mathbf{0}. \tag{3.18}$$

Hence,

$$\left(\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{y}}\right)^{-1} = \mathsf{S}^{-1}. \tag{3.19}$$

According to Clarke[12] it is justified to approximate S^{-1} by the pseudo-inverse S^{+}, if the scatter matrix S is (nearly) singular. Furthermore,

$$\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{\Phi}}{\partial \mathsf{S}} \frac{\partial \mathsf{S}}{\partial \boldsymbol{x}}, \tag{3.20}$$

where $\frac{\partial \boldsymbol{\Phi}}{\partial \mathsf{S}}$ is an $N \times N^2$ matrix (N is the number of parameters, i.e. the length of $\boldsymbol{y}$)

$$\begin{aligned}\frac{\partial \boldsymbol{\Phi}}{\partial \mathsf{S}} &= \begin{bmatrix} \frac{\partial \Phi_1}{\partial S_{11}} & \frac{\partial \Phi_1}{\partial S_{12}} & \cdots & \frac{\partial \Phi_1}{\partial S_{1N}} & \frac{\partial \Phi_1}{\partial S_{21}} & \cdots & \frac{\partial \Phi_1}{\partial S_{2N}} & \cdots & \frac{\partial \Phi_1}{\partial S_{NN}} \\ \frac{\partial \Phi_2}{\partial S_{11}} & \frac{\partial \Phi_2}{\partial S_{12}} & \cdots & \frac{\partial \Phi_2}{\partial S_{1N}} & \frac{\partial \Phi_2}{\partial S_{21}} & \cdots & \frac{\partial \Phi_2}{\partial S_{2N}} & \cdots & \frac{\partial \Phi_2}{\partial S_{NN}} \\ \vdots & & & & \ddots & & & & \vdots \\ \frac{\partial \Phi_N}{\partial S_{11}} & \frac{\partial \Phi_N}{\partial S_{12}} & \cdots & \frac{\partial \Phi_N}{\partial S_{1N}} & \frac{\partial \Phi_N}{\partial S_{21}} & \cdots & \frac{\partial \Phi_N}{\partial S_{2N}} & \cdots & \frac{\partial \Phi_N}{\partial S_{NN}} \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{y}^T & \boldsymbol{0}^T & \cdots & \boldsymbol{0}^T \\ \boldsymbol{0}^T & \boldsymbol{y}^T & \cdots & \boldsymbol{0}^T \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0}^T & \boldsymbol{0}^T & \cdots & \boldsymbol{y}^T \end{bmatrix}\end{aligned} \tag{3.21}$$

and $\frac{\partial \mathsf{S}}{\partial \boldsymbol{x}}$ is an $N^2 \times NM$ matrix (M is the number of data sets $\boldsymbol{x}_i$)

$$\frac{\partial \mathsf{S}}{\partial \boldsymbol{x}} = \begin{bmatrix} \left(\frac{\partial S_{11}}{\partial \boldsymbol{x}_1}\right)^T & \left(\frac{\partial S_{11}}{\partial \boldsymbol{x}_2}\right)^T & \cdots & \left(\frac{\partial S_{11}}{\partial \boldsymbol{x}_M}\right)^T \\ \left(\frac{\partial S_{12}}{\partial \boldsymbol{x}_1}\right)^T & \left(\frac{\partial S_{12}}{\partial \boldsymbol{x}_2}\right)^T & \cdots & \left(\frac{\partial S_{12}}{\partial \boldsymbol{x}_M}\right)^T \\ \vdots & \vdots & \ddots & \vdots \\ \left(\frac{\partial S_{NN}}{\partial \boldsymbol{x}_1}\right)^T & \left(\frac{\partial S_{NN}}{\partial \boldsymbol{x}_2}\right)^T & \cdots & \left(\frac{\partial S_{NN}}{\partial \boldsymbol{x}_M}\right)^T \end{bmatrix} \tag{3.22}$$

According to Equation 3.15 the Jacobian of the linear-least-squares fitting process is then given as

$$\mathsf{J}_{\mathrm{LLS}} = -\left(\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{y}}\right)^{-1} \frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{x}} = -\mathsf{S}^{-1} \frac{\partial \boldsymbol{\Phi}}{\partial \mathsf{S}} \frac{\partial \mathsf{S}}{\partial \boldsymbol{x}} \tag{3.23}$$

3.4 Summary and Conclusion

Two methods to estimate the uncertainty of a processing result from the input data and the uncertainty of the input data have been presented. The first method, Monte-Carlo simulation, is based on simulated measurement data. The uncertainty of the processing

result is thereby computed from the synthetically large population of output data. The second method utilizes the first-order Taylor approximation of the process function to analytically derive an estimation of the uncertainty of the processing result.

The advantages of each method are outlined in the following (the negated expression of each item can be considered as a disadvantage of the other method):

- Monte-Carlo simulation:
    1. The computational process is not approximated.
    2. There is no assumption on the error distribution of the input data. Any probability distribution function (PDF) of the error in the input data can be modelled.
    3. Assuming that the correct PDF of the error in the input data is modelled, the PDF of the error in the result can be estimated in addition to the covariance matrix of the result.
    4. The implementation of Monte-Carlo simulation is simple.
- Analytic estimation:
    1. Once implemented for a specific process, the analytic estimation of the uncertainty in the result is computationally fast. This is especially advantageous if many parameters need to be varied over a large measurement range.

Chapter 4

Uncertainty of Lines and Derived Quantities

In this chapter, the uncertainties associated with lines and quantities derived from line parameters are investigated. The error propagation within the following processes is considered:

1. Fitting a line to a set of noisy data points.
2. Computing the intersection point of two lines, which exhibit uncertainties in their parameters.
3. Computing the distance between two noisy intersection points.

The estimation of the uncertainty in the results of these processes is performed using the analytic approach based on first-order Taylor-series expansion presented in Section 3.3. A general model for the input data (and the associated uncertainty) of each of these processes is established, and the uncertainty expressions of all results are symbolically derived.

The results are numerically verified by means of Monte Carlo simulation. A simple measurement task is simulated to create numerical input data for the verification. The measurement task is very similar to the measurement problem in the video-extensometer application (see Chapter 8). It is briefly outlined in the following section.

4.1 Evaluation Procedure of the Simulated Measurement Task

In the simulated measurement example the task is to compute the distance between two vertices and its uncertainty. Each vertex is defined as the intersection point of two

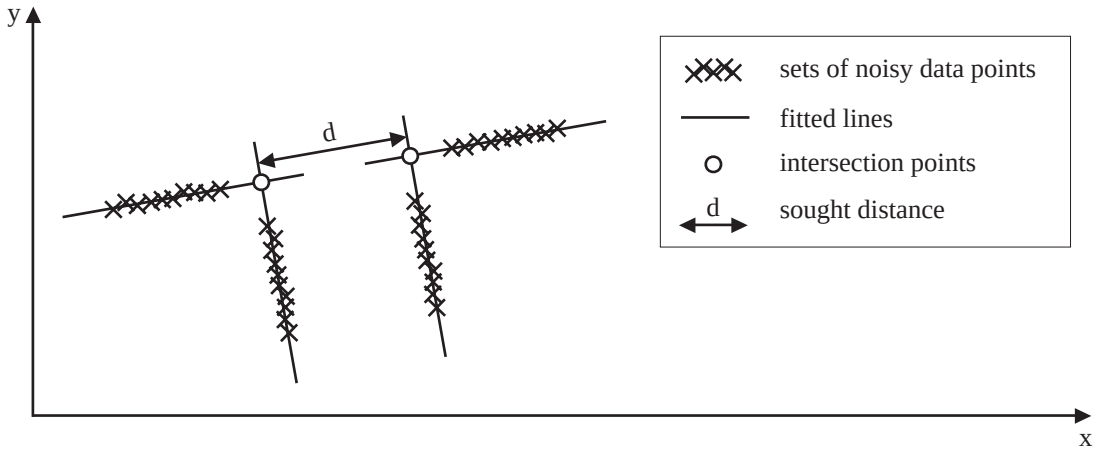

Figure 4.1: Simulated measurement example.

straight edges, which are each represented by a set of noisy points. Figure 4.1 visualizes the measurement situation.

The evaluation is performed in the following processing steps:

1. Fitting a line to each set of data points.
2. Computing the intersection points of each pair of lines.
3. Computing the distance between the two intersection points.

4.2 Covariance Matrix of Line Parameters

The estimation of the uncertainty associated with the parameters of a line, which is fitted to a set of noisy data points, is presented in this section. A general model for the point set is established and the analytical approach to model the error propagation is used to symbolically derive a functional relationship between the covariance matrix of the line parameters and the parameters of the point set model. The result is verified numerically by means of Monte-Carlo simulation.

The point set model is parameterized as follows: A set of M points is given. The points are equally distributed on a segment of the line with length s. The centroid of the points is $\boldsymbol{p}_m = [x_m,\, y_m]^T$. The orientation of the line (equivalently the major orientational axis of the point set) is given as the angle α to the x-axis. The model is visualized in Figure 4.2. According to this model, the expected coordinates (i.e. without deviation due to noise) of a single point $\boldsymbol{p}_i$, whereby $i = \{0 \dots M-1\}$, are given as

$$\boldsymbol{p}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} t_i \cos(\alpha) + x_m \\ t_i \sin(\alpha) + y_m \end{bmatrix}, \text{ where } \quad t_i = -\frac{s}{2} + \frac{s}{M-1} i. \tag{4.1}$$

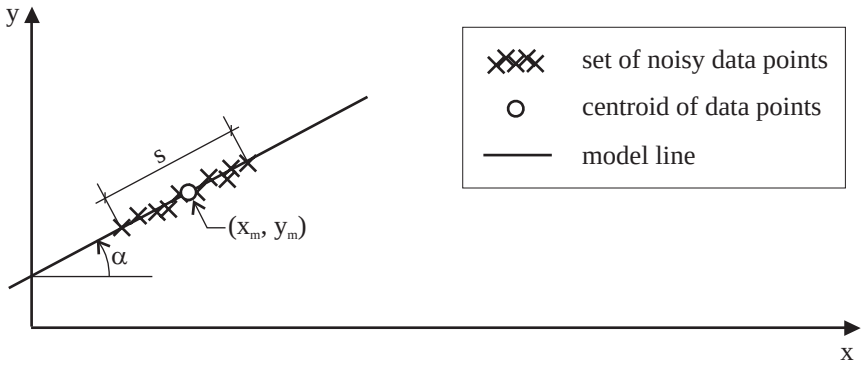

Figure 4.2: Parameters of the point set model

The uncertainty of the coordinates of the points is assumed to be equal in each direction, independent, and Gaussian distributed with standard deviation σ. Thus the covariance matrix of a single point is given as

$$\Lambda_{\boldsymbol{p}_i} = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix} \tag{4.2}$$

Knowing the model for the point set, the expected line parameters of the line fitted to this set is obtained by

$$\boldsymbol{l} = \begin{bmatrix} \sin(\alpha) \\ -\cos(\alpha) \\ \sin(\alpha)x_m - \cos(\alpha)y_m \end{bmatrix} \tag{4.3}$$

To estimate the covariance matrix of the line parameters analytically, the assumed line fitting routine (see Section 2.2) is broken down into its individual steps. The normalization of the data (see Section 2.1) is neglected for this error propagation analysis, since scaling is done for numerical reasons only and does not influence the uncertainty of the result, and shifting the centroid to the origin is already inherent to the fitting routine due to the orthogonal residualization of the design matrix.

4.2.1 Uncertainty of the Mean Free Points

In the course of redefining the design matrix by orthogonal residualization (cp. Equations 2.13 and 2.19), the points are made mean free, i.e. shifted to the origin:

$$\hat{\boldsymbol{p}}_i = \boldsymbol{p}_i - \boldsymbol{p}_m = \begin{bmatrix} \hat{x}_i \\ \hat{y}_i \end{bmatrix} = \begin{bmatrix} t_i \cos(\alpha) \\ t_i \sin(\alpha) \end{bmatrix} \tag{4.4}$$

However, this does not change the uncertainty of an individual point. Thus,

$$\Lambda_{\hat{\boldsymbol{p}}_i} = \Lambda_{\boldsymbol{p}_i} = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix} \tag{4.5}$$

4.2.2 Uncertainty of the Orientational Line Parameters

The orientational line parameters $\boldsymbol{l}_{12} = [l_1, l_2]^T$ are found by solving the homogeneous linear least squares problem defined in Equation 2.20.

According to Equation 3.8 the merged covariance matrix $\Lambda_{\boldsymbol{p}}$ of all points and the Jacobian J_{LLS} of the linear least squares problem are needed to estimate the error propagation. The matrix $\Lambda_{\boldsymbol{p}}$ is obtained by the cascading of the covariance matrices of the individual points to a $2M \times 2M$ diagonal matrix with all diagonal entries equal to σ^2:

$$\Lambda_{\boldsymbol{p}} = \begin{bmatrix} \Lambda_{\boldsymbol{p}_1} & & 0 \\ & \ddots & \\ 0 & & \Lambda_{\boldsymbol{p}_M} \end{bmatrix} = \begin{bmatrix} \sigma^2 & 0 & & & 0 \\ 0 & \sigma^2 & & & \\ & & \ddots & & \\ & & & \sigma^2 & 0 \\ 0 & & & 0 & \sigma^2 \end{bmatrix} \tag{4.6}$$

The Jacobian J_{LLS} is found as described in Subsection 3.3.3: In this special case the scatter matrix S turns out as

$$\mathsf{S} = \mathsf{D}^T\mathsf{D} = \frac{s^2 M(M+1)}{12(M-1)} \begin{bmatrix} \cos^2\alpha & \cos\alpha\sin\alpha \\ \cos\alpha\sin\alpha & \sin^2\alpha \end{bmatrix}. \tag{4.7}$$

Since the model points are perfectly aligned along a line, the scatter matrix S is singular and does not have an inverse. Thus the S^{-1} is approximated by its pseudo-inverse S^+:

$$\mathsf{S}^{-1} \approx \mathsf{S}^+ = \frac{12(M-1)}{s^2 M(M+1)} \begin{bmatrix} \cos^2\alpha & \cos\alpha\sin\alpha \\ \cos\alpha\sin\alpha & \sin^2\alpha \end{bmatrix}. \tag{4.8}$$

Furthermore $\frac{\partial \boldsymbol{\Phi}}{\partial \mathsf{S}}$ and $\frac{\partial \mathsf{S}}{\partial \boldsymbol{p}}$ are needed as defined in Equations 3.21 and 3.22:

$$\frac{\partial \boldsymbol{\Phi}}{\partial \mathsf{S}} = \begin{bmatrix} \boldsymbol{l}_{12}^T & \boldsymbol{0}^T \\ \boldsymbol{0}^T & \boldsymbol{l}_{12}^T \end{bmatrix} = \begin{bmatrix} \sin\alpha & -\cos\alpha & 0 & 0 \\ 0 & 0 & \sin\alpha & -\cos\alpha \end{bmatrix}, \tag{4.9}$$

$$\frac{\partial \mathsf{S}}{\partial \boldsymbol{p}} = \begin{bmatrix} 2\hat{x}_1 & 0 & 2\hat{x}_2 & 0 & \cdots & 2\hat{x}_M & 0 \\ \hat{y}_1 & \hat{x}_1 & \hat{y}_2 & \hat{x}_2 & \cdots & \hat{y}_M & \hat{x}_M \\ \hat{y}_1 & \hat{x}_1 & \hat{y}_2 & \hat{x}_2 & \cdots & \hat{y}_M & \hat{x}_M \\ 0 & 2\hat{y}_1 & 0 & 2\hat{y}_2 & \cdots & 0 & 2\hat{y}_M \end{bmatrix}. \tag{4.10}$$

Thus,

$$\mathsf{J}_{\text{LLS}} = -\mathsf{S}^+ \frac{\partial \boldsymbol{\Phi}}{\partial \mathsf{S}} \frac{\partial \mathsf{S}}{\partial \boldsymbol{p}} = -\begin{bmatrix} \mathsf{J}_1 & \mathsf{J}_2 & \cdots & \mathsf{J}_M \end{bmatrix}, \tag{4.11}$$

where

$$\mathsf{J}_i = \frac{12(M-1)}{s^2 M(M+1)} \begin{bmatrix} t_i \cos\alpha\sin\alpha & -t_i\cos^2\alpha \\ t_i \sin^2\alpha & -t_i\cos\alpha\sin\alpha \end{bmatrix}, \text{and} \quad t_i = -\frac{s}{2} + \frac{s}{M-1} i. \tag{4.12}$$

Hence the covariance matrix of the orientational line parameters is:

$$\Lambda_{\boldsymbol{l}_{12}} = \mathsf{J}_{\mathrm{LLS}} \Lambda_{\boldsymbol{p}} \mathsf{J}_{\mathrm{LLS}}^T = \frac{12\sigma^2(M-1)}{s^2 M(M+1)} \begin{bmatrix} \cos^2\alpha & \cos\alpha\sin\alpha \\ \cos\alpha\sin\alpha & \sin^2\alpha \end{bmatrix}. \tag{4.13}$$

4.2.3 Covariance of all Line Parameters

So far the orientational line parameters $\boldsymbol{l}_{12}$ and their covariance matrix $\Lambda_{\boldsymbol{l}_{12}}$ were determined. The complete line vector $\boldsymbol{l}$ is determined by the following functional relationship:

$$\boldsymbol{l} = \boldsymbol{f}(l_1, l_2, x_m, y_m) = \begin{bmatrix} l_1 & l_2 & -(l_1 x_m + l_2 y_m) \end{bmatrix}^T. \tag{4.14}$$

The Jacobian $\mathsf{J}_{\boldsymbol{l}}$ of this function $\boldsymbol{f}$ is

$$\mathsf{J}_{\boldsymbol{l}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -x_m & -y_m & -l_1 & -l_2 \end{bmatrix} \tag{4.15}$$

The covariance matrix of the input parameters $\Lambda_{\boldsymbol{l}\boldsymbol{x}_m}$ is obtained by merging $\Lambda_{\boldsymbol{l}_{12}}$ and the covariance matrix of the the centroid $[x_m,\, y_m]^T$. As the centroid coordinates are merely the mean values of the point coordinates, their variances are $\sigma^2_{x_m} = \sigma^2_{y_m} = \frac{\sigma^2}{M}$. According to the model, the individual point coordinates are uncorrelated, hence their mean values are as well. Furthermore the centroid coordinates are not correlated with the orientational line parameters. Thus the merged covariance matrix is

$$\Lambda_{\boldsymbol{l}\boldsymbol{x}_m} = \begin{bmatrix} \Lambda_{\boldsymbol{l}_{12}} & \mathsf{0} \\ \mathsf{0} & \begin{bmatrix} \frac{\sigma^2}{M} & 0 \\ 0 & \frac{\sigma^2}{M} \end{bmatrix} \end{bmatrix} \tag{4.16}$$

Finally the covariance matrix of the line parameters is

$$\Lambda_{\boldsymbol{l}} = \mathsf{J}_{\boldsymbol{l}} \Lambda_{\boldsymbol{l}\boldsymbol{x}_m} \mathsf{J}_{\boldsymbol{l}}^T = k \begin{bmatrix} \cos^2\alpha & \cos\alpha\sin\alpha & -x_m\cos^2\alpha - y_m\cos\alpha\sin\alpha \\ & \sin^2\alpha & -x_m\cos\alpha\sin\alpha - y_m\sin^2\alpha \\ \text{sym.} & & (x_m\cos\alpha + y_m\sin\alpha)^2 + \frac{\sigma^2}{kM} \end{bmatrix}, \tag{4.17}$$

where $k = \frac{12(M-1)}{s^2 M(M+1)}\sigma^2$.

Analyzing the factor k, which is common to all matrix entries, the following conclusions can be drawn:

1. The variances and covariances of the line parameters increase linearly with the variance σ^2 of the individual points.
2. Neglecting the fraction $\frac{M-1}{M+1}$, which converges to 1 for a large number of points M, the entries of the covariance matrix decrease linearly with M.

3. The uncertainty of the line parameters decreases quadratically with the length s of the line segment on which the points are distributed.

4. Considering Conclusions 2 and 3 with respect to image processing, the length of a straight image feature (e.g. an edge or a laser line) has an inverse cubic influence on the uncertainty of the line fitted to it. This is due to the fact that in general with increasing length s the number of extracted pixel coordinates (i.e. the number of input points M) increases linearly.

The individual entries of the covariance matrix reveal the following:

1. The position of the centroid $\boldsymbol{p}_m$ does not influence the uncertainty of the orientational line parameters, but only the variance of the third line parameter and its associated covariances.

2. The variance of the third line parameter $\Lambda_{l(3,3)}$ comprises two parts: The distance of the centroid from the origin in the direction of the line orientation has a quadratic influence. Additionally the uncertainty of the centroid $\frac{\sigma^2}{M}$ increases the uncertainty of $\Lambda_{l(3,3)}$.

4.2.4 Numerical Verification

Monte-Carlo simulation is used to numerically determine the uncertainty of the line covariance matrix. The result is then compared to the analytically derived covariance matrix. Table 4.1 summarizes the chosen parameter values for numerical verification.

Parameter	Value	Parameter	Value
M	50	x_m	50.37 units
s	50 units	y_m	93.91 units
α	7°	σ^2	25 units^2

Table 4.1: Parameters for the numerical verification of the covariance matrix of a fitted line.

The analytic estimation of the covariance matrix $\Lambda_{\boldsymbol{l}\text{an}}$ is immediately obtained using Equation 4.17. For Monte-Carlo simulation, 1000 noisy point sets are created and a line is fitted to each set. The numerical estimation of the covariance matrix $\Lambda_{\boldsymbol{l}\text{MC}}$ is then obtained by numerically evaluating Equation 3.3. The results of both estimation methods are:

$$\Lambda_{\boldsymbol{l}\text{an}} = \begin{bmatrix} 0.0023 & 0.0003 & -0.1406 \\ 0.0003 & 0.0000 & -0.0173 \\ -0.1406 & -0.0173 & 9.2049 \end{bmatrix}, \quad \Lambda_{\boldsymbol{l}\text{MC}} = \begin{bmatrix} 0.0024 & 0.0003 & -0.1470 \\ 0.0003 & 0.0000 & -0.0187 \\ -0.1470 & -0.0187 & 9.6982 \end{bmatrix} \tag{4.18}$$

The relative error between the two results is

$$e_{\text{rel}} = \frac{\|\Lambda_{\boldsymbol{l}\text{an}} - \Lambda_{\boldsymbol{l}\text{MC}}\|_2}{\|\Lambda_{\boldsymbol{l}\text{MC}}\|_2} = 5.09\%, \tag{4.19}$$

indicating that the analytical estimation yields a reasonable approximation of the covariance matrix. The differences between the Monte-Carlo- and the analytical estimations can be explained as follows: The approximation of the implicit fitting function ($\boldsymbol{\Phi} = \mathsf{S}\boldsymbol{y} = \boldsymbol{0}$ instead of $\boldsymbol{\Phi} = \mathsf{S}\boldsymbol{y} - \lambda\boldsymbol{y} = \boldsymbol{0}$, cp. Equation 3.18) leads to an error in the analytic estimation. The approximation $\lambda \approx 0$ is too unprecise, due to the high noise level of the input points, which was chosen to gain an obvious effect in the visualization of the results

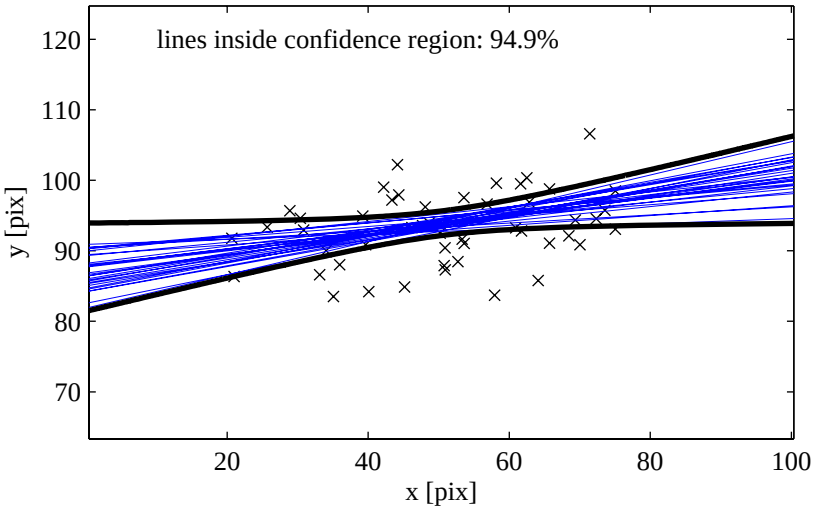

Figure 4.3: Uncertainty of a fitted line: Example point set (crosses), 30 randomly selected lines of a total of 1000 obtained by Monte-Carlo simulation (thin lines) and the 95%-confidence region of the fitted line (thick curve).

The results of the numerical verification of the covariance matrix estimation are visualized in Figure 4.3. An example point set, a random selection of 30 lines obtained from Monte-Carlo simulation and a hyperbolic approximation of the 95%-confidence region of the fitted lines are shown. The confidence region is based on the Mahalanobis distance and created using the analytically estimated covariance matrix of the line parameters[49]. In this special case the duality between points and lines[25] in projective geometry is used to transform the actually elliptical confidence region of the line parameters (i.e. in line space) to its hyperbolic representation in the point space[12]. A further evaluation exhibits, that 94.9% of the 1000 lines of Monte-Carlo simulation lie inside the 95%-confidence region, which is based on the analytically derived covariance matrix. This confirms the near equivalence of the two estimations of the covariance matrix.

Relating to the simulated measurement example, the covariance matrices of the other three lines are also analytically and numerically estimated. A comparison of the results obtained with the two methods indicates similar relative errors of approximately 5% for all lines.

4.3 Uncertainty of an Intersection Point of two Lines

Given two lines $\boldsymbol{l}$ and $\boldsymbol{g}$ and their covariance matrices $\Lambda_{\boldsymbol{l}}$ and $\Lambda_{\boldsymbol{g}}$, respectively, the intersection point (represented in homogeneous coordinates) is found by calculating the cross product of the line vectors:

$$\boldsymbol{p}_h = \begin{bmatrix} x_h \\ y_h \\ w_h \end{bmatrix} = \boldsymbol{l} \times \boldsymbol{g} = \begin{bmatrix} l_2 g_3 - l_3 g_2 \\ l_3 g_1 - l_1 g_3 \\ l_1 g_2 - l_2 g_1 \end{bmatrix} \tag{4.20}$$

To analytically estimate the covariance matrix $\Lambda_{\boldsymbol{p}_h}$ of the intersection point coordinates, the merged covariance matrix of the two lines

$$\Lambda_{\boldsymbol{lg}} = \begin{bmatrix} \Lambda_{\boldsymbol{l}} & 0 \\ 0 & \Lambda_{\boldsymbol{g}} \end{bmatrix}, \tag{4.21}$$

and the Jacobian of the cross product

$$\mathsf{J}_\times = \begin{bmatrix} 0 & g_3 & -g_2 & 0 & -l_3 & l_2 \\ -g_3 & 0 & g_1 & l_3 & 0 & -l_1 \\ g_2 & -g_1 & 0 & -l_2 & l_1 & 0 \end{bmatrix} \tag{4.22}$$

are required. The covariance matrix of the intersection point is then computed as

$$\Lambda_{\boldsymbol{p}_h} = \mathsf{J}_\times \Lambda_{\boldsymbol{lg}} \mathsf{J}_\times^T, \tag{4.23}$$

which delivers a lengthy expression, that can easily be evaluated with a symbolic math processor. The affine coordinates of the intersection points are determined as

$$\boldsymbol{p}_a = \begin{bmatrix} x_a & y_a \end{bmatrix}^T = \begin{bmatrix} \frac{x_h}{w_h} & \frac{y_h}{w_h} \end{bmatrix}^T. \tag{4.24}$$

The Jacobian of this operation is

$$\mathsf{J}_a = \begin{bmatrix} \frac{1}{w_h} & 0 & -\frac{x_h}{w_h^2} \\ 0 & \frac{1}{w_h} & -\frac{y_h}{w_h^2} \end{bmatrix} \tag{4.25}$$

Finally the covariance matrix of the intersection point, represented in affine coordinates, is determined as

$$\Lambda_{\boldsymbol{p}_a} = \mathsf{J}_a \Lambda_{\boldsymbol{p}_h} \mathsf{J}_a^T. \tag{4.26}$$

This expression can also be evaluated using a symbolic math processor.

4.3.1 Numerical Verification

The result of the analytical estimation of the covariance matrix of an intersection point is also numerically verified utilizing Monte-Carlo simulation.

The input data to the simulation is chosen as follows: The first line is equivalent with the line in the numerical example of Section 4.2. The second line is determined by the centroid of a point set at $\boldsymbol{p}_m = [106.09,\ 50.37]$ and an angle to the x-axis of $\alpha = 97°$. The other parameters of the point set are equal to the parameters of the point set according to the first line.

The analytic estimates of line covariance matrices are used as the input to determine the analytic estimate of the covariance matrix of the intersection point. For Monte-Carlo simulation the lines obtained by the line-fitting simulation are reused rather than creating new noisy lines.

The numerical evaluation yields the following results:

$$\Lambda_{\boldsymbol{p}_a\mathrm{an}} = \begin{bmatrix} 6.2647 & 0.0000 \\ 0.0000 & 6.2647 \end{bmatrix}, \quad \Lambda_{\boldsymbol{p}_a\mathrm{MC}} = \begin{bmatrix} 6.8650 & 0.1725 \\ 0.1725 & 6.5472 \end{bmatrix} \tag{4.27}$$

The relative error is:

$$e_{\mathrm{rel}} = \frac{\|\Lambda_{\boldsymbol{p}_a\mathrm{an}} - \Lambda_{\boldsymbol{p}_a\mathrm{MC}}\|_2}{\|\Lambda_{\boldsymbol{p}_a\mathrm{MC}}\|_2} = 9.74\%, \tag{4.28}$$

The difference between the two estimated covariance matrices of the intersection points is caused by the nonlinearity of the process function. This nonlinearity is neglected in the analytic estimation procedure, due to the first-order approximation in the Taylor-series expansion.

The results are presented in Figure 4.4. The intersection points obtained by Monte-Carlo simulation, the 95%-line confidence regions and the 95%-point confidence region are shown. A further evaluation reveals, that 94.7% of the Monte-Carlo intersection points lie within the 95%-confidence region that is determined by the analytically estimated covariance matrix of the intersection point coordinates.

4.4 Uncertainty of the Distance between two Points

The third evaluation step in the measurement example is to compute the distance between the two intersection points and its uncertainty (variance).

The Euclidian distance between two points $\boldsymbol{p}_1 = [x_1,\ y_1]$ and $\boldsymbol{p}_2 = [x_2,\ y_2]$ is

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}. \tag{4.29}$$

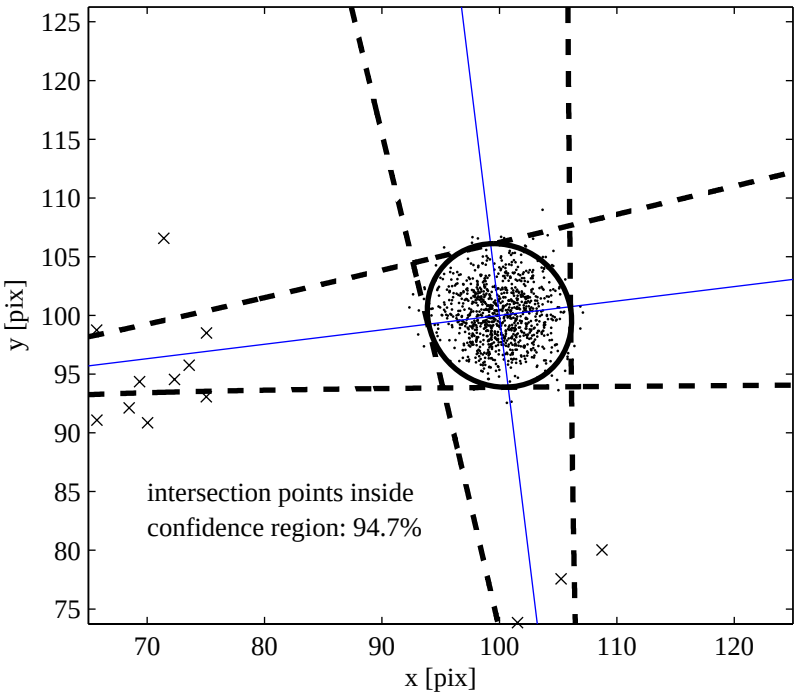

Figure 4.4: Uncertainty of an intersection point: Intersection points obtained by Monte-Carlo simulation (small dots), 95%-confidence limit of the intersection point (thick curve) and 95%-confidence limit of the intersecting lines (thick dashed curves).

The Jacobian of this evaluation turns out to be

$$\mathsf{J}_d = \frac{1}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}} \begin{bmatrix} (x_1 - x_2) & (y_1 - y_2) & -(x_1 - x_2) & -(y_1 - y_2) \end{bmatrix}. \quad (4.30)$$

Knowing the covariance matrices $\Lambda_{\boldsymbol{p}_1}$ and $\Lambda_{\boldsymbol{p}_1}$ of the two points, they can be merged to

$$\Lambda_{\boldsymbol{p}_{12}} = \begin{bmatrix} \Lambda_{\boldsymbol{p}_1} & 0 \\ 0 & \Lambda_{\boldsymbol{p}_2} \end{bmatrix}, \quad (4.31)$$

and the uncertainty of the distance is found as

$$\Lambda_d = \sigma_d^2 = \mathsf{J}_d \Lambda_{\boldsymbol{p}_{12}} \mathsf{J}_d^T \quad (4.32)$$

This yields a lengthy expression again, which can be evaluated with a symbolic math processor.

4.4.1 Numerical Verification

The analytically estimated variance of the distance is again verified by Monte-Carlo simulation. The positions of the two points as modelled in the measurement example are $\boldsymbol{p}_1 = [100,\ 100]^T$ and $\boldsymbol{p}_2 = [149.63,\ 106.09]^T$. The associated covariance matrix has been

estimated in the previous numerical verification steps (cp. Equation 4.27). The results from analytical estimation and Monte-Carlo simulation are:

$$\sigma^2_{d,an} = 12.53, \quad \sigma^2_{d,MC} = 13.39. \tag{4.33}$$

The relative error between the two estimates is

$$e_{\text{rel}} = \frac{\sigma^2_{d,an} - \sigma^2_{d,MC}}{\sigma^2_{d,MC}} = -6.42\% \tag{4.34}$$

Figure 4.5 summarizes the numerical evaluations of a simulated distance measurement example. Exemplary point sets, 95%-line confidence regions, and 95%-intersection point confidence regions are shown, and the estimated standard deviations of the measured distance are given.

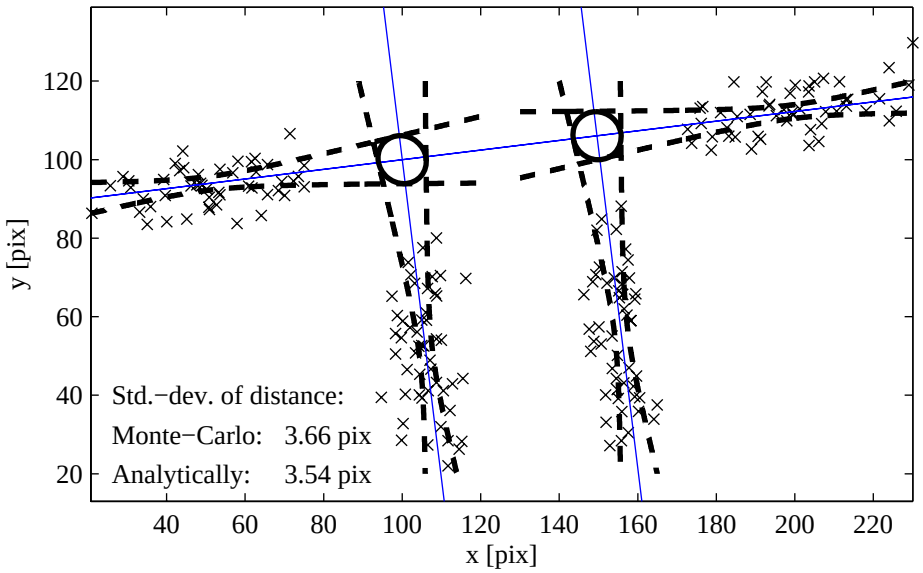

Figure 4.5: Uncertainty of distance calculation: Point sets (crosses), 95% line confidence regions (thick dashed curves), and 95% intersection point confidence regions (thick curves).

4.5 Summary and Conclusions

In this chapter the uncertainties in (1) line parameters estimated by fitting, (2) the intersection point of two lines, and (3) the distance between two points are analytically derived. The results are verified by means of Monte-Carlo simulation. From the analytical results and the comparison with the numerical Monte-Carlo results, the following conclusions can be drawn:

1. A relatively simple symbolic expression of the covariance matrix of the parameters of a line fitted to scattered data points is analytically derived. This expression

reveals, that the uncertainty in the individual data points has a linear influence on the covariance of the line parameters. Furthermore, it can be concluded from this expression, that the length of a straight image feature has an inverse cubic influence on the uncertainty of the line fitted to it.

2. Lengthy expressions are obtained for the covariance matrices of intersection points and for the variance of the distance between two points. However they can easily be handled using a symbolic math processors and enable a fast estimation of the uncertainties in the results of these computations.

3. The analytical approach delivers a good approximation of the uncertainty in the results of the discussed evaluation processes. Considering line fitting, deviations from the uncertainty estimates obtained by Monte-Carlo simulation are caused by the approximation of the implicit fitting function. The nonlinearity of the operations to compute the intersection point of two lines and the distance between two points leads to a deviation of the analytical uncertainty estimate from the Monte-Carlo estimate.

Chapter 5

Statistical Uncertainty Analysis of Circle Fitting

Fitting circles to noisy data points is a common task in image processing and metric vision. Circles are used as calibration marks and measurement marks (e.g. [20, 60]). Circular profiles are encountered in numerous machine vision applications (e.g. [40]) and rotation-symmetric parts are common in production processes. Along with the number of applications, the task of fitting circles is extensively covered in established and recent literature[2, 9, 24, 38] and a variety of algorithms, both algebraic and geometric, are available. However the issue of error propagation in the fitting process is neglected in literature, although the uncertainty of the fitting result is (together with the computational effort) a crucial selection criterion of a fitting algorithm. Even if independent isotropic Gaussian noise of the input data is assumed (which is common and justified in a bulk of machine vision applications), the distribution of the error in the fitting parameters may be anisotropic, non-Gaussian, and not independent. The resulting dependence (i.e. correlation) between fitting parameters is often overlooked, but drastically influences the uncertainty of results that are derived from these parameters.

This chapter is intended to analyze the error distribution and correlation of fitting parameters and the error propagation in different circle fitting algorithms. For these investigations Monte-Carlo simulation is preferred to the analytical approach, since, firstly, some operations in the circle fitting process are nonlinear, and secondly, Monte-Carlo simulations permits the determination of bias in the results. Common characteristics to all tests are that the data points are distributed equally on a circle or circular arc and that they are perturbed with isotropic independent Gaussian noise.

In the first part the tangent position is introduced as a new characteristic result of fitting circular arcs. It is a measure of the position of the fitted circle. Due to the correlation between center point coordinates and the radius of the fitted circle, the tangent position exhibits a lower uncertainty and its error distribution is, in contrast to the center point coordinates, strictly Gaussian.

The second part of this chapter investigates the sensitivity of four circle fitting algorithms on the following properties of the input data: (1) the number, (2) the uncertainty, and

(3) the subtending arc angle of the data points. Similar tests have been performed and published by Chernov *et al.*, but they investigated other fitting methods and their focus was on convergence properties of iterative methods [7, 8, 10]. In addition, in this work the error analysis of the circle fitting methods is explicitly broken down into bias and standard deviation of the result.

5.1 The Significance of the Tangent Position

In the case of fitting a circular arc (i.e. fitting a circle to noisy data points arranged along a circular arc), the center point of the found circle is statistically a poor estimate of the position of the arc, as will be shown within this section. The tangent point is introduced as a quantity that delivers a more accurate measure of the position.

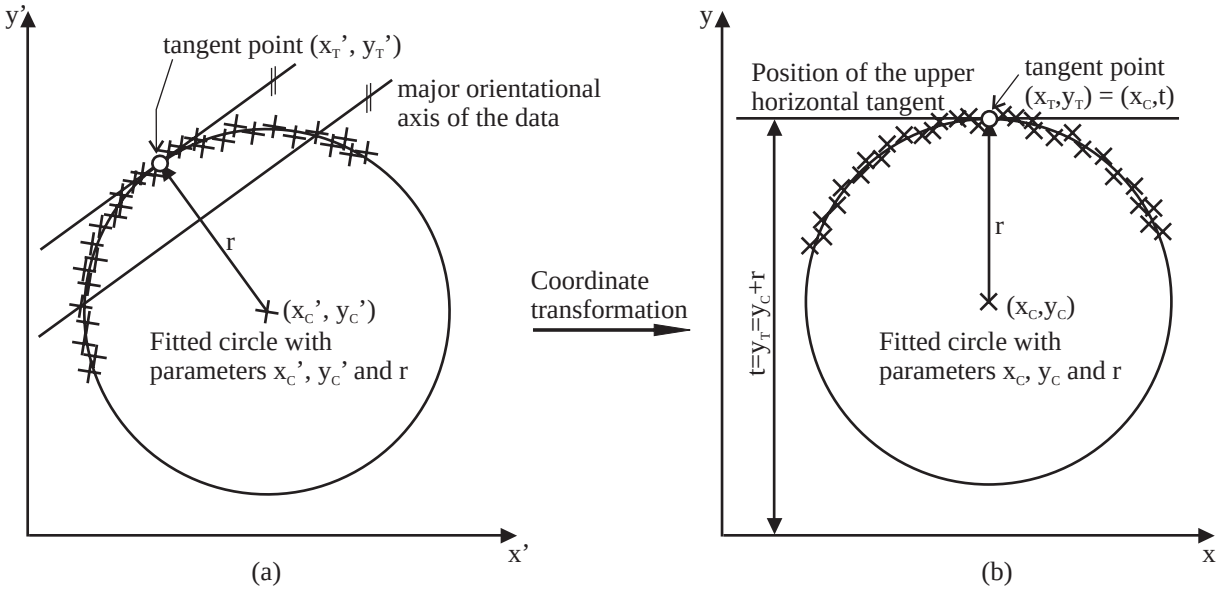

Figure 5.1: Definition of the tangent point (a) and the tangent position (b).

The tangent point is defined here as the contact point between the tangent parallel to the major orientational axis of the data and the circle (see Figure 5.1a). To simplify analysis of statistical properties of this tangent point, the data is assumed to be oriented, such that the tangent is parallel to the x-axis. This simplifies analysis without loss of generality. The tangent point can then be characterized by a single parameter t defined here as the tangent position (see Figure 5.1b):

$$t = y_c + r \tag{5.1}$$

5.1.1 Uncertainty and Distribution of the Tangent Position

Uncertainty

A Monte-Carlo simulation is performed to estimate the uncertainty of the tangent position and to compare it to the uncertainty of the center point and the radius: 100 points are equally distributed on a 90°-arc of the unit circle and perturbed with mean free Gaussian noise with a standard deviation of 0.05 (i.e. 5% of the radius). The circular arc is aligned as shown in Figure 5.1b. 10000 such point sets are generated and a circle is fitted to each set. The method of total least squares (see Subsection 2.3.2) is chosen for all circle fitting tasks in this investigation of the tangent position.

The results of the simulation show, that the error in the coordinates of the center point has an anisotropic distribution. The major axis of the center point distribution is aligned with the minor axis of the input data. This is the y-axis in the example shown in Figure 5.2. The standard deviation of the y-coordinate of the center point y_c is 0.061 and thus slightly larger than that of the individual input points. The uncertainty of the tangent position t is 0.008. This is approximately a factor of 8 better than the center point and almost a factor of 7 better than the individual points.

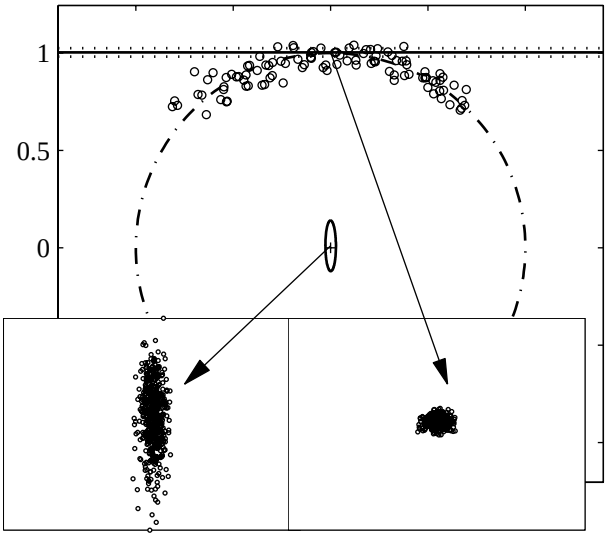

Figure 5.2: Results of the Monte-Carlo simulation: The uncertainty of the y-coordinate of the center point (left sub-plot) is approximately by a factor of 8 larger than the uncertainty of the tangent position (right sub-plot).

The correlation coefficients of the circle parameters x_c, y_c and r are shown in Table 5.1. There is a significant negative correlation between y_c and r, i.e. if a fit overestimates the radius, this will be compensated by a shift of the center point and vice versa. Therefore the tangent position, which is the sum of these quantities (cp. Equation 5.1), varies within closer limits and its standard deviation is smaller than that of y_c and r.

	x_c	y_c	r
x_c	1	−0.0179	0.0173
y_c	−0.0179	1	−0.9960
r	0.0173	−0.9960	1

Table 5.1: The correlation matrix for the evaluated parameters x_0, y_0 and r .

The covariance matrix of the fitting parameters y_c and r, which is obtained from the Monte-Carlo simulation is

$$\Lambda_{y_c,r} = \begin{bmatrix} \sigma^2_{y_c} & \sigma^2_{y_c,r} \\ \sigma^2_{y_c,r} & \sigma^2_r \end{bmatrix} = \begin{bmatrix} 0.00371 & -0.00337 \\ -0.00337 & 0.00308 \end{bmatrix}. \tag{5.2}$$

The magnitude of the off-diagonal entries $\sigma^2_{y_c,r}$ in this matrix also indicate the negative correlation between y_c and r. It can be shown, that this negative correlation is the reason for the lower uncertainty of the tangent position, by applying the theory of linear error propagation discussed in Chapter 3. The Jacobian of the tangent position with respect to y_c and r is (cp. Equation 5.1):

$$\mathsf{J}_t = \begin{bmatrix} \frac{\partial t}{\partial y_c} & \frac{\partial t}{\partial r} \end{bmatrix} = \begin{bmatrix} 1 & 1 \end{bmatrix}. \tag{5.3}$$

Thus, according to Equation 3.8, the variance of the tangent position is:

$$\sigma^2_t = \mathsf{J}_t \Lambda_{y_c,r} \mathsf{J}_t^T = \sigma^2_{y_c} + \sigma^2_r + 2\sigma^2_{y_c,r} = 5.6 \cdot 10^{-5}, \tag{5.4}$$

i.e. due to the negative and large (relative to the variances $\sigma^2_{y_c}$ and σ^2_r) value of $\sigma^2_{y_c,r}$, the uncertainty of the tangent position is small.

Distribution

The histogram of each circle parameter is shown in Figure 5.3. The histograms of y_c and r indicate a deviation from an ideal normal distribution. To verify this optical impression a χ^2-distribution test [49, 54] is performed for each parameter. The null hypothesis H_0 (a parameter is normally distributed) is tested against the alternative hypothesis H_A (it is not normally distributed). A significance level of $\alpha = 0.01$ is chosen and the data is grouped into $k = 30$ intervals. If an interval contains less than 5 elements, it is merged with an adjacent interval. The number of remaining intervals k_m is then used to select the limiting χ^2-quantile. The number of estimated distribution parameters is $c = 2$ (i.e. mean and standard deviation). Thus, the null hypothesis H_0 is rejected in favor of the alternative hypothesis H_A, if

$$\chi^2 > \chi^2_{1-\alpha,k_m-c-1} = \chi^2_{0.99,k_m-3} \tag{5.5}$$

The result of the χ^2-distribution test is shown in Table 5.2. It is confirmed, that the distribution of the y-coordinate and the radius is not Gaussian, but the x-coordinate and the tangent position are normally distributed.

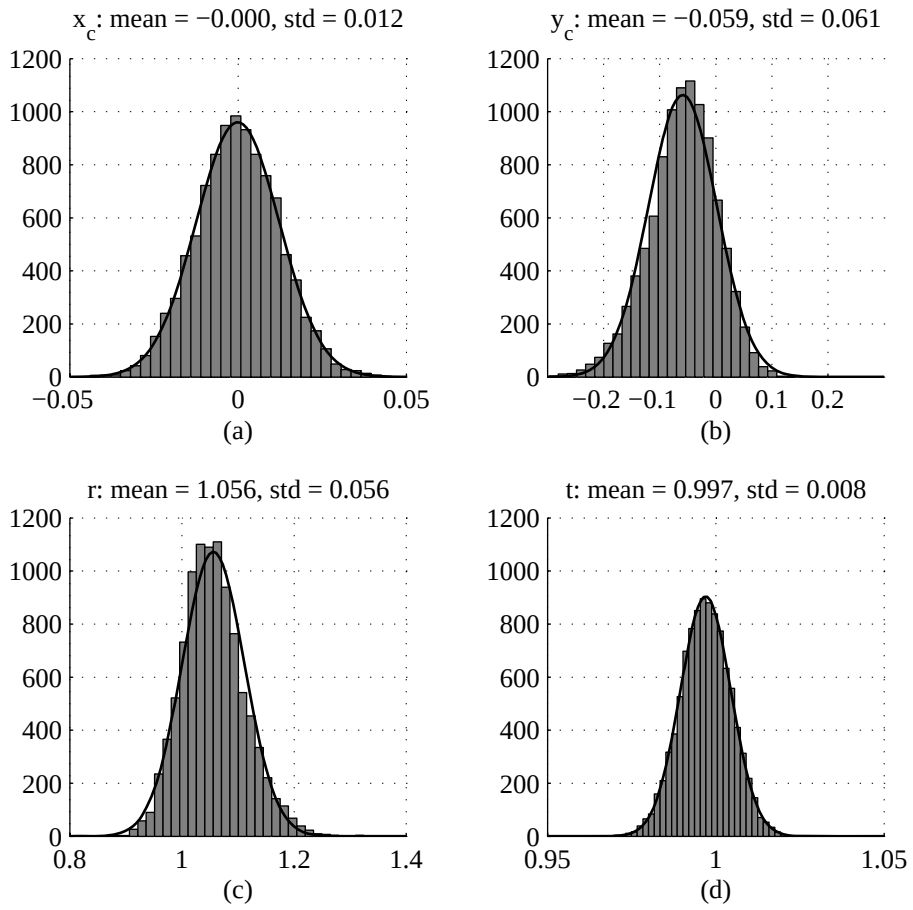

Figure 5.3: Distribution of the circle parameters estimated by Monte-Carlo simulation (histogram) in comparison to a normal distribution with the same mean and standard deviation (solid line).

	k_m	χ^2	$\chi^2_{0.99,k_m-3}$	H_0: normal distribution
x_c	29	38.5	45.6	accept
y_c	26	250.7	41.6	reject
r	26	521.8	41.6	reject
t	29	19.5	45.6	accept

Table 5.2: Result of χ^2-distribution test of the circle parameters

Recapitulating, the Monte-Carlo simulation results show, that the position of a circle, which is found by fitting circular arc data, can be described by the tangent point, i.e the point of contact of the tangent parallel to the main orientational axis of the data. The coordinates of this point exhibit a lower uncertainty than the center point of the circle, and they are normally distributed, whereas the center point coordinates do not feature a normal distribution.

5.1.2 Measurement Examples

The theoretically predicted lower uncertainty of the tangent point with respect to the center point of a circle fitted to circular arc data is verified with real measurement data obtained from repeatability measurements on a cylinder. Another example demonstrates a measurement task, where incorporating the tangent point into the evaluation decreases the uncertainty of the measurement result.

Repeatability measurement on a cylinder

The first measurement test is performed by repeatedly measuring a stationary cylindrical target with a light sectioning measurement head (see Section 7.1). The center point coordinates, radius and tangent position of 300 acquired and rectified light section profiles are evaluated. The individual measurement results are shown in Figure 5.4. The mean, standard deviation, maximum and minimum values have been calculated; the results are shown in Table 5.3: it can clearly be seen that the the standard deviation and the difference between maximum and minimum value is smallest for the tangent position.

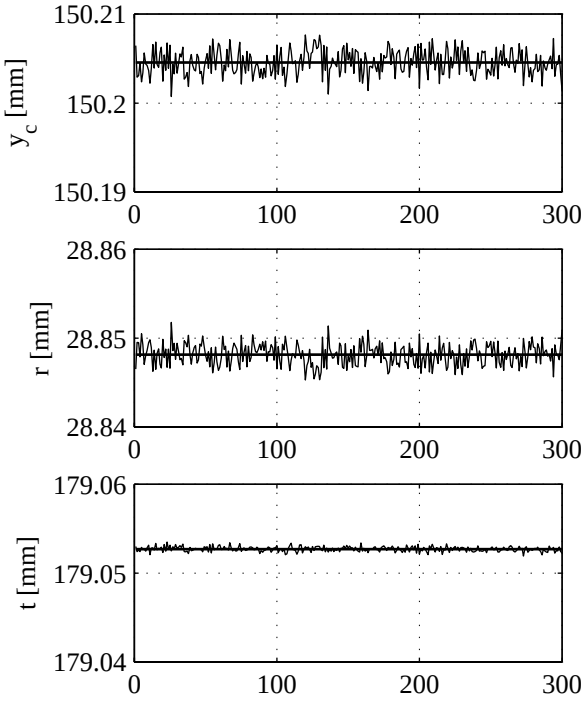

Figure 5.4: The results of 300 sequential measurements of the coordinate of the centre point y_c, radius r, and the calculated tangent position t.

The correlation matrix of the measured parameters (see Table 5.4) shows the very strong correlation between the radius r and y_c. The results of the real measurement are consistent with the results of the Monte-Carlo simulation.

	mean	std. dev.	max	min	max - min
r	28.8481	0.0012226	28.8518	28.8453	0.0064592
x_c	−6.9595	0.0004586	−6.9580	−6.9609	0.0028565
y_c	150.2046	0.0013986	150.2077	150.2007	0.0069241
t	179.0527	0.0002818	179.0535	179.0519	0.0015667

Table 5.3: Repeatability measurement for the radius, x- and y-coordinates of the centre points and the position of the tangent point. Mean value, standard deviation, maximum value minimum value, and difference, for the measured radius.

	x_c	y_c	r
x_c	1	−0.4894	0.4944
y_c	−0.4894	1	−0.9858
r	0.4944	−0.9858	1

Table 5.4: The correlation matrix for the measured parameters.

Measuring the distance between moving cylinders

Another example of a real measurement with a light sectioning measurement head is performed[1] to confirm the statistical advantage of the tangent position compared to the center point. In this example the distance between two cylinders is repeatedly measured. The position of the cylinders to each other is fixed, but they are moved with respect to the measurement head. Furthermore the real radius of each cylinder is known. Figure 5.5 shows the acquired raw data of all 29 measurements. A pair of corresponding data of one measurement is emphasized. The cylinders are imaged as circular arcs.

Two methods of distance evaluation are compared:

1. **Fitted center points:** For each individual measurement, a circle is fitted to each arc and the distance between the center points of the fitted circles is evaluated.

2. **Modified center points:** Similar to the first method, a circle is fitted to each arc. Then the tangent position is determined using the center point and radius of the fitted circle. A modified center position of each circle is calculated by subtracting the (known) real radius from the tangent position. The distance between the two modified center points is evaluated.

The results of both evaluation methods are shown in Figure 5.6. Evaluating the tangent position and modifying the center point clearly reduces the standard deviation of the evaluated distance values from $1.43\mu m$ to $0.60\mu m$.

[1]The actual measurements were taken by TBK Automatisierung und Messtechnik GmbH, Graz, Austria. Evaluation and interpretation of the results were solely performed by the author.

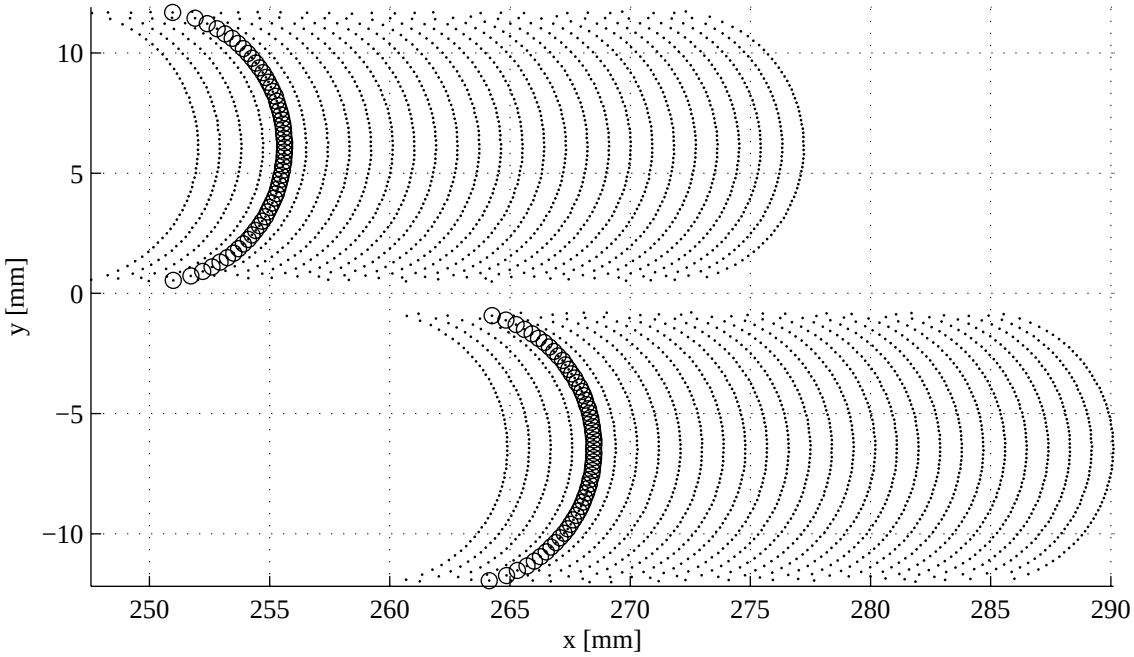

Figure 5.5: Raw data of 29 measurements of two moving cylinders with a light sectioning measurement head. A pair of corresponding data of one measurement is emphasized.

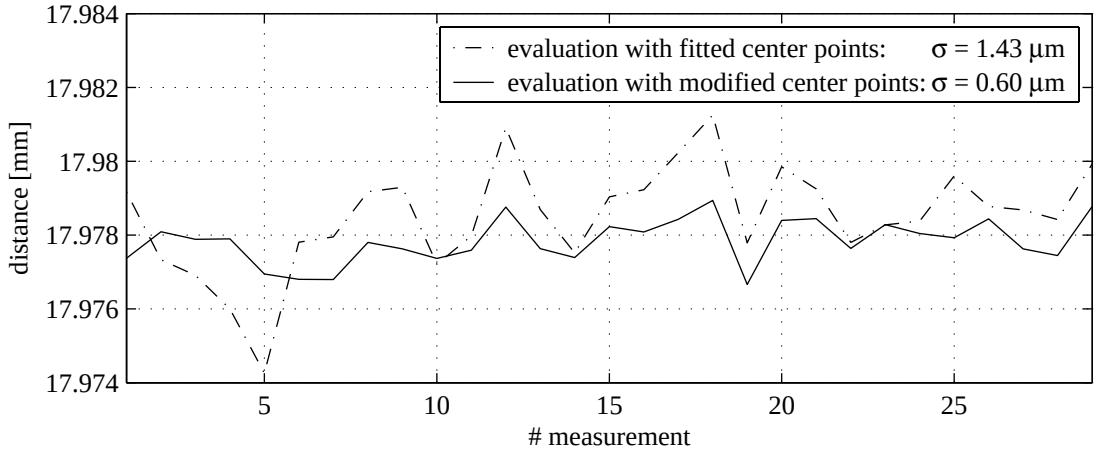

Figure 5.6: Results of the repeated distance measurements of cylinders. Evaluating the tangent position and modifying the center point coordinates reduces the uncertainty of the distance by 58%.

5.2 Sensitivity of Circle Fitting Algorithms

When fitting circles to noisy data points, the error in the points is propagated to the resulting circle parameters (center point $\boldsymbol{p}_c = [x_c, y_c]^T$, radius r, and tangent position t). The uncertainty in these results is thereby affected by the following factors of influence:

1. uncertainty level in the data points,
2. number of data points,
3. subtending arc angle of data points,
4. error measure (algebraic or geometric), that is minimized by the used fitting algorithm.

Monte-Carlo simulation on synthetically created input data is used to determine the degree of influence of the above listed factors on the standard deviation and the bias of the circle parameters. The usage of Monte-Carlo simulation in favor of the analytic approach to estimate the error propagation is motivated by the fact that the identification of the bias error is enabled. Comparing the mean of the circle parameters estimated through fitting with the model parameters used to create the input data yields this bias error. As an unbiased error model for the input data is chosen (i.e. the data is perturbed with mean free Gaussian noise), any resulting bias of the circle parameters is solely introduced by the used fitting algorithm and thus provides a crucial selection criterion for the algorithm.

The following circle fitting algorithms are considered for evaluation:

1. Direct algebraic methods
   (a) Simple algebraic fit (SAF, cf. Subsection 2.3.1).
   (b) Total least squares algebraic fit (TLS, cf. Subsection 2.3.2)
   (c) Partitioned total least squares algebraic fit (PTLS, cf. Subsection 2.3.3)
2. Nonlinear iterative geometric method
   (a) Gauss-Newton optimization (NL, cf. Subsection 2.3.4)

To provide a close estimation of the standard deviation and bias of the resulting parameters, each Monte-Carlo test is performed with 10000 data sets. The radius of the model circle is chosen to be 100 units and its center is at the origin. These model parameters are somewhat arbitrary, but due to the preceding normalization (see Section 2.1) of each data set, the fitting algorithms are anyway invariant to translational and scaling transformations. The noise, number, and subtending arc angle of the points is varied according to the specific test case. The points are distributed equally on the subtending arc angle and symmetrically with respect to the y-axis (as shown in Figure 5.2). The error of the input points is modelled as a perturbation of the x- and y-coordinate with mean-free independent Gaussian noise. The noise level is expressed as standard deviation and all error measures (standard deviation and bias) are expressed in percentage of the radius. The influence of the number of data points and the noise level is investigated for both full circles and circular arcs. The influence of the subtending arc angle applies to circular arcs only.

5.2.1 Full Circles

Influence of the number of data points

To estimate the influence of the number of data points on standard deviation and bias of the centroid coordinates and the radius of the fitted circle, the standard deviation of the coordinates of the data points $\sigma_{\boldsymbol{p}}$ is set to 5% of the radius and the number of data points N is varied between 10 and 500.

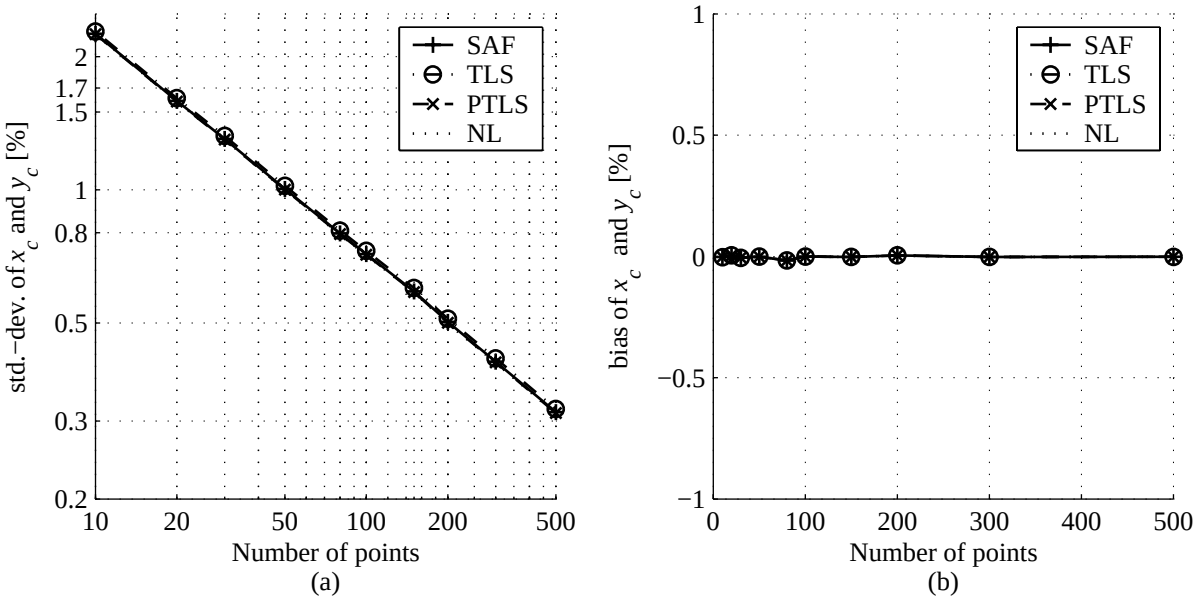

Figure 5.7: Standard deviation (a) and bias (b) of the center coordinates x_c and y_c of the circle against the number of data points (full circles)

The results regarding both center point coordinates x_c and y_c are equivalent due to the rotational symmetry of a full circle. As can be seen in Figure 5.7 the results of all four fitting algorithms are equivalent. The influence on the standard deviation is shown in Figure 5.7a. On a double-logarithmic scale the standard deviation follows a linear course, thus it can be modelled as:

$$\sigma_{x_c} = \sigma_{y_c} = d \cdot N^k, \tag{5.6}$$

where k denotes the slope of the line and d is the y-offset at $N = 10^0$. Linear regression exhibits that in this case $k = -\frac{1}{2}$ (decades per decade) and $d = 7.1$. Expressing d as a multiple of $\sigma_{\boldsymbol{p}}$ (i.e. the standard deviation of the data points), Equation 5.6 yields:

$$\sigma_{x_c} = \sigma_{y_c} = 7.1 \cdot N^{-\frac{1}{2}} = \frac{1.42\sigma_{\boldsymbol{p}}}{\sqrt{N}}, \tag{5.7}$$

thus the standard deviation of the fitted center point coordinates is by a factor of 1.42 larger than the standard deviation of the mean of the data points.

As depicted in Figure 5.7b, there is no systematic bias in the center point coordinates.

Regarding the standard deviation, the influence of the number of data points on the uncertainty of the radius (see Figure 5.8a) is similar to their influence on the center point: The results of all algorithms are equivalent and a similar functional relation can be modelled:

$$\sigma_r = \frac{1.0\sigma_{\boldsymbol{p}}}{\sqrt{N}}, \tag{5.8}$$

i.e. the inverse square-root-relation with N remains, but the pre-factor to the uncertainty of the mean is approximately 1.

However, in contrast to the center point coordinates the radius is biased, as shown in Figure 5.8b. The magnitude of the bias hardly depends on the number of data points, but is a characteristic of the fitting algorithm. It is largest when using the SAF-algorithm, but also remains on a significant level (with respect to the standard deviation) with the geometric NL-algorithm.

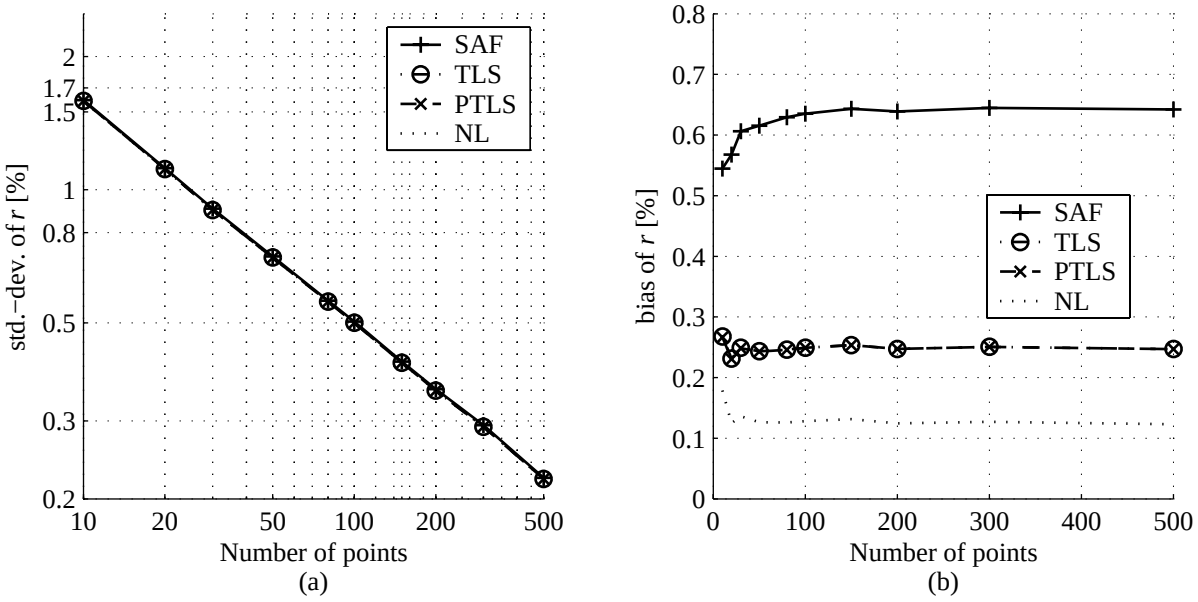

Figure 5.8: Standard deviation (a) and bias (b) of the radius r of the circle against the number of data points (full circles)

Bias introduced by the NL-algorithm

Although the bias of the radius introduced by the algebraic algorithms is explained by the non-geometric error measure that is minimized during fitting, the biased radius obtained by the geometric NL-algorithm seems to be a surprising result. The reason for this bias is the curvature of the circle. Assuming independent Gaussian noise in the x-and y-coordinates of the input data points, the probability that a point is outside the circle is slightly larger than the probability that the point is inside the circle as depicted in Figure 5.9. This effect is intensified with an increasing ratio of point uncertainty and circle

radius. Thus, the least-mean-squares geometric distance of all points to the fitted circle is obtained with a radius that is larger than the radius of the model circle.

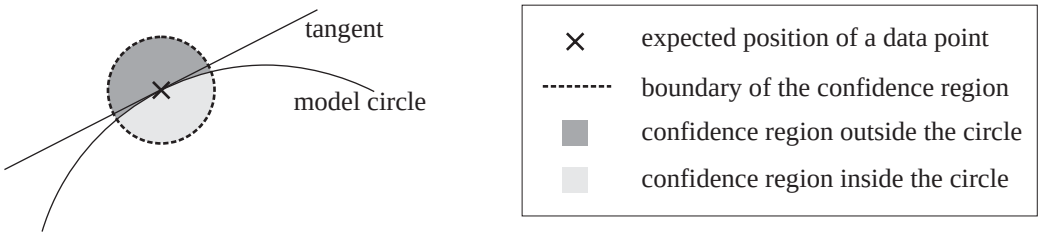

Figure 5.9: Assuming independent Gaussian noise in the x- and y-coordinate of a data point, results in an raised probability that the point lies outside the circle.

This influence test is repeated with a modified error model, that guarantees that the probability of an inside or outside position of the noisy point is equal: Instead of perturbing the x- and y-coordinate of the points, the radial position of the points is perturbed. Equivalent to the previous test, the noise level is chosen to be 5% of the radius.

Considering the influence of the number of data points on the center point coordinates, this test yields exactly the same results as the previous test (see Figure 5.7). The results of this test regarding the error of the radius are shown in Figure 5.10. The standard deviation of the fitted radius is the same as for the previous test (Figure 5.10a). In Figure 5.10b it can be seen that the geometric NL-algorithm delivers an unbiased result and in comparison with the previous test, the bias introduced by the algebraic algorithms is also reduced (approximately by the same amount as for the NL-algorithm).

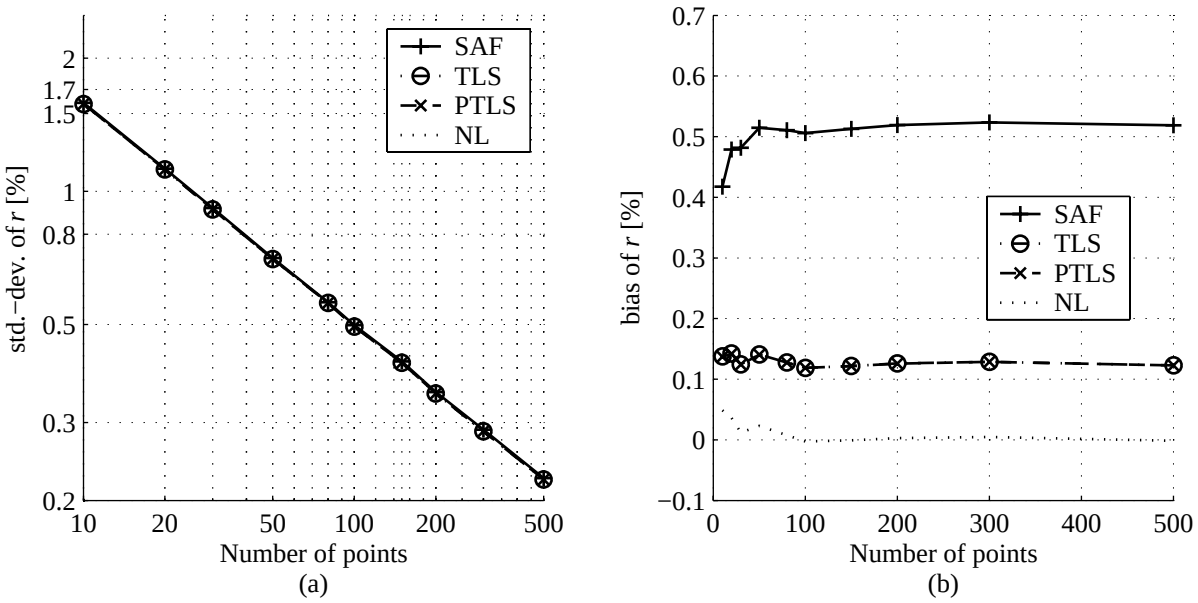

Figure 5.10: Standard deviation (a) and bias (b) of the radius r of the circle against the number of data points (full circles) when applying a radial error model.

The subsequent Monte-Carlo tests are performed with the original error model (perturbation of the x- and y-coordinates) for the following reasons:

1. For real measurement data, a perturbation in x- and y-coordinates of the individual points is more likely than a perturbation in radial direction only.
2. The standard deviations are not affected by a change of the error model.
3. If a radial error model is justifiable, the diagrams presented here, can yet be used to estimate the errors in the results: The bias introduced by the NL-algorithm is approximately zero. The bias of the algebraic algorithms can be approximated by reducing the values presented here by the correspondingly given bias of the NL-algorithm

Influence of noise level

Several Monte-Carlo tests are performed with a varying noise level of the data points between 0.1 and 10% of the radius of the model circle. A number of $N = 100$ data points is used for all tests.

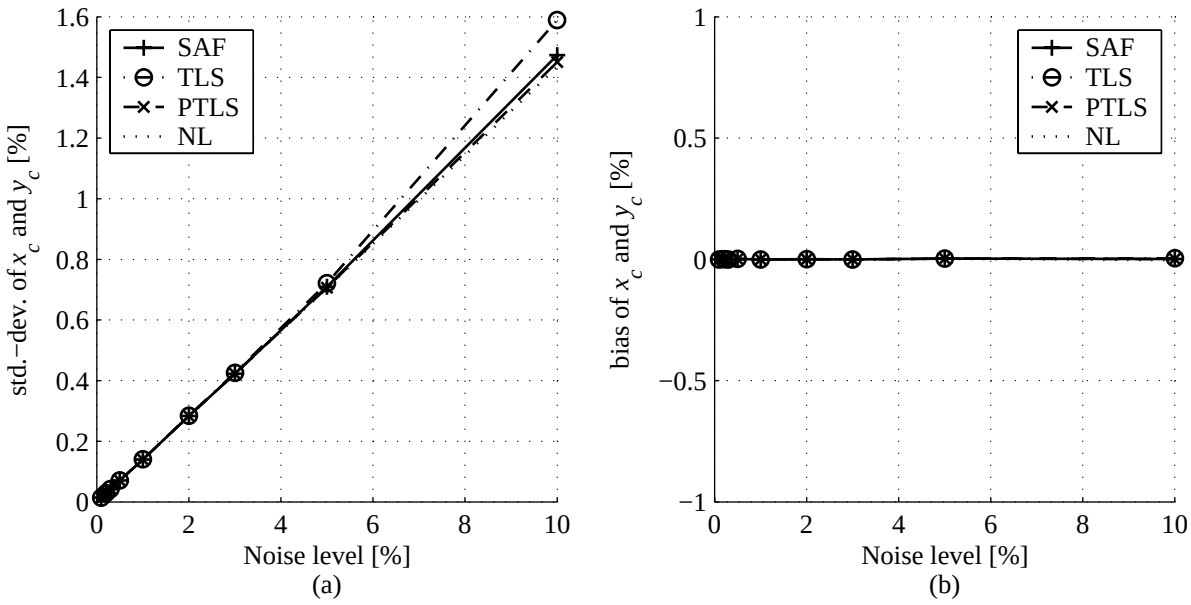

Figure 5.11: Standard deviation (a) and bias (b) of the center coordinates x_c and y_c of the circle against the noise of data points (full circles)

The influence of the noise level of the data points on the standard deviation of the center point coordinates of the fitted circle is visualized in Figure 5.11a and exhibits a linear behavior. It is consistent with Equation 5.7. The results of the different fitting algorithms are nearly equivalent, only the TLS-fit exhibits a small upwards deviation at high noise level of the input points. According to Figure 5.11b the center point coordinates are unbiased.

There is also a linear relationship between the noise level of the data points and the standard deviation of the radius (see Figure 5.12a), which is consistent with Equation 5.8. Figure 5.12b shows that the radius is biased. The bias increases approximately quadratically with the noise level and is also dependent on the fitting algorithm. It is largest with the SAF-algorithm and smallest, but still significant, with the NL-algorithm.

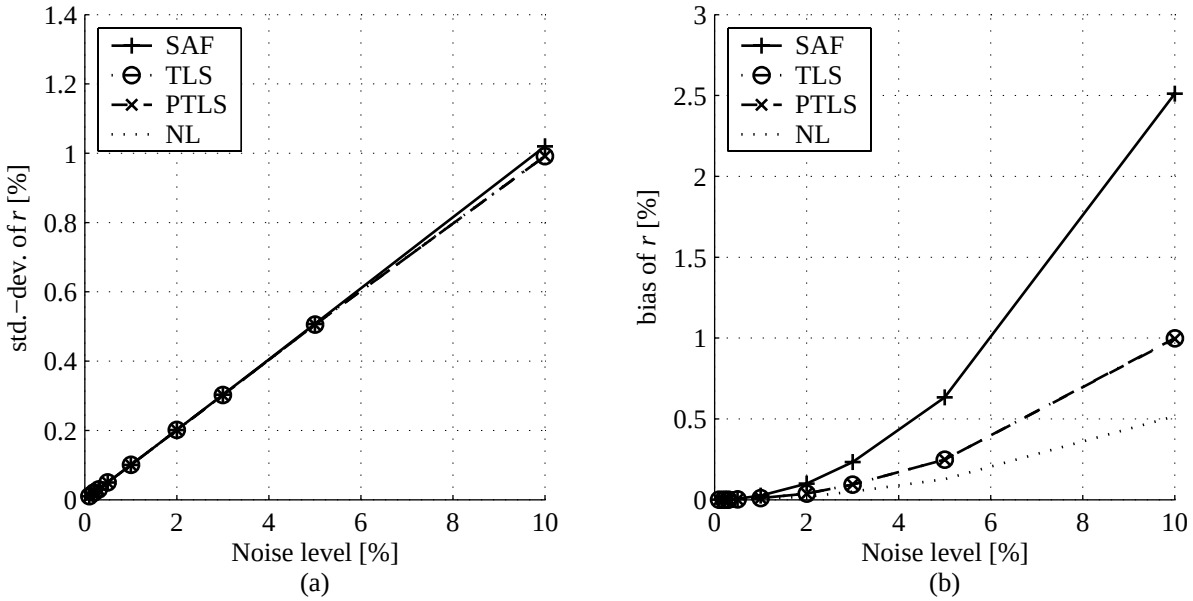

Figure 5.12: Standard deviation (a) and bias (b) of the radius r of the circle against the noise of data points (full circles)

5.2.2 Circular Arc Data

When testing the influence factors and the fitting algorithms on circular arc data, the influence on the center point coordinates x_c and y_c is presented separately, as there are large differences in their uncertainties. Similar to the tests presented in Section 5.1, the x-axis is parallel to the major orientational direction of the data, and the y-axis is perpendicular to it. In addition the uncertainty of the tangent position t is evaluated and presented.

Influence of the number of data points

The data points are distributed along a 120°-circular arc and perturbed with mean-free Gaussian noise with a standard deviation of 5% of the radius of the model circle. Similar to the tests with points on a full circle, the number of data points is varied between 10 and 500.

The test results about the influence of the number of data points are shown in Figures 5.13 to 5.16 and exhibit the following characteristics:

1. There is an inverse square-root relation between the standard deviation of each circle parameter and the number of data points, whereby the PTLS-algorithm yields a slightly lower standard deviation than the other algorithms.

2. The scaling factors between the standard deviations of the mean of the input points and the circle parameters are approximately:

$$\frac{\sigma_{\boldsymbol{p}}}{\sqrt{N}} : \sigma_{x_c} : \sigma_{y_c} : \sigma_r : \sigma_t \approx 1.0 : 1.8 : 6.4 : 5.8 : 1.4 \tag{5.9}$$

3. All circle parameters except the x-coordinate are biased, if an algebraic fitting algorithm is used.

4. The radius and the tangent position are even biased, if the geometric NL-algorithm is used, albeit at a low level (with respect to the standard deviation).

5. An increasing number of data points does not reduce the bias.

6. Considering the results of the algebraic fitting routines only, SAF yields the lowest bias and PTLS performs worst for all circle parameters.

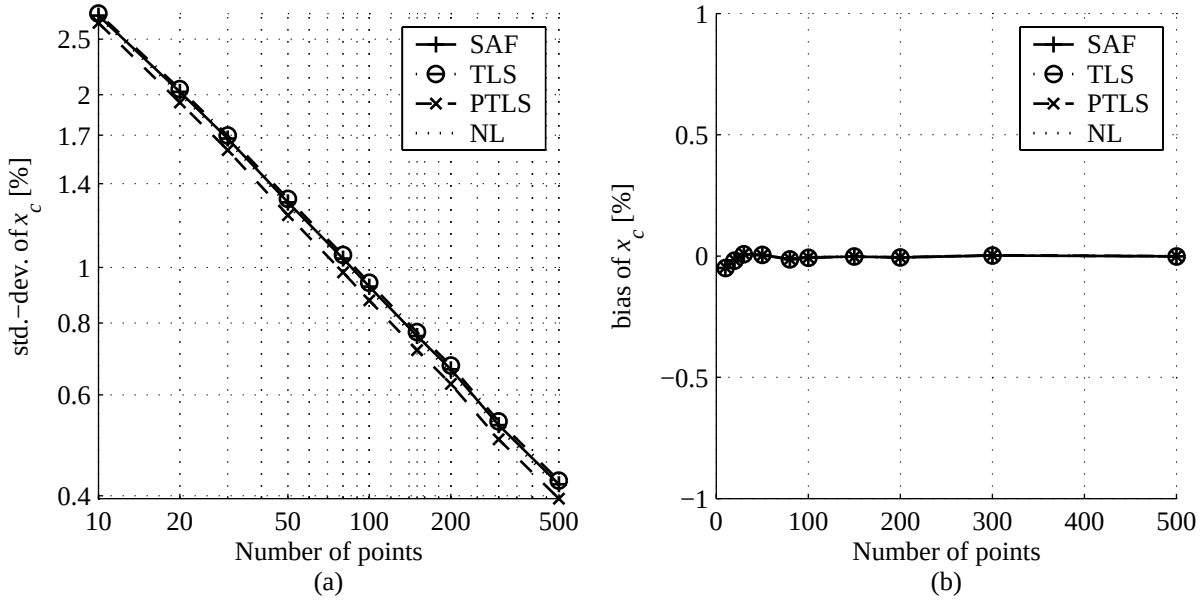

Figure 5.13: Standard deviation (a) and bias (b) of the center coordinate x_c of the circle against the number of data points (circular arcs).

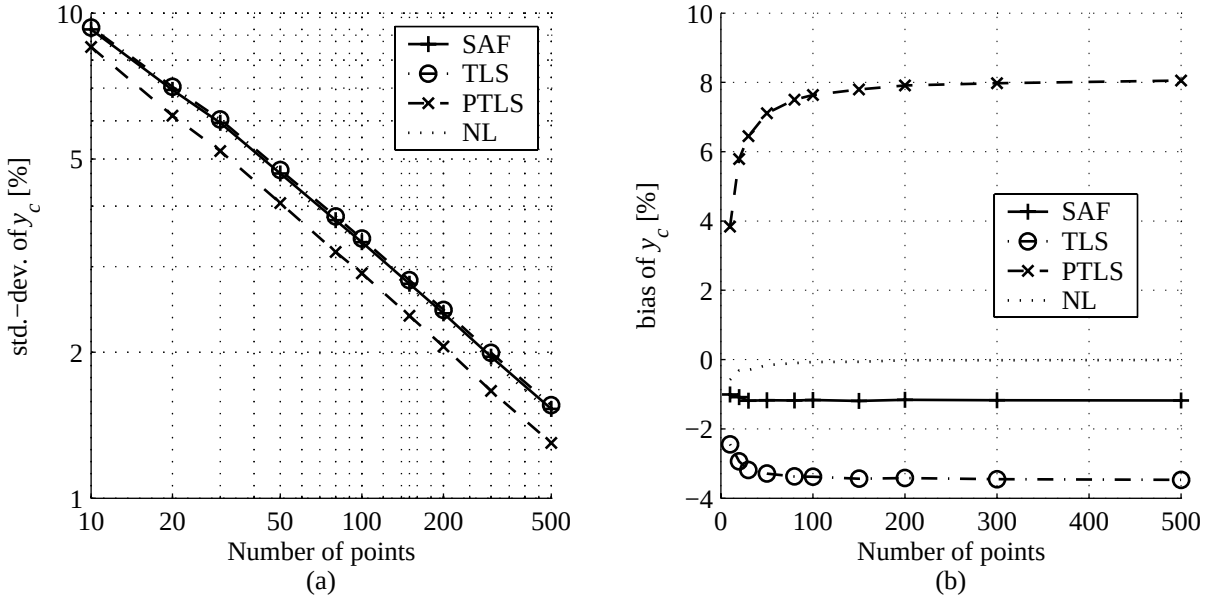

Figure 5.14: Standard deviation (a) and bias (b) of the center coordinate y_c of the circle against the number of data points (circular arcs).

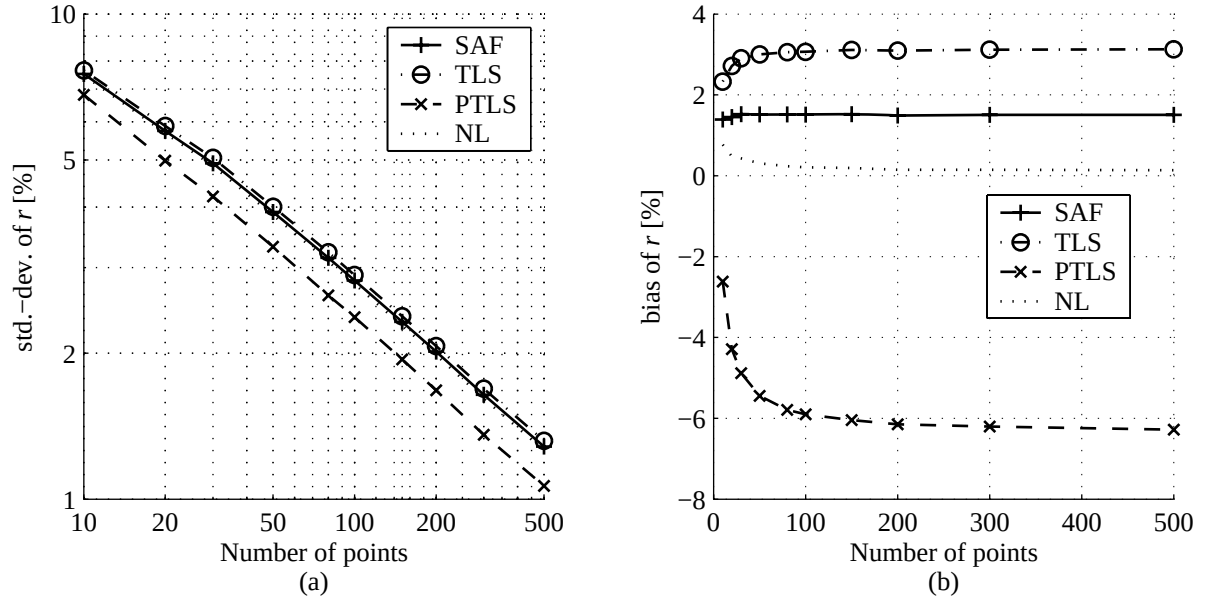

Figure 5.15: Standard deviation (a) and bias (b) of the radius r of the circle against the number of data points (circular arcs).

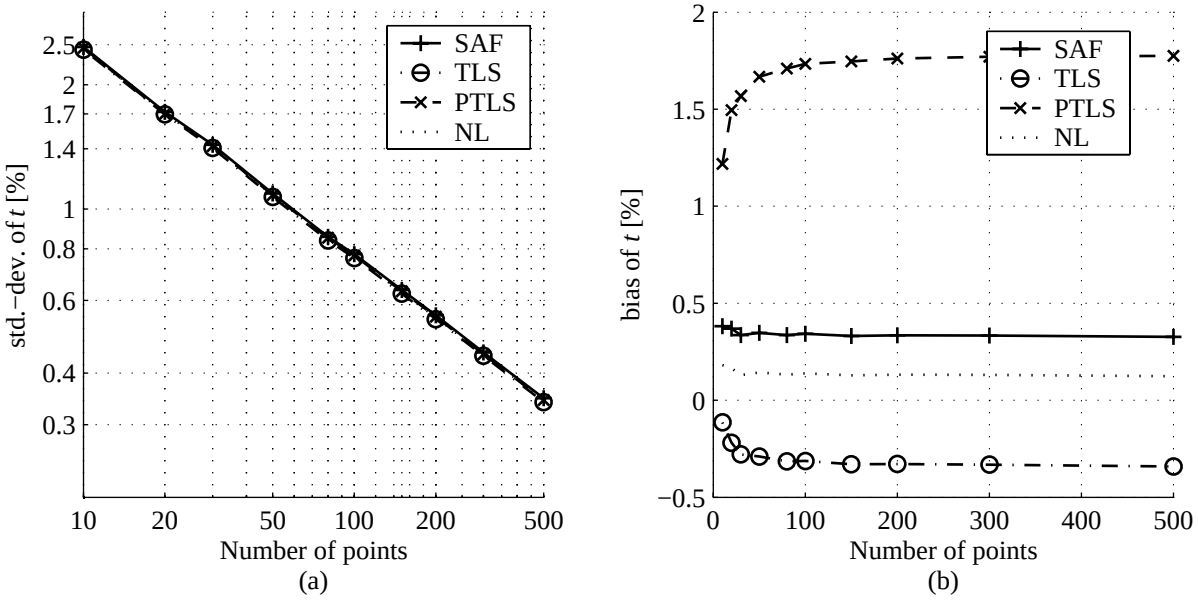

Figure 5.16: Standard deviation (a) and bias (b) of the tangent position t of the circle against the number of data points (circular arcs).

Influence of noise level

To test the influence of the noise level, a number of $N = 100$ input points are distributed on a 120°-circular arc. The standard deviation of the input point coordinates is varied between 0.1 and 10% of the radius of the model circle.

The test results about the influence of the noise level of the input points are shown in Figures 5.17 to 5.20 and exhibit the following characteristics:

1. There is an approximately linear relation between the standard deviation of each circle parameter and the number of input points, whereby at a high noise level ($>$ 3% of the radius) the PTLS-algorithm clearly deviates downwards, and the TLS-algorithm deviates upwards from the linear course.

2. The scaling factors between the standard deviations of the mean of the input points and the circle parameters are consistent with the scaling factors obtained from the number-of-points-influence test (Equation 5.9).

3. Also consistent with the previous test, all circle parameters except the x-coordinate are biased, if an algebraic fitting algorithm is used, and

4. the radius and the tangent position are even biased, if the geometric NL-algorithm is used.

5. The bias increases approximately quadratically with the noise level in the input data, thus gaining a significant portion of the absolute error at high noise levels.

6. Consistent with the number-of-points-influence test, SAF yields the lowest bias of all algebraic fitting algorithms and PTLS performs worst.

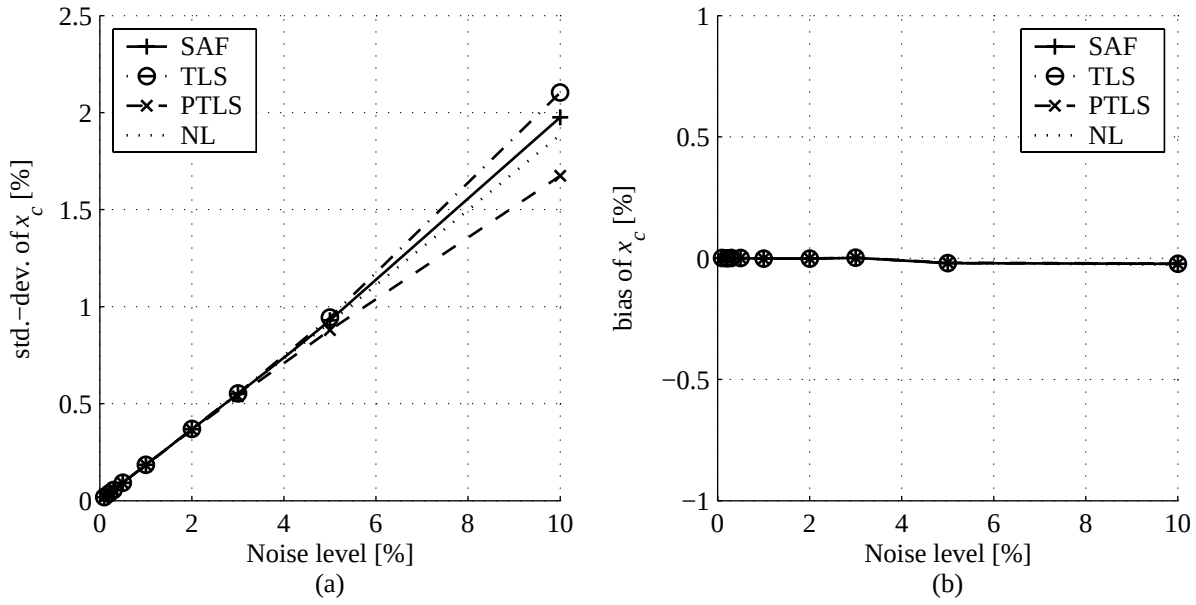

Figure 5.17: Standard deviation (a) and bias (b) of the center coordinate x_c of the circle against the noise level of data points (circular arcs).

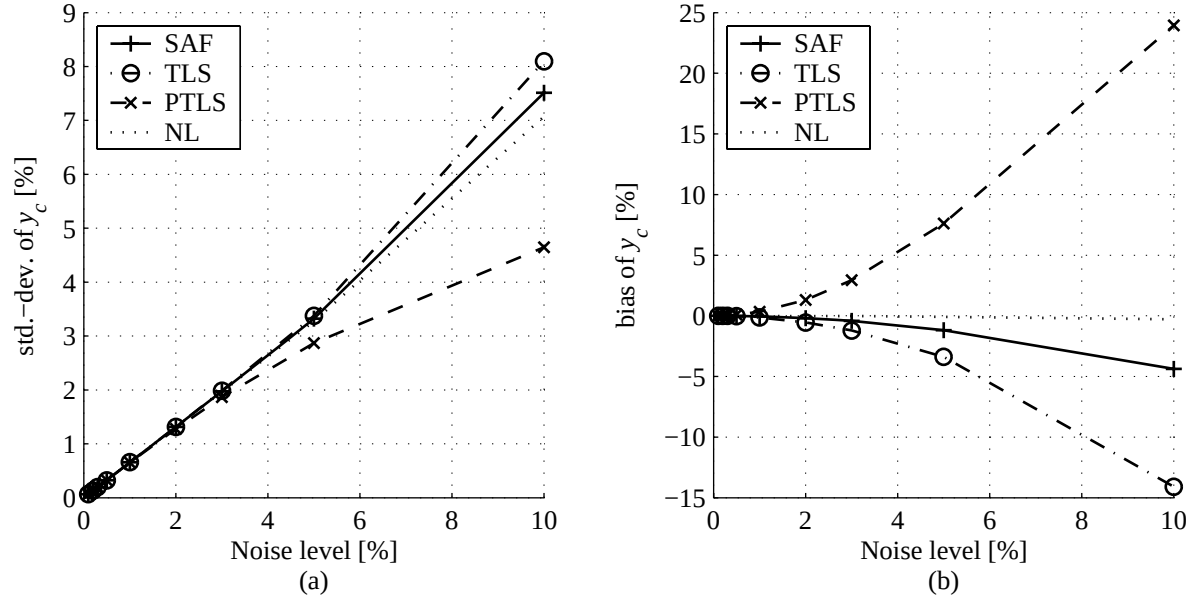

Figure 5.18: Standard deviation (a) and bias (b) of the center coordinate y_c of the circle against the noise level of data points (circular arcs).

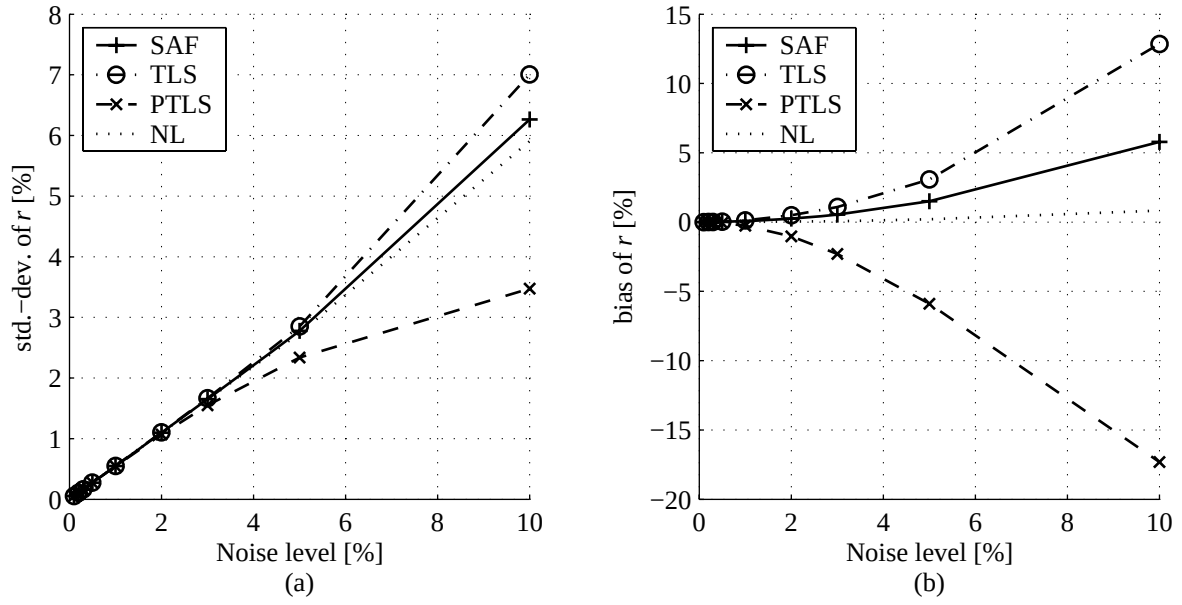

Figure 5.19: Standard deviation (a) and bias (b) of the radius r of the circle against the noise level of data points (circular arcs).

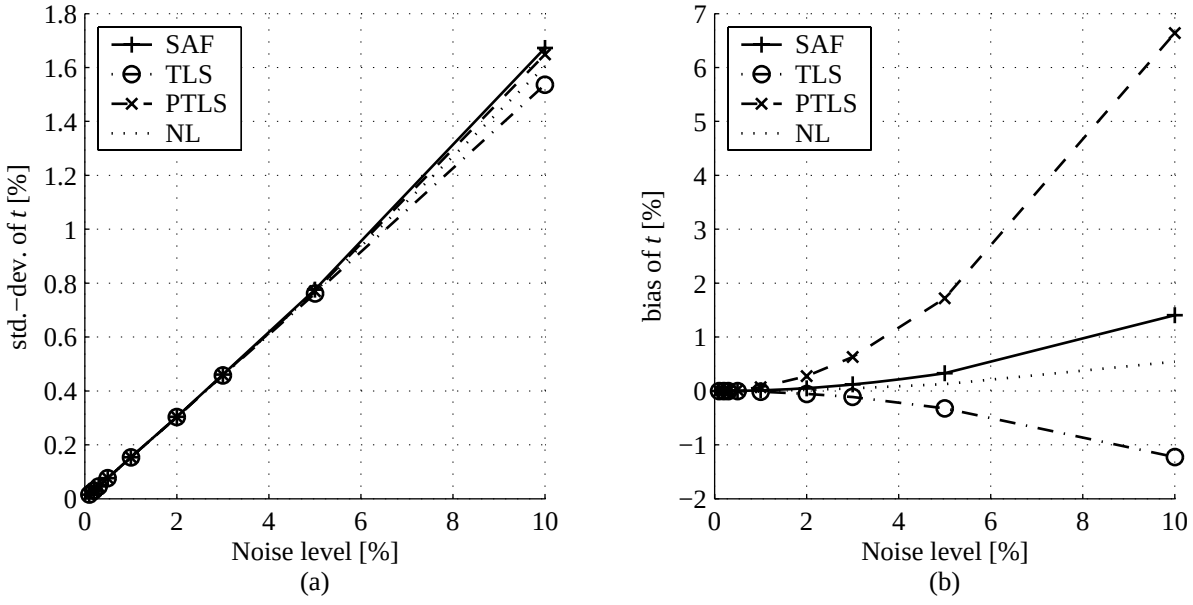

Figure 5.20: Standard deviation (a) and bias (b) of the tangent position t of the circle against the noise level of data points (circular arcs).

Influence of the arc angle

This test is performed by arranging $N = 100$ input data points with a coordinate uncertainty of 1% on a circular arc. The arc angle is varied from 20° to 270°.

The results about the influence of the arc angle α on the uncertainty of the circle param-

eters are visualized in Figures 5.21 to 5.24 and can be summarized as follows:

1. Considering the standard deviations of the circle parameters, the algorithms SAF, TLS and NL yield equivalent results, which exhibit the following characteristics within an angle range of 45° and 180°:

$$\sigma_{x_c} \propto \frac{1}{\alpha}, \quad \sigma_{y_c} \propto \sigma_r \propto \frac{1}{\alpha^2}, \quad \text{and} \quad \sigma_t \approx \text{const.} \tag{5.10}$$

2. PTLS yields significantly smaller standard deviations for angles below 60°, but this advantage is by far cancelled by the large bias of the circle parameters obtained with this algorithm.

3. Each algorithms delivers an unbiased estimate of the x-coordinate of the center point.

4. The bias of y_c and r increases drastically with decreasing arc angle (especially below 45°), whereby the NL-algorithm delivers the results with the lowest bias and PTLS performs worst. SAF yields the best performance of the algebraic fitting algorithms.

5. With SAF, TLS and NL, the bias of the tangent position does not depend on the arc angle and is small referred to the standard deviation. The bias obtained with PTLS increases sharply with decreasing arc angle.

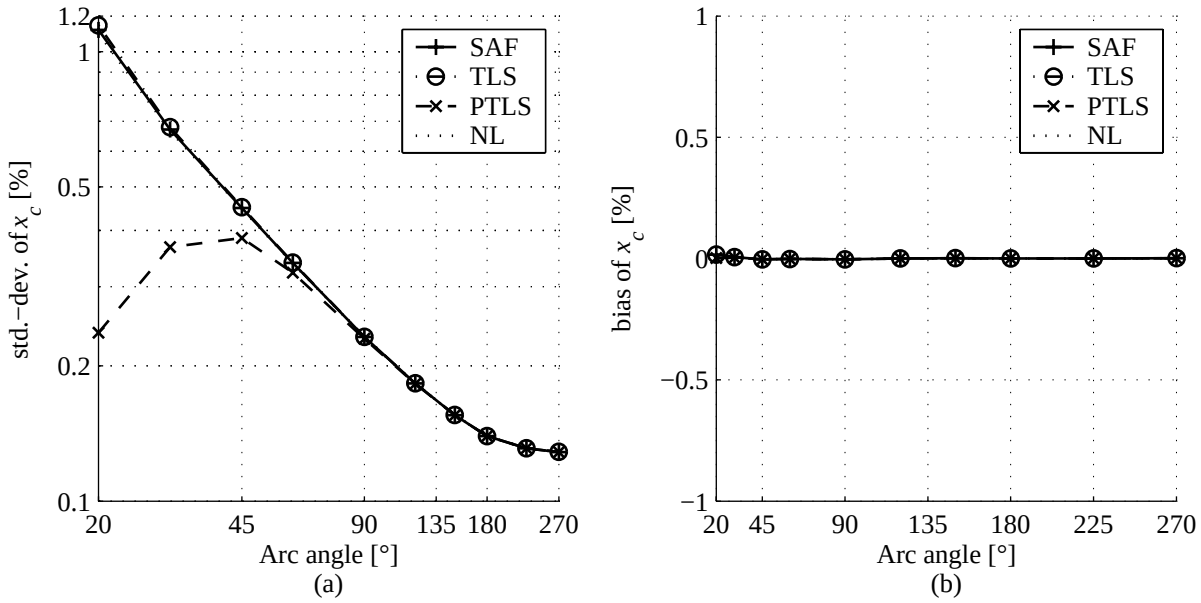

Figure 5.21: Standard deviation (a) and bias (b) of the center coordinate x_c of the circle against the subtending arc angle.

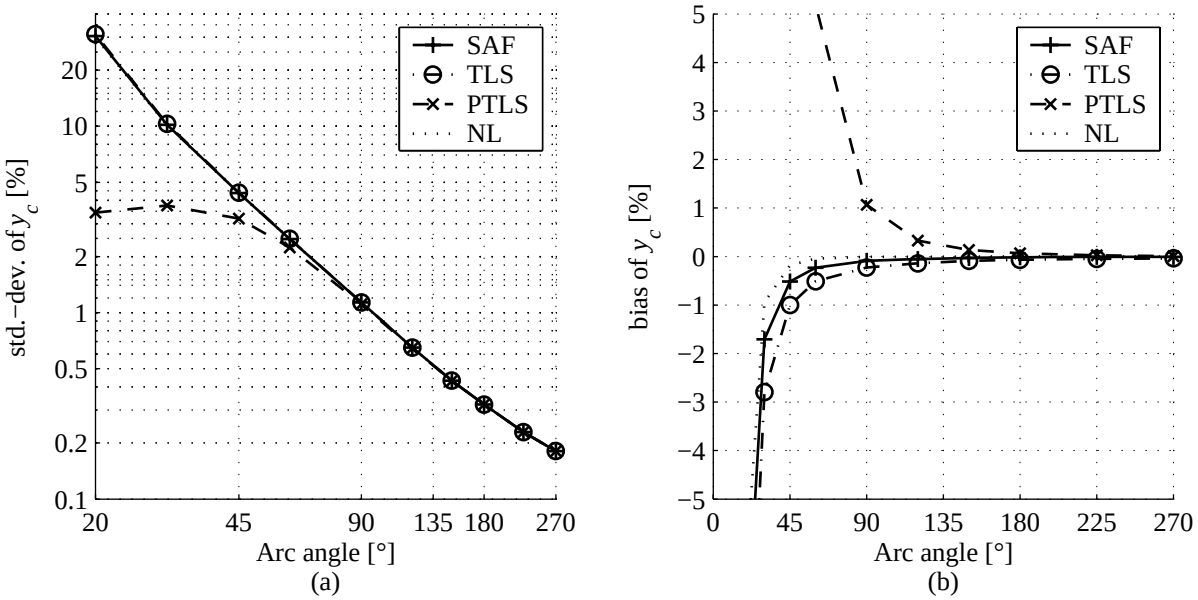

Figure 5.22: Standard deviation (a) and bias (b) of the center coordinate y_c of the circle against the subtending arc angle.

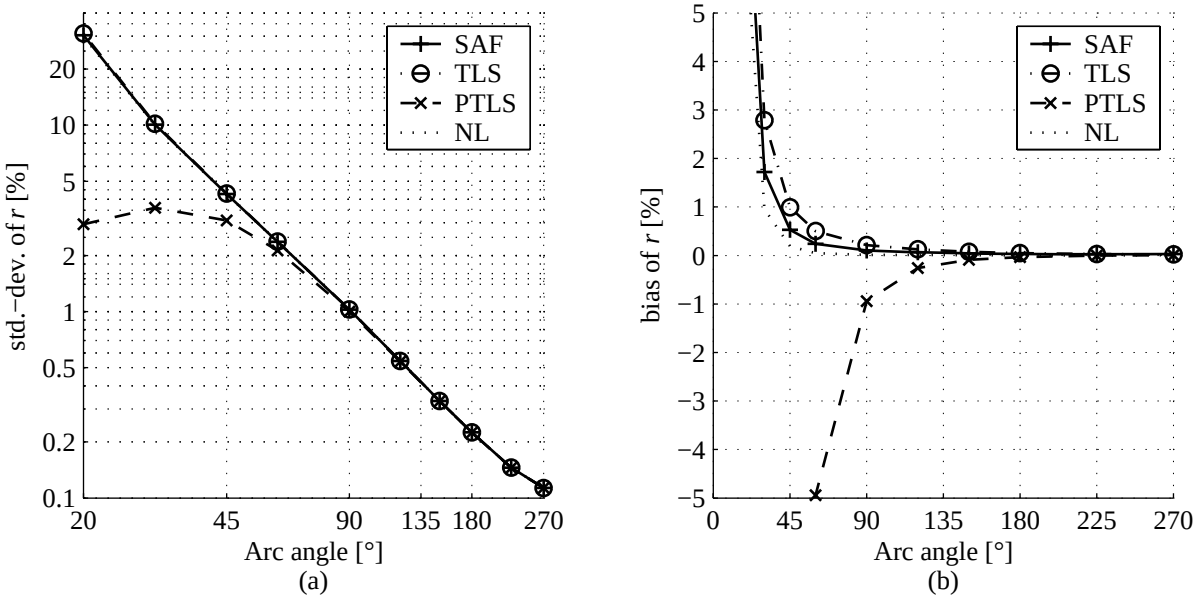

Figure 5.23: Standard deviation (a) and bias (b) of the radius r of the circle against the subtending arc angle.

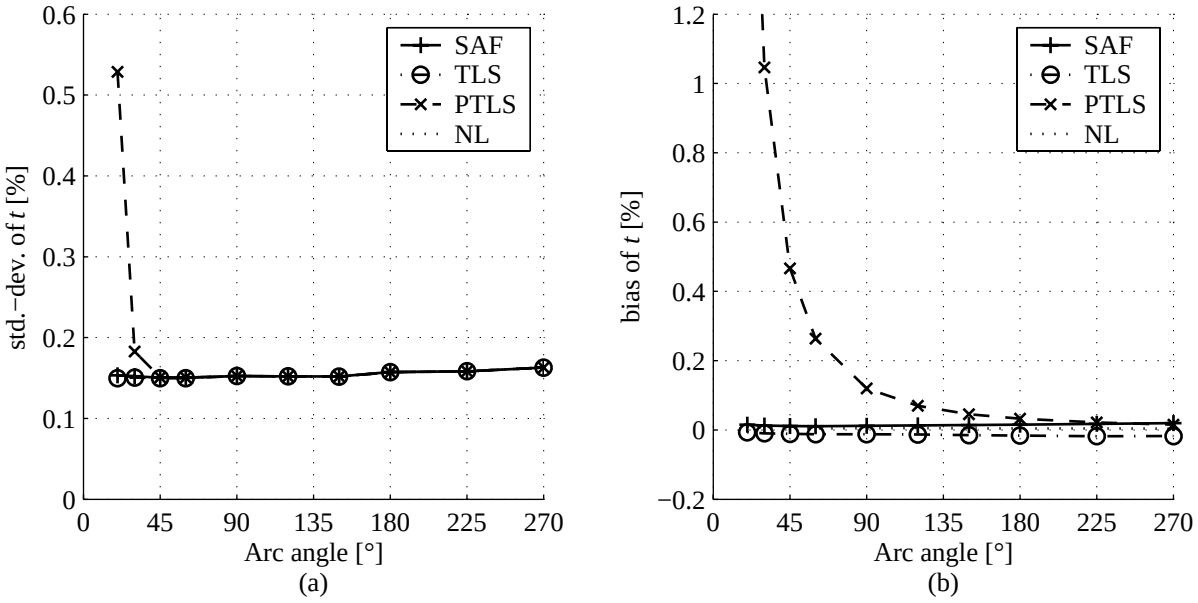

Figure 5.24: Standard deviation (a) and bias (b) of tangent point t of the circle against the subtending arc angle.

5.2.3 Conclusions on Sensitivity of the Fitting Algorithms

Fitting methods

1. The selection of the fitting method has a negligible influence on the standard deviation of the results. In some cases PTLS delivers results with significantly lower standard deviation than the other algorithms. However, this advantage is outweighed by the accompanying large bias of these results.

2. Considering the bias of the results, the NL-algorithm delivers the best performance for all test cases. Nevertheless, depending on the error distribution in the input points, the obtained radius and tangent position might be biased.

3. When fitting circles to data that is arranged over the full circumference, the PTLS-algorithm is the best algebraic alternative to NL-fitting and should be used if the computational effort of a nonlinear algorithm is too high.

4. The best algebraic alternative to the NL-algorithm for fitting circles to circular arc data is the SAF-algorithm.

Influence factors

1. The number of data points N has an inverse square-root relation to the standard deviation of the results:

$$\sigma_{x_c} \propto \sigma_{y_c} \propto \sigma_r \propto \sigma_t \propto \frac{1}{\sqrt{N}} \tag{5.11}$$

   There is no influence of the number of data points on the bias of the fitting results for $N > 80$. For $N < 80$ the influence on the bias is differing (both increasing and decreasing is possible) and depends also on the fitting algorithm.

2. The noise level $\sigma_{\boldsymbol{p}}$ of the input data has a linear influence on the standard deviation of the results and an approximately quadratical influence on the bias. Thus the bias is gaining a significant portion of the absolute error at high noise levels of the input data.

3. Within the interval $45° < \alpha < 180°$, the influence of the subtending arc angle α on the standard deviation of the x-coordinate of the center point is approximately

$$\sigma_{x_c} \propto \frac{1}{\alpha}, \tag{5.12}$$

   and on the y-coordinate and the radius

$$\sigma_{y_c} \propto \sigma_r \propto \frac{1}{\alpha^2}. \tag{5.13}$$

   The standard deviation of the tangent position is approximately constant with respect to the arc angle α.

   There is no common relation describing the influence of the arc angle α on the bias of the y-coordinate and the radius. With decreasing α, the bias increases rapidly, but algorithm-related at differing order of magnitude. For $\alpha < 30°$, even the results obtained by NL-fitting are significantly biased. The bias of the tangent position is not influenced by α, except if PTLS is used for fitting.

Chapter 6

Curve from Curvature

In this chapter methods to reconstruct the shape of a closed planar curve from curvature values, that are discretely sampled over the complete circumference of the curve, are introduced. Reconstructing a curve from curvature data is a subtask of measuring cross-sectional profiles of rotating objects (see Chapter 7). It is separately discussed here for the following reasons:

1. The reconstruction step is detached from the actual instrumentation for measuring the curvature. It may also apply to systems with a curvature measurement sensor other than a light sectioning head as presented in Chapter 7.
2. In addition to the derivation of the reconstruction algorithms, this chapter includes a section on testing the algorithm with synthetic input data. Thereby the goodness of the reconstruction is proved by comparing the reconstruction results with the model curve from which the input data was created.

The two reconstruction algorithms presented within this chapter are adapted to the specific sample spacing of the measured curvature values. Considering the measurement of local curvatures on a rotating profile, the sample spacing (i.e. the distribution of the measurement positions over the circumference) is determined by the way the rotation of the object is driven. This is explained in detail in the first section. Thereafter the derivation of the reconstruction algorithms is presented and results on testing the methods with synthetic input data are shown.

In contrast to the modality in this work, the term "curve reconstruction" is commonly used in literature (e.g. in [6, 16, 26]) to describe the problem of finding a shape (e.g. a polygon) that best approximates a finite set of points. Within this chapter the terms "curve reconstruction" and "curve from curvature" are used in the sense of reconstructing the shape of a curve from curvature data. The reconstruction results are determined up to a Euclidean transformation, i.e. the absolute position and orientation of the reconstructed curve is ambiguous. The curves that are considered to be reconstructed are C^1-continuous convex Jordan curves[50].

6.1 Object Rotation and Sample Spacing

It is assumed, that a stationary measurement device is used to acquire the local curvature values of a cross-sectional profile. The profile is rotated to sample the curvatures over the complete circumference of the curve. Two ways to drive the rotation, which cover the mechanically most relevant situations, are considered, as schematically depicted in Figure 6.1:

1. **Roller driven rotation:** The object is driven by non-slip contact to a circular roller with constant rotational speed (Figure 6.1a). The sampling position is assumed to be at the point of contact, slightly shifted in axial direction. It is assumed, that within this shift, the object is a right generalized cylinder. Thus, in case of constant acquisition rates, the measured curvatures are sampled at positions of equally spaced arc lengths segments. The resulting spacing is equivalent if the profile stands still and the measurement device is moved along the curve with constant speed.

2. **Shaft driven rotation:** If the object is mounted on a rotating shaft, it is possible to accurately control the rotational angle for each acquisition (Figure 6.1b). In case of shaft driven rotation, the actual sampling position (i.e. the fixed position of the measurement device) is arbitrary. Provided constant acquisition rates and constant rotational speed of the shaft, the measured curvatures are sampled at positions of equally spaced rotational angles. A simple way to implement such a setup is to lay the object on a rotating table, whereby a non-slip contact between the object and the table is required. Equivalently, when considering a motionless profile, the measurement device could be rotating around the object on a circular course to obtain the same sample spacing.

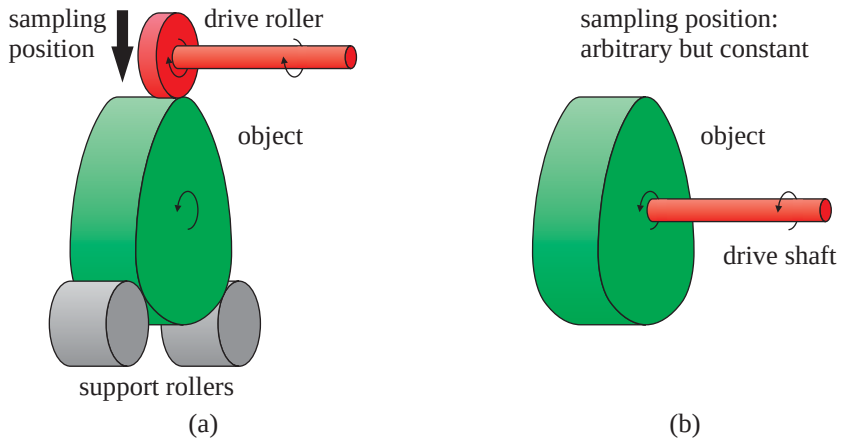

Figure 6.1: Two possible rotations of the object: roller driven rotation (a); and shaft driven rotation (b).

The differing arrangements of the sample position, according to the way of driving the rotation, are exemplarily demonstrated in Figure 6.2 for an elliptical curve. The more regular sample spacing obtained by the roller driven rotation is apparent.

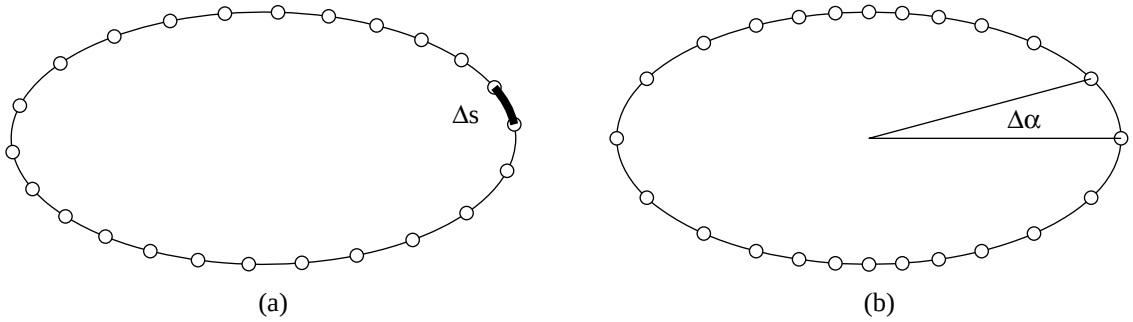

Figure 6.2: Ellipse with different arrangements of the sample positions: roller driven rotation yields equally spaced arc length segments Δs (a) and shaft driven rotation causes equally spaced rotational angles $\Delta\alpha$ (b).

6.2 Curve from Curvature Algorithms

The curvature κ is defined as

$$\kappa = \frac{1}{\rho} = \frac{\mathrm{d}\phi}{\mathrm{d}s}, \tag{6.1}$$

where ρ denotes the radius of curvature; ϕ and s are the tangential angle and the arc length, respectively.

If a curve C is parameterized in the parameter t, $C \triangleq \boldsymbol{r}(t) = [x(t),\, y(t)]^T$, Equation 6.1 can be formulated as[61]:

$$\kappa(t) = \frac{1}{\rho(t)} = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{\frac{3}{2}}}, \tag{6.2}$$

where " $'$ " denotes the derivative with respect to the parameter t.

6.2.1 Curvature Data Spaced at Constant Arc-Length Segments

Considering the measurement of curvature values over the complete circumference of a rotating profile and driving the rotation of the object with contacting rollers, a set of N curvature values κ_i, whereby $i = \{1 \ldots N\}$, spaced at constant arc length segments Δs, is obtained. In the continuous case, this is equivalent to knowing the curvature as a function of the arc length s,

$$\kappa = \kappa(s), \tag{6.3}$$

which is called the *intrinsic equation* or *natural equation* of a curve. The determination of the parameter equation of the curve, $C \triangleq \boldsymbol{r}(s) = [x(s),\, y(s)]^T$, is performed as shown in the following[58, 59]:

Rearranging 6.1 to

$$\mathrm{d}\phi = \kappa\, \mathrm{d}s, \tag{6.4}$$

and integrating with respect to s, delivers:

$$\phi(s) = \int_0^s \kappa(s)\,\mathrm{d}s + \phi_0, \tag{6.5}$$

Furthermore, it is known that

$$\frac{\mathrm{d}x}{\mathrm{d}s} = \cos(\phi), \text{ and } \frac{\mathrm{d}y}{\mathrm{d}s} = \sin(\phi). \tag{6.6}$$

Consequently,

$$\mathrm{d}x = \cos(\phi)\,\mathrm{d}s,\ \mathrm{d}y = \sin(\phi)\,\mathrm{d}s. \tag{6.7}$$

Integrating with respect to s yields the parameter equation of the curve,

$$x(s) = \int_0^s \cos(\phi)\,\mathrm{d}s + x_0 \tag{6.8}$$

$$y(s) = \int_0^s \sin(\phi)\,\mathrm{d}s + y_0. \tag{6.9}$$

The resulting curve is determined up to a Euclidean transformation. The translational part of this transformation is determined by the choice of the integration constants x_0 and y_0. The rotational part implicitly depends on the choice of ϕ_0. Since only the shape of the cross-sectional profile and not its absolute position and orientation are of interest, all three values can be set to zero.

Returning to the discrete case of N measurements of the curvature κ_i sampled at constant arc length Δs over one complete circumference of the curve, the discrete equivalent formulations of Equations 6.5, 6.8 and 6.9 are

$$\phi_k = \sum_{i=1}^{k} \kappa_i \Delta s \tag{6.10}$$

$$x_k = \sum_{i=1}^{k} \cos(\phi_i)\Delta s \tag{6.11}$$

$$y_k = \sum_{i=1}^{k} \sin(\phi_i)\Delta s \tag{6.12}$$

It is assumed that the curvature values are sampled over exactly one complete revolution of the object, thus $\phi_N = 2\pi$. Therewith and with Equation 6.10, Δs can be computed as follows:

$$\phi_N = 2\pi = \sum_{i=1}^{N} \kappa_i \Delta s = \Delta s \sum_{i=1}^{N} \kappa_i, \tag{6.13}$$

hence,

$$\Delta s = \frac{2\pi}{\sum_{i=1}^{N} \kappa_i}. \tag{6.14}$$

Finally, the curve is centered at the origin:

$$\tilde{x}(k) = x(k) - \bar{x}(k), \tag{6.15}$$
$$\tilde{y}(k) = y(k) - \bar{y}(k). \tag{6.16}$$

MATLAB® code

Using the variable names `K` and `R` for κ and ρ, respectively, the algorithm is formulated in MATLAB® code as follows:

```
function [x, y] = reconstructRS( R );
% reconstruct a curve from radii of curvature
% sampled at constant arc length segments
K = R.^(-1);
deltaS = 2*pi/(sum( K ));
phi = cumsum( K ) * deltaS;
xT = cumsum( cos( phi )) * deltaS;
yT = cumsum( sin( phi )) * deltaS;
x = xT - mean( xT );
y = yT - mean( yT );
```

Listing 6.1: reconstructRS.m

A note on the interpretation of the algorithm

A possible interpretation of the algorithm is (see also Figure 6.3):

1. A set of N vectors with length Δs and gradient angles ϕ_k, where $k = \{1 \dots N\}$ and $\phi_k = \{0 \dots 2\pi\}$, is created (see Figure 6.3a).
2. The cumulative sum of these vectors (i.e. the x- and y-components of the vectors) leads to the points on the curve (see Figure 6.3b).

6.2.2 Curvature Data Sampled at Points with Known Rotational Angle

Measuring local curvatures on a cross-sectional profile of a rotating object with shaft driven rotation, delivers a set of curvature values κ_i at positions of known rotational

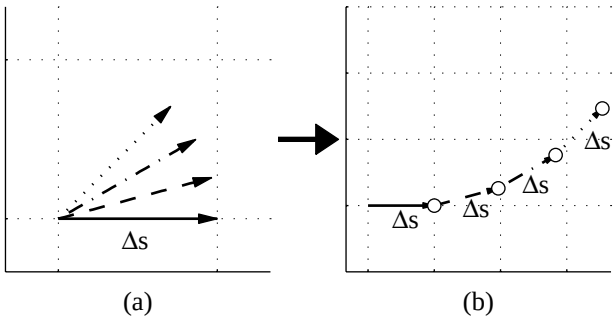

Figure 6.3: A possible interpretation of the reconstruction algorithm: Part of a set of vectors with gradient angles ϕ_k (a) and the cumulative sum of these vectors reconstructing points on the curve (small circles) (b).

angle α_i. Unlike the arc-length s (see Equation 6.1), there is no general mathematical relation between the rotational angle α and the curvature κ (or the radius of curvature ρ). Thus a a metrological solution to reconstruct the curve is proposed:

1. In addition to the local curvature, the local tangential angle $\phi_{m,i}$ to the curve at the sampling position is measured. This angle is related to the reference coordinate system of the measurement device.

2. According to Figure 6.4, the reference to the rotating coordinate system of the object is found by subtracting the rotational angle α_i:

$$\phi_i = \phi_{m,i} - \alpha_i \tag{6.17}$$

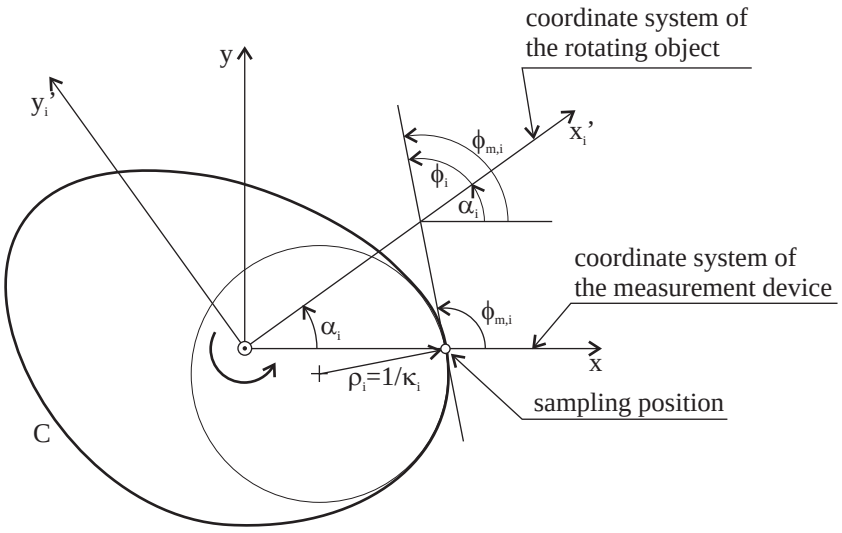

Figure 6.4: The tangential angle ϕ is transformed to the rotating coordinate system of the specimen.

Now the problem can be reformulated to reconstructing a curve using curvature values κ_i at points with known tangential angle ϕ_i. The reconstruction algorithm is developed for the continuous case and then a discrete equivalent formulation is given:

Rearranging 6.1 to

$$\rho(\phi) = \frac{\mathrm{d}s}{\mathrm{d}\phi} \tag{6.18}$$

and using 6.6, it follows:

$$\frac{\mathrm{d}x}{\mathrm{d}\phi} = \frac{\mathrm{d}x}{\mathrm{d}s}\frac{\mathrm{d}s}{\mathrm{d}\phi} = \cos(\phi)\rho(\phi), \text{ and } \frac{\mathrm{d}y}{\mathrm{d}\phi} = \frac{\mathrm{d}y}{\mathrm{d}s}\frac{\mathrm{d}s}{\mathrm{d}\phi} = \sin(\phi)\rho(\phi), \tag{6.19}$$

multiplying by $\mathrm{d}\phi$ yields

$$\mathrm{d}x = \cos(\phi)\rho(\phi)\,\mathrm{d}\phi \text{ and } \mathrm{d}y = \sin(\phi)\rho(\phi)\,\mathrm{d}\phi. \tag{6.20}$$

Integrating with respect to ϕ delivers

$$x(\phi) = \int_0^{\phi} \cos(\phi)\rho(\phi)\,\mathrm{d}\phi + x_0, \tag{6.21}$$

and for y,

$$y(\phi) = \int_0^{\phi} \sin(\phi)\rho(\phi)\,\mathrm{d}\phi + y_0, \tag{6.22}$$

As in Subsection 6.2.1 the resulting curve is determined up to a Euclidean transformation. Since the absolute position and orientation of the curve is not of interest, the integration constants x_0 and y_0 can be set to zero again.

The discrete equivalent formulation of Equations 6.21 and 6.22 is:

$$x_k = \sum_{i=1}^{k} \cos(\phi_i)\rho_i\Delta\phi_i \tag{6.23}$$

$$y_k = \sum_{i=1}^{k} \sin(\phi_i)\rho_i\Delta\phi_i \tag{6.24}$$

whereby in general,

$$\Delta\phi_i = \phi_{i+1} - \phi_i. \tag{6.25}$$

In the special case of N linearly distributed values of ϕ between 0 and 2π, $\Delta\phi_i$ is constant:

$$\Delta\phi_i = \Delta\phi = \frac{2\pi}{N}. \tag{6.26}$$

It can be seen from Equations 6.23 and 6.24, that the radii of curvature $\rho_i = \frac{1}{\kappa_i}$ are required rather than the curvatures.

MATLAB® code

Using the variable name `R` for the radius of curvature ρ, the algorithm is formulated in MATLAB® code as follows:

```
function [x, y] = reconstructRA( R, phi_m, alpha )
% reconstruct a curve from radii of curvature
% sampled at positions of known rotational angle
phi = phi_m - alpha;
deltaPhi = diff( phi );
xT = cumsum( cos( phi )) .* R .* deltaPhi;
yT = cumsum( sin( phi )) .* R .* deltaPhi;
x = xT - mean( xT );
y = yT - mean( yT );
```

Listing 6.2: reconstructRA.m

Limitations of the algorithm

The algorithm presented in this section exhibits a major disadvantage in comparison to the reconstruction with curvatures spaced at constant arc length segments: The reconstruction does not work, if the curve features straight parts with zero curvature. At such positions the radius of curvature ρ is infinite and $\Delta\phi$ becomes zero. Mathematically the product $\rho\,\Delta\phi$ is then undefined. Due to noise, the measurement will deliver some large value for ρ and a (possibly even negative) value near zero for $\Delta\phi$. The product delivers an arbitrary value and hence the length of the straight part is incorrect. Due to the integrating nature of the reconstruction algorithm, this error propagates to the end of the reconstructed curve.

6.3 Reconstruction Tests with Synthetic Data

To test the algorithms presented in the previous section, synthetic curvature data sets are created. Two geometric shapes, each with different parameterization (i.e. simulated sample positions), are investigated:

- Ellipses
- Curves of constant width (CCW)

6.3.1 Tests with Elliptical Data

Reconstruction tests are performed with computed curvature data of an ellipse. An ellipse has been chosen, since the radii of curvature can be computed exactly. Furthermore both

nearly circular ellipses with rather constant radii of curvature and extreme flat ellipses with a broad range of curvatures can be investigated.

The following test parameters are selected:

1. Distribution of sample points, at which the curvatures are computed: Equally spaced arc length and equally spaced rotational angle.
2. The ratio of semimajor axis a and semiminor axis b of the ellipse: $b : a = 0.7$ and $b : a = 0.4$.

Creating test data

Using the following parameterization to describe the ellipse with semimajor axis a and semiminor axis b,

$$x = a\cos(t),\ y = b\sin(t),\ \text{where } t = \{0 \dots 2\pi\}, \tag{6.27}$$

according to Equation 6.2 the curvature κ is computed as

$$\kappa(t) = \frac{ab}{\left(b^2\cos^2(t) + a^2\sin^2(t)\right)^{\frac{3}{2}}}. \tag{6.28}$$

The relation between the parameter t and the rotational angle α is given by

$$\alpha(t) = \arctan\left(\frac{b}{a}\tan(t)\right) \quad \text{and} \quad t(\alpha) = \arctan\left(\frac{a}{b}\tan(\alpha)\right). \tag{6.29}$$

To retrieve a set of sample positions $\boldsymbol{r}_i = [x,\, y]^T$ and curvatures κ_i sampled at constant change of rotational angle $\Delta\alpha$ around the circumference of an ellipse, a linearly spaced set of α_i-values between 0 and 2π is generated and the corresponding parameter values t_i are calculated with Equation 6.29. These t_i are used in Equations 6.27 and 6.28 to determine $\boldsymbol{r}_i$ and κ_i.

The arc length of a curve in parameterized form $C \triangleq \boldsymbol{r}(t) = [x(t),\, y(t)]^T$ between two parameter values t_1 and t_2 is given with:

$$s = \int_{t_1}^{t_2} \left(x'(t)^2 + y'(t)^2\right)^{\frac{1}{2}}\, \mathrm{d}t \tag{6.30}$$

In case of the ellipse (Equation 6.27) the arc length is:

$$s = \int_{t_1}^{t_2} \left(a^2\sin^2(t) + b^2\cos^2(t)\right)^{\frac{1}{2}}\, \mathrm{d}t \tag{6.31}$$

As there is no closed form solution for this integral, it is evaluated numerically. Consequently a general explicit expression for $t(s)$ cannot be given. To create test data with points approximately sampled at constant arc length, a simple iterative workaround is used: A circle with its center at an arbitrary point on the ellipse, and with a radius of the desired length of an arc segment, is intersected with the ellipse. One of the intersection points is used as a center to construct a new circle and intersect it with the ellipse. This procedure is repeated until the starting point is reached again (see Figure 6.5).

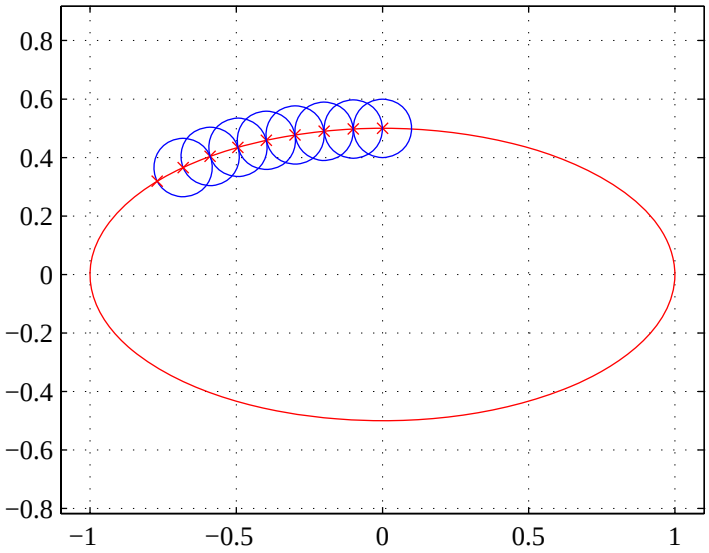

Figure 6.5: The first 8 steps in iteratively constructing sample points with approximately constant arc length distribution.

The intersection points $\boldsymbol{r}_i$ obtained in this manner are then used to calculate the according parameters t_i and hence, the curvatures κ_i.

Results of ellipse reconstruction

The results of the reconstruction with the algorithms presented in Subsections 6.2.1 ("arc-length"-algorithm) and 6.2.2 ("rotational-angle"-algorithm) are shown in Figures 6.6 to 6.9. In all figures the crosses mark the sample positions on the original curve, at which the curvatures are computed. The solid line is the curve, reconstructed with the "arc-length"-algorithm, and the dashed line is the curve, reconstructed with the "rotational-angle"-algorithm.

In the example shown in Figure 6.6 the curvature values of the ellipse are computed at 72 positions with a constant arc length spacing. When utilizing the "arc-length"-algorithm, the curve reconstructed from these curvature values is virtually identical with the model ellipse. Applying the "rotational-angle"-algorithm on this curvature data, yields a curve which is smaller than the original model.

A similar result is shown in Figure 6.7. Here the 72 original points (i.e. sampling positions) are spaced at constant rotational angles. The appropriate "rotational-angle"-algorithm

delivers a correct result, whereas the curve obtained with the "arc-length"-algorithm is too large.

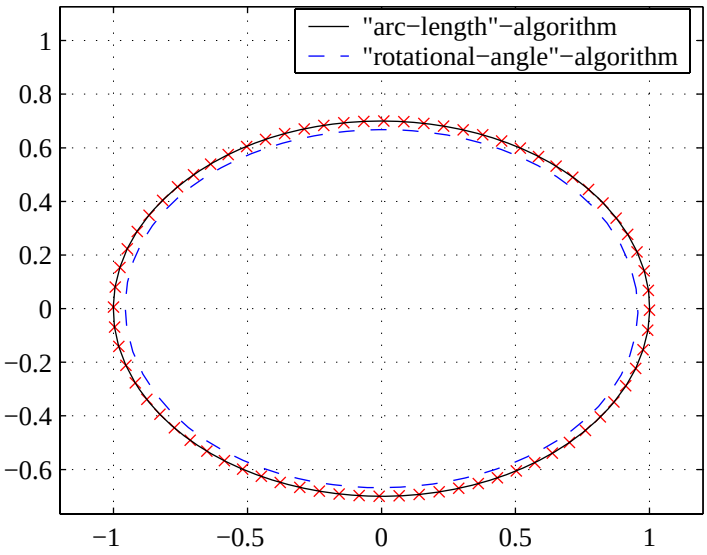

Figure 6.6: Ellipse reconstruction ($b : a = 0.7$). Sample points (crosses) are spaced at constant arc length segments.

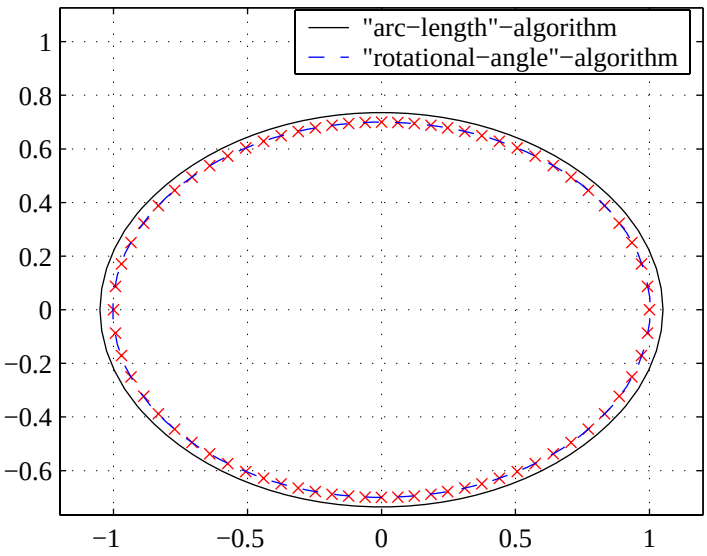

Figure 6.7: Ellipse reconstruction ($b : a = 0.7$). Sample points (crosses) are spaced at constant change of rotational angle.

The tests are repeated with a more shallow ellipse as shown in Figures 6.8 and 6.9. When using the inappropriate algorithm, the deviation (in size and shape) from the original model ellipse becomes more apparent. Considering the constant-angle spacing in Figure 6.9, even the curve reconstructed with the appropriate "rotational-angle"-algorithm does not perfectly match the original sample points. This is caused by the uneven distribution

of the sampling points, which is especially sparse at the principal vertices of the ellipse. Increasing the number of sample points would improve the accuracy of the reconstruction result.

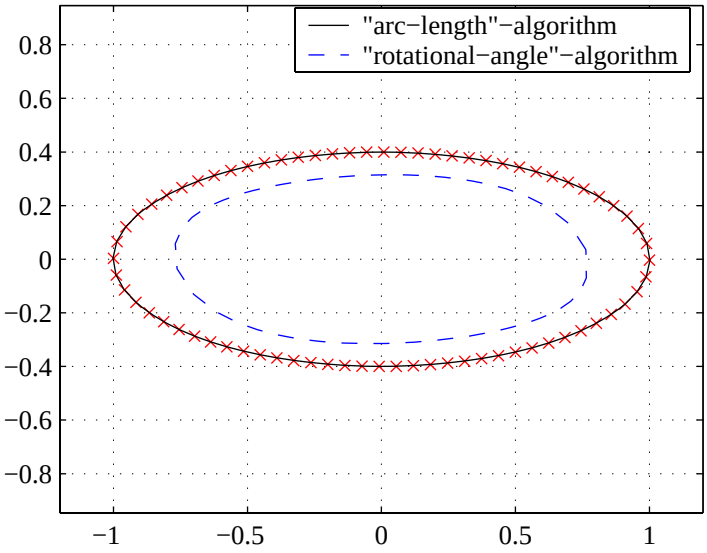

Figure 6.8: Ellipse reconstruction ($b : a = 0.4$). Sample points (crosses) are spaced at constant arc length segments.

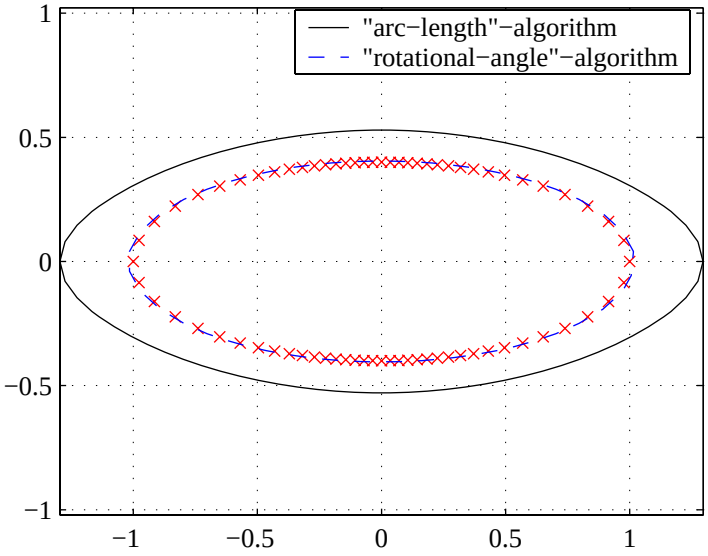

Figure 6.9: Ellipse reconstruction ($b : a = 0.4$). Sample points (crosses) are spaced at constant change of rotational angle.

6.3.2 Tests with Data of a Curve of Constant Width

The second group of geometric objects, which are investigated, are curves of constant width (CCW). CCWs are characterized by the fact that the distance between two parallel tangents is constant. CCWs are considered for testing the reconstruction algorithms, as they are encountered in the pipe measurement application (Section 7.3).

A possible parameterized form of a special symmetric type of CCW is:

$$x(t) = \{1 + b\cos(n\,t)\}\cos(t) + b\,n\sin(n\,t)\sin(t), \tag{6.32}$$

$$y(t) = \{1 + b\cos(n\,t)\}\sin(t) - b\,n\sin(n\,t)\cos(t), \tag{6.33}$$

where

$$b < \frac{1}{n^2 - 1} \quad \text{and} \quad n \in [1, 3, 5, \ldots. \tag{6.34}$$

The parameter n characterizes the number of vertices of the curve. The value of b determines the deviation from the ideal unit circle (which is encountered if $b = 0$).

The following test parameters are selected:

1. Distribution of sample points, at which the curvatures are computed: Equally spaced arc length and equally spaced rotational angle.
2. Two pairs of the values n and b: $n = 3$, $b = 0.08$ and $n = 5$, $b = 0.02$.

Creating test data

Utilizing Equation 6.2 the curvature κ is computed as:

$$\kappa(t) = \left| \frac{1}{1 + (1 - n^2)b\cos(nt)} \right| \tag{6.35}$$

The rotational angle α is:

$$\alpha(t) = \arctan\left(\frac{y(t)}{x(t)}\right) = \arctan\left(\frac{\{1 + b\cos(n\,t)\}\sin(t) - b\,n\,\sin(n\,t)\cos(t)}{\{1 + b\cos(n\,t)\}\cos(t) + b\,n\,\sin(n\,t)\sin(t)}\right) \tag{6.36}$$

Parameter values t for specific rotational angles α are obtained by numerically solving Equation 6.36 for t.

The arc length of a CCW is computed by evaluating Equation 6.30:

$$s(t) = \frac{\{n^2 - 1\}b\,\sin(n\,t) - n\,t}{n} \tag{6.37}$$

The parameter value t corresponding to a specific arc length is computed numerically from Equation 6.37.

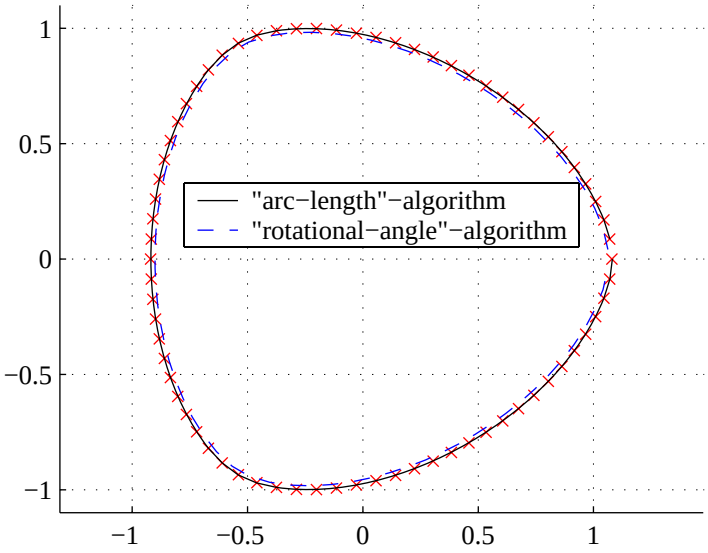

Figure 6.10: CCW reconstruction ($n = 3$, $b = 0.08$). Sample points (crosses) are spaced at constant arc length segments.

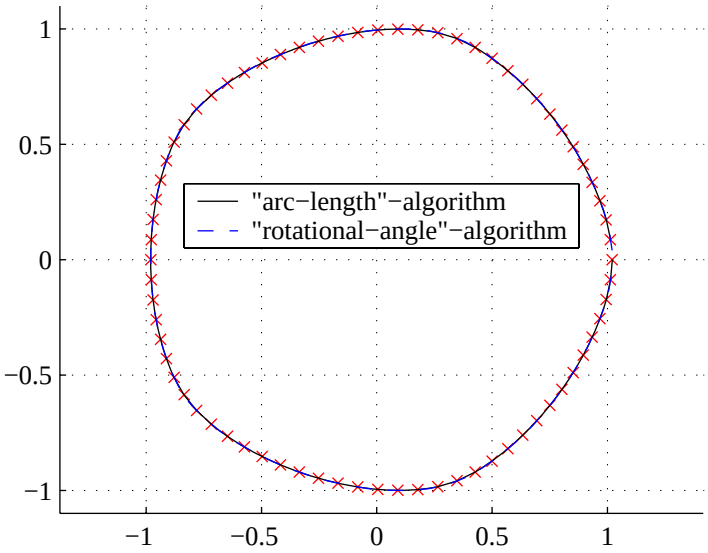

Figure 6.11: CCW reconstruction ($n = 5$, $b = 0.02$). Sample points (crosses) are spaced at constant arc length segments.

Reconstruction results

The reconstruction results with curvature data from a CCW are presented in Figures 6.10 to 6.13. Similar to the ellipse reconstruction plots, the crosses mark the sample positions at which the curvatures are computed. The solid line is the curve, reconstructed with the "arc-length"-algorithm, and the dashed line is the curve, reconstructed with the "rotational-angle"-algorithm.

Similar to the tests with elliptical data, using the algorithm appropriate to the spacing of

the sample positions leads to a reconstruction result that matches the shape of original curve. Unlike the elliptical curves, the difference between arc-length-spacing and arc-angle-spacing is less distinct for CCWs. Thus, even when using the inappropriate algorithm, the deviation of the reconstructed curve from the original curve is small. Especially for the 5-vertex CCW (Figures 6.11 and 6.13), which exhibits a near circular shape, both reconstruction algorithms deliver virtually the same result, independent from the actual sample spacing.

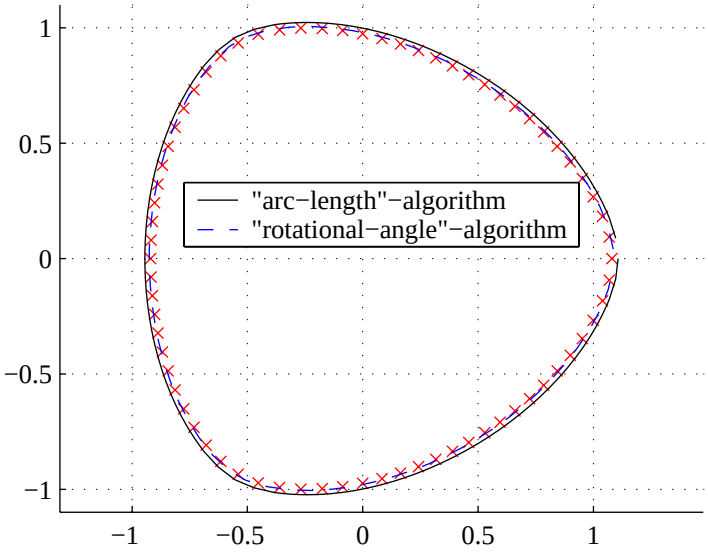

Figure 6.12: CCW reconstruction ($n = 3$, $b = 0.08$). Sample points (crosses) are spaced at constant change of rotational angle.

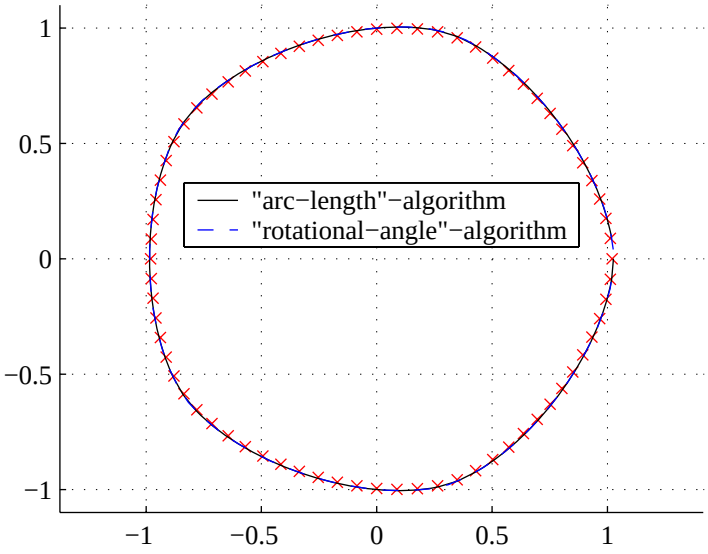

Figure 6.13: CCW reconstruction ($n = 5$, $b = 0.02$). Sample points (crosses) are spaced at constant change of rotational angle.

6.4 Summary of the Algorithms and the Results

The findings about curve reconstruction from curvature data are summarized and conclusions are drawn from the presented results:

1. The way of driving the rotation of the measurement object determines the spacing of the positions, where the curvatures of a cross-sectional profile are measured. Constant arc-length-spacing and constant rotational-angle-spacing are obtained with the two most relevant mechanical driving situations.

2. Two reconstruction algorithms are derived, each appropriate for a specific sample spacing (i.e. arc-length or rotational-angle-spacing). Compared to the "arc-length"-algorithm, the "rotational-angle"-algorithms exhibits two drawbacks: Firstly, in addition to the radius of curvature, the tangential angle is required at each sample position, and secondly, the algorithm cannot be applied to reconstruct curves which exhibit straight parts.

3. Reconstruction tests with synthetically created input data show, that the results obtained with the appropriate algorithm are a close approximation of the original model curve.

4. If the curve is near-circular and the variation of the curvature values over the circumference is small, the sample positions obtained by arc-length-spacing and rotational-angle-spacing are very similar, and thus both reconstruction algorithm deliver equivalent results. This fact can be advantageously used, if near-circular shapes need to be measured: In most cases it is mechanically easier to perform a shaft driven rotation, but the arc-length algorithm is more convenient as only the curvatures need to be measured. According to the test results, combining these two merits is appropriate in case of near-circular curves. An exemplary task for such a simplified procedure is testing cross-sectional profiles for circularity.

Part II

Metric Vision Applications

Chapter 7

Profile Measurement

In this chapter a new approach to measure cross-sectional profiles of rotating objects based on light sectioning, circle fitting and curve reconstruction is presented.

Iteratively points on a locally confined part of the rotating surface are measured with a light sectioning measurement head and approximated by a circle. The radius of the circle is used as a measure of the local radius of curvature of the material. In this manner the radii of curvature over one complete revolution of the profile is acquired and then used to reconstruct the original shape, i.e. to determine a set of points that describes the shape.

The motivation for this work is founded by the need to measure cross-sectional profiles of parts in industrial production, mainly within the quality control process. Existing optical measurement methods for this task can be classified in three main groups:

1. Multiple light sectioning heads, typically four or more, are used to acquire the complete cross section simultaneously[52]. The disadvantage of this method is, that the relative positions of the measurement heads need to be stabilized with respect to each other. This normally requires a complicated mechanical mounting, which is held at constant temperature, to avoid the problems associated with the thermal expansion. This, and the large number of heads, leads to high costs.

2. Shadow measurement systems. A single shadow head cannot measure curves of constant width. It will misinterpret a curve of constant width as being an eccentric object. Solutions have been found with multiple heads[28], but have similar problems to the above solution. Furthermore, only convex profiles can be measured.

3. Rotating the object and using optical triangulation to measure the position of a point on the surface with respect to some reference point. This method, however, is subject to errors resulting from the eccentricity of the object.

The proposed measurement procedure based on iterative curvature measurement and curve reconstruction is advantageous for the following reasons:

1. The curvature measurement system is more compact, as it does not need to clasp around the complete profile, but measures at one single position.

2. A frequent requirement is to integrate a measurement system in an existing production plant. Due to its compactness the integration of a curvature measurement system is mechanically easier and thus more cost efficient.

3. To measure the complete circumference of a vertically oriented profile at once, the measurement must be partially performed from the bottom up. In such an orientation, especially optical measurement system are prone to dust and dirt. The orientation of the curvature measurement system is preferably from the top down.

4. The complete circumference might be inaccessible due to the production process or the production safety.

5. The rotational movement of the material, which is required for the proposed measurement principle, is inherent in many production processes (e.g. turning, grinding, etc.).

7.1 Instrumentation

A light sectioning measurement head[27] as shown in Figure 7.1 is used to repeatedly acquire the local geometry of the rotating measurement object. It contains a laser with line optics, a camera with lens and an optical interference filter optimized for the wavelength of the laser light. Intelligent measurement heads are additionally equipped with a processor to perform basic image processing tasks[1]. The measurement principle, optical requirements and two possible calibration techniques are discussed in the following subsections.

Figure 7.1: Light sectioning measurement head[1]

7.1.1 Principle of Light Sectioning

Light sectioning is a range imaging method to optically determine sections of an object.

[1]image used with permission of Hot Vision Research GmbH, 8700 Leoben, Austria

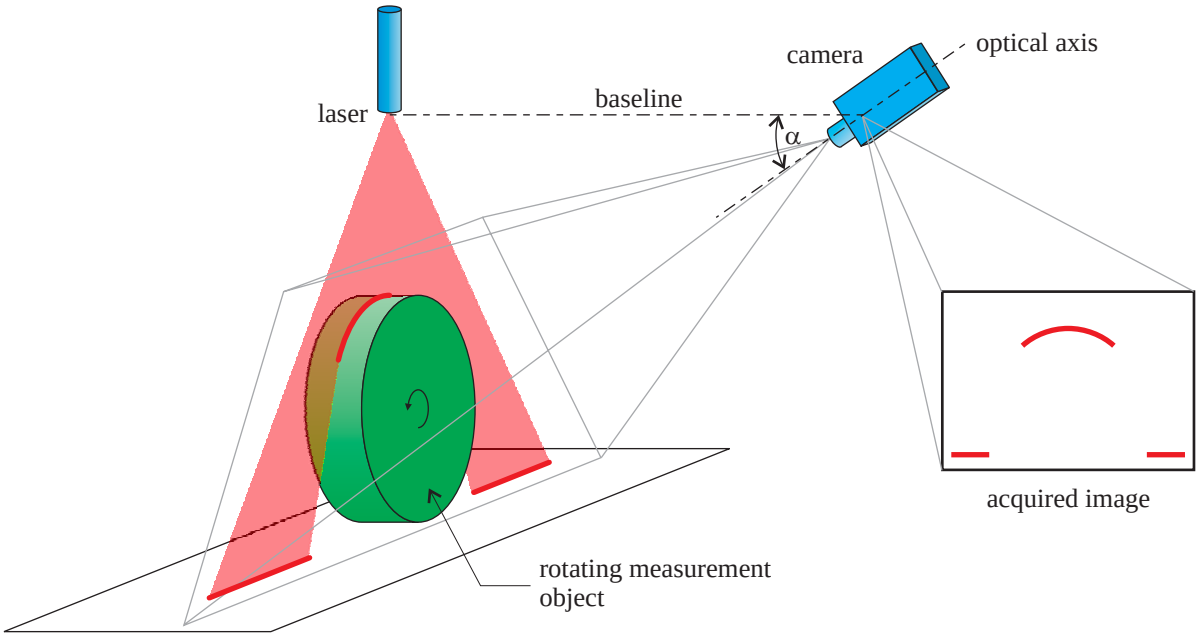

Figure 7.2: Principle of light sectioning

The principle optical arrangement of a light sectioning measurement system is shown in Figure 7.2. A laser, with cylindrical optics, projects a light plane onto the surface of the object. The laser optics is selected such that the Rayleigh region[17] lies in the planned measurement range; this results in the maximum measurement resolution. The Rayleigh region of the laser, is that region in which the laser line is optimally focused and has the minimum width.

The plane of light illuminates the surface of the object and the camera acquires the light scattered from the surface. The disturbing effects of external light are reduced by placing an optical interference filter in front of the camera lens. The narrow pass-band ($\pm 10nm$) of the optical filter, is matched to the emission wavelength of the laser.

7.1.2 The Scheimpflug Condition

All points observed by the camera lie in the plane of the laser light and the camera chip itself is a planar device. Consequently, the optics are imaging one plane onto a second plane. The Scheimpflug condition is implemented to optimize the depth of focus (see Figure 7.3). The Scheimpflug condition requires the image plane, the laser plane and the lens plane to intersect at one and the same line (approximation for thin lenses)[43]. To fulfill this condition, the camera chip must be tilted with respect to the lens plane (this is the plane perpendicular to the optical axis, passing through the center of the lens). The optimal tilt angle β is determined by the following relationship:

$$\tan \beta = \frac{b}{a} \tan \alpha, \tag{7.1}$$

whereby a is the object distance, b is the image distance and α is the angle between the baseline and the optical axis (and therewith the angle between laser plane and lens plane; refer to Figures 7.2 and 7.3). Standard C-mount camera/lens connections[15], as used in the majority of machine vision applications, do not provide the possibility to tilt the camera chip with respect to the lens plane. Thus standard optics can only be used, if the ratio of image distance to object distance $b : a$ is small and therewith only a small tilt angle β is required. Focussing the complete measurement range for close-up range imaging, will always require a significant tilt angle β and the usage of a special mechanical mounting.

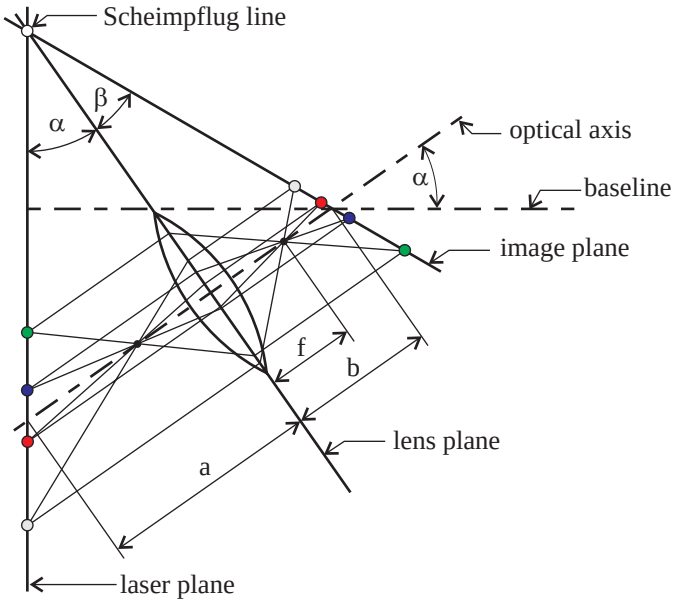

Figure 7.3: The Scheimpflug condition (approximation for thin lenses): laser plane, lens plane and image plane intersect at one and the same line.

7.1.3 Calibration

Ideally, the imaging of the laser-illuminated points on the object surface onto the camera chip can be described by a planar homography between the laser plane and the imaging plane. To rectify the image coordinates, the inverse transformation of this imaging-homography is required, i.e. the calibration homography. This calibration homography has 8 degrees of freedom and can be represented by a 3×3 matrix H. It can be determined or estimated from four or more correspondences of points on the laser plane and the imaging plane (see Subsection 2.4). An approach to identify such point correspondences, utilizing a calibration target with two or more subsidiary planes, is presented by Fauster *et al.*[20].

However, this solution ignores the fact that the optics are not perfect: spherical aberrations of the camera lens and non ideal behavior of the cylindrical laser-optics, lead to a nonlinear mapping between the laser plane and the lens plane. Thus the linear rectification with the calibration homography introduces a source of measurement errors.

An alternative approach that overcomes this problem is to calibrate the light sectioning

measurement head using a precision $x-y$ table: A calibration-object is moved through the full field of view of the measurement head, and measured at each point. The positions of the table and corresponding pixel coordinates are used to generate a look-up table, additionally bi-linear interpolation is used to generate sub-pixel resolution. Calibrating in this manner eliminates the errors associated with the non ideal optics.

Besides this advantage, the $x-y$ table calibration exhibits some disadvantages over the homography based target calibration:

1. The calibration procedure with the $x-y$ table is performed under laboratory conditions, since a precise positioning of the $x-y$ table with respect to the measurement system in an industrial environment is hardly possible. Thus, it can only be applied to closed measurement heads with a fixed optical arrangement. The target calibration can also be applied on-site; this is especially useful to calibrate test setups with varying optical arrangements.

2. The memory usage of the look-up table obtained from the $x-y$ table calibration is huge compared to 9 entries of the 3×3 homography matrix. Considering a camera chip with an 1400×1000 pixel array, storing the corresponding real-world positions (x- and y-coordinates, assuming 4 bytes each) requires approximately 10MB of memory space. In embedded computer systems this amount of memory might not be available.

7.2 Evaluation Procedure

Each step of the evaluation procedure presented in this section is demonstrated using real measurement data from the pipe quality control system presented in Section 7.3.

7.2.1 Acquisition and Data Extraction

An image of the projection of the laser plane onto the rotating object is acquired. The image is represented as a discrete intensity function $I(x,y)$, whereby $x \in \mathbb{N}$ is the column index and $y \in \mathbb{N}$ is the row index. The center of gravity $y_{COG}(x)$ of pixel intensities is determined for each column $x = \{1 \dots N\}$ of the image:

$$y_{COG}(x) = \frac{\sum\limits_{\substack{y=1\dots M \\ I(x,y)>I_{thresh}}} y \cdot (I(x,y) - I_{thresh})}{\sum\limits_{\substack{y=1\dots M \\ I(x,y)>I_{thresh}}} (I(x,y) - I_{thresh})} \quad \forall \quad I(x,y) > I_{thresh}, \tag{7.2}$$

whereby M is the number of rows, N is the number of columns and I_{thresh} is an intensity threshold. Pixel with an intensity below this threshold are neglected. It is assumed that they are not illuminated by the laser and thus they would distort the position of the

center of gravity. Their intensity is either caused by ambient light, which is not completely removed by the optical filter, or by residual dark charge (thermal noise)[15].

The obtained positions of the center of gravity $y_{COG}(x)$ together with the number of the corresponding column x form the image coordinates of the points that represent the light-section profile (see Figure 7.4).

Figure 7.4: Acquired image with extracted light-section profile (red dots).

7.2.2 Rectification

The image coordinates of the light-section profile are rectified to a metric coordinate frame (see Figure 7.5). Depending on the type of calibration (cp. Subsection 7.1.3), this is performed either by using the calibration homography H, or by selecting the corresponding metric coordinates from a look-up table (and using bi-linear interpolation for sub-pixel resolution).

7.2.3 Circle and Tangent Fitting

The determination of the local curvature of the section profile is performed by fitting a circle to the rectified points. The selection of the fitting algorithm is based on the findings presented in Chapter 5 and depends on the available processing time:

- A nonlinear iterative method (e.g. Gauss-Newton as presented in Subsection 2.3.4) is used, if the processing time is sufficient. This delivers the best results with respect to bias and standard deviation. In case of online evaluation, the processing time is restricted by the acquisition interval. Thus the used processing time is an additional stop criterion of the iteration.

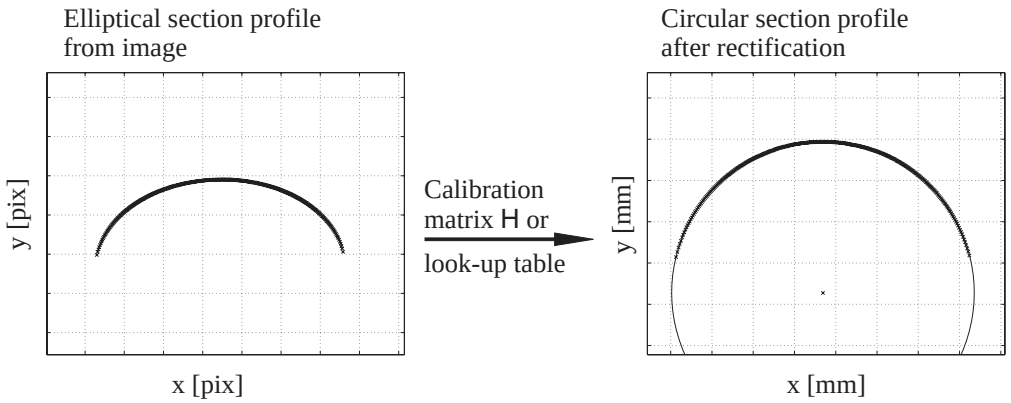

Figure 7.5: Rectification of the section profile. Left: The image of the section profile shows an elliptically arranged cloud of points. Right: Transformation of the points exhibits the real near-circular shape of the local cross section.

- The simple algebraic fitting routine (SAF, refer to Subsection 2.3.1 for a detailed description) is used in case of limited processing time. It performs best of all algebraic methods when fitting circles to circular arc data and is also adequate for online processing.

The radius of the fitted circle is used as an estimate for the local radius of curvature ρ and the tangent position t determines the eccentricity of the rotating object. The eccentricity will be used to determine the period of revolution. The progression of these two signals (as exemplarily shown in Figure 7.6) is recorded during the rotation of the object.

7.2.4 Signal Conditioning

The next processing steps regard to signal conditioning. Measurement outliers in both signals are detected and removed. The number of measurements corresponding to one complete revolution of the pipe is determined and finally the noise components in the signals are suppressed by harmonic filtering.

Outlier detection

To detect outliers in the measurement values, the following conditions for both the radius and the tangent position are checked:

1. Range of the values. For both signals the range of possible measurement values is physically limited. The actual value of the limits are determined individually for each measurement problem. Measurement values beyond these limits indicate a false measurement and are removed.

2. Gradient of the values. The difference between to subsequent measurement values must not exceed a certain limit, as both signals are continuous in their nature.

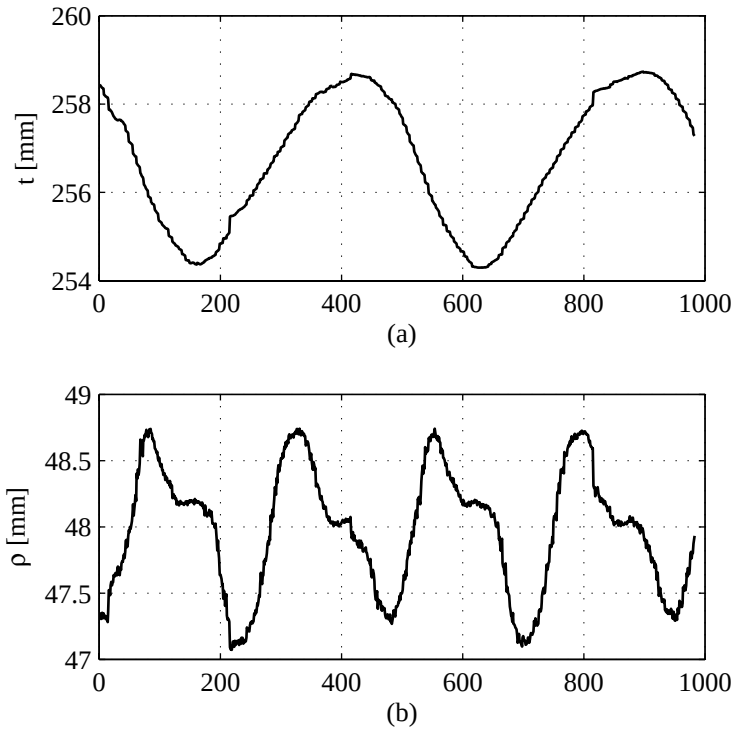

Figure 7.6: The measurement signals: (a) tangent position of the rotating material, (b) radius of curvature.

Abrupt changes in the tangent position or in the radius of curvature are indications of false measurements. Thus measurement values with gradients above a defined threshold are removed.

Determining the period

The exact number of individual measurements that corresponds to one complete revolution of the object may vary due to different diameters. To determine the length of one rotational period, the normalized aperiodic autocorrelation sequence[47] for the tangent position signal t is calculated (n corresponds to the total number of individual measurements):

$$t_{autocorr}(k) = \frac{1}{n-k} \sum_{j=1}^{n} t(j)\, t(k+j), \quad \text{for} \quad -(n-1) \leq k \leq (n-1) \tag{7.3}$$

The results of this calculation for both the tangent position and the radius of curvature are shown in Figure 7.7. The number of measurements $\tilde{n}$ between two subsequent maxima of the autocorrelated tangent position signal can be clearly identified and corresponds to one revolution of the object.

A discrete periodic sequence for the tangent position and the radius of curvature can now

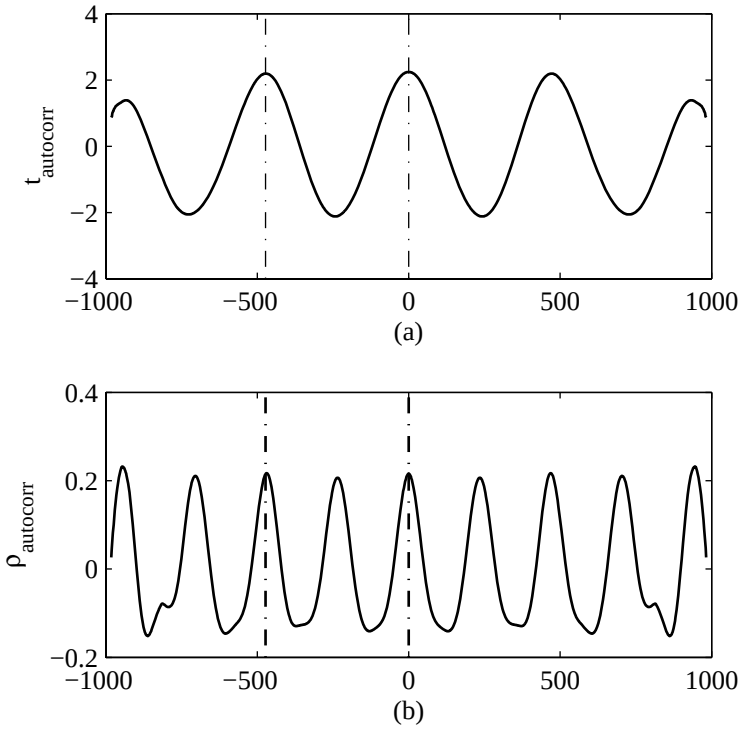

Figure 7.7: Aperiodic auto-correlation sequences for tangent position (a) and radius of curvature (b). The period of revolution has been identified.

be extracted:

$$\tilde{\rho} = \rho(1 \ldots \tilde{n}) \tag{7.4}$$

$$\tilde{t} = t(1 \ldots \tilde{n}) \tag{7.5}$$

Harmonic filtering

Both sequences of measurements, $\tilde{t}$ and $\tilde{\rho}$, are the sum of a perfectly periodic signal and a stochastic component (noise). This enables the application of harmonic filtering, without the necessity for windowing, since perfectly periodic signals do not lead to the Gibb's phenomena[47]. A number of N harmonic coefficients for the radius of curvature signal are determined as follows,

$$C_k = \sum_{i=1}^{\tilde{n}} \tilde{\rho}(i) \mathrm{e}^{-j\left\{\frac{2\pi ik}{\tilde{n}}\right\}} \quad \text{for} \quad 0 \leq k \leq N. \tag{7.6}$$

The actual number of harmonic coefficients used, is application dependent. For the presented example 10 harmonic coefficients are determined. The filtered sequence is now synthesized using,

$$\tilde{\rho}_h(k) = \sum_{i=1}^{N} C_i \, \mathrm{e}^{-j\left\{\frac{2\pi ik}{\tilde{n}}\right\}}. \tag{7.7}$$

The result of this harmonic filtering is shown in Figure 7.8. The systematic portion of the data is maintained, whereby the high frequency components, primarily noise, are suppressed. The same procedure can be applied to the tangent position signal.

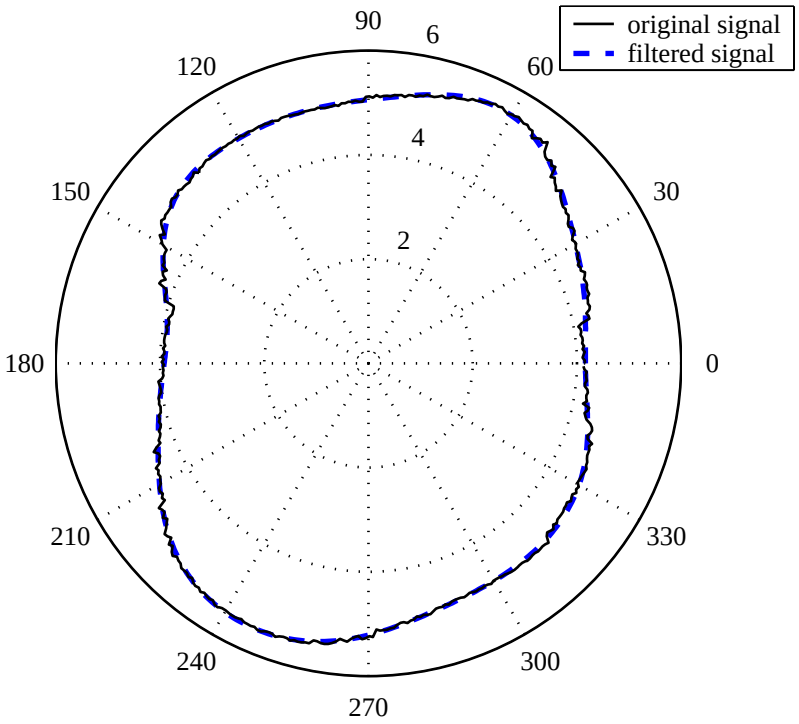

Figure 7.8: Polar plot of the original and filtered sequence of the radius of curvature over one revolution of the object (90% of the mean value have been subtracted to emphasize the variation).

7.2.5 Profile Reconstruction

The final step in the measurement evaluation is the reconstruction of the cross-sectional profile from the measured radii of curvature. The conditions of rotating the measurement object and the appropriate reconstruction algorithms are thoroughly discussed in Chapter 6. Depending on the way of driving the rotation, the appropriate algorithm is chosen: The "arc-length"-algorithm (see Subsection 6.2.1) is chosen, if the object is rotated by contact to circular rolls, whereas the "rotational-angle"-algorithm (see Subsection 6.2.2) is appropriate, if the rotation is shaft-driven.

Example results of the reconstruction with both algorithms are shown in Figure 7.9. Due to the near circular cross-section of the measurement object, the reconstruction results obtained with both algorithms virtually coincide. To visualize both the deviation from an ideal circle and the negligible difference of the reconstruction results, the profile is scaled such that 97% of the mean radius are subtracted.

All necessary steps for profile measurement of rotating objects have been discussed so far.

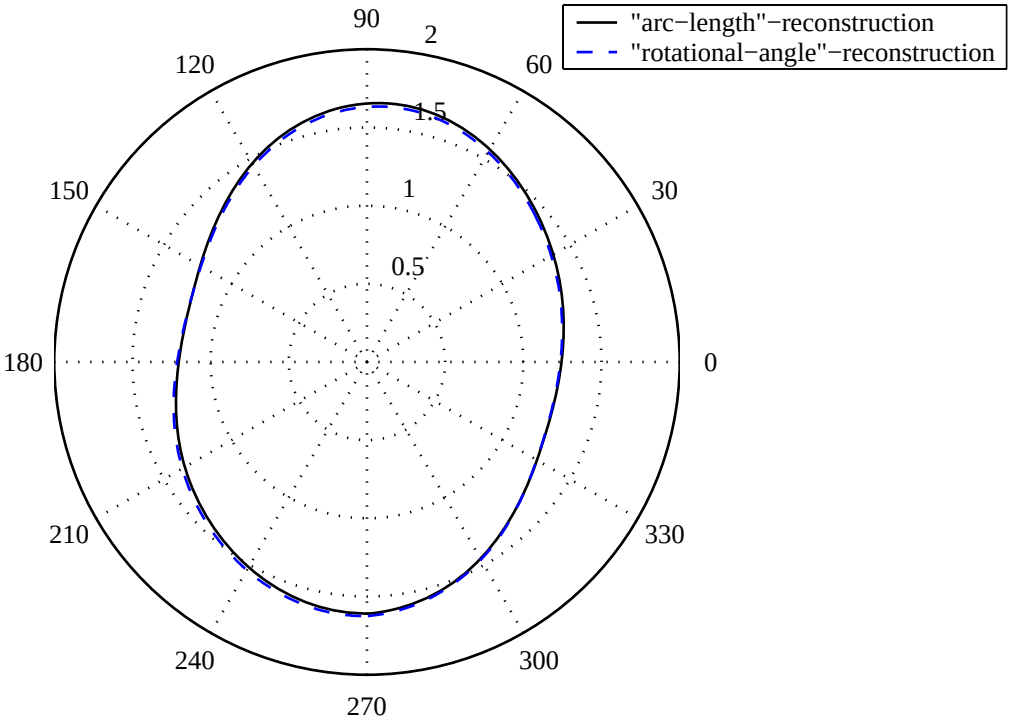

Figure 7.9: Polar plot of the cross-sectional profile, reconstructed with "arc-length"-algorithm and "rotational-angle"-algorithm. 97% of the mean radius are subtracted to visualize the deviation from an ideal circle. Due to the near circular shape both algorithms deliver virtually the same result.

The following section describes an industrial implementation of such a profile measurement system.

7.3 Industrial Implementation as a Quality Control System for Pipes

7.3.1 Problem Statement

Seamless steel pipes are a highly stressed component in the oil production industry. Two main quality features of such pipes are their straightness and the circularity of their cross-sectional profile. The straightness crucially determines the critical load, which a pipe can withstand before buckling. Furthermore both the straightness and the circularity have an influence on the quality of threads that are cut at the end of the pipes for screwed, leak-proof pipe connections.

Although the final step of the pipe production is an adjustment at a rolling-straightener, the ends of the pipes, where not all rolls are simultaneously engaged, may exhibit a significant bending. As this bending occurs randomly and cannot be effectively influenced

by process parameters, a straightness inspection after this final production step is required for each pipe (100%-quality control). Therefor the pipes are placed on radial rollers and rotated around their longitudinal axis. The eccentricity, the rotating pipe exhibits at its end, is used as a direct measure of its straightness. If a certain eccentricity level is exceeded, the pipe must be re-straightened.

Up to now a mechanical tactile sensor is used to measure the eccentricity of the rotating pipe. However this system is error-prone due to particles, such as scale, on the surface and has high maintenance costs due to abrasion and associated wear of the contacting probe. Moreover, this system is not able to measure the circularity of the cross-sectional profile of the pipe.

Machine vision based solutions for straightness inspection, which are contact-less and therefore not susceptible to abrasion, are barely discussed in literature. Lu *et al.*[42] present an experimental solution based on light sectioning. Several triangulation sensors are placed along the longitudinal axis of the pipe and measure the 3D center position of the section profile by applying an ellipse fitting routine on the data. The deviation of these positions from a straight line is used as measure for the straightness of the pipe. However, the number of required measurement sensors depends on the length of the pipe and the proposed setup is not applicable to measure the circularity of the cross section.

As a new approach, the presented profile measurement system is applied to determine both the straightness and the circularity of the pipes. The system is industrially implemented in an existing pipe production plant.

7.3.2 Industrial Measurement Setup

The measurement setup is shown in Figure 7.10. Consecutively both ends of the pipe are positioned under a profile measurement head. The pipe is rotated around its longitudinal axis for at least one complete revolution. The rotation is driven by contact to circular drive rolls.

The deployed measurement heads are intelligent sensors with processing capabilities. Both image acquisition and data extraction are performed by the measurement head. Thus only the extracted points of interest need to be transmitted to a central processing node, where further evaluation steps are performed. Due to this evaluation design up to 500 images can be acquired and processed per second.

The dedicated camera is based on the imaging device MT9M413 from Micron Technology, Inc., with a resolution of 1280×1024 pixels, and 500 full frames per second. A field programmable gate array, within the camera, processes the image data at full frame rate; delivering the position of the laser line in each column of the image.

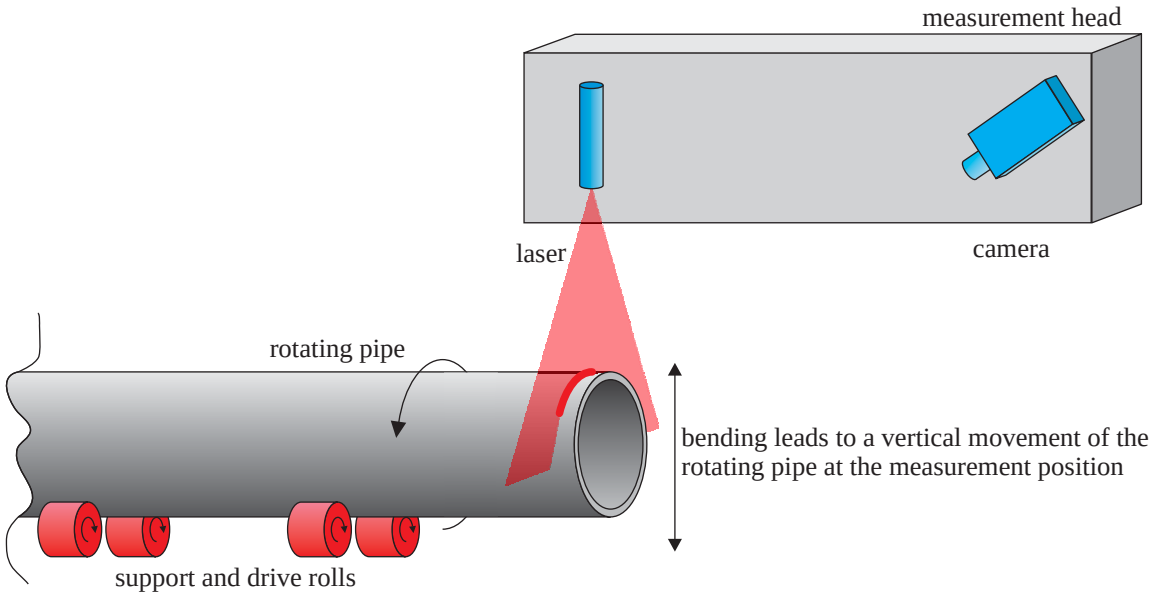

Figure 7.10: Schematic view of the measurement setup: The pipe is rotated by contact to circular drive rolls. The profile measurement head repeatedly measures the local geometry and the vertical position of the rotating pipe.

7.3.3 Evaluation of the Straightness

The straightness measurement with the profile measurement head is based on the same principle as with the tactile sensor: Bending of the pipe leads to an eccentric movement at the end of the pipe. This movement is optically measured by determining the variation of the tangent position of the circle fitted to the acquired data. The maximum deflection is used as an inverse measure of the straightness of the pipe.

Within the evaluation procedure of the profile measurement system, the progression of the tangent position of the circle fitted to the section profile is determined (see Subsection 7.2.3). The eccentric movement of the pipe is determined from this sequence of measurements (see Figure 7.11):

$$ecc = \frac{t_{max} - t_{min}}{2}. \tag{7.8}$$

This value is then utilized as a measure of the straightness of the pipe and compared to a tolerance limit, which is specifically defined for each class of pipes. If the measured eccentricity value exceeds the tolerance limit, the pipe is sorted out for repeated straightening.

7.3.4 Evaluation of the Circularity

The profile measurement head iteratively measures the local radii of curvatures. The real shape of the cross-sectional profile is then reconstructed from the curvature data und used

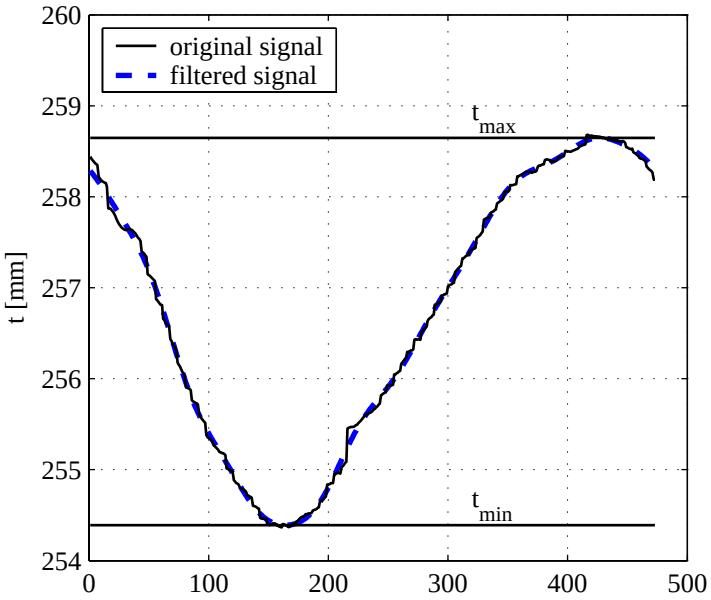

Figure 7.11: Originally measured and filtered tangent position signal.

to determine the circularity of the pipe.

The reconstruction is performed using the "arc-length"-algorithm (see Subsection 6.2.1). Although the measurement position is at the top side of the pipe and the drive roll contacts the pipe at the bottom side (cp. Figure 7.10), it can be assumed, that the measurement positions are sampled at constant arc-length segments. This is due to the nearly perfect circularity of the cross-sectional profile of the pipes.

Simulated caliper measurement

A first characteristic value of the circularity of the cross-sectional profile is found by simulating the measurement with a caliper on the reconstructed shape of the profile and evaluating the difference between the maximum and minimum caliper diameter. A caliper measures the distance between two parallel tangents which lie on the surface of the material. This data can be extracted from the reconstructed x and y coordinates which are available as functions of the tangential angle ϕ (see Figure 7.12):

$$d_{Caliper}(\phi) = \sqrt{\{x(\phi) - x(\phi + 180°)\}^2 + \{y(\phi) - y(\phi + 180°)\}^2} \tag{7.9}$$

Identifying curves of constant width

Some care needs to be taken when interpreting the results of a caliper measurement: curves of constant width (CCWs)[41, 58] are commonly encountered when rolling or straightening a pipe with three rolls each separated by 120° (see Figure 7.13).

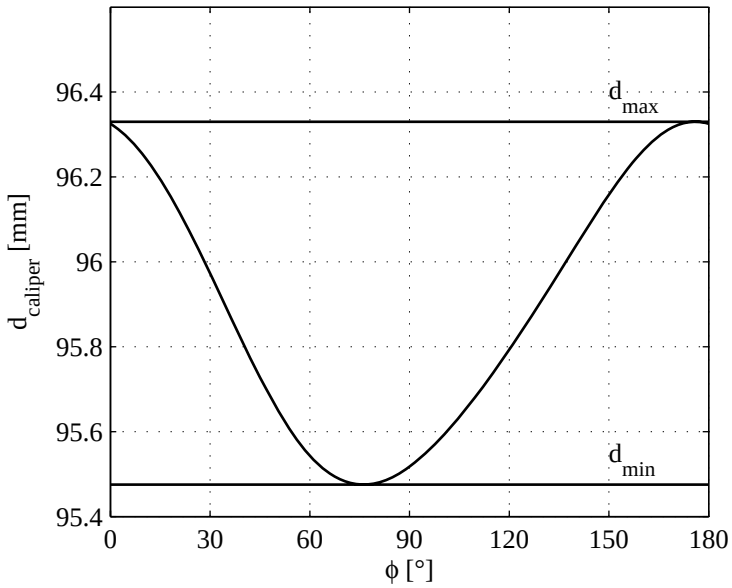

Figure 7.12: Diameter as would be measured using a caliper.

(a) (b)

Figure 7.13: Rolling a pipe with three rolls (a); resulting profile: a curve of constant width with $n = 3$ vertices (b).

CCWs are characterized by the fact that the distance between two parallel tangents is constant. Thus the measured caliper diameter is constant over the complete circumference of the curve, although the curve may clearly deviate from an ideal circle.

Considering the parameterization of a CCW as presented in Subsection 6.3.2 (see Equations 6.32 and 6.33), the radius of curvature is given by:

$$\rho(t) = |\, 1 + (1 - n^2)\, b\, \cos(n\, t)\, |, \tag{7.10}$$

whereby n determines the number of vertices and $|b| < \frac{1}{n^2-1}$ is a measure of the deviation from an ideal circle. Consequently, in the presence of a CCW with this characteristic, the auto-correlation sequence of the measured radius of curvature will exhibit a third harmonic, relative to the period of revolution of the material. This case can be seen in the measurement result shown in Figures 7.14a and 7.14b.

Equation 7.10 gives the key to finding a least-square approximation for the CCW to a sequence of measurements of the radius of curvature ρ. The CCW, described by Equations 6.32 and 6.33, is fully and uniquely described by the Elliptical Fourier Descriptor[53] with the zeroth, C_0, and n^{th}, C_n, coefficients, since Equation 7.10 can be fully described by these Fourier components. The result of such an approximation is shown in Figure 7.14c. Considering the case of a pipe rolled or straightened with three rolls, the absolute value of the third coefficient is an indication of the deviation of the CCW from an ideal circle.

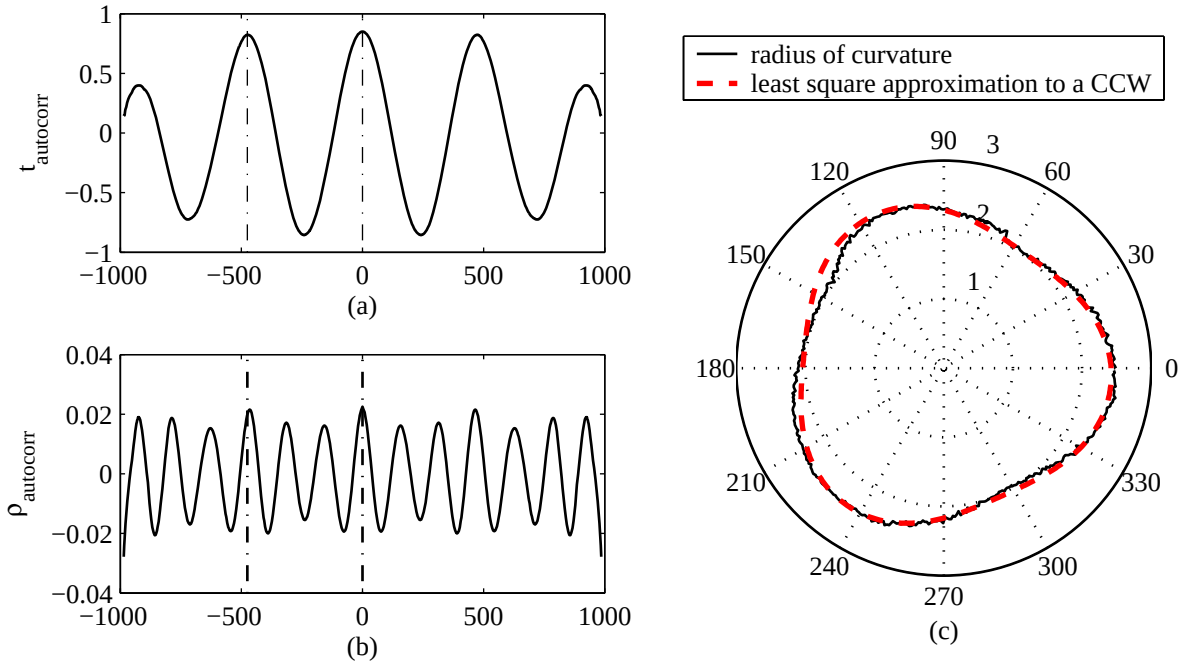

Figure 7.14: Aperiodic auto-correlation sequences for tangent (a) position and radius of curvature (b). The third harmonic relationship between tangent position and radius of curvature suggests that a CCW has been measured. Least square approximation of the radius sequence to a CCW (c).

7.3.5 System Configuration and Plant Integration

An overview of the system configuration is presented in Figure 7.15. The core components of the system are two intelligent profile measurement heads (one for each end of the pipe) and the central processing and visualization PC. The measurement heads perform image acquisition, data extraction and segmentation. The image coordinates of the points describing the light-section profile are transmitted to the PC. The PC performs rectification, circle and tangent fitting, and the straightness and circularity evaluation. The results from both ends of a pipe are finally used to make a quality decision.

For field level communication the measurement system is equipped with a National In-

struments Fieldpoint^{TM}-module.[2]. There are input signals of two categories: Signals containing information about the production plant and signals containing information about the measurement system. The former are used for triggering the measurement process, the latter mainly for security reasons, such as turning off the laser, when someone enters the measurement area. Output signals are used to communicate with the production plant to initiate sorting out of pipes with a negative quality decision.

An RS232 serial connection to the company wide quality control system enables monitoring of the performance of the production plant and integration of the measurement results to the production log of the pipes.

Remote access to the measurement system is provided over a TCP/IP network connection. This is mainly used for remote visualization and software maintenance reasons.

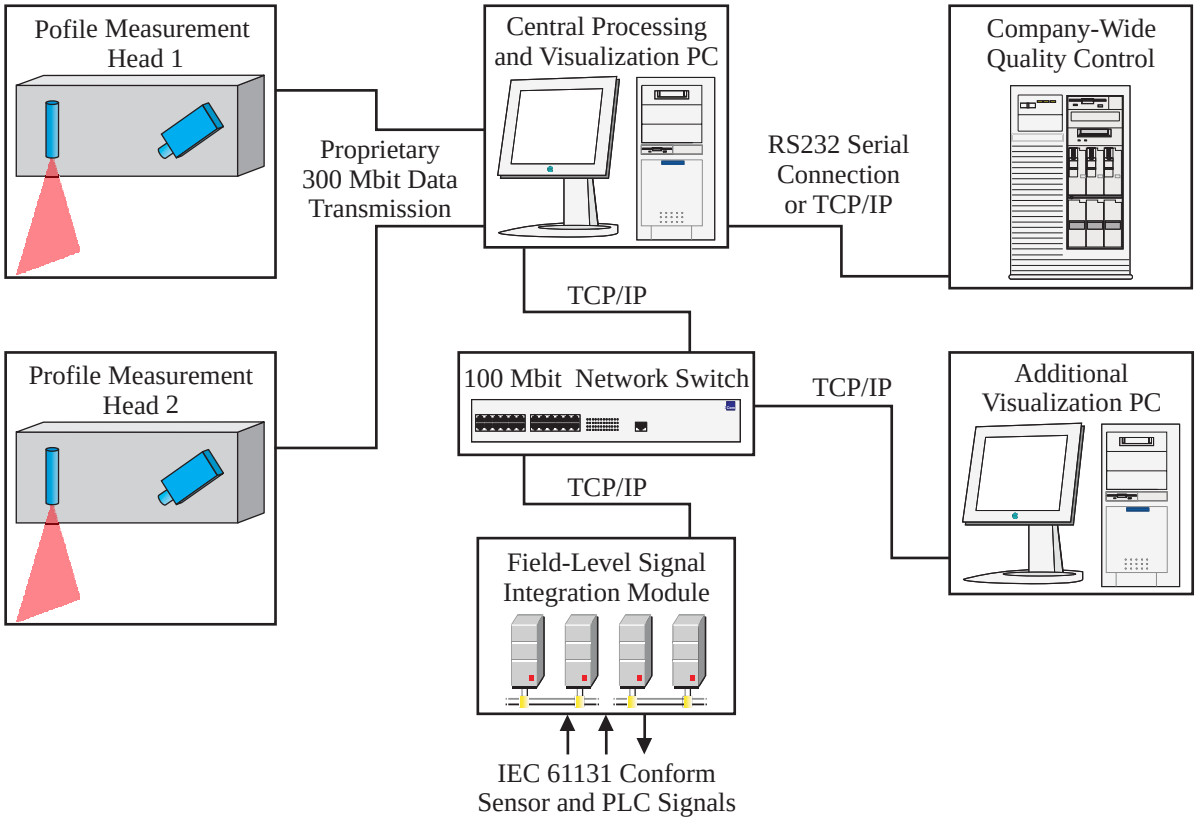

Figure 7.15: Overview of the system configuration.

The pipe quality measurement system has been embedded in an existing pipe production plant. An enclosed measurement station is shown in Figure 7.16.

[2]see: http://www.ni.com/products

Figure 7.16: Measurement station in the plant

7.4 Summary and Conclusion

A new approach to measure cross-sectional profiles of rotating objects is presented. The measurement is based on reconstructing the shape of the profile from local radii of curvatures measured with a single light sectioning system. It is a viable alternative to conventional solutions to measuring cross-sections, whereby the costs for instrumentation are significantly lower with the new solution.

The local geometry is iteratively measured during the rotation of the object. The radius of curvature is estimated by fitting a circle to the acquired and rectified data.

Fourier harmonic filtering can be applied to suppress noise in the cross-section measurement. There is no Gibbs power since the periodicity of the measurement data can be identified via aperiodic normalized autocorrelation.

The measurement system has been successfully integrated in a pipe production plant. Two quality features, straightness of the pipe and circularity of its cross-section, are determined.

The identification of curves of constant width is possible with the proposed method. Such curves are encountered when measuring the cross-sectional profile of pipes, which were rolled or straightened with an odd number of rollers. A least-squares approximation of curves of constant width can be made using Elliptical Fourier descriptors.

Chapter 8

High Temperature Video Extensometry

A video-extensometer system for optical strain measurement is presented in this chapter. The system is designed to perform measurements at temperatures of up to 1400°C. The motivation for this work arises from the need for a contact-less extension measurement system to obtain material properties of refractory material at its typical working temperature. The material tests are performed by applying a load on a specimen and measuring the deformation of the specimen. There is a lack of tactile probes capable of performing reliable deformation measurements at the high temperature range. Presently, the deformation of the specimen is estimated by measuring the movement of the testing machine. To determine material properties such as the modulus of elasticity more accurate information about the real deformation is required. The presented video extensometer system is a new approach to measure the real deformation of the specimen at elevated temperatures. Besides the discussion of the measurement evaluation procedure, this work focusses on

1. the hardware requirements regarding the high temperature and associated thermal radiation,
2. the identification of sources of measurement errors and potentials to increase the accuracy of the system.

8.1 Problem Statement

Testing of refractory material is performed with specimens as shown in Figure 8.1. A wedge is driven into the gap of the specimen at a controlled speed and the applied force is measured with a load cell. The transmitted load leads to a deformation of the specimen. The sought deformation at the gap opening is called *crack mouth opening displacement* (CMOD). There are two phases of specimen deformation: At first the gap of the specimen is elastically widened. In this test phase, a relatively high load leads to a small deformation. The second test phase starts, when high local stress initiates a crack at the tip of the gap;

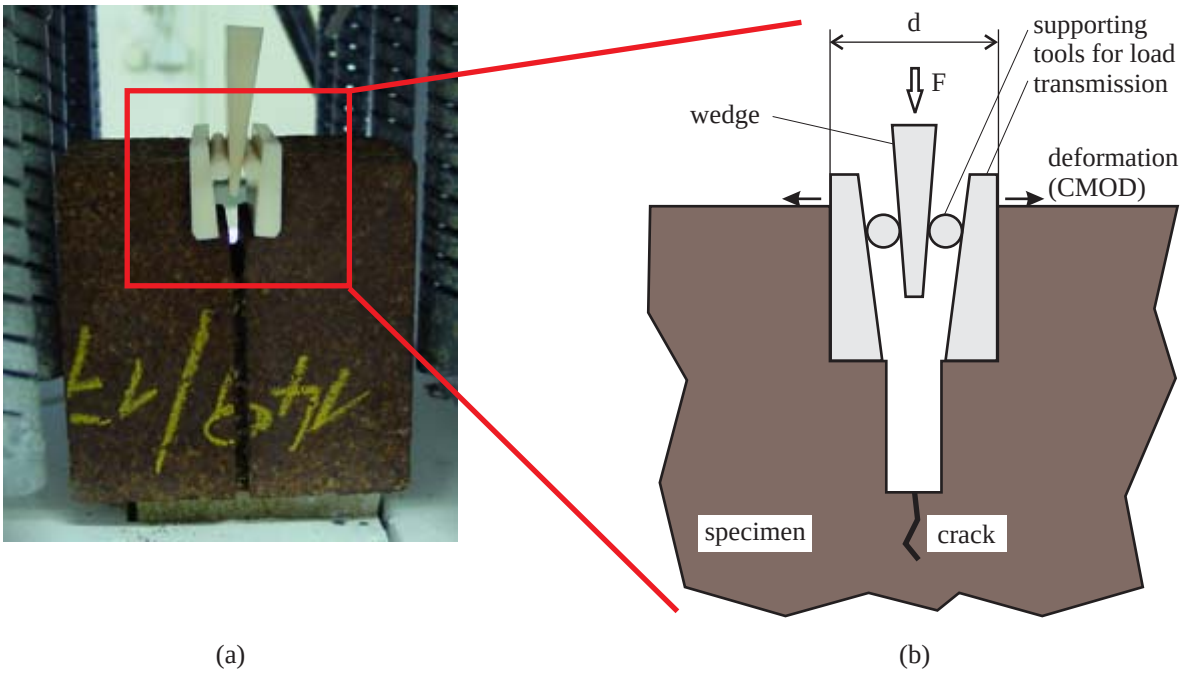

Figure 8.1: Foto (a) and geometry (b) of the specimen

in this phase the crack is growing until the specimen breaks. During the crack growth, the applied load is decreasing, but the deformation of the specimen is progressing.

Measuring the movement of the testing machine yields a poor estimate of the CMOD. The obtained information is suitable to roughly determine the absorbed fracture energy, but it does not allow the determination of the elasticity of the material. Especially in the first phase of the testing procedure (before the crack is initialized), where the relevant data to determine the elasticity is to be collected, the movement of the testing machine and the deformation of the specimen do not correspond.

With the proposed video extensometer system images of the specimen are acquired during the material testing procedure and the CMOD is measured directly using metric vision techniques. The aims of the video extensometer measurements are:

1. Increasing the accuracy of the fracture toughness determination;
2. Enabling the measurement of the modulus of elasticity.

8.2 Hardware Setup

The instrumentation and hardware setup of the video-extensometer is shown in Figure 8.2. The test-bench setup for heating and loading the specimen at the left hand side of the figure is basically the same as without the usage of the video-extensometer. One side of the oven has been opened and equipped with a quartz glass window to enable a free view

of the camera to the specimen. The optical setup and the image acquisition and evaluation hardware are discussed in detail.

8.2.1 Optical Setup

The optical setup is guided by the following requirements:

1. Measurements are made at temperatures from room temperature up to 1400°C; hence the camera must be isolated from the heat in the oven.
2. The specimen is emitting visible and infrared heat radiation. The intensities in the infrared range dominate the emitted radiation spectrum by far. The CCD chip of the camera must be suitably protected from this radiation.
3. The amount of near-infrared and red radiation must be reduced to preserve visible contrast between the specimen and the background or supporting tools in the image.

The following optical components are used to meet these requirements (see Figure 8.2):

1. **Quartz glass**: A quartz glass isolates the interior of the oven from the environment. It withstands temperatures up to 1600°C and exhibits suitable transmission properties for visible light in the complete temperature range at which measurements are performed. Images are acquired from outside the oven through the quartz glass.
2. **Hot mirror**: A hot mirror[55] is a glass with dielectric coating, that reflects up to 90% of infrared radiation with a wavelength above 700nm. This component protects the imaging device from a major part of the radiation energy. Due to the reflection rather than absorbtion of the radiation, the hot mirror and its housing are not critically heated.
3. **Optical filter**: An optical filter reduces the amount of emitted visible red light that reaches the camera chip. This light would reduce the contrast in the image.

8.2.2 Acquisition and Evaluation Hardware

The proposed video-extensometer is a PC-based system. Image acquisition is performed with a JAI CV-S3200[1] color camera (PAL standard) and a national instruments NI PCI 1411[2] color frame grabber. External signals such as the load cell data can be accessed via Beckhoff fieldbus components[3]: Analog input and output modules and an ethernet bus coupler enable the integration of these signals over a TCP/IP network connection. The evaluation of the CMOD is performed online on the PC, where the corresponding load and deformation data is also merged.

[1] www.jai.com
[2] www.ni.com
[3] www.beckhoffautomation.com

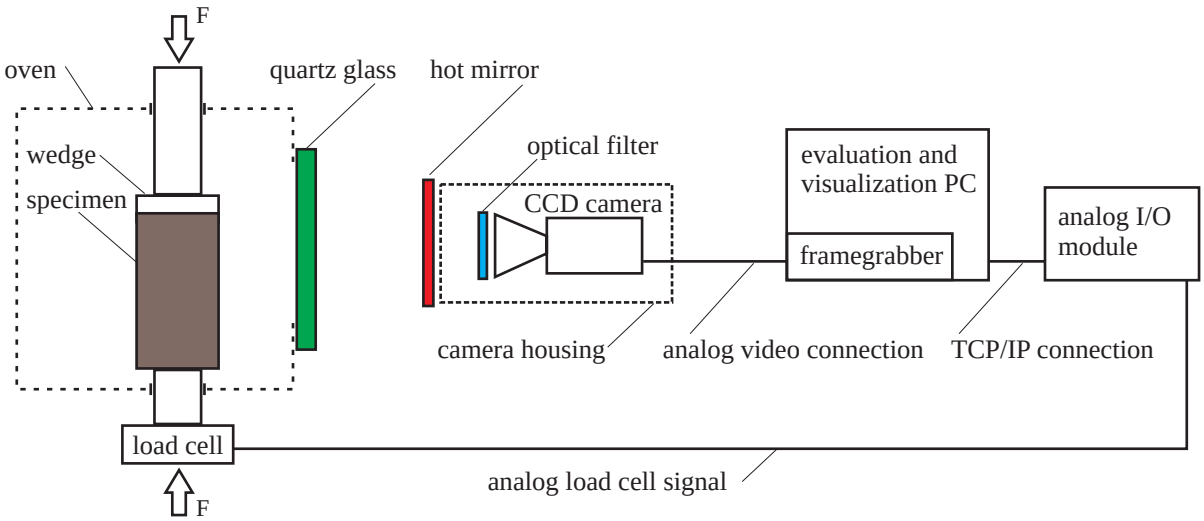

Figure 8.2: Measurement setup

8.3 Measurement Evaluation

8.3.1 Evaluation Procedure

A series of images is acquired and processed online during the testing procedure. In each image the distance d, as shown in Figure 8.1b, is calculated and the change of that distance (i.e. the CMOD value) is recorded.

The principal steps of processing the acquired images are (see Figure 8.3):

1. Extract the points, which describe relevant edges of the specimen (see below for a detailed description).
2. Abstract the edges by fitting lines to the extracted points (see Section 2.2).
3. Intersect the fitted lines to obtain the corners $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ at the crack mouth opening.
4. Calculate the distance $d = \|\boldsymbol{p}_1 - \boldsymbol{p}_2\|$ (whereby $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ are expressed in affine coordinates) between these corners . The change of this distance in the sequence of images is the sought CMOD.

The extraction of points is performed in regions of interest (ROIs) of the image. These ROIs, which contain the relevant edges, are initially selected by the user before starting the material testing process. Then the ROIs are automatically tracked as the positions of the edges change in the image throughout the testing process. Ideally each ROI contains a nearly vertical or horizontal edge and the processing steps are uniform for each ROI (see Figure 8.4):

1. **Smoothing**: A 5x5 averaging filter is applied to reduce pixel noise.

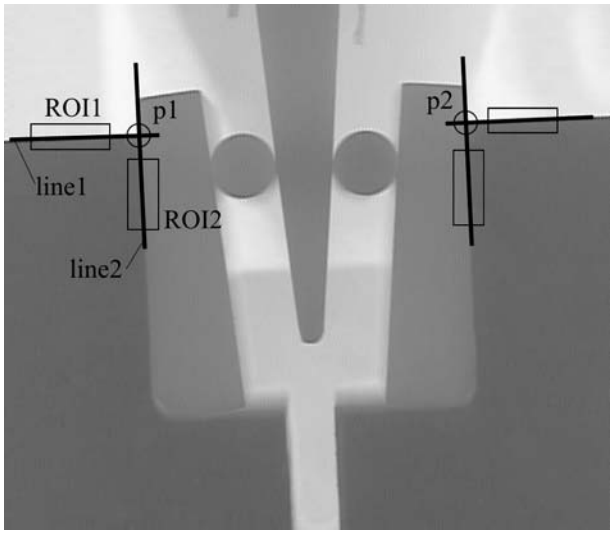

Figure 8.3: Evaluation of an image.

2. **Edge enhancement**: A 3x3 Prewitt filter[57] enhances the edges in each ROI. The orientation of the Prewitt filter is chosen according to the orientation of the edge (horizontal or vertical, positive or negativ) in the ROI.

3. **Point extraction** The extraction of a set of edge points in each ROI is performed by calculating the center of gravity (COG, cp. Subsection 7.2.1 and [45]) in each row/column (depending on the edge orientation). One coordinate of each edge point is the corresponding index of the row/column. The second coordinate is the result of the COG calculation.

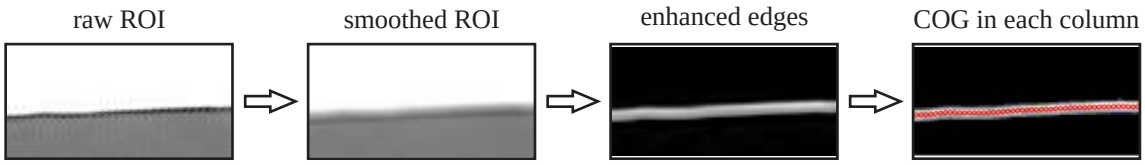

Figure 8.4: Illustration of the evaluation of the ROI1 (cp. Figure 8.3)

8.3.2 Calibration

The complete evaluation procedure as described in the previous Subsection is performed in the image coordinate frame. Thus the obtained CMOD values are in the unit of pixel. A relation between pixel-values and millimeter (mm)-values is found by a simple calibration routine prior to each measurement: The initial width of the gap d_R of the specimen at room temperature T_R is measured with a caliper. Thermal expansion of the gap, when the specimen is heated to the testing temperature T_T, is included:

$$d_T = (1 + \alpha\,(T_T - T_R))\,d_R, \tag{8.1}$$

where d_T is the calculated widths of the gap at testing temperature, $T_T - T_R$ is the temperature difference, and α is the thermal expansion coefficient of the specimen material.

Before the specimen is loaded and deformed, it is also measured with the video extensometer. This yields the initial width of the gap d_V in pixel units.

The scaling factor between pixel- and mm-values is then computed as $s = \frac{d_T}{d_V}$. This factor is used in the subsequent measurement process to transform the CMOD values from pixel- to mm-units.

This simple calibration approach is based on the following assumptions:

1. The camera axis and the front face of the specimen are perpendicular.
2. The distance between the front face of the specimen and the camera remains constant during the testing procedure.
3. There is a linear relation between the pixel- and mm-values (i.e. the optical distortion of the lens is low).

8.3.3 Exemplary Result of a Real Measurement

The plot of an exemplary evaluation result of a CMOD measurement obtained with the developed video extensometer at a temperature of 1100°C is depicted in Figure 8.5. It shows the measured load against the measured CMOD-values. To obtain a smoother curve plot, the CMOD values are filtered with an adjacent averaging filter with a width of 11 values.

The slope of the curve at the beginning of the measurement is evaluated and yields a measure of the elasticity of the material. The approximated line is determined with the same fitting algorithm (see also Section 2.2) that is used to extract the edge positions in the images.

Furthermore, the area under the curve is evaluated and delivers a measure of the fracture energy absorbed by the specimen during the testing procedure. This is an indication of fracture toughness of the material.

8.4 Identification of Error Sources and Accuracy Improvements

During the development of the video extensometer system several sources of measurement errors have been identified. In this section the influence of three particular error sources are evaluated and potentials to increase the accuracy of the measurement results are elaborated.

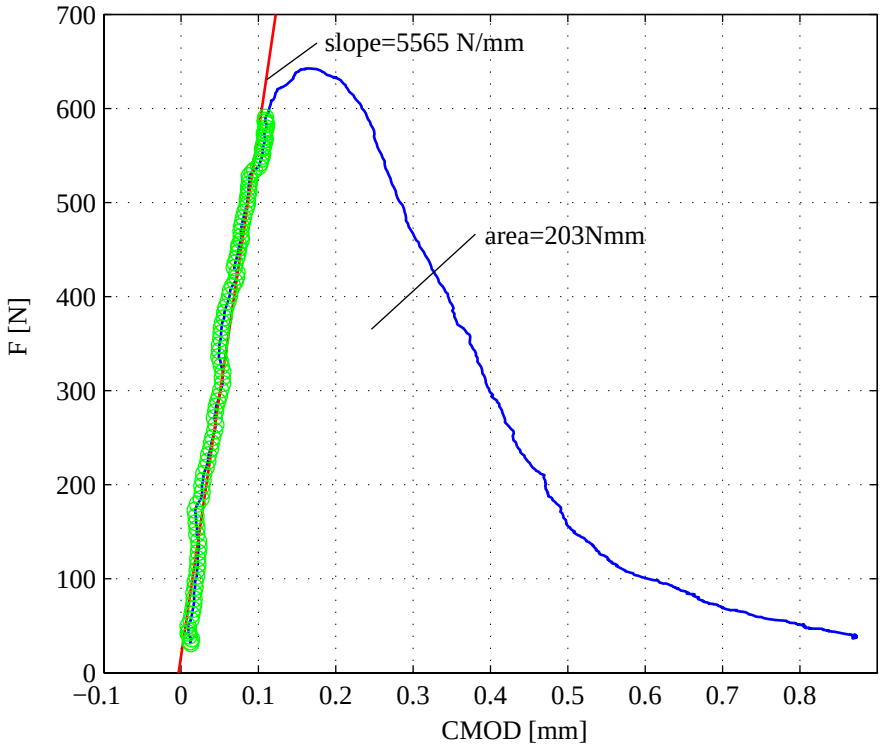

Figure 8.5: Exemplary evaluation result of the video extensometer measurement.

8.4.1 Orientation of the Edges in the Image

A perfect alignment of edges to the orientation of the pixel array of a CCD chip leads to significant errors in determining the position of the edge. The error is caused by the optically inactive area between the pixels. This effect is schematically illustrated in Figure 8.6a. The change of the position of a moving edge is not detected, as long as the edge is imaged in the interpixel area. A continuously moving edge causes a step-like curve of the measured edge position. This eventually influences the progression of the measured CMOD-values. The result of a real measurement with edges aligned to pixel grid orientation is shown in Figure 8.6b.

Two steps are taken to reduce the described effect:

1. The lens is not perfectly focussed on the front face plane of the specimen. Hence, edges are not imaged sharply between the active pixel area, and a change in the position of the edge causes a response in the adjacent pixel.

2. The orientation of the CCD-array is rotated with an angle of approximately 7° relative to the edge orientation. Assuming an ROI of 100 pixel height, a vertical edge is then imaged on 12 pixel columns over its length. Even if some pixel rows in the ROI do not respond to a change of position of the edge in two subsequent images, the change will be detectable in the majority of rows.

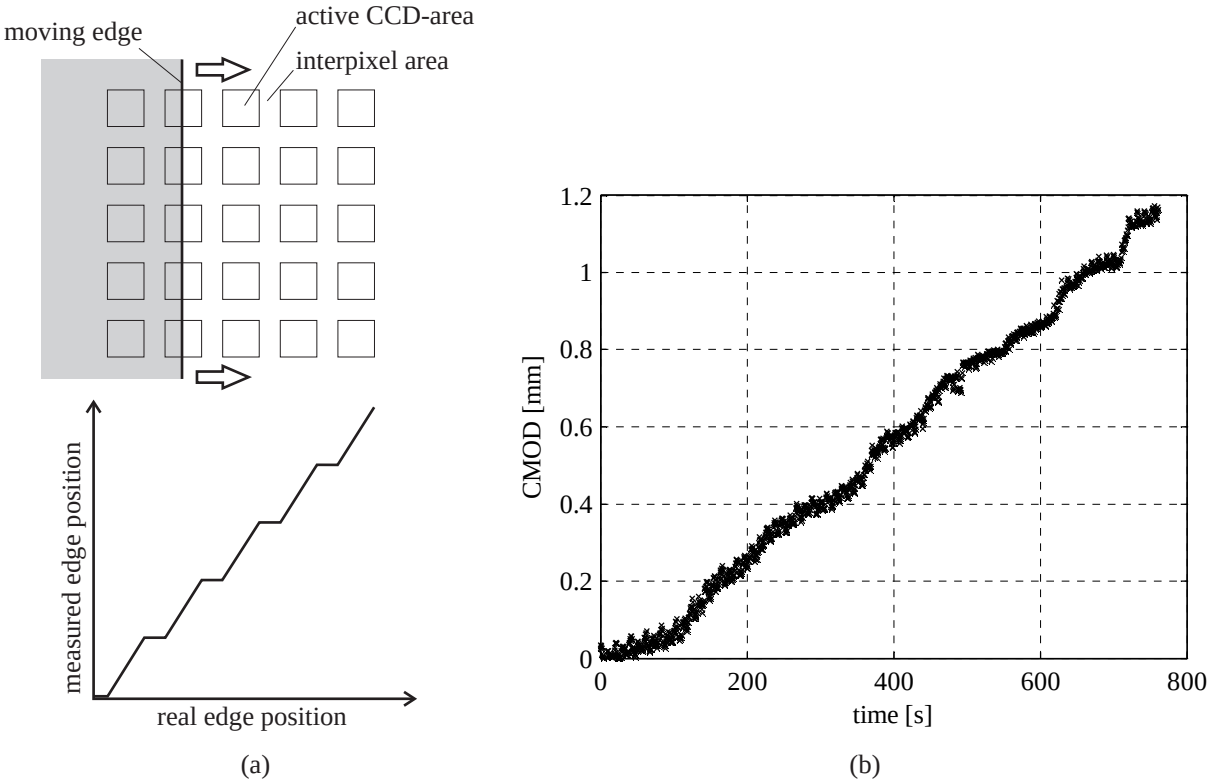

Figure 8.6: a) Schematic drawing to illustrate the effect of edges perfectly aligned to pixel orientation on the evaluation of the edge position. b) Effect of alignment on a real CMOD measurement.

The proposed measures are sufficient to eliminate this error source in the CMOD measurement.

8.4.2 Air Turbulence

A severe effect that increases the uncertainty of the measurement result is turbulence of hot air. The temperature gradient near the oven leads to movement of air at different temperatures and hence to a changing index of refraction. This causes an inaccuracy of the imaged scene with respect to positions of image features such as edges and corners.

The effect of air turbulence cannot be completely isolated from other effects, such as e.g. mechanical vibrations. Investigations of the influence on air turbulence are performed by comparing measurement results obtained at elevated temperature of 1100°C and at room temperature.

Typical CMOD measurements, for high temperature and room temperature measurements, are shown in Figures 8.7a and 8.7b, respectively. The uncertainty of an individual measurement value is determined by fitting a line to a part of the measurement points, which is assumed to follow a straight course. The root-mean-square-distance of the measurement points from the fitted line is evaluated and used as a measure of their uncertainty.

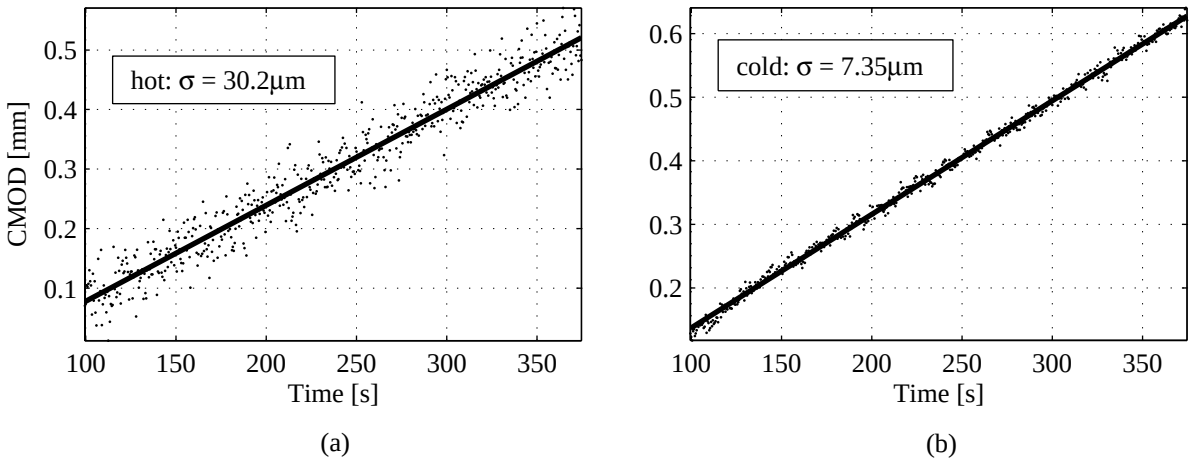

Figure 8.7: Evaluation of measurement noise for measurements at 1100°C (a) and at room temperature (b).

Comparing the evaluated quantities shows, that the uncertainty in measurements at high temperatures is approximately four times as large as at room temperature. A simple solution to reduce the measurement uncertainty at high temperatures has not been found so far. A proposal to reduce or even eliminate the air turbulence is to perform the measurements in an evacuated chamber. However, this would dramatically increase the costs for video extensometer measurements.

8.4.3 Influence of the line fitting routine

The influence of the used line fitting algorithm on the uncertainty of the evaluated CMOD values is investigated utilizing the method of Monte-Carlo simulation (see Section 3.2).

Two fitting approaches are considered:

1. **Independent line fit**: The lines are fitted individually to each point set extracted from the ROIs.
2. **Orthogonal line fit**: It is assumed that the two specimen edges, according to each corner point, are perpendicular to each other, and remain perpendicular during the test procedure. Thus it is valid to fit a pair of orthogonal lines (see Subsection 2.2.3) to each pair of edges.

The configuration of a simulated test data set is similar to the real measurement situation (compare Figures 8.3 and 8.8): Each simulated ROI contains 100 points with an uncorrelated standard deviation of 1 unit in the x- and y-coordinate. The length of each ROI is 10 units and distance from the ROIs to the simulated corner point is 5 units. The distance between the two corner points is 25 units. 10000 noisy data sets are created and each one is processed with both line fitting approaches.

The resulting uncertainty of the intersection points (99%-confidence regions) and the uncertainty of the distance between the intersection points are shown in Figure 8.8. Fitting the lines independently to each ROI point set, leads to an isotropic and uncorrelated noise in the x- and y-coordinates of the intersection points (Figure 8.8a), whereas the uncertainty in the x- and y-coordinates of the intersection points of orthogonally fitted lines is anisotropic and correlated (Figure 8.8b).

The uncertainty of the measured distance between the corner points is $\sigma_{CMODi} = 0.535$, if the corner points are determined by intersection independently fitted lines. The uncertainty is lower ($\sigma_{CMODo} = 0.387$), if the perpendicularity of the edges is considered and orthogonal lines are fitted to the points. This is due to the resulting correlations in the uncertainty of the corner point coordinates and the orientation of the connecting line between the corner points. Thus using the orthogonal line fitting algorithm reduces the uncertainty of the measured distance by approximately 25%.

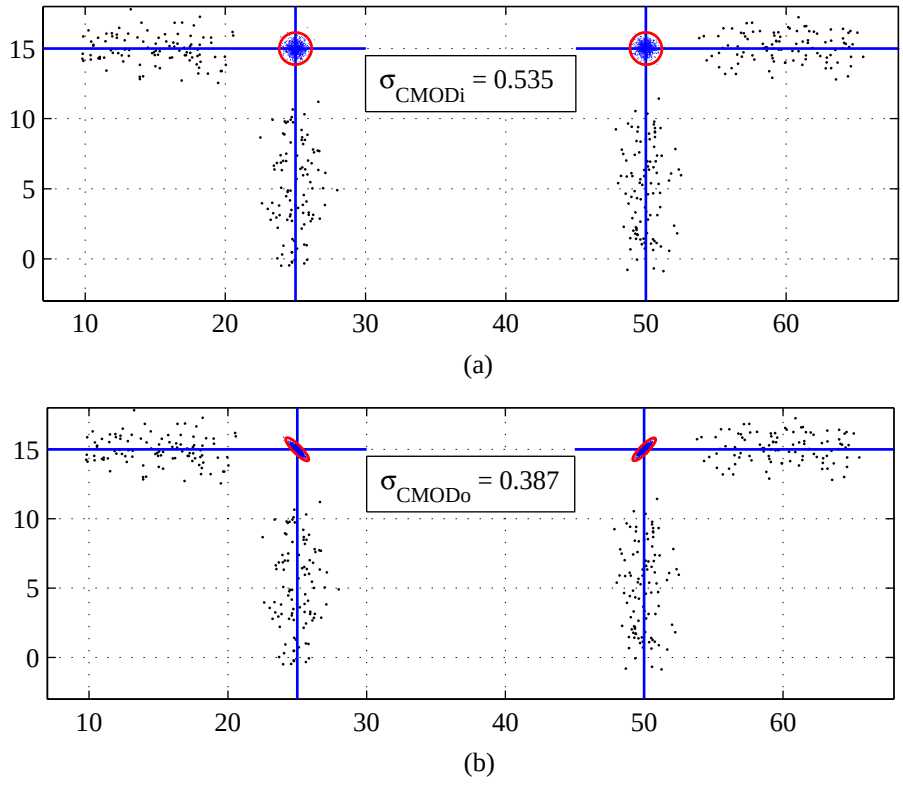

Figure 8.8: Comparison of the Monte-Carlo simulation results (99%-confidence region of intersection points and standard deviation of their distance) obtained with independent line fits (a) and orthogonal line fits (b).

8.5 Summary and Conclusions

The developed video extensometer turns out to be a possible solution to perform CMOD measurements at elevated temperatures of up to 1400°C.

Hardware adaptations concerning the optics are necessary to preserve sufficient contrast in the image and to protect the camera chip from the heat radiation, which is emitted from the hot specimen.

In contrast to the previous approach of estimating the deformation of the specimen from the movement of the testing machine, the direct measurement of the deformation enables the determination of the modulus of elasticity of the material. However, the uncertainty in the measured modulus of elasticity is high. This is mainly caused by the turbulence of hot air in the optical path. A practical solution to reduce this effect has not been found so far.

A perfect alignment of the specimen edges and the orientation of the pixel grid, introduces an error in the measurement of edge positions, due to the inactive interpixel area on the camera chip. This error source can be eliminated through a slight rotation of the camera around the optical axis and through avoiding perfectly sharp imaging of the front face plane of the specimen.

The uncertainty in the CMOD measurements can be reduced by approximately 25%, if the perpendicularity of the specimen edges is considered and the edges are abstracted by orthogonal lines. The intersection of orthogonally fitted lines results in a correlation of the uncertainty in the coordinates of the intersection points. Due to this correlation and the orientation of the connecting line between the intersection points, the uncertainty of the distance is lower compared to the distance of points with uncorrelated uncertainty in the coordinates.

Chapter 9

Conclusion and Future Work

9.1 Conclusions

The following main conclusions, regarding the uncertainty in measurements obtained with metric vision methods, can be drawn from this work:

1. Two methods to estimate the uncertainty in processing results from the input data and its associated uncertainty are presented: An analytical method based on first-order Taylor-series expansion, and the well known Monte-Carlo simulation, i.e. a numerical method. The analytical method turns out to be a valuable alternative to Monte-Carlo simulation, since it is computationally fast and enables the derivation of symbolic expressions of the uncertainty in results.

2. The uncertainty in the fit results, when fitting lines to noisy data points, is analytically investigated. The derived expression of the covariance matrix of the line parameters reveals, that the uncertainty in the individual data points has a linear influence on the covariance of the line parameters. Furthermore, it can be concluded from this expression, that the length of a straight image feature has an inverse cubic influence on the uncertainty of the line fitted to it. The numerical verification of the analytic results shows, that there is only a small error in the analytic estimate of the uncertainty.

3. The sensitivity of four circle fitting algorithms is statistically tested. The input parameters that have been varied during the tests are: The number of input points, the noise in the input points and the subtending arc angle (when considering circular arc data). The standard deviations of the fit results are very similar for all fitting algorithms, whereas the bias in the results is characteristic for each algorithm. As expected, the geometric fitting algorithm performs best for all kind of input data. The partitioned total least squares algorithm is the best algebraic alternative to geometric algorithms, when the data is distributed over the complete circumference of the circle. Considering circular arc data, the simple algebraic fit delivers the best results of the tested algebraic methods.

4. It is important to consider correlations in fit results for the following reasons: (1) A correct estimate of error in results derived from the fit parameters is only obtained, if the correlations are included in the analysis, and (2) the correlations can be advantageously used to derive results exhibiting less uncertainty than the original fit parameters. This is explicitly shown for the newly introduced tangent position of circles, fitted to circular arc data.

5. An acquisition-specific phenomenon is encountered in the video-extensometer application: The alignment of object edges with the orientation of the pixel grid introduces an additional error source. An incorrect position of the edge is measured, due to the inactive inter-pixel area on the camera chip. Rotating the camera with respect to the edge orientation and/or avoiding completely focussed imaging of the edge, may eliminate this error source.

The work on profile measurement based on light sectioning and profile reconstruction leads to the following conclusions:

1. The new approach turns out to be a possible way to measure cross-sectional profiles of rotating objects. It features a compact measurement instrumentation and a simple setup compared to existing profile measurement systems.

2. Differential geometry is successfully applied to reconstruct the shape of the profile from measured radii of curvature. The derived reconstruction algorithms are compatible with the specific setup to drive the rotation of the measurement object and the resulting parameterization of the positions, where the radii of curvature are sampled.

3. Curves of constant width, which are encountered as an unwanted result in several technical processes, such as drilling or rolling, can be detected. These curves exhibit a constant diameter over the complete circumference, however, they may clearly deviate from an ideal circle. A least-squares approximation of curves of constant width can be made using Elliptical Fourier descriptors.

9.2 Future work

Metric vision is a broad and open field of research and development with respect to both mathematical evaluation techniques and possible applications. Two distinct topics within this field have been thoroughly discussed in this work. However there still remain open issues related to these topics:

1. The uncertainty of fitting lines and circles, which are the most important objects in metric vision, has been investigated. However, analyzes of the errors associated with fitting of other geometric objects (such as general conics, polynomials, splines, etc.) are necessary.

2. Another open issue is analyzing the second branch of errors in the measurement chain of metric vision systems: The error propagation in the calibration procedure, the uncertainty of the calibration information and its influence on the uncertainty in the measurement result. In this context, the uncertainties associated with the determination of homographies are of particular importance, since the error structure in the rectified data is influenced by uncertainty of the original input data and by uncertainty of the used homography.

3. Considering the proposed method of curvature measurement with a light sectioning measurement head, it is clearly not possible to measure the radius of curvature for an infinitely short arc length. Actually, all circle fitting algorithms are more stable for large subtending angles, typically 90° to 120°; this can be regarded as a measurement window and is equivalent to convoluting the measurement result with a rectangular window in the frequency domain. This leads to a "$\sin(x)/x$"-distortion of the spectrum. Future work on the profile measurement system will concentrate on correcting this effect using pre-distortion, i.e. de-convoluting the signal with the known measurement window.

Author's Publications

Journal Publications

1. P. Schalk, E. Fauster, P. O'Leary, and M. Weiss, Framework for automatic quality control in industrial environments using distributed image processing, *Journal of Electronic Imaging*, 13(3):504-514, 2004.

2. P. Schalk, R. Ofner, and P. O'Leary, Pipe eccentricity measurement using laser triangulation, *Image and Vision Computing, Special Issue on Computer Vision Applications*, Article in Press.
[Online] Available October 6, 2006: http://dx.doi.org/10.1016/j.imavis.2006.04.021

3. P. Schalk, P. O'Leary, R. Ofner, and A. Gfrerrer, Measuring and analyzing cross-sectional profiles of rotating objects using light sectioning, submitted to the *IEEE Transactions on Instrumentation and Measurement*, 2007.

Conference Proceedings

1. P. Schalk, E. Fauster, and P. O'Leary, High-temperature video-extensometry for material testing of refractories, In *IS&T / SPIE 17th Annual Symposium on Electronic Imaging*, pages 129-139, San Jose, California, USA, 2005.

2. P. Schalk, M. Weiss, E. Fauster, and P. O'Leary, Videoextensometrie als Messverfahren zur Prüfung glühender Werkstoffe, In *XXIII. Verformungskundliches Kolloquium*, pages 209-227, Donnersbach, Austria, 2004.

3. P. O'Leary, P. Schalk, R. Ofner, and A. Gfrerrer, Instrumentation and Analysis-Methods for the Measurement of Profiles using Light Sectioning, In *Proceedings of the 23rd IEEE Instrumentation and Measurement Technology Conference*, pages 1108-1113, Sorrento, Italy, April 2006.

4. P. O'Leary, P. Schalk, R. Ofner, and A. Gfrerrer, Measuring the Straightness and Cross-Sections of Rolled Seamless-Pipes, In *XXV. Verformungskundliches Kolloquium*, pages 37-50, Donnersbach, Austria, 2006.

5. E. Fauster, P. Schalk, and P. O'Leary, Evaluation and calibration methods for the application of a video-extensometer to tensile testing of polymer materials, In *IS&T / SPIE 17th Annual Symposium on Electronic Imaging*, pages 187-198, San Jose, California, 2005.

6. M. Weiss, A. Schiller, P. O'Leary , E. Fauster, P. Schalk, Development of a distributed vision system for industrial conditions, In *6th Int. Conf. on Quality Control by Artificial Vision*, pages 91-102, Gatlingburg, USA, 2003.

7. E. Fauster, P. Schalk, M. Tratnig, Calibration method for light sectioning measurement systems, In *2nd WSEAS Int. Conf. on Signal, Speech and Image Processing*, pages 6-11, Skiathos, Greece, 2002.

Oral Presentations

1. Measuring the Straightness and Cross-Sections of Rolled Seamless-Pipes, *XXV. Verformungskundliches Kolloquium*, Donnersbach, Austria, 2006.

2. Automatic measurement of pipe eccentricity using digital image processing, *IS&T / SPIE 17th Annual Symposium on Electronic Imaging*, San Jose, California, USA, 2005.

3. High-temperature video-extensometry for material testing of refractories, *IS&T / SPIE 17th Annual Symposium on Electronic Imaging*, San Jose, California, USA, 2005.

4. Videoextensometrie als Messverfahren zur Prüfung glühender Werkstoffe, *XXIII. Verformungskundliches Kolloquium*, Donnersbach, Austria, 2004.

Technical Papers

1. M. Weiss, A. Schiller, P. O'Leary , E. Fauster, P. Schalk, A distributed vision system for quality control at a wire rolling mill, *Society of Manufacturing Engineers*, paper number MV03-250, USA, 2003.

References

[1] Sick IVP AB, Company. IVC 3D, 2006. [Online]. Available: http://www.sick.de/ivp/products/overview/ivc/en.html.

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

[3] W. Alt. *Nichtlineare Optimierung*. Vieweg Verlag, Braunschweig, 2002.

[4] M. Brandner and T. Thurner. Uncertainty in optical measurement applications: A case study. *IEEE Trans. Instrum. Meas.*, 55(3):713–720, 2006.

[5] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[6] S.-W. Cheng, S. Funke, M. Golin, P. Kumar, S.-H. Poon, and E. Ramos. Curve reconstruction form noisy samples. *Computational Geometry*, 31:63–100, 2005.

[7] N. Chernov and C. Lesort. Least squares fitting of circles and lines. Technical report, Dept. Mathematics, University of Alabama at Birmingham, AL, USA, Jan 1, 2003.

[8] N. Chernov and C. Lesort. Statistical efficiency of curve fitting algorithms. Technical report, Dept. Mathematics, University of Alabama at Birmingham, AL, USA, Mar 19, 2003.

[9] N. Chernov and C. Lesort. Least squares fitting of circles. *Journal of Mathematical Imaging and Vision*, 23:229–225, 2005.

[10] N. Chernov, C. Lesort, and N. Simányi. On the complexity of curve fitting algorithms. Technical report, Dept. Mathematics, University of Alabama at Birmingham, AL, USA, Aug 15, 2003.

[11] W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. Revisiting hartleys normalized eight-point algorithm. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(9):1172–1177, 2003.

[12] J. C. Clarke. Modelling uncertainty: A primer. Technical report 2161/98, Dept. Engineering Science, University of Oxford, UK, 1998.

[13] A. Criminisi, I. Reid, and A. Zisserman. A plane measuring device. *Image and Vision Computing*, 17(8):625–634, 1999.

[14] G. Csurka, C. Zeller, Z. Zhang, and O. D. Faugeras. Characterizing the uncertainty of the fundamental matrix. *Computer Vision and Image Understanding*, 68(1):18–36, 1997.

[15] C. Demant, B. Streicher-Abel, and P. Waszkewitz. *Industrielle Bildverarbeitung*. Springer, Berlin Heidelberg, 1998.

[16] T. K. Dey, K. Mehlhorn, and E. A. Ramos. Curve reconstruction: Connecting dots with good reason. In *Symposium on Computational Geometry*, pages 197–206, Miami Beach, FL, USA, 1999. ACM Press.

[17] J. Eichler and H. J. Eichler. *Laser. Bauformen, Strahlführung, Anwendungen*. Springer, Berlin Heidelberg, 3rd edition, 1998.

[18] O. D. Faugeras. *Three-Dimensional Computer Vision - A Geometric Viewpoint*. Artificial Intelligence. The MIT Press, Cambridge, MA, USA, 1993.

[19] E. Fauster, P. Schalk, and P. O'Leary. Evaluation and calibration methods for applications of a video-extensometer to tensile testing of polymer materials. In *IS&T / SPIE 17th Annual Symposium on Electronic Imaging*, pages 187–198, San Jose, CA, USA, 2005. SPIE.

[20] E. Fauster, P. Schalk, and M. Tratnig. Calibration method for light sectioning measurement systems. In *2nd WSEAS Int. Conf. on Signal, Speech and Image Processing (ICOSSIP 2002)*, pages 6–11, Skiathos, Greece, 2002. WSEAS Press.

[21] R. S. Figliola and D. E. Beasley. *Theory and Design for Mechanical Measurements*. John Wiley & Sons, New York, 3rd edition, 2000.

[22] J. Gallier. *Geometric Methods and Applications*. Springer, New York, 2001.

[23] W. Gander, G. H. Golub, and R. Strebel. Fitting of circles and ellipses least squares solution. Technical report, ETH Zürich, Dept. Informatik, November 1994. [Online]. Available: ftp://ftp.inf.ethz.ch/doc/tech-reports/2xx/217.ps.

[24] W. Gander, G. H. Golub, and R. Strebel. Least-squares fitting of circles and ellipses. *BIT*, 34:558–578, 1994.

[25] A. Gfrerrer. *Analytische Projektive Geometrie*. Lecture notes, Graz University of Technology, Austria, 2003.

[26] J. Giesen. *Curve reconstruction*. Dissertation, ETH Zürich, Switzerland, 2000.

[27] Hot Vision Research GmbH, Company. Hot profile: Laserprofilsensor, 2006. [Online]. Available: http://www.hotvision.at/hotprofile.html.

[28] LAP GmbH, Company. Vorrichtung zur Messung der Unrundheit von länglichen Werkstücken, 2003. German Patent, Number: DE 200 23 127 U 1.

[29] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, 3rd edition, 1996.

[30] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley, Reading, MA, 1992.

[31] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau. Demosaicking: Color filter array interpolation. *IEEE Signal Processing Magazine*, 22(1):44–54, 2005.

[32] R. Halíř and J. Flusser. Numerically stable direct least squares fitting of ellipses. In V. Skala, editor, *6th Int. Conf. in Central Europe on Computer Graphics, Visualization and Interactive Digital Media*, pages 125–132, Plzen, Czech Republic, 1998.

[33] M. Harker and P. O'Leary. Computation of homographies. In W. F. Clocksin, A. F. Fitzgibbon, and P. H. S. Torr, editors, *British Machine Vision Conference*, volume 1, pages 310–319, Oxford, UK, 2005.

[34] M. Harker, P. O'Leary, and P. Zsombor-Murray. Incremental matrix orthogonalization with an application to curve fitting. In C.A. Bouman, editor, *Computational Imaging III*, volume 5674, pages 354–361, San Jose, USA, 2005. SPIE.

[35] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, 2nd edition, 2003.

[36] G. C. Holst. *CCD Arrays, Cameras and Displays*. SPIE Optical Engineering Press, Bellingham, WA, 1996.

[37] ISO. *Guide to the Expression of Uncertainty in Measurements*. International Organization for Standardization, Geneva, Switzerland, 1st edition, 1993.

[38] I. Kasa. A circle fitting procedure and its error analysis. *IEEE Trans. Instrum. Meas.*, 25:8–14, 1976.

[39] F. Klein. *Elementarmathematik vom höheren Standpunkte aus II*. Springer, Berlin, 1968 reprint edition, 1925.

[40] N. Koller, R. Ofner, P. O'Leary, and E. Fauster. Optical servoing for industrial surface machining. In *IS&T / SPIE 18th Annual Symposium on Electronic Imaging*, pages 1–11, San Jose, CA, USA, 2006. SPIE.

[41] A. A. Ligun and A. Shumeiko. Description of convex curves. *Ukrainian Mathematical Journal*, 52(7):1040–1057, 2000.

[42] R. S. Lu, Y. F. Li, and Q. Yu. On-line measurement of the straightness of seamless steel pipes using machine vision technique. *Sensors and Actuators A*, 94:95–101, 2001.

[43] H. M. Merklinger. The Scheimpflug's Patent. *Photo Techniques*, 17(6):56, 1996.

[44] Y. Nievergelt. Hyperspheres and hyperplanes fitted seamlessly by algebraic constrained total least-sqares. *Linear algebra and its applications*, 331:43–59, 2001.

[45] R. Ofner. *Three-Dimensional Measurement via the Light-Sectioning Method*. Doctoral thesis, University of Leoben, Austria, 2000.

[46] P. O'Leary, M. Harker, and P. Zsombor-Murray. Direct and least square fitting of coupled geometric objects for metric vision. *IEE Proc.- Vis. Image Signal Process.*, 152(6):687–694, 2005.

[47] A. V. Oppenheim and R.W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall, Engelwood Cliffs, 1989.

[48] K. N. Plataniotis and A. N. Venetsanopoulos. *Color Image Processing and Applications*. Digitital Signal Processing. Springer, Berlin Heidelberg, 2000.

[49] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, 2nd edition, 1995.

[50] A. Pressley. *Elementary Differential Geometry*. Springer, London, 2001.

[51] T. Rabbani and F. v. d. Heuvel. Efficient Hough Transform for automatic detection of cylinders in point clouds. In *ISPRS Laser scanning 2005*, Enschede, the Netherlands, 2005.

[52] Joanneum Research, Company. 2D-Profilvermessung, 2000. [Online]. Available: http://dib.joanneum.ac.at/bv_main_d.html.

[53] J. C. Russ. *The Image Processing Handbook*. CRC Press, Boca Raton, FL, USA, 2nd edition, 1995.

[54] L. Sachs. *Angewandte Statistik*. Springer, Berlin, 8th edition, 1997.

[55] C. M. Sallinger. *Inspection and repair of refractories using a telerobot guided by panoramic optical servoing*. Doctoral thesis, University of Leoben, Austria, 2004.

[56] P. Schalk, E. Fauster, and P. O'Leary. High-temperature video-extensometry for material testing of refractories. In *IS&T / SPIE 17th Annual Symposium on Electronic Imaging*, pages 129–139, San Jose, CA, USA, 2005. SPIE.

[57] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision*. Brooks/Cloe Publishing Company, 2nd edition, 1999.

[58] K. Strubecker. *Differentialgeometrie. 1. Kurventheorie der Ebene und des Raumes*. de Gruyter, Berlin, 1964.

[59] S. Sy. General helicis and other topics in the differential geometry of curves. Masters thesis, Michigan Technological University, Houghton, MI, 2001.

[60] M. Tratnig. *Calibration and Registration Approaches for Light-Sectioning Setups Featuring Small Fields of View*. Doctoral thesis, University of Leoben, Austria, 2005.

[61] E. W. Weisstein. Curvature. From M*athWorld*–A Wolfram Web Resource, 1999. [Online]. Available: http://mathworld.wolfram.com/Curvature.html.

[62] Z. Zhang. Parameter estimation techniques: A tutorial with application to conic fitting. Rapport de Recherche 2676, Institut National de Recherche en Informatique et en Automatique (INRIA), Sophia Antipolis, France, October 1995.