

Web-Based Reference Software For Characterisation Of Surface Roughness

Dem Fachbereich Maschinenbau und Verfahrenstechnik
der technischen Universität Kaiserslautern
zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

vorgelegte

Dissertation

von

Frau

M.Sc. Bini Alapurath George

aus Bangalore

Datum der wissenschaftlichen Aussprache: 28.09.2017

Betreuer der Dissertation: Prof. Dr.-Ing. Jörg Seewig

Acknowledgment

Firstly, I would like to express my gratitude and sincere thanks to my doctoral advisor Prof. Dr.-Ing. Jörg Seewig for his advice and guidance during my research and for giving me the opportunity to pursue my PhD.

I would like to thank my colleagues from the Institute of Measurement and Sensor Technology for the support and discussions during my research.

I wish to extend my thanks to Mr. Timo Wiegel for supporting me with my research, as part of his internship program in the Institute of Measurement and Sensor Technology, T.U. Kaiserslautern during summer 2012.

I am thankful to my husband, Dr. Mahesh Poolakkaparambil for his support and encouragement. A special thanks to my mother, Valsala George for her love and prayers. I am indebted to them for their help and support.

I take this opportunity to thank all my friends for the memorable time during my stay in Kaiserslautern.

Finally, I would like to assert that all the contents and pictures in this thesis are from the knowledge of literatures, publications and paper, which are mentioned in the references and naturally my knowledge from the study and research in this work.

Kaiserslautern, _____ Signature: _____

Abstract

This research explores the development of web based reference software for characterisation of surface roughness for two-dimensional surface data. The aim of the thesis is to develop a reference software which makes the evaluation methods for verification of surface characteristics easier for clients. The Institute of Measurement and Sensor Technology (MTS) is working in collaboration with the Physikalisch Technische Bundesanstalt-Braunschweig, the German National Metrology Institute on the implementation of appropriate algorithms. The algorithms used in this software are based on International ISO standards. The ISO standards used in the software are mentioned in the upcoming chapters. Most software used in industrial measuring instruments may give variations in the parameters calculated due to numerical changes in calculation. Such variations can be verified using the proposed reference software.

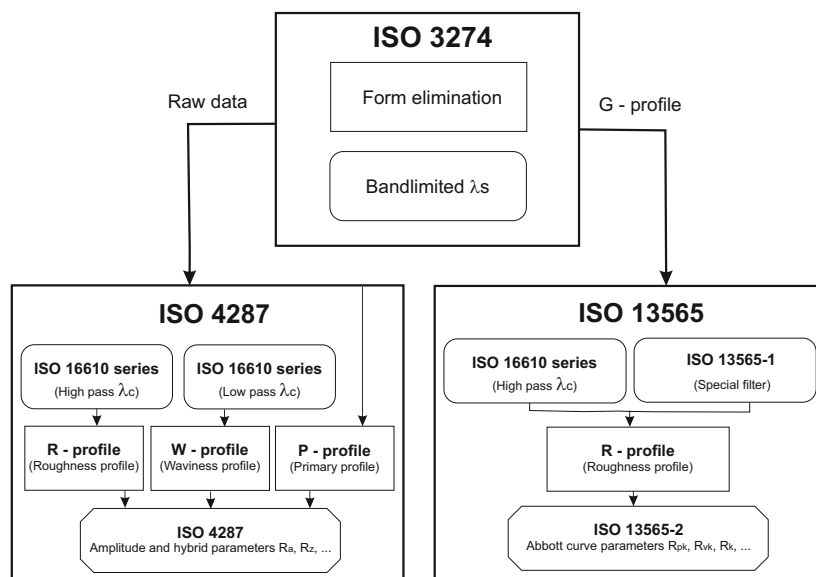


Figure 0.1: Work flow of the software

Surface texture is a result of the manufacturing process, it affects the functionality of the part with its environment. Turning, grinding, milling and other factors like cutting tool, speed, feeds and vibrations give rise to surface irregularities. These irregularities are termed as "Roughness" and are due to the movement of the cutting tool. On the other hand "Waviness" is the result of vibrations, that are caused between the cutting tool and the workpiece during machining process. "Form" are errors due to lack of straightness and flatness on the surface. The methods opted to measure surface texture strongly depends upon the surface requirement. The cutoff wavelength

and filter technique used for analysis allows the surface texture to be separated into three components: roughness, waviness and form. Each of these component is the result of the manufacturing process and will in return affect the functionality of the surface. The cutoff wavelength and filter method must be chosen according to the surface type. Inappropriate value and method can give rise to error prone results. Clients with no experience in filtration must use the default cutoff value (0.8 mm) and method (*Gaussian filter*) as defined in DIN EN ISO 4288 [DIN98d]. Before applying a Gaussian filter, it is better to remove the form from the surface profile using form eliminator (*F-operator*). It is described briefly in the upcoming chapter. And parameters are values that describe its characteristics. The most important parameter describing surface integrity is surface roughness, it strongly affects the performance of a workpiece as its size has been reduced to micrometers and nanometers these days. Therefore measuring surface roughness is vital for quality control of a workpiece. Such surfaces have higher friction. In general, the surface texture must be a compromise between sufficient roughness for wear and sufficient smoothness for expected life span. There are several parameters specified in the International standards, out of which R_a or average roughness is the most widely measured parameter. R_z is also widely used parameter in preference to R_a . The overall work flow of the software is shown in chapter 5.

The evaluation of surface roughness is carried out in four major steps: data capture, data align, data filtering and parameter calculation. This work walks through the flow of the program shown in figure 0.1, explaining how surface profiles were evaluated by preprocessing steps called fitting and filtering. This process is followed by parameter evaluation according to DIN EN ISO 4287 [DIN09] and DIN EN ISO 13565-2 [DIN98c] standards to extract important information from the profile to characterise surface roughness.

Kurzbeschreibung der Arbeit in Deutsch

Die Bewertung der Oberflächenrauheit an technischen Funktionsflächen und der Vergleich von Rauheitskennwerten mit Toleranzvorgaben ist ein fester Bestandteil der industriellen Qualitätssicherung. Die Auswertung der Rauheit erfolgt dabei in vier Schritten: Messdaten erfassen, Messdaten ausrichten, Messdaten filtern, Anwendung von Kenngrößenoperatoren. Insbesondere die Normen zu Filteralgorithmen und Kenngrößen werden derzeit in der ISO TC 213 WG 15 und WG 16 neu überarbeitet. Die flächenhafte Auswertung der Oberflächenrauheit ist in der neuen Normenreihe ISO 25178 verankert.

Entscheidend für die Vereinheitlichung der Rauheitsauswertung sind sogenannte Referenzalgorithmen. Der Lehrstuhl für Messtechnik und Sensorik arbeitet kontinuierlich mit der Physikalisch Technischen Bundesanstalt in Braunschweig an der Implementierung entsprechender Algorithmen. Für die praxisorientierte Verwendung der Referenzalgorithmen ist es entscheidend, dass dem wissenschaftlichen und industriellen Umfeld eine allgemein zugängliche Plattform zur Verfügung gestellt wird.

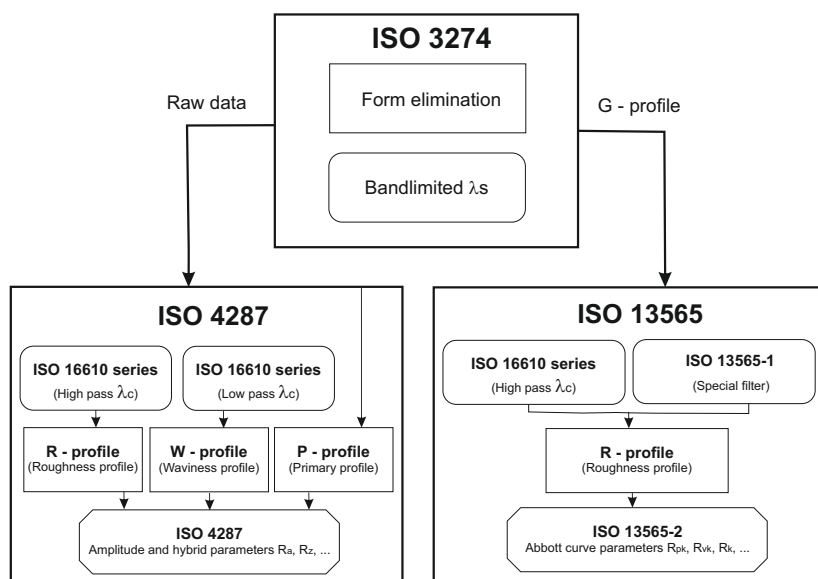


Figure 0.2: Blockdiagramm der Arbeit

Contents

1	Introduction	1
1.1	Surface profile measurement	2
1.1.1	Stylus instruments	2
1.1.2	Optical instruments	2
1.2	Surface profile characterisation	3
1.2.1	Form removal	5
1.2.2	Filtering	5
1.2.3	Parameter calculation	6
1.3	Organisation	6
2	State of the art	8
2.1	Measurement standard	8
2.1.1	Software review	10
2.2	Preprocessing	11
2.2.1	Geometrical fitting of straight line	12
2.2.2	Geometrical fitting of circle	18
2.3	Filter application	21
2.3.1	2RC filter	22
2.3.2	Gaussian filter	23
2.3.3	Special filter	24
2.3.4	Spline filter	25
2.3.5	Robust Gaussian regression filter	27
2.3.6	Selection of filter	29
2.4	Parameter analysis	29
2.4.1	Primary surface profile	30
2.4.2	Amplitude parameters	31
2.4.3	Spacing parameters	35
2.4.4	Hybrid parameters	35
2.4.5	Abbott-Firestone curve parameters	37
3	Corrigendum of the ISO filter	40
3.1	Gaussian profile filter	40
3.1.1	Gaussian profile filter for an open profile - Planar surface	40
3.1.2	Gaussian profile filter for a closed profile - Cylindrical surface	42
3.1.3	Periodic Gaussian profile filter for closed profiles	44
3.2	Spline filter	45

3.3	Robust regression filter	48
3.3.1	Filter equations	48
3.4	Sorting algorithm	52
4	Measurement uncertainty	55
4.1	Uncertainty	55
4.2	Uncertainty of roughness parameters	57
5	Reference software	59
5.1	Patterns and frameworks	59
5.1.1	Web-applications	59
5.1.2	Programming languages	60
5.1.3	Software patterns and frameworks	61
5.2	Implementation	63
5.2.1	Client implementation	64
5.2.2	Server implementation	70
5.3	Java and C++ interoperation	76
5.3.1	Integration	76
6	Soft gauges	82
6.1	Comparison with existing software	83
6.1.1	Primary profile parameters	84
6.1.2	Waviness profile parameters	85
6.1.3	Roughness profile parameters	86
7	Summary and future work	89
7.1	Summary and conclusion	89
A	Glossary	98
B	Appendix	100
B.1	M-script for total least square fit	100
B.2	M-script for circular fit	101
B.3	Play framework deployment settings	105
B.4	Java Native Interface (JNI)	106

1 Introduction

The main aim of the thesis is to develop a web based reference software for characterisation of surface roughness for two dimensional dataset. The surface of a workpiece often appears to be smooth to the human eye, however a clear examination shows chattermarks on the surface which are caused by the machining tools. These chattermarks often affect the performance of the workpiece. To avoid the irregularities caused by the machining tools, it is necessary to first measure the surface. As the proposed software only deals with two dimensional data, surface is referred as profiles. Surface profiles are generally acquired using industrial measuring instruments. These instruments capture data from the surface depending on its size and shape. The datapoints measured are further processed by the reference software to obtain parameters that give a summary of information contained in the surface.

Before the data is processed, it is important to make sure that the form is eliminated from the surface profile. They can be removed by a preprocessing step called fitting. Fitting is followed by a process called filtering to extract useful surface information from the profile. It is generally based on DIN EN ISO 16610 series. Special filters like DIN EN ISO 13565-1 is also used in the process. The idea behind measuring technical surfaces is to describe them with its characteristic parameters. Parameters are calculated from the filtered profile based on DIN EN ISO 4287 and DIN EN ISO 13565-2 standards. DIN EN ISO 4287 specifies the terms and definitions of parameters that are currently used in industry for stylus methods. The parameters calculated give informations about the functional properties of the surface, hence its important to make sure that the informations are not lost during the analysis. By undergoing the process according to ISO standards, it helps the clients to compare their results with other similar measurements taken using different measuring instruments.

The steps carried out in the web application are:

1. Uploading data files,
2. Form removal (fitting),
3. Filtering profiles according to ISO standard,
4. Calculating the standard surface profile parameters according to ISO standard,
5. Exporting the results.

The proposed software is tested and compared using soft gauges. Evaluation of surface profile parameters using soft gauges is compared with the software to check for its robustness and is tabulated in chapter 6.

1.1 Surface profile measurement

Surface measurement includes a range of measurement technologies. It is one of the main features that can affect the analysis process. There are several instruments that are available in industry today to measure surface profiles. This section highlights the most commonly used measuring instruments like stylus instruments and optical instruments. Both these instruments have its own principle and are used depending on the surface to be measured.

1.1.1 Stylus instruments

Stylus instruments are based on the principle of running a probe across a surface to detect variations in height as a function of distance $z(x)$. The part of the stylus in contact with the surface to be measured is a stylus tip. Its traversing direction is generally perpendicular to the direction of lay i.e. in the direction of the surface pattern. A transducer is used to convert its vertical movement into an electrical signal. The traced profile consists of form deviations, waviness and roughness components (DIN EN ISO 3274). Stylus instruments tend to introduce some errors into roughness measurement due to factors like stylus tip radius (ranging from $2\mu\text{m}$ to $10\mu\text{m}$), stylus load (0.75 mN) and lateral deflection by asperities. The effect of stylus is notable when the curvature of the peaks and valleys increases i.e. when they are steeper thereby not allowing the stylus to pass into it. Hence certain parameters will be affected by it. The effect of stylus load can leave a trace on soft surfaces giving rise to surface roughness, it can also cause damage to the surface being measured. Measurement errors can also arise when the stylus loses its contact with the surface as a result of high speed [Tho82]. The DIN EN ISO 4287 standard specifies the terms, definitions and parameters used in the evaluation of surface profile by stylus instruments. Stylus instruments are so far the widely used instruments for measuring surfaces, but the time taken by a stylus instrument to measure a large surface is considered to be a major drawback in production area these days, hence optical instruments take advantage over the stylus instruments.

1.1.2 Optical instruments

Unlike stylus instruments, measurements by optical instruments does not involve direct contact with surfaces, instead a beam of electromagnetic radiation is projected onto the surface from a point light source. Measurement is based on the reflection of light from the surface. The light can be reflected specularly, diffusely or both. If the surface is rough, an undulating pattern will be observed and a smooth surface creates specular reflection. Thus, the amount of specular and diffuse reflection can be

used to determine surface roughness [Tho82]. There are several optical measurements technologies that have come into existence. Traceability in such instrument is achieved through optical methods like confocal microscopy, white light interferometry, focus variation and chromatic confocal sensor. The important properties of microscopes for measuring surface topography are vertical resolution, lateral resolution, slope range, speed and the cost.

Confocal microscopy can measure height variation $z(x,y)$ with vertical resolution and lateral resolution depending on the magnification and the objective directly independent of illumination and shadowing effects. But the signal decreases and becomes unreliable for high surface slopes leading to outliers. White light interferometry is comparable to confocal microscopy, but has issues with steep sloped surfaces due to complex sensor signal. On the other hand, one promising technique is the chromatic confocal microscopy. It is a measuring technique comprising of a chromatic confocal probe and a lateral scanning system. The probe senses each point of the workpiece and extracts its height and light intensity. The lateral scanning system allows the measurement of a line profile, areal surface, roundness profile and form. As a single point measuring technique, the process is similar to that of a stylus profilometer and as a confocal probe, the chromatic confocal microscopy is insensitive to ambient light and scattered reflection from the workpiece. On comparison with confocal microscopy, chromatic confocal microscopy does not require any vertical scanning to capture the surface height. This section only highlights the basic principles of optical methods. For more information about optical traceability, refer [JS08], [Jor08] and [Jor04].

Optical instruments can cover a wide range of area. Its speed, advantage of depth section imaging and high resolution makes it well suited in the field of biology. They are also applicable to surface profilometry to measure surface height, its non contact type measurement prevents the surface from scratches. Despite their advantages, there are few demerits faced. Firstly, optical instruments require a large amount of light to be reflected from the surface. Therefore low reflective surface measurement cannot be reliable. Secondly when there are steep slopes on the surface, light cannot pass through it easily thereby giving rise to non measured points called outliers on the surface data.

1.2 Surface profile characterisation

Surface measurements are taken using stylus or optical instruments depending on the surface type. These measurements play a vital role in industry, as they provide useful information about the surface measured. To obtain the desired information it is

necessary to undergo three main processes: fitting, filtering and parameterisation. Each of these analysis methods are described in detail in chapter 2 section 2.2, 2.3 and 2.4. It is necessary to ensure that sufficient datapoints are measured so that even if the outliers (Non Measured Points) are knocked out, enough datapoints are available for the analysis. Too many outliers can affect the quality of fit. Data normalisation generally reduces noise in the dataset and enables direct comparisons between the raw data values that may have been measured across different ranges. It is important to assure that the profile analysis confirms to ISO standards. Figure 1.1 shows the overall view of the application and subsections explain in brief the procedures followed in the application.

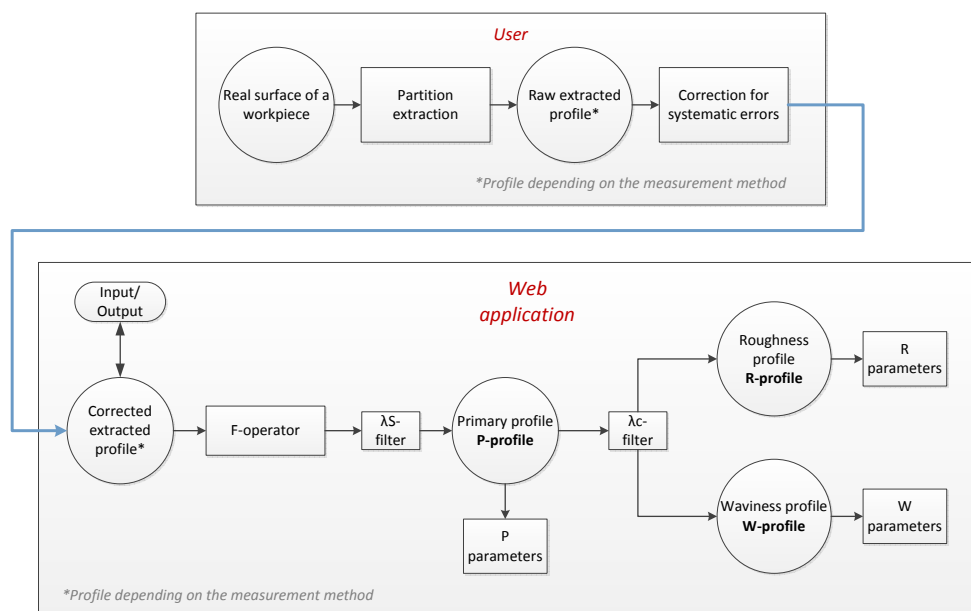


Figure 1.1: Overall view of the analysis

It is important to explain some of the general terms and definitions of surface measurement to understand the basic language of surface metrology. To begin with, the total length of the profile measured is called the traverse length, which includes the start area, evaluation length and the stop area. The evaluation length (L_n) is five times the sampling length (L_r) and the sampling length equals the cutoff wavelength (L_c) of the filter used in the analysis. In some of the industrial measuring instruments and in the proposed reference software the start and stop area are set to L_c . The default number of sampling lengths for the primary profile is one and the

default number of sampling lengths for roughness and waviness profile is five (DIN EN ISO 4287). There are two types of profile filters used by industrial instruments for measuring surface profiles, the ($\lambda_s = L_s$) filter and the ($\lambda_c = L_c$) filter. L_s defines the intersection between the roughness profile and shorter wavelength components whereas L_c filter defines the intersection between the roughness and waviness profile in a surface. Profile parameters are calculated based on sampling length or the evaluation length. If evaluated on sampling length, an average of those five sampling lengths is considered.

1.2.1 Form removal

"Fitting" referred as F-operator in the analysis is used to remove form from the surface. In other words, a straight line is removed from the dataset to avoid the influence of the workpiece being tilted with respect to the mounting table. The nominal form from the surface profile has to be eliminated before L_s filtering, hence this step in the process is mandatory. The nominal form is eliminated by a compensation element. Once the form has been removed, the surface profile consists of three main components. These are the primary profile, waviness profile and roughness profile. The instruments used these days for surface measurement have options for form removal of the surface. Even after form removal, some fractions of form remain in the surface profile which has to be taken into account. There are several algorithms available to remove form from the surface profile, the most commonly used is the least square straight line fit. Recent ISO standard use the total least square fit as default fitting method for surface characterisation. The form removal methods used in the software and its mathematical representations are explained in detail in chapter 2 section 2.2. In this thesis, least square method is denoted as (LS) and total least square method as (TLS).

1.2.2 Filtering

Filters are basically used to suppress unwanted data components into different wavelength bands. They are widely used in industry starting with electrical filters to digital filters. From figure 1.1, its clear that the primary profile (P-profile) is filtered to separate the waviness profile and roughness profile. A Gaussian filter is used as default filter in the application and is according to ISO standard. The definition of Gaussian filter is given in ISO 16610-21. They are based on moving a Gaussian weighted average through the primary profile, thereby resulting in the waviness profile (W-profile). The roughness profile (R-profile) is the difference between the waviness profile and the primary profile. Filtering is mainly based on the cutoff wavelength L_c . Earlier investigations led to the standard sampling length of 0.8mm.

For process like grinding, determination of sampling length is difficult as it is hard to predict the number of cycles in it. For convenience, there was a selection of sampling length made available by the British Standard BS 1134, the US Standard B46 and ISO recommendations R468 to suit the process being measured. The cutoff wavelengths recognized have been 2.5mm, 0.8mm and 0.25mm, they roughly represent the boundaries of waviness for manufacturing surfaces [Whi94]. The filter cutoff value can have a huge impact on the evaluation of roughness and waviness profiles. Choosing a smaller cutoff value can result in smaller roughness values. It is very important to carefully consider the filter options and values for accurate and robust results. Choosing a wrong filter type and value can lead to error prone results, hence it is recommended that the client use the default settings of the software when they are not really sure about the filter options. Filters used in the software and its mathematical representations are explained in detail in chapter 2 section 2.3.

1.2.3 Parameter calculation

The last and most important part of the analysis is "*Parameter calculation*". The functional behavior of the surface is known through it. Once the surface is filtered and profiles are obtained, the next important task is to calculate the characteristic parameters. The simplest of parameters is the total height of the profile, i.e. the "peak to valley" height. For the primary profile the total peak to valley height is designated as P_t , for the waviness profile it is W_t and for the roughness profile it is R_t . The first letter always designates the profile type. R_t reacts sensitively to outliers and is used particularly to detect cracks in the boundary layer [See96]. The most commonly used roughness parameter is the average roughness R_a . The average roughness calculates the average distance between the surface and the reference line along all the points in the profile. Since its the average distance (with respect to reference line), several profiles can have same R_a values. The parameter R_a is often used when tolerancing rough surfaces [See96]. Parameter calculations based on DIN EN ISO 4287 and DIN EN ISO 13565 are described in detail in chapter 2 section 2.4.

1.3 Organisation

This thesis is organised into seven chapters, starting with introduction and state of the art solution. Chapter 2 describes the preprocessing steps used in the software. It summarizes the algorithms that are available in literature. Section 2.2 describes the various fitting methods used for form removal. Section 2.3 elaborates the filtering techniques used in the software according to ISO 16610 series and DIN EN ISO 13565-1 (special filter). Section 2.4 contains the parameter calculations according to DIN EN ISO 4287 and Abbott curve parameters according to DIN EN ISO 13565-

2. Chapter 3 walks through the corrigendum of the ISO filters and also focuses on the sorting algorithm used to evaluate Abbott curve parameters. Chapter 4 provides an insight to uncertainty in measurements and presents some of the uncertainty in roughness parameters. Chapter 5 focuses on the development and deployment of the reference software and its application. It was developed using Java programming language with Eclipse as IDE. Several frameworks like the Play framework and JFreeChart were used to make this software a user friendly application. Chapter 6 includes the experimental results and comparison of the software (soft gauge). Finally chapter 7 describes the conclusion and summary of this thesis.

2 State of the art

Most of the algorithms used to evaluate surface parameters are from the International ISO standards. These standards describe the properties of instruments, measurement standards, filter functions and the definitions and calculations of surface parameters. Unfortunately all the standards have ambiguous definitions, the implementations are not clearly described. It is therefore necessary to verify the precision of the software used in the measuring instruments. Most of the software used in the instruments are manufactured based on ISO 4287 [DIN09]. Each of these instruments have their own analysis software. Due to ambiguous definitions in the standards, different implementation can provide different results. Therefore there is a need for a standard reference software that can compare the results of software under test and the software used in the measuring instruments.

The instruments commonly used to measure surface texture are computer based. When a surface is measured it collects the data, filters and calculates the surface parameters to get the functional properties of the surface. The analysis and evaluation carried out by the software is important for accurate results and performance of the surface. The correctness of the software is to be verified for its robustness. Hence comparability of the measurement results play a vital role in surface metrology.

There are some institutes and organisations likes National Institute of Standards and Technology (NIST) and Physikalisch Technische Bundesanstalt (PTB) that have developed such a software. The NIST in US have reported on the development of Internet based surface metrology algorithm testing system and the Physikalisch Technische Bundesanstalt in Germany has developed reference software for roughness analysis. Both these software are available online on respective websites. This chapter reviews the software that are available and compares the difference between them with the proposed reference software.

2.1 Measurement standard

ISO 5436-2 defines two measurement standards, Type F1 - reference data sets and Type F2 - reference software against which the software under test (instruments) is verified.

Type F1: This measurement standard is used to test the software by passing it into the software under test as input data and verifying the test results with the soft gauge. The software used in the instruments have there own data formats. A standard file format with an extension *.smd* is recommended for Type F1 to provide a

unique format for data handling. Figure 2.1 is an example for type F1 measurement standard.

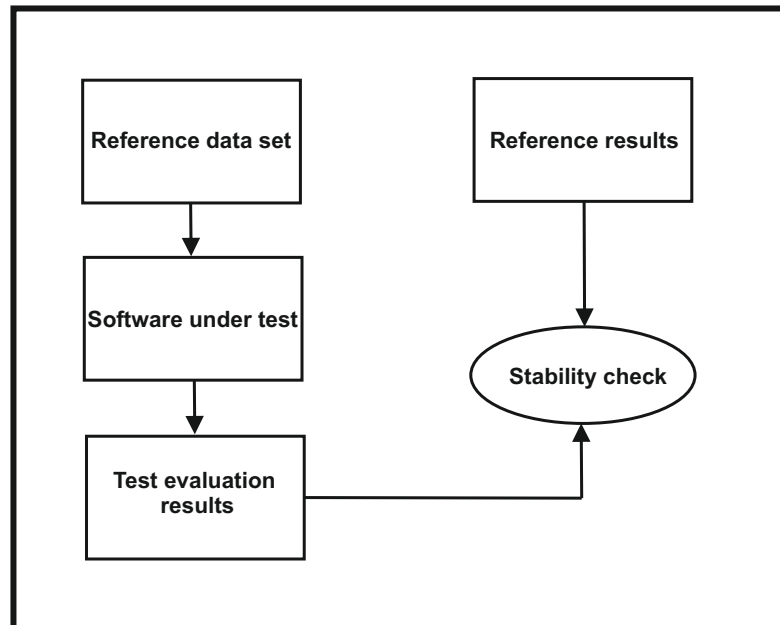


Figure 2.1: Type F1 measurement standard

Table 2.1 is an example for Type F1 measurement standard. The reference data set used in the table was downloaded from the website of National Physical Laboratory (NPL) in the UK [NPL11]. From the table it is clear that the results obtained are comparable. There are cases where the parameter values obtained under same evaluation conditions are different. They are possibly due to ambiguous definitions in the ISO 4287 standard and due to different ways of analysing the parameters. Therefore it is necessary to have unambiguous definitions to get the actual property of surface parameters.

Type F2: Reference software is a software against which the software under test is compared. The reference software compared must be consistent with each other. The software proposed in this thesis provides mathematically well defined and unambiguous definitions of surface parameters. It is compared with other software like NIST and PTB, the results are tabulated in chapter 6. Figure 2.2 is an example for type F2 measurement standard.

Table 2.1: Type F1 (Profile: sine wave, filter: Gaussian , Lc = 0.8 mm)

R-profile	Reference data set	Software under test
R_a	0.636 μm	0.64 μm
R_q	0.707 μm	0.71 μm
R_z	2.000 μm	2.00 μm
R_p	1.000 μm	1.00 μm
R_v	1.000 μm	1.00 μm
R_t	2.000 μm	2.00 μm
R_c	2.000 μm	2.00 μm
R_{sk}	1.704E-16	0.00
R_{ku}	1.500	1.50
RS_m	160.0 μm	0.16 mm

2.1.1 Software review

NIST: The internet based system by NIST was developed using Java Servlet. There are few reference data set available on the NIST website, some of them are random profile, sine wave, square wave and impulse. The user has the possibility to submit profiles into the system. The system supports ASCII and SDF data formats. Depending upon the selections made by the user (curvature removal and filter options), the primary, waviness and roughness profiles are displayed and the surface parameters are calculated. Parameters include height, spacing and hybrid parameters. Parameters are calculated based on evaluation length or sampling length as per user selection. Power spectral density (PSD) and auto correlation can also be evaluated using the NIST testing system. The report can be viewed or downloaded as a pdf document. The data files can be downloaded and saved in Surface Data File (SDF) and Extensible Markup Language (XML) format.

PTB: PTB software is based on the International standards for roughness measurement. The reference software by PTB is available online on the PTB website. It can process *.smd* data format as defined in the measurement standard (ISO 5436-2) and also the *.pr* data files. Like the NIST software, the report analysis can be downloaded as pdf document or sent via email to the user. This software does not seem to work fine for all *.smd* data formats. More often give rise to "filesize is 0 Bytes - upload failed" error message. Only 2RC and Gaussian filter (ISO11562) are used for filter application (status: 08/2014). The PTB software was recently updated with additional features like tilt correction, filter options and parameter evaluation (status: 02/2018). In chapter 6, table 6.4 and 6.5 compares the roughness parameters of a

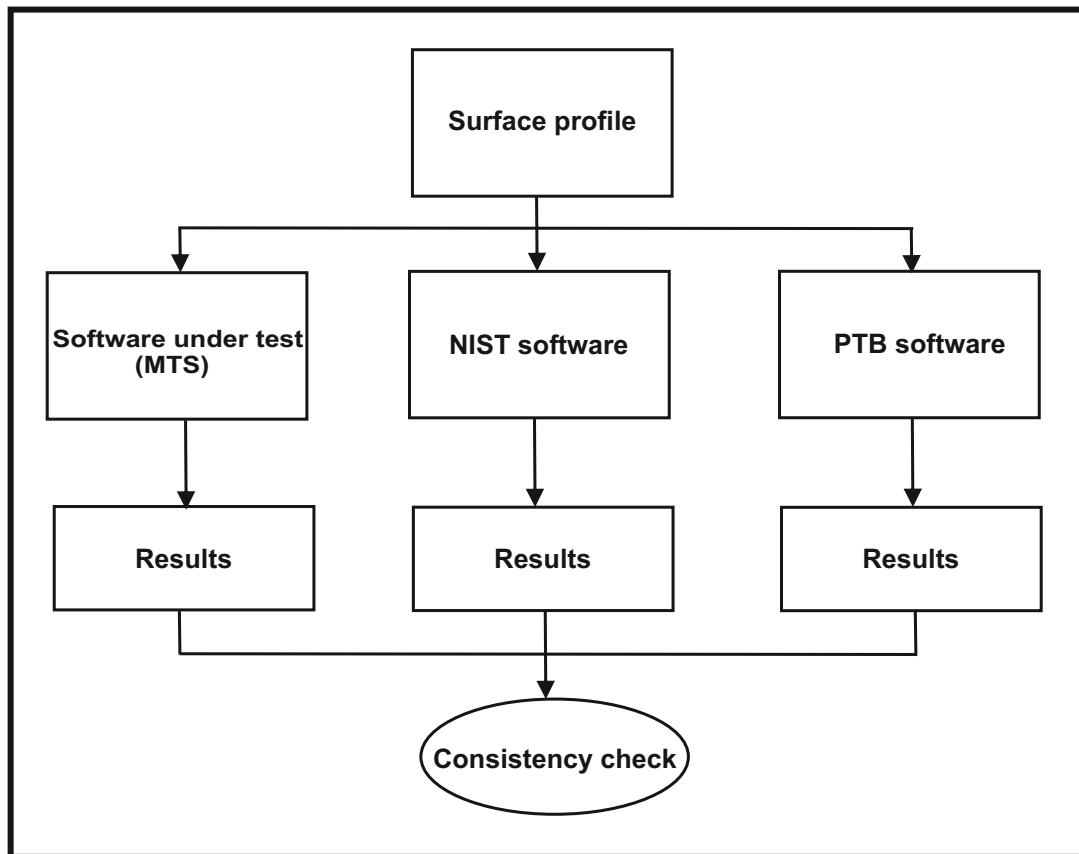


Figure 2.2: Type F2 measurement standard

profile evaluated using the proposed software and the updated version of the PTB software.

The National Physical Laboratory (NPL) in the UK, has also developed a stand alone software which can be downloaded freely from the internet. It can also process *.smd* data files. As already mentioned, the reference data set used in table 2.1 was downloaded from NPL website for comparison with the software under test. To conclude, the main goal of this thesis was to develop a reference software that can have well defined unambiguous definitions of parameters such that the software used in various measurement instruments and other software can be compared.

2.2 Preprocessing

In some cases the surface to be measured is tilted relative to the measuring device or in some cases it may be curved. In either cases the underlying geometry must be removed. Hence a reference line or circle is used depending upon the surface type. The most commonly used fitting methods are the least squares method and the total least squares method. These methods are used to fit a reference line or

circle to remove the form from the profile. The least squares and the total least squares methods access the fitting accuracy in different ways. The least squares method minimizes the sum of the squared vertical distances from the datapoints to the fitting line. While the total least squares method minimizes the sum of the squared orthogonal distances from the datapoints to the fitting line. In the past, the least square method was generally used to solve the fitting problems. Today the ISO standard uses the total least squares fit as the default fitting method.

This section deals with 2D fitting problems. It includes the simple and robust algorithm for fitting a reference line and circle, the algorithms for least square orthogonal fitting of ellipse, hyperbola and parabola are covered in other books and papers (refer [ARW01]). This section is mainly based on the recent advances made by author S. J. Ahn. The authors Least squares orthogonal distance fitting approaches are used in the software proposed. The profile used in this section is measured from the workpiece KNT 4058 (S1) roughness measurement standard using stylus instrument T8000 at the laboratory of Measurement and Sensor Technology (MTS), T.U. Kaiserslautern.

2.2.1 Geometrical fitting of straight line

Mean line fitting

There are several methods to fit a straight line to a set of datapoints. Mean line fitting is one such fitting method which is used in the proposed software. This method evaluates the mean of the data values and the mean value is subtracted from the datapoints to get a set of values which represents the form removed profile. This fitting method is not widely used today. The most commonly used fitting methods are the least squares method and the total least squares method. They are described in sub-sections below.

Figure 2.3 shows a profile with mean line fitting (red) and the plot below it is the form removed profile. The residuals above and below the fitted mean line is called the primary profile. A short wavelength filter is sometimes used to remove the noise from the primary profile. Once the geometry is removed, the waviness and roughness in the surface is to be separated from the primary profile. This part of the analysis is covered in the upcoming sections.

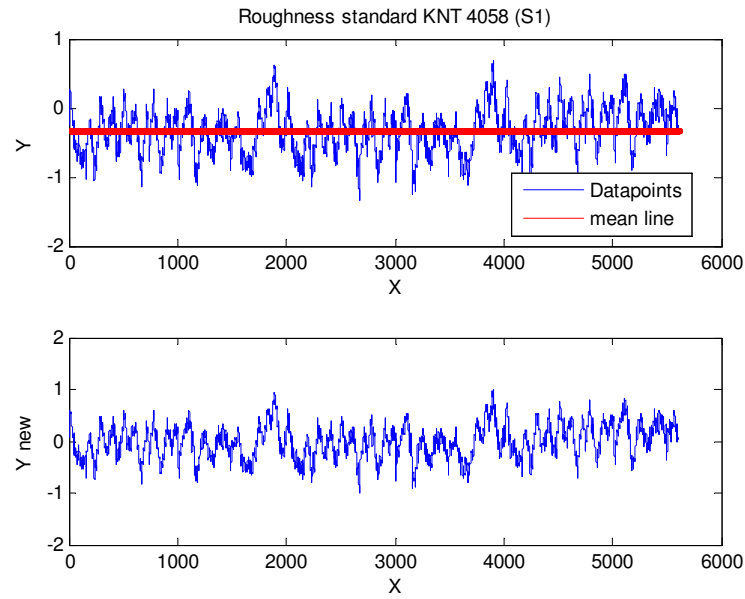


Figure 2.3: Primary profile after mean line fitting

Least square fitting

The least squares fitting minimizes the sum of squared residuals. Generalizing from a straight line to a k^{th} degree polynomial.

$$Y = a_0 + a_1x + a_2x^2 + \dots + a_kx^k \quad (2.1)$$

to approximate the given set of data $(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)$ where $(n \geq k + 1)$, the fitting curve $f(x)$ has the least square residual

$$R^2 = \sum_{i=1}^n [y_i - f(x_i)]^2 = \sum_{i=1}^n [y_i - (a_0 + a_1x_i + a_2x_i^2 + \dots + a_kx_i^k)]^2 \quad (2.2)$$

$a_0, a_1, a_2 \dots a_k$ are the unknown coefficients to be obtained. By evaluating the partial derivatives of the equation 2.2 with respect to $a_0, a_1 \dots a_k$ and equating the derivatives to zero leads to the following k equations.

The partial derivatives are:

$$\frac{\partial(R^2)}{\partial a_0} = -2 \sum_{i=1}^n [y_i - (a_0 + a_1x_i + a_2x_i^2 + \dots + a_kx_i^k)] = 0 \quad (2.3)$$

$$\frac{\partial(R^2)}{\partial a_1} = -2 \sum_{i=1}^n [y_i - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_k x_i^k)] x_i = 0 \quad (2.4)$$

$$\frac{\partial(R^2)}{\partial a_k} = -2 \sum_{i=1}^n [y_i - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_k x_i^k)] x_i^k = 0 \quad (2.5)$$

Expanding the above equations:

$$\sum_{i=1}^n y_i = a_0 \sum_{i=1}^n 1 + a_1 \sum_{i=1}^n x_i + a_2 \sum_{i=1}^n x_i^2 + \dots + a_k \sum_{i=1}^n x_i^k \quad (2.6)$$

$$\sum_{i=1}^n x_i y_i = a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 + a_2 \sum_{i=1}^n x_i^3 + \dots + a_k \sum_{i=1}^n x_i^{k+1} \quad (2.7)$$

$$\sum_{i=1}^n x_i^k y_i = a_0 \sum_{i=1}^n x_i^k + a_1 \sum_{i=1}^n x_i^{k+1} + a_2 \sum_{i=1}^n x_i^{k+2} + \dots + a_k \sum_{i=1}^n x_i^{2k} \quad (2.8)$$

in matrix form,

$$\begin{bmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i y_i \\ \vdots \\ \sum_{i=1}^n x_i^k y_i \end{bmatrix} = \begin{bmatrix} n & \sum_{i=1}^n x_i & \dots & \sum_{i=1}^n x_i^k \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \dots & \sum_{i=1}^n x_i^{k+1} \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{i=1}^n x_i^k & \sum_{i=1}^n x_i^{k+1} & \dots & \sum_{i=1}^n x_i^{2k} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} \quad (2.9)$$

This is a Vandermonde matrix. The matrix for a least squares fit, given n points and fitting with polynomial coefficients $a_0 a_1 \dots a_k$ is

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 & \dots & x_1^k \\ 1 & x_2 & \dots & x_2^k \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & \dots & x_n^k \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} \quad (2.10)$$

This can be solved by multiplying the transpose X^T on both sides of the equation,

$$X^T y = X^T X a \quad (2.11)$$

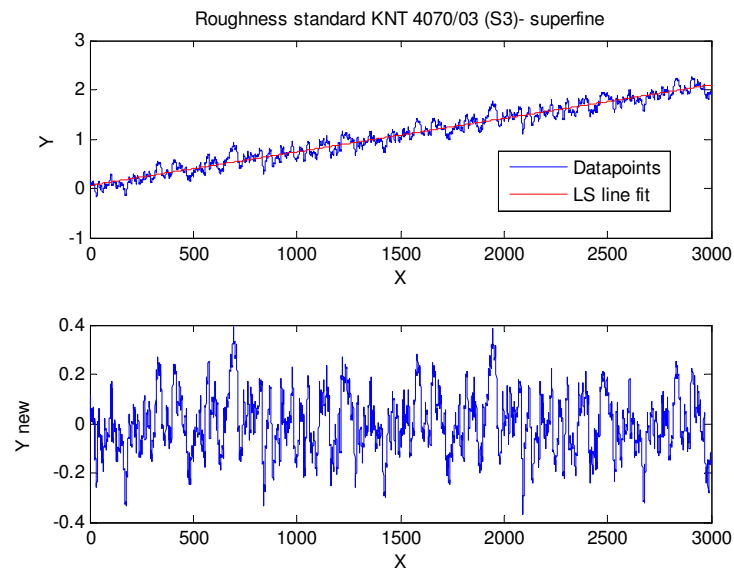


Figure 2.4: Primary profile after least square fitting

This matrix equation can be solved numerically to obtain the vector

$$a = (X^T X)^{-1} X^T y \quad (2.12)$$

Setting $k=1$ in the above equation reproduce the linear solution [Wei99].

The "best fit" line will have minimum error between the line and the datapoints. Figure 2.4 shows a profile with the least square line fit (red) and the plot below it is the form removed profile.

Total least square fitting

The total least squares method minimizes the sum of the squared orthogonal distances from the datapoints to the fitting line. It is the most accurate method of finding the best fit and is ISO recommended. The total least square method is based on the algorithm proposed by S.J. Ahn [Ahn04]. The algorithm was used to implement the implicit form of a straight line as shown in figure 2.5. The plot above (blue) is the raw profile superimposed on the raw data is a total least square line (red) which is implemented by the method described in this sub-section. The plot below it is the total least squares removed profile (primary profile) which will be used for further analysis.

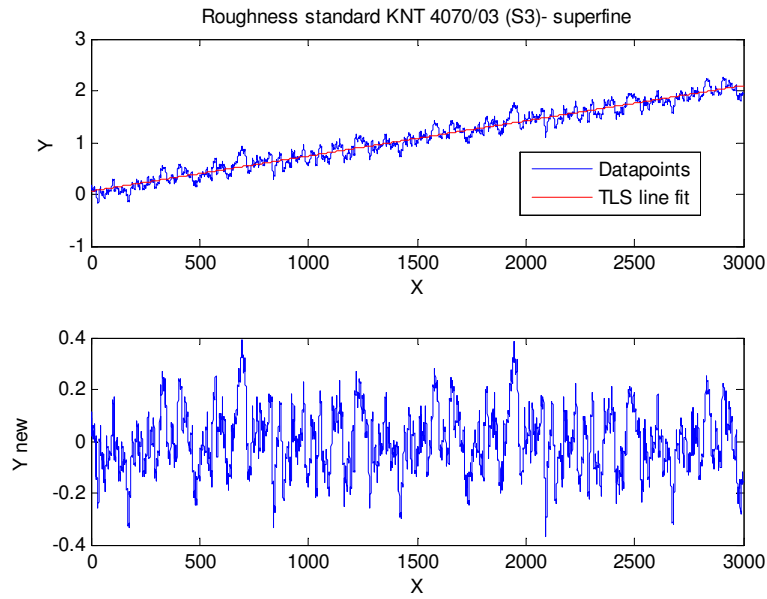


Figure 2.5: Primary profile after total least square fitting

Lets consider the total least squares method for finding the line that minimizes the sum of squares of true distances. If $X = X_c + t \cdot r$ is the line passing through the point X_c and parallel to the unit vector r in n -dimensional space ($n \geq 2$), then the square sum of the orthogonal distances from m given points to the line is given as:

$$\sum_{i=1}^m \|e_i\|^2 = \sum_{i=1}^m \|(X_i - X_c) \cdot r\|^2 \quad (2.13)$$

If r is parallel to one of the axis, the coordinate value of X_c is set along that axis. Therefore we get:

$$X_c = \frac{1}{m} \sum_{i=1}^m X_i \quad (2.14)$$

$$Y_c = \frac{1}{m} \sum_{i=1}^m Y_i \quad (2.15)$$

if we set,

$$x_i = X_i - X_c, \quad y_i = Y_i - Y_c \quad (2.16)$$

$$M_{xx} = \sum_{i=1}^m x_i^2 \quad (2.17)$$

$$M_{yy} = \sum_{i=1}^m y_i^2 \quad (2.18)$$

$$M_{xy} = \sum_{i=1}^m x_i y_i \quad (2.19)$$

then the matrix H is defined as:

$$H = \begin{pmatrix} M_{yy} & -M_{xy} \\ -M_{xy} & M_{xx} \end{pmatrix}$$

The square matrix H for n -dimensional space ($n \geq 2$) can be decomposed using Singular Value Decomposition (SVD). Therefore,

$$H = V_H W_H V_H^T \quad (2.20)$$

where,

W_{H1}, \dots, W_{Hn} = singular values

V_{H1}, \dots, V_{Hn} = orthogonal vectors

if r is parallel to one of the principal axes V_{Hj} , then the squares sum of the orthogonal distances in equation 2.13 becomes

$$\sum_{i=1}^m \|e_i\|^2 = r^T H r = V_{Hj}^T V_H W_H V_H^T V_{Hj} = W_{Hj} \quad (2.21)$$

the index j is chosen to be the smallest W_{Hj} . Hence, the total least square method is to calculate r that minimizes the equation. Therefore the geometric fitting line is:

$$X = X_c + t \cdot V_{Hj} \quad (2.22)$$

$$Y = Y_c + t \cdot V_{Hj} \quad (2.23)$$

with error variance

$$\sigma^2 = \frac{W_{Hj}}{m} \quad (2.24)$$

The parameter covariance matrix is given by:

$$Cov(a) = (J^T J)^{-1} = (VWU^T U W V^T)^{-1} = V W^{-2} V^T \quad (2.25)$$

$$Cov(a_j, a_k) = \sum_{i=1}^q \left(\frac{V_{ji} V_{ki}}{w_i^2} \right), j = 1, \dots, q, k = 1, \dots, q. \quad (2.26)$$

Consequently, the variance of the estimated parameters is:

$$\sigma^2(a_j) = \frac{\sigma_0^2}{p - q} Cov(a_j, a_k), j = 1, \dots, q \quad (2.27)$$

The geometric fitting line of the given datapoints in n-dimensional space passes the center of gravitation and is parallel to one of the principal centroidal moment [ARR99]. The difference in the results using least squares line fit and total least squares line fit is found to be relatively small, as it depends on the inclination. Matlab script for TLS fit is available in appendix B.1 for reference.

2.2.2 Geometrical fitting of circle

The geometrical fitting of circle is based on the coordinate description of the corresponding point on the circle for the given datapoint, where the distance between the two points is the shortest path from the given point to the circle. In the past, fitting problems have been usually solved through the Least Squares Method (LSM). The geometric fitting of circle is a nonlinear problem and must be solved iteratively. There are several kinds of iteration techniques like the Newton, Gauss-Newton, and Steepest descent iterations. Here the Gauss-Newton iteration is used. Fitting requires a set of starting values to start the iteration process, poor starting values can affect the quality of fit. The algorithm used in the software for circular fit is as described in the paper [ARR99]. Figure 2.6 shows the circular view of the surface profile measured.

Iterative method

The circle in n- dimensional space ($n \geq 2$) with center at X_c and radius R can be described as:

$$X_c = \frac{1}{m} \sum_{i=1}^m X_i \quad (2.28)$$

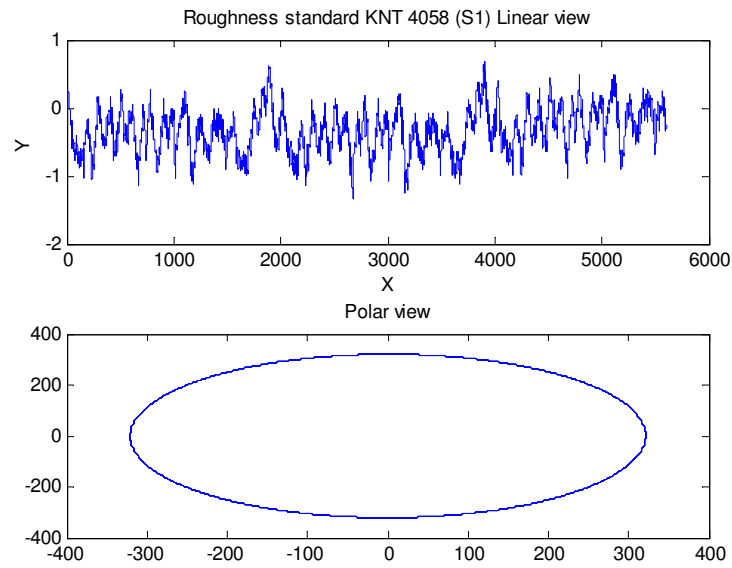


Figure 2.6: Circular view of the surface profile

$$R^2 = \| X - X_c \|^2 \quad (2.29)$$

$$R = \sqrt{\frac{1}{m} \sum_{i=1}^m \| X_i - X_c \|^2} \quad (2.30)$$

For a given point X_i ($\neq X_c$), the nearest corresponding point X'_i on the circle is:

$$X'_i = X_c + R \frac{X_i - X_c}{\| X_i - X_c \|}, \quad i = 1, \dots, m \quad (2.31)$$

with orthogonal error distance vector

$$X''_i = X_i - X'_i = [\| X_i - X_c \| - R] \frac{X_i - X_c}{\| X_i - X_c \|} \quad (2.32)$$

From equation 2.31, the Jacobian matrix J for the Gauss-Newton iteration can be directly derived. Once the Jacobian matrix is known, it can be decomposed using Singular Value Decomposition (SVD).

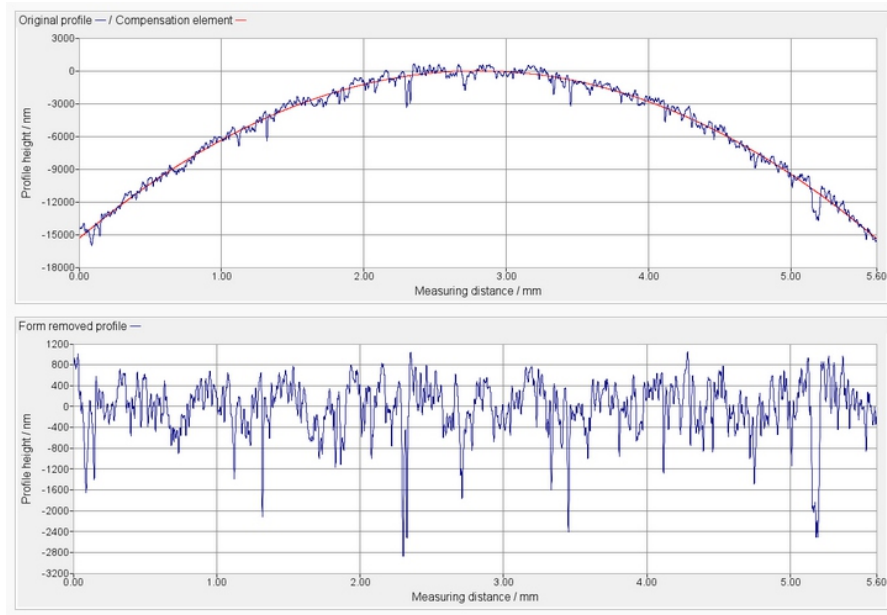


Figure 2.7: Circular fit of a surface profile (cylinder) measured using T8000

$$J_{X'_i, a} = \frac{\partial}{\partial a} \left[X_c + R \frac{X_i - X_c}{\|X_i - X_c\|} \right] \quad (2.33)$$

$$J_{X'_i, a} = \frac{\partial X_c}{\partial a} + \frac{X_i - X_c}{\|X_i - X_c\|} \frac{\partial R}{\partial a} - \frac{R}{\|X_i - X_c\|} [I - D] \frac{\partial X_c}{\partial a} \quad (2.34)$$

where:

$$D = \frac{(X_i - X_c)(X_i - X_c)^T}{\|X_i - X_c\|^2}. \quad (2.35)$$

The parameter vector a is defined as:

$$a^T = (R, X_c^T) = (R, X_c, Y_c). \quad (2.36)$$

We have,

$$\frac{\partial R}{\partial a} = (1 \mid 0^T), \quad \frac{\partial X_c}{\partial a} = (0 \mid I). \quad (2.37)$$

With the Jacobian matrix $J_{X'_i, a}$ and the error distance vector X''_i at each point X'_i , we construct $p = (m \times n)$ linear equations for the m given n -dimensional points. The nearest point on a circle from a given point can be located without any difficulty from

equation 2.31. The initial parameter vector starting the Gauss-Newton iteration may be supplied from an algebraic fitting. Here the center of the circle and the RMS central distances are used as initial parameter values as in equation 2.36. After solving the linear equation, the parameter vector a is updated with a step size parameter $\lambda=1.3$.

After successful termination of the iteration, information about the quality of the fit is provided. The parameter covariance matrix is given by:

$$Cov(a) = (J^T J)^{-1} = (VWU^T U W V^T)^{-1} = V W^{-2} V^T \quad (2.38)$$

$$Cov(a_j, a_k) = \sum_{i=1}^q \left(\frac{V_{ji} V_{ki}}{w_i^2} \right), j = 1, \dots, q, k = 1, \dots, q. \quad (2.39)$$

Consequently, the variance of the estimated parameters is:

$$\sigma^2(a_j) = \frac{\sigma_0^2}{p-q} Cov(a_j, a_k), j = 1, \dots, q \quad (2.40)$$

In summary, the geometric fitting requires an iterative process to find the best fit to the datapoints. The higher the iteration, the greater is the accuracy but performance is slow. On the other hand the lower the iteration, faster is the fitting process but less accurate. The algorithm proposed by S.J. Ahn is robust when compared to the least square method [ARR99]. Matlab script for circular fit is available in appendix B.2 for reference.

2.3 Filter application

This section provides a brief overview about various kinds of filters used in the proposed software. They are basically used to extract surface roughness by separating the data into three components. Most of the filters used today in surface metrology belong to the class of linear filters. The most commonly used filtering technique is the Gaussian filter. Gaussian filters were introduced to overcome the disadvantages caused by 2RC filters, however they give rise to end effects. These end effects are due to convolution. i.e. the convolution between the input data and the filter weighting function. To overcome the drawbacks of Gaussian filter, spline filter algorithm was introduced by M. Krystek [Kry96]. T. Goto [TG04] proposed the robust spline filter which is less prone to outliers and the Gaussian regression filter was proposed by H. Bodschwinn and J. Seewig in the year 1996 [See05]. These filters are designed

for overcoming the problems like end effect, distortions and to improve the computational efficiency and robustness in surface measurement. Robust filters are found to be insensitive to extreme points, whereas the Gaussian linear filter is sensitive to extreme points.

2.3.1 2RC filter

The 2RC filter was the first filter to be used in surface metrology. It was used to obtain the roughness profile from raw profile [MR09]. A short description about 2RC filter is given below.

The weighting function of a 2RC roughness filter is:

$$S(x) = \frac{A}{\lambda_c} \left(2 - A \frac{|X|}{\lambda_c} \right) \exp \left(-A \frac{|X|}{\lambda_c} \right) \quad (2.41)$$

where,

$A = 3,64$ for 75% transmission at λ_c

λ_c = is the cutoff wavelength

x = the position in millimeters from the origin of the weighting function

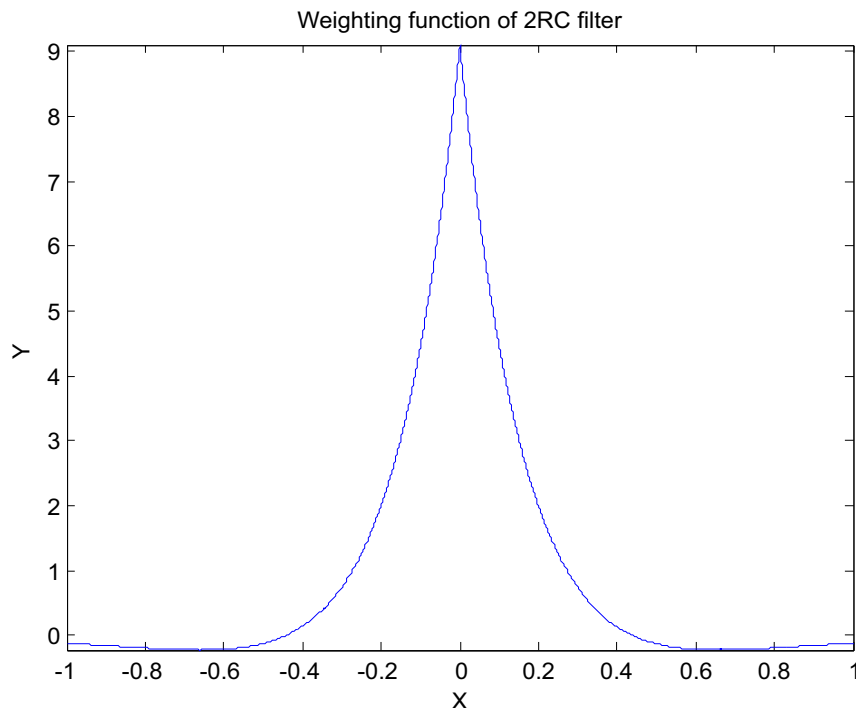


Figure 2.8: 2RC Weighting Function

Convolving the primary profile with the weighting function given in equation 2.41 results in 2RC filter mean line. The filter mean line generated by the 2RC filter does not represent the waviness profile. Subtracting the filter mean line from the profile provides the roughness profile. The 2RC filter is currently discarded from the International ISO standards, therefore it is not used in the software. The transmission characteristics of the filter can be obtained from the Fourier transform of equation 2.41. Phase distortion of the filter mean line with respect to the primary profile is considered to be an undesirable feature of the 2RC filter. Despite its limitations, the 2RC filter was historically used for several years. Figure 2.8 shows the weighting function of a 2RC roughness filter.

2.3.2 Gaussian filter

Gaussian filters are used to separate the profiles into waviness and roughness components. The weighting function of Gaussian filters is symmetrical, it is as shown in figure 3.1. Since the profile is defined for finite length, the Gaussian filter shows end effects at the end of the profile. They are generally reduced by discarding the data at the beginning and the end of the filtered profile. The discarded area is called the end effect region. ISO 16610-28 [Int10a] standard describes the end effect correction methods, they are generally used when the entire measurement data set is to be used for further analysis. In this filter method, the waviness and roughness components can be added back together to recreate the original profile [DIN12] [MR09]. The Gaussian filter used is based on ISO 16610-21 standard.

End effects

Convolution of a Gaussian filter with the data profile results in distortions at the end of the profile. The distortions at the end of the profile are termed as "*end effects*". Figure 2.9 shows end effects caused by Gaussian filtering at the ends of a honed cylinder profile measured using Hommel T8000 at the Institute of Measurements and Sensor Technology. These end effects can be reduced by considering longer profiles during measurements, such that the end effect region in the profile can be discarded. But in cases where small measurements are to be made, end effect correction methods as described in ISO 16610-28 are to be used. The spline filter and robust profile filters like Gaussian regression filter and robust spline filter described in sections below have an automatic end effects correction.

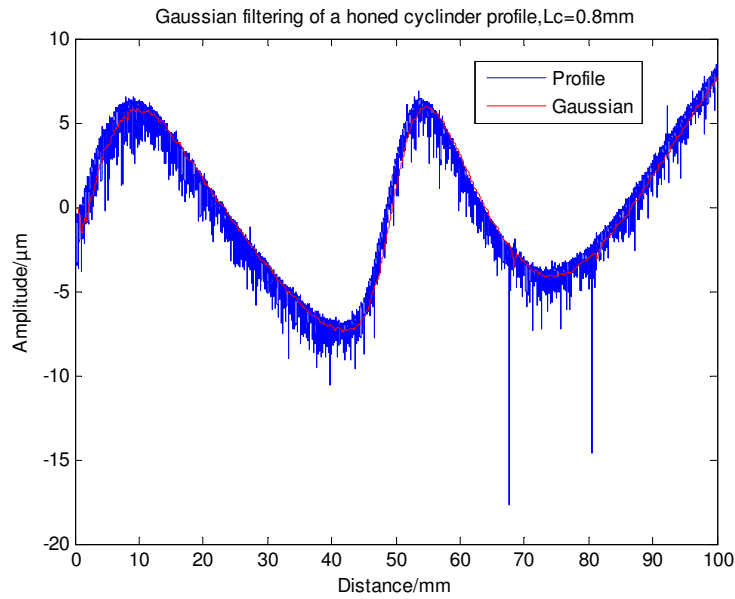


Figure 2.9: Gaussian filtering, $L_c=0.8\text{mm}$

Outliers

Outliers are spikes that are created on the surface profiles but are not seen on the actual workpiece. They are generally visible on profiles when measured using optical instruments. As already mentioned in chapter 1, these instruments require sufficient amount of light to be reflected from the sample to the detector. At times, the light does not pass through the surface with deep valley, hence its nearly impossible to get the height information from the surface profile and thereby provides incorrect value at those points. The value at these points can either be higher or lower than the surrounding values. This give rise to a sharp peak or valley in the profile which is termed as "*outliers*". The implementation of Gaussian filter is explained in chapter 3 section 3.1.1 for open profiles and section 3.1.2 for closed profiles.

2.3.3 Special filter

Special filter is a filter for surface profiles with stratified functional properties i.e. surfaces with minimal waviness. It is used to suppress the valley influence on the mean line so that a satisfactory line is generated. DIN EN ISO 13565-1 termed as special filter is based on the Gaussian filter. It is also called "*Double filtering*".

In this filter the surface profile measured is filtered by the Gaussian filter and all parts of the profile below the mean line obtained using Gaussian filter are removed. The profile left behind is filtered again by the Gaussian filter. The resulting mean

line is the waviness profile. The roughness profile is obtained by subtracting the waviness profile from the measured profile. The roughness profiles evaluated from surfaces with deep valleys suffer from undesired distortions when generated using Gaussian filter. ISO 13565-1 filtering method is used to reduce these distortions. Although this filter provides acceptable results, it has restrictions because it only suppresses valleys. The parameters evaluated from these profiles are defined as ISO 13565-2 parameters. It is used in the software according to DIN EN ISO 13565-1 standard [DIN98b]

2.3.4 Spline filter

A spline filter is a digital filter. The mean line obtained from the spline filter is a spline, hence the name *Spline filter*. The most commonly used spline filter is the cubic spline, i.e. a cubic spline made of cubic polynomials. Cubic splines are curves that smoothly connects all the datapoints in a profile. This filter was developed to overcome the disadvantages caused by the Gaussian filter. Spline filter shows no end effects due to automatic end effect correction, hence it provides mean line for the entire profile and no information is lost from the surface profile measured. Figure 2.11 shows spline filtering of a honed cylinder profile measured using Hommel T8000 at the Institute of Measurements and Sensor Technology. Spline filter used in the software is based on ISO 16610-22 standard [Int06].

Weighting function

The transfer function of the cubic spline filter is steeper than the Gaussian filter according to ISO 11562 [DIN98a], which means that spline filters have better selectivity than the Gaussian filters. The difference between the transfer functions of the spline and Gaussian filter is relatively small, the spline filter does not have significant merits over the Gaussian filter. Spline filter was developed to approximate the Gaussian filter.

The weighting function of the spline profile filter is:

$$S(x) = \frac{\pi}{\lambda_c} \sin\left(\sqrt{2} \frac{\pi}{\lambda_c} |x| + \frac{\pi}{4}\right) \exp\left(-\sqrt{2} \frac{\pi}{\lambda_c} |x|\right) \quad (2.42)$$

where,

λ_c = the cutoff wavelength of the profile filter

x = the position in millimeters from the origin of the weighting function

The weighting function of a spline profile filter is as shown in figure 2.10

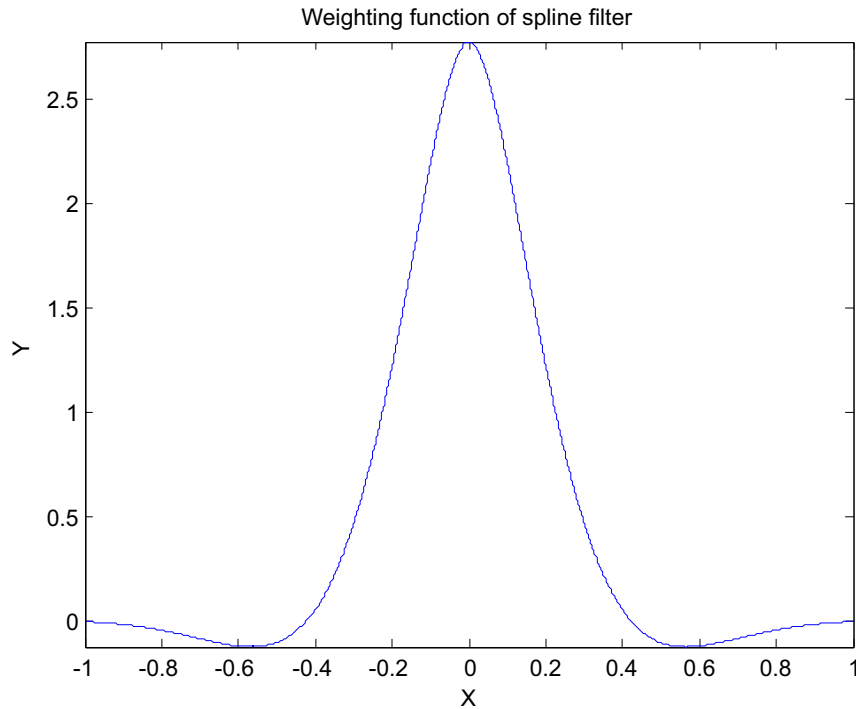


Figure 2.10: Weighting function of spline filter

Transmission characteristics

The transmission characteristics for spline profile filter are based on cardinal cubic splines. As mentioned above, a spline filter is a digital filter, it is not necessary to sample the weighting function. Therefore spline filters are defined by filter equations and not by weighting functions. Fourier transform of the spline filter equation gives the filter characteristics.

For the long wave component the transmission characteristics is given by:

$$\frac{a1}{a0} = \left[1 + \beta \alpha^2 \sin^2 \left(\frac{\pi \Delta x}{\lambda} \right) + 16(1 - \beta) \alpha^4 \sin^4 \left(\frac{\pi \Delta x}{\lambda} \right) \right]^{-1} \quad (2.43)$$

where,

$a0$ = amplitude of the profile before filtering

$a1$ = amplitude of the filtered profile

λ = wavelength of the profile

Δx = sampling interval

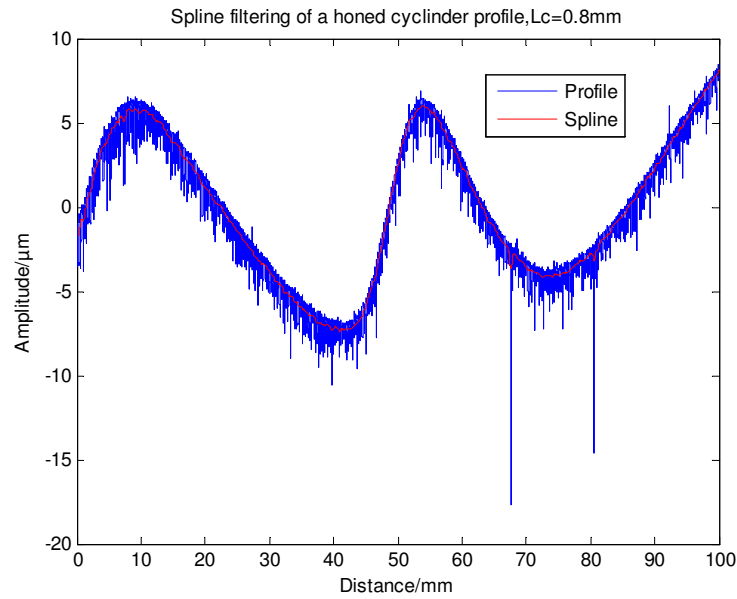


Figure 2.11: Spline filtering, $L_c=0.8\text{mm}$

Transmission characteristics of the short wave component is complementary to the long wave component and is given by:

$$\frac{a2}{a0} = 1 - \frac{a1}{a0} \quad (2.44)$$

where,

$a2$ = amplitude of the filtered profile

$a0$ = amplitude of the profile before filtering

λ = wavelength of the profile

Δx = sampling interval

Figure 2.12 shows the transmission characteristics of the long wave spline filter along with the Gaussian long wave filter characteristics for comparison, it is clear from the figure that the transfer function of the spline filter is sharper and steeper than the Gaussian filter. Hence the separation between the two wavelength bands is precise. However their difference would be too small for any characteristic changes. The two parameters α and β are selected such that the effect of the tension parameter β on the filter transfer capacity is controllable for the cutoff wavelength λ_c . The implementation of spline filter is described in chapter 3 section 3.2.

2.3.5 Robust Gaussian regression filter

Robust filters are improvement over other filters and handles the outliers efficiently. It shows no unwanted deviation caused by deep holes or scratches on the surface.

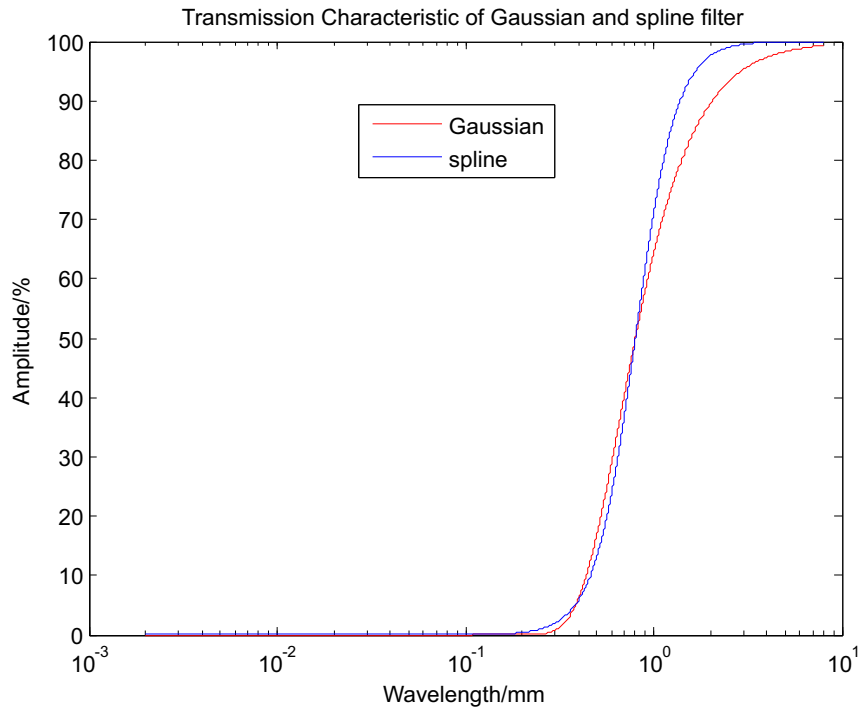


Figure 2.12: Transmission characteristic of Gaussian and spline filter

Outliers are usually seen in plateau honed surfaces. The waviness profile obtained using Gaussian filter is not robust with respect to outliers. Outliers leads to distorted roughness profile, which in turn affects the parameter calculation. The R_k filter is an alternative to this problem but the distortions are not completely removed. The outliers problem caused by the standardised Gaussian filter has motivated the development of new filters like the robust Gaussian filter and robust spline filter. The robust spline filter is removed from the ISO standard, hence its not considered in the thesis. The advantage of Gaussian regression filter over Gaussian filter is that there are no end effects, hence the complete profile can be used for further analysis and can approximate form upto p^{th} degree. For roughness measurement, it is recommended to use the 1^{st} degree filter and the 2^{nd} degree filter can be used for both roughness and form filter applications [See05]. The implementation for robust regression filter is detailed in chapter 3 section 3.3

Figure 2.13 shows the zoomed view of a random sinusoidal signal with deep holes in the profile. Its clear that the robust Gaussian regression filter shows no deviation towards the holes.

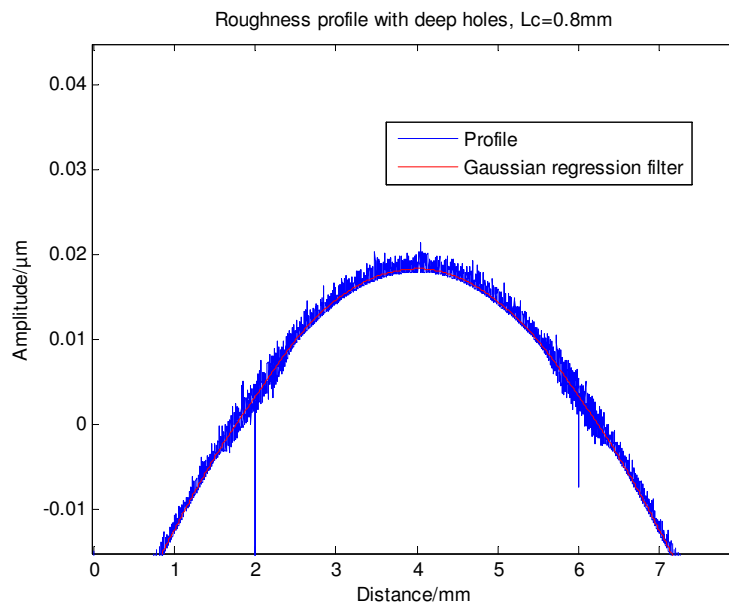


Figure 2.13: Gaussian regression filter

2.3.6 Selection of filter

Linear profile filters are commonly used for filtering surface profile elements with uniformly distributed width i.e. for homogeneous roughness structures like ground, turned and milled surfaces. On the other hand the robust profile filters are best suited for inhomogeneous roughness structures, where the width of the profile element varies strongly. Typical examples are sintered, honed and ALSi particle surfaces [VDI08].

2.4 Parameter analysis

The surface profile traced using industrial instruments consists of form, waviness and roughness components. Such profile traces are due to the intersection of the skin model by an intersection plane perpendicular to the skin model in a specified direction [Pri15]. They are referred as mechanical profile in the new profile standard. Form deviations are due to errors in the machine tool whereas waviness in the profile are produced by the vibrations in the machine tool and roughness components are often random in nature [VDI08]. The surface profile after filter application is completely free from form and waviness and is ready for parameter evaluation. This section presents the new profile standard and the parameters used in the software.

2.4.1 Primary surface profile

It is a surface profile obtained when a profile trace is represented as the mathematical model with a specified cutoff wavelength. A profile S-filter (L_s filter) is used to obtain the primary surface profile. The influence of L_s filter is noticed when a surface is measured using different measuring instrument of various stylus tip radius. For some applications like multi scale analysis, the profile S-filter is not used. In such cases the cutoff wavelength is set to zero. L_s profile filter defines the intersection between roughness profile and short wavelength components. Figure 2.14 shows the relationship between the mechanical profile and the primary surface profile. There are several parameters evaluated from the primary surface profile, each having its own functional properties. To understand these parameters, it is important to know some terms like the *Sampling length* and *Evaluation length*. Sampling length is the length across x-axis used for identifying the irregularities in the profile under evaluation. Evaluation length is the total length across x-axis of the profile under evaluation. There are three kinds of profiles defined in the International ISO standards.

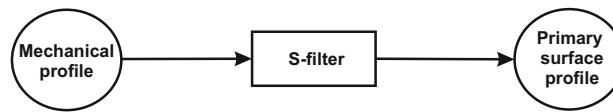


Figure 2.14: Relationship between the mechanical profile and the primary surface profile [Pri15]

Primary profile (P-profile): Primary profiles are profiles derived from the primary surface profile by removing form using profile F-operator as a prefilter. Some profile filters are sensitive to form and requires the primary surface profile to be prefiltered before being used. The primary parameters are calculated from the P-profile $p(x)$, the first letter always represent the profile type. For example P_t is the total peak to valley height of primary profile. In primary profiles the sampling length L_r is equal to the evaluation length L_n .

Waviness profile (W-profile): Waviness profiles are obtained from the primary surface profile by removing small scale lateral components by a profile S-filter with a specified cutoff wavelength. It defines the intersection between the roughness and waviness components. The Gaussian filter is used as a default filter. The waviness parameters are extracted from the W-profile $w(x)$ according to ISO 4287 [DIN09]. In waviness profiles the sampling length L_r is equal to the cutoff wavelength L_c .

Roughness profile (R-profile): Roughness profiles are obtained from the primary surface profile by removing large scale lateral components by a profile L-filter with the same specified cutoff wavelength as for the profile S-filter. They are usually superimposed on the waviness profile. The Gaussian filter is used as a default filter. The roughness parameters are extracted from the R-profile $r(x)$ according to ISO 4287 [DIN09]. In roughness profile the sampling length L_r is equal to the cutoff wavelength L_c . For roughness profiles the standard number of sampling length is five. Figure 2.15 shows the measuring chain to derive the P-profile, W-profile and R-profile.

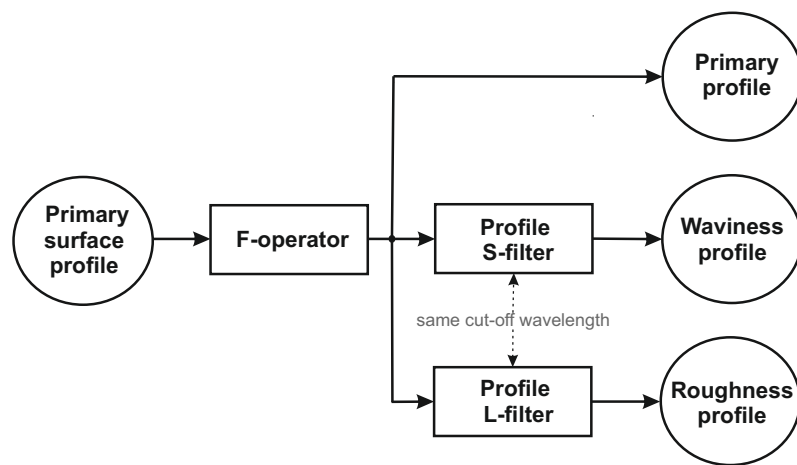


Figure 2.15: Measuring chain to obtain P-profile, W-profile and R-profile [Pri15]

Figure 2.16 shows a profile filtered using spline filter with a cutoff wavelength $L_c = 0.8\text{mm}$. The profile used here is a roughness measurement standard (S1) measured using stylus instrument T8000.

The most commonly used ISO parameters are discussed below. The Letter ' X ' is used in this section to represent parameters evaluated within sampling length and evaluation length. ISO 4287 defines parameters evaluated within sampling length and some across the evaluation length, by taking an average of parameters from each sampling length (refer table 2.2 and 2.3).

2.4.2 Amplitude parameters

Amplitude parameters give information about the height and depth measure of the profile from the mean line. The most commonly used amplitude parameters and the ones used in the proposed software are described below.

Table 2.2: Surface parameters, ISO 4287:1997

Parameters	Description	Evaluation length	Sampling length
R_a	Arithmetic mean deviation of the assessed profile		X
R_q	Root mean square deviation of the assessed profile		X
R_z	Maximum height of profile		X
R_p	Maximum profile peak height		X
R_v	Maximum profile valley depth		X
R_t	Total height of profile	X	
R_c	Mean height of profile elements		X
R_{sk}	Skewness of the assessed profile		X
R_{ku}	Kurtosis of the assessed profile		X
R_{dq}	Root mean square slope of the profile		X
R_{sm}	Mean width of profile elements		X

Table 2.3: Additional surface parameters

Parameters	Description	Evaluation length	Sampling length
R_{max}	Maximum peak to valley depth		X
R_{da}	Arithmetic mean slope of the profile		X
R_{la}	Arithmetic mean wavelength of the profile		X
R_{lq}	Root mean square wavelength of the profile		X

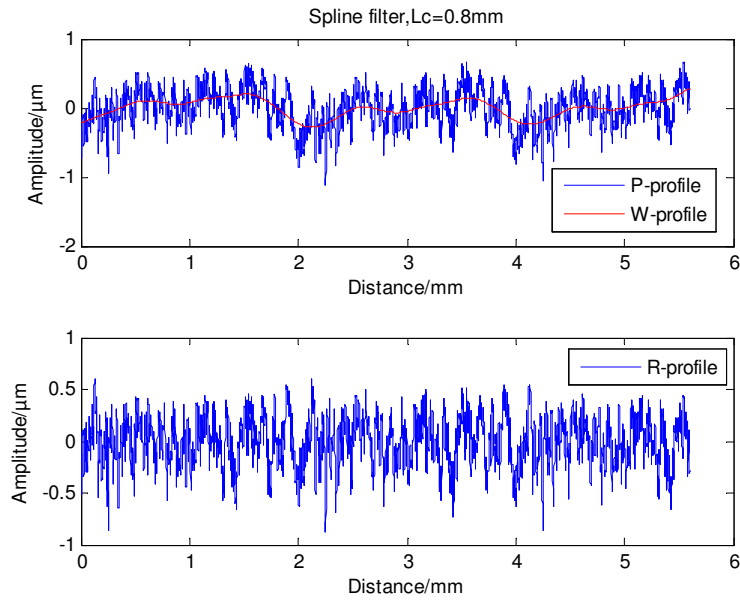


Figure 2.16: Surface profile

Roughness average X_a : X_a is defined as the arithmetic average of the absolute values of the profile z measured from the mean line.

$$X_a = \frac{1}{n} \int_n |z(x)| dx \quad (2.45)$$

Root mean square roughness X_q : X_q is defined as the root mean square average of the absolute values of the profile z measured from the mean line.

$$X_q = \sqrt{\frac{1}{n} \int_n z(x)^2 dx} \quad (2.46)$$

Peak height X_p : X_p is defined as the distance to the highest profile peak from the mean line of the surface profile within one sampling length. It is as shown in figure 2.17.

Valley depth X_v : X_v is defined as the distance to the deepest profile valley from the mean line of the surface profile within one sampling length. It is as shown in figure 2.18.

Mean profile depth X_z : X_z is defined as the arithmetic mean value of the successive distance between the highest peak and deepest valley of consecutive sampling lengths. It is as shown in figure 2.19.

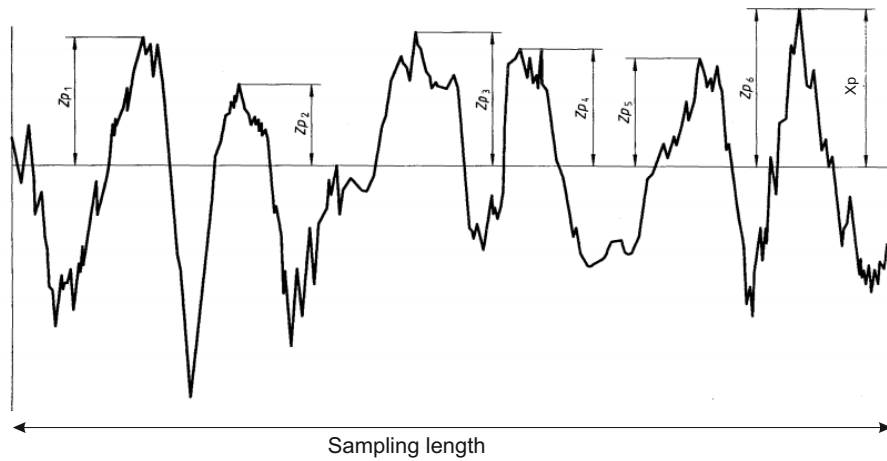


Figure 2.17: Peak height of profile, from DIN EN ISO 4287

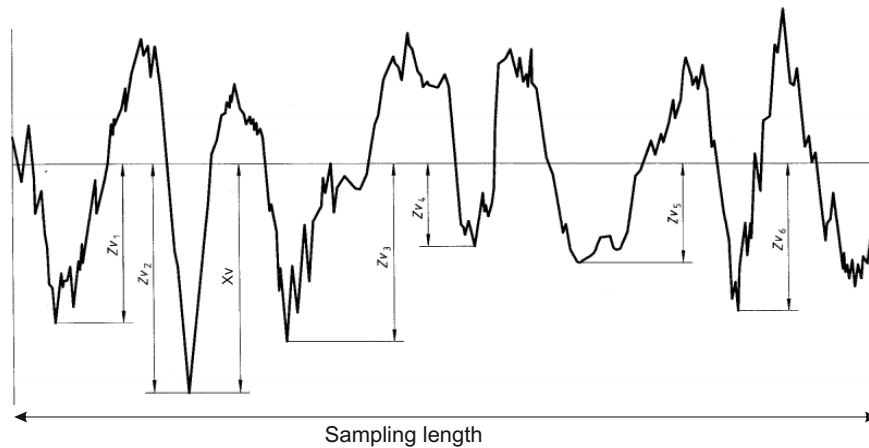


Figure 2.18: Valley depth of profile, from DIN EN ISO 4287

$$X_z = \frac{1}{n} (X_{z1} + X_{z2} + X_{z3} + \dots + X_{zn}) \quad (2.47)$$

where,

X_{zi} = is the distance between the highest peak and deepest valley depth within a sampling length.

Maximum height X_t : X_t is defined as the distance between the highest peak and the deepest valley of the surface profile z within the evaluation length.

$$X_t = \max(z(i)) - \min(z(i)) \quad (2.48)$$

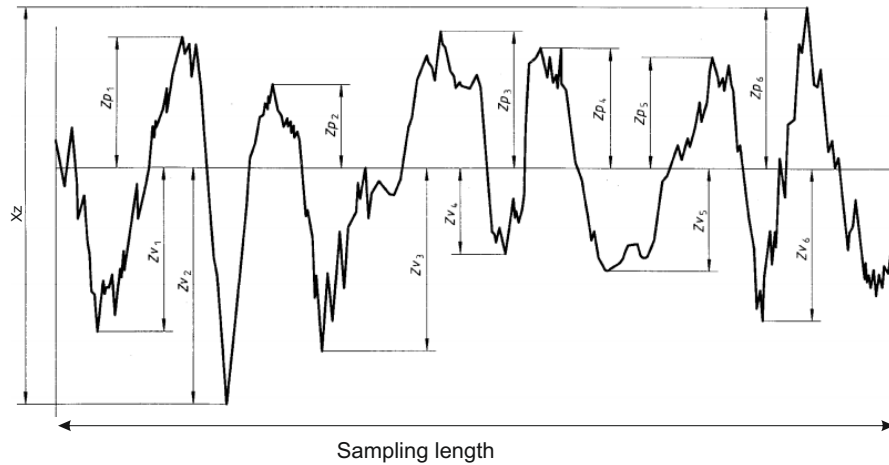


Figure 2.19: Maximum height of profile, from DIN EN ISO 4287

Maximum peak to valley depth X_{max} : X_{max} is defined as the largest peak to valley height of the surface profile z within the sampling length.

2.4.3 Spacing parameters

As amplitude parameters give information about the height and depth measures of the surface, the spacing parameters are used to describe the spacing between the peaks and valleys. If the peaks in the profile are close to each other, the surface material can sustain leaks.

Peak count X_c : X_c is defined as the number of profile peaks above a threshold value t_h within the evaluation length.

Mean width of profile X_{sm} : X_{sm} is defined as the arithmetic average of the width of the surface profile (see figure 2.20).

$$X_{sm} = \frac{1}{n} \sum_{i=1}^n X_s(i) \quad (2.49)$$

where n is the number of profile elements and $X_s(i)$ is the width of the i^{th} element within a sampling length.

2.4.4 Hybrid parameters

This sub-section describe two types of hybrid parameters, which are used in the software evaluation.

Skewness X_{sk} : X_{sk} is defined as a measure of the asymmetry of the profile.

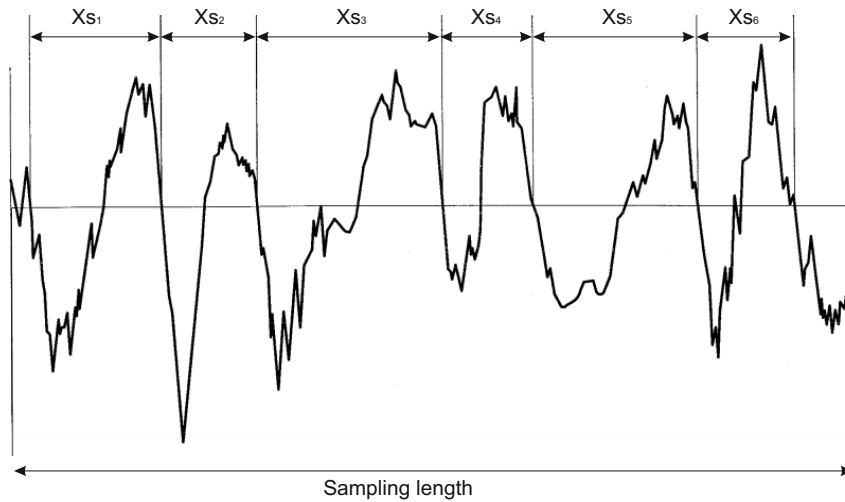


Figure 2.20: Mean width of profile, from DIN EN ISO 4287

$$X_{sk} = \frac{1}{nX_q^3} \sum_{i=1}^n z(i)^3 \quad (2.50)$$

Kurtosis X_{ku} : X_{ku} is defined as a measure of the peakness of the profile.

$$X_{ku} = \frac{1}{nX_q^4} \sum_{i=1}^n z(i)^4 \quad (2.51)$$

Note: Skewness and kurtosis are strongly influenced by peaks and valleys in the profile. Negative skew value indicates good bearing properties.

Arithmetical mean slope of the profile X_{da} : This expresses the arithmetic mean of the ordinate slopes dz/dx in a sampling length.

$$X_{da} = \frac{1}{n} \int_n \left| \frac{d}{dx} z(x) \right| dx \quad (2.52)$$

Root mean square slope of the profile X_{dq} : This expresses the root mean square of the ordinate slopes dz/dx in a sampling length.

$$X_{dq} = \sqrt{\frac{1}{n} \int_n \left(\frac{d}{dx} z(x) \right)^2 dx} \quad (2.53)$$

Arithmetical mean wavelength of the profile X_{Ia} : Is a parameter that relates the spacing between local peaks and valleys weighted by their individual frequencies and amplitudes.

$$X_{Ia} = 2\pi \frac{X_a}{X_{da}} \quad (2.54)$$

Root mean square wavelength of the profile X_{Iq} : Is a parameter that relates the root mean square spacing between local peaks and valleys weighted by their individual frequencies and amplitudes.

$$X_{Iq} = 2\pi \frac{X_q}{X_{dq}} \quad (2.55)$$

2.4.5 Abbott-Firestone curve parameters

The Abbott-Firestone curve is sometimes referred to several other names like the Material Ratio Curve and Bearing Area Curve (BAC). It calculates the ratio of air to material at any level starting from the highest peak down to the lowest valley depth [MGC01]. It is also called the Bearing Area Curve as it emphasizes the bearing surface. The Abbott Firestone curve describes the height distribution of the surface to be characterised. In general the waveform is made up of amplitude (height) and wavelength (spacing) features. Among these two the height information is relatively important as it seems to relate more readily to functional importance [Whi94]. For a Gaussian surface, the curve is said to have an S-shaped appearance [Bhu99].

Surface finish of machined parts is important in determining the quality of the surface. It defines the capacity of wear, lubrication, resistance to load etc. The imperfections or irregularities on the surfaces generally have the form of peaks and valleys of varying height and width. Rough surfaces wear more quickly than smooth surfaces [KMN⁺]. Roughness is difficult to be controlled during manufacturing, but investigations reveal that increasing the cutting speed can eventually improve the quality of the surface and in turn reduces the surface roughness [MBM97]. It plays an important role in industry to estimate the area in contact between two surfaces and to assess the rate of wear. The area in contact can be obtained from the Abbott curve.

Parameter according to DIN EN ISO 4287 [DIN09]:

The material ratio is calculated using equation 2.56. Figure 2.21 explains the parameters used in the equation. $L_1, L_2 \dots L_n$ are the distances between the peaks that is to be truncated at a cutting depth C . The cutting depth C is the distance between the evaluated intersection line and the specified reference line C_{ref} . The material ratio can also be evaluated on the primary and waviness profiles (P_{mr} and W_{mr}).

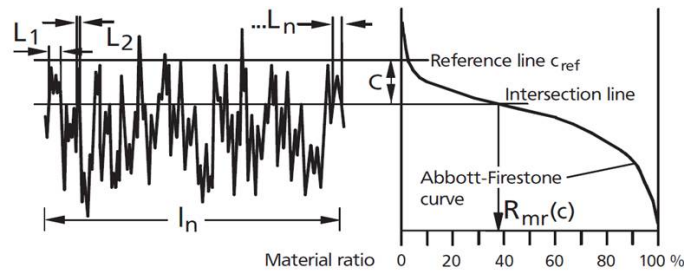


Figure 2.21: Abbott curve of a profile at a cutting depth C [Mahr]

$$R_{mr}(C) = \frac{(L_1 + L_2 + \dots + L_n)}{l_n} \cdot 100 \text{ in } \% \quad (2.56)$$

where,

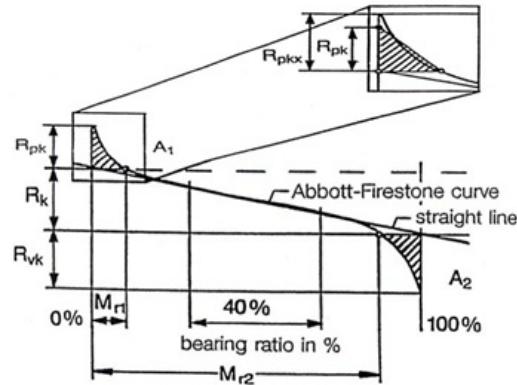
C = cutting depth

R_{mr} = material ratio

l_n = evaluation length

Parameters according to DIN EN ISO 13565:

The roughness profile is generated by special filtering techniques according to ISO 13565-1 [DIN98b]. The most widely used surface parameters R_a and R_z do not define the surface in detail, as different profiles can have the same average roughness. Hence more information about the surface can be obtained by the R_k parameters shown in table 3.1. A straight line approximately 40% divides the Abbott-Firestone curve into three regions as shown in figure 2.22 [MGC01]. Where, R_{pkx} is the total peak height. Once the material ratio is calculated and plotted, the parameters according to DIN EN ISO 13565-2 [DIN98c] are calculated. They are as shown in equation 2.57 and equation 2.58 [BH92].

Figure 2.22: Abbott curve with R_k parameters [Mahr]

$$A_1 = \frac{R_{pk} \cdot M_{r1}}{2 \cdot 100\%} \cdot 1000 \text{ in } \frac{\mu\text{m}^2}{\text{mm}} \quad (2.57)$$

$$A_2 = \frac{R_{vk} \cdot (100\% - M_{r2})}{2 \cdot 100\%} \cdot 1000 \text{ in } \frac{\mu\text{m}^2}{\text{mm}} \quad (2.58)$$

where,

A_1 = peak area

A_2 = valley area

A new algorithm called the sorting algorithm was proposed to characterise the surface roughness. It is described briefly in chapter 3 section 3.4. Due to ambiguous definitions in the standards, different implementation method can provide different results. Therefore there is a need for a standard reference software that can have unambiguous definition of parameters which can be compared with the results of software under test and the software used in the measuring instruments.

3 Corrigendum of the ISO filter

Chapter 2 section 2.3 gives an insight about the available filters and its benefits. This chapter focus more on the implementation of filters discussed in section 2.3. Depending on the surface type, one of these filters can be used to separate the waviness and roughness profiles from the primary profile. A new approach to calculate Abbott curve parameters from the roughness profile is discussed in section 3.4.

3.1 Gaussian profile filter

3.1.1 Gaussian profile filter for an open profile - Planar surface

Open profiles are profiles having finite length. They are measured only in one section of the surface. Therefore in such cases the evaluation length is reduced by the end effect region. The weighting function of Gaussian profile filter is shown in figure 3.1.

The weighting function of a Gaussian filter for an open profile is:

$$s(x) = \frac{1}{\alpha \cdot \lambda_c} \exp \left[-\pi \left(\frac{x}{\alpha \cdot \lambda_c} \right)^2 \right] \quad (3.1)$$

$$\alpha = \sqrt{\frac{\ln 2}{\pi}} \quad (3.2)$$

where,

$$\alpha = 0.4697$$

λ_c = the cutoff wavelength

x = the position in millimeters from the origin of the weighting function

Transmission characteristics of an open profile:

The transmission characteristics in figure 3.2 and figure 3.3 are determined by Fourier transform of the weighting function $S(x)$. The characteristics of long wave and short wave components are complementary to each other. They are as described below.

Transmission characteristics of the long wave component (mean line):

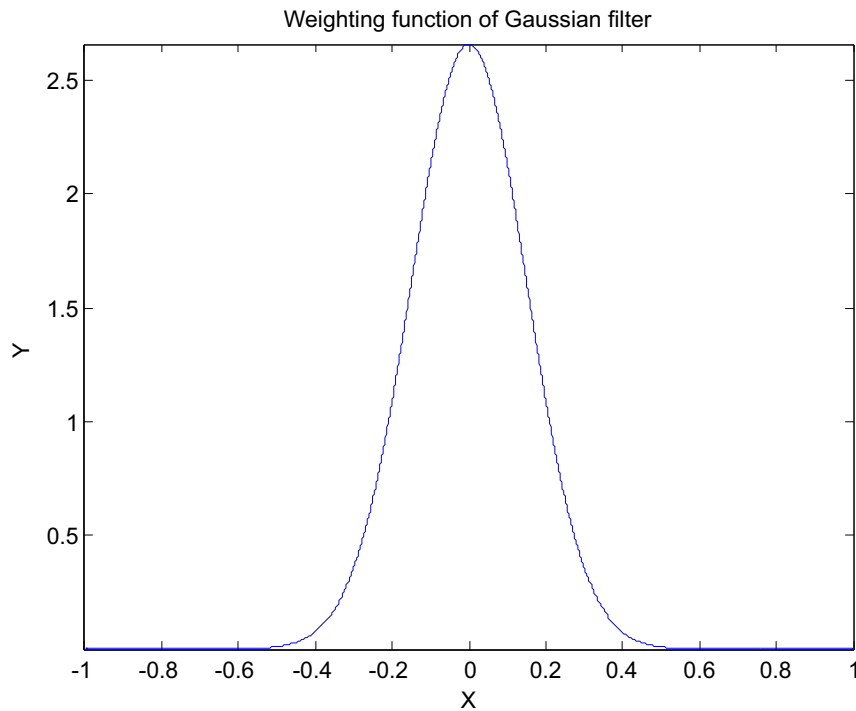


Figure 3.1: Weighting function of a Gaussian profile filter

$$\frac{a1}{a0} = \exp \left[-\pi \left(\frac{\alpha \cdot \lambda c}{\lambda} \right)^2 \right] \quad (3.3)$$

where,

$a0$ = amplitude of the profile before filtering

$a1$ = amplitude of the filtered profile

λ = wavelength of the profile

Transmission characteristics of short wave component:

$$\frac{a2}{a0} = 1 - \exp \left[-\pi \left(\frac{\alpha \cdot \lambda c}{\lambda} \right)^2 \right] \quad (3.4)$$

where,

$a2$ = amplitude of the filtered profile

$a0$ = amplitude of the profile before filtering

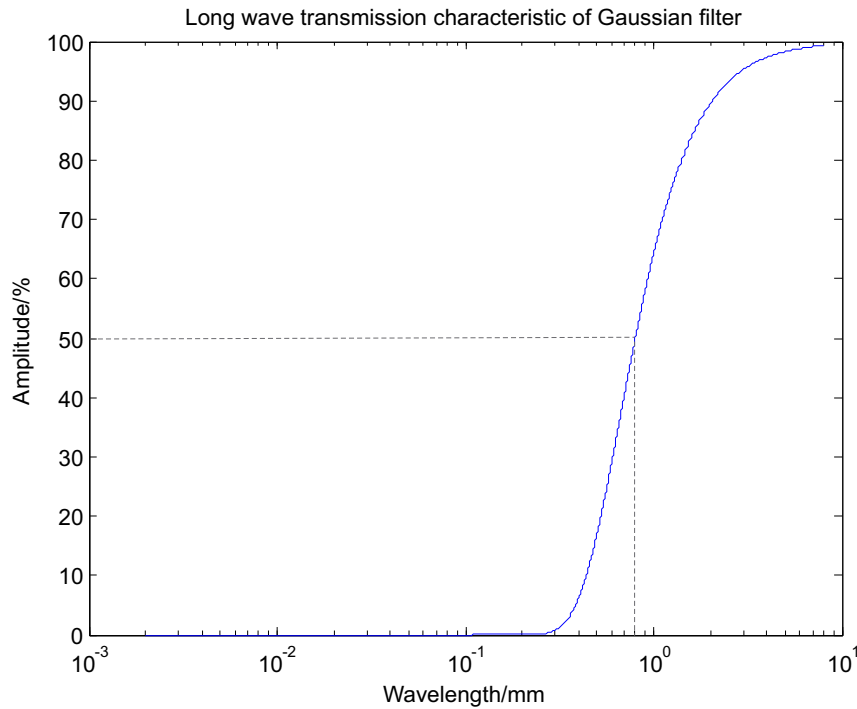


Figure 3.2: Long wave transmission characteristics of the Gaussian filter

λ = wavelength of the profile

3.1.2 Gaussian profile filter for a closed profile - Cylindrical surface

A closed profile is a profile with infinite evaluation length, for example roundness measurement. Its measured around the entire circumference of a surface. A Gaussian closed profile filter is not recommended if the length of the profile is less than $2\lambda_c$. The profile is too short to apply the Gaussian filter weighting function. However a closed profile with small diameter can be measured as an open profile of infinite length L as published by author Jörg Seewig and Matthias Eifler [SE14]. See section 3.1.3.

The weighting function of a Gaussian filter for a closed profile is:

$$s(x) = \begin{cases} \frac{f_c}{\alpha \cdot L} \cdot \exp \left[-\pi \left(\frac{x \cdot f_c}{\alpha \cdot L} \right)^2 \right] & \left(-\frac{L_c \cdot L}{f_c} \leq x \leq \frac{L_c \cdot L}{f_c} \right) \\ 0 & \text{Otherwise} \end{cases} \quad (3.5)$$

where,

x = distance along the closed profile from the center of the weighting function

f_c = cutoff frequency in undulations per revolution

L = length of closed profile

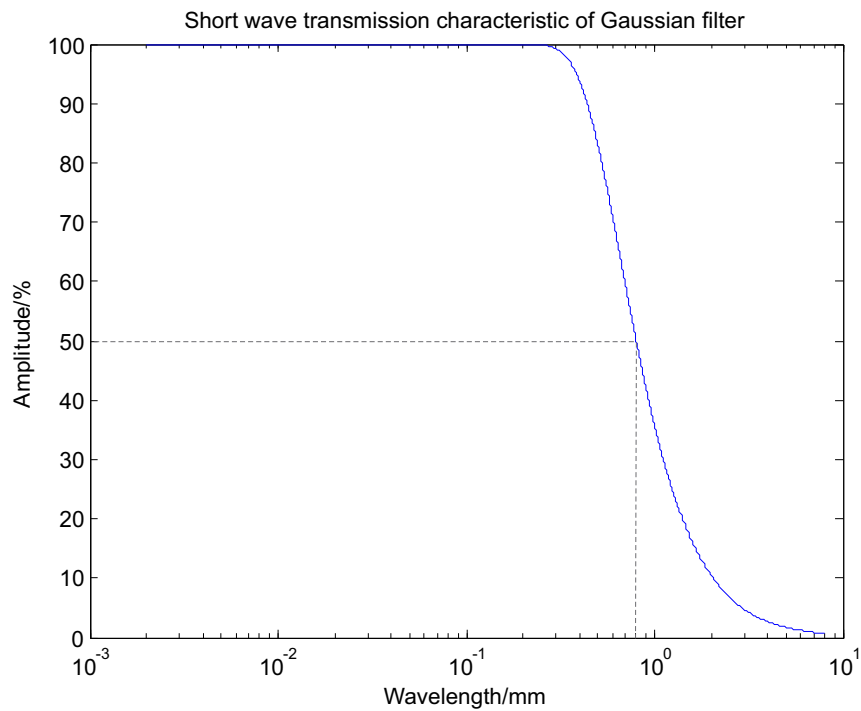


Figure 3.3: Short wave transmission characteristics of the Gaussian filter

L_c = truncation constant (refer [DIN12])

$\alpha = 0.4697$

Transmission characteristics of a closed profile:

The filter characteristics when $\lambda_c \ll L$ can be approximated as shown below.

Transmission characteristics of the long wave component (mean line):

$$\frac{a1}{a0} = \exp \left[-\pi \left(\frac{\alpha \cdot f}{f_c} \right)^2 \right] \quad (3.6)$$

where,

$a0$ = amplitude of the profile before filtering

$a1$ = amplitude of the filtered profile

f = frequency of the profile in undulations per revolution

Transmission characteristics of short wave component:

$$\frac{a_2}{a_0} = 1 - \exp \left[-\pi \left(\frac{\alpha \cdot f}{f_c} \right)^2 \right] \quad (3.7)$$

where,

a_2 = amplitude of the filtered profile

a_0 = amplitude of the profile before filtering

f = frequency of the profile in undulations per revolution

3.1.3 Periodic Gaussian profile filter for closed profiles

For closed profiles with short measurement area, it is nearly impossible to use the standard Gaussian filter with a high cutoff value. In such cases closed profiles can be measured as an open profile of infinite length. i.e. a closed profile with length L can be periodically represented as:

$$\tilde{z}(x) = \tilde{z}(x + k \cdot L) \quad (3.8)$$

where k is a random even number that adds a multiple of the period length L . And the convolution integral is expressed as:

$$w(x) = \sum_{k=-\infty}^{\infty} \int_0^L z(\xi + k \cdot L) \cdot s(x - (\xi + k \cdot L)) \cdot d\xi \quad (3.9)$$

For closed profiles equation 3.8 can be used into the convolution integral as:

$$w(x) = \sum_{k=-\infty}^{\infty} \int_0^L \tilde{z}(\xi) \cdot s(x - (\xi + k \cdot L)) \cdot d\xi \quad (3.10)$$

Exchanging the sum and the integral results in the periodic convolution integral:

$$w(x) = \int_0^L \tilde{z}(\xi) \cdot \sum_{k=-\infty}^{\infty} s(x - (\xi + k \cdot L)) \cdot d\xi \quad (3.11)$$

The new term of the weighting function is periodic and is defined as the periodic weighting function:

$$\tilde{s}(x) = \sum_{k=-\infty}^{\infty} s(x - k \cdot L) \quad (3.12)$$

The above periodic weighting function $\tilde{s}(x)$ can therefore be applied to closed profiles, whose length is less than $2\lambda_c$. And the periodic convolution integral can be rewritten as:

$$\tilde{w}(x) = \int_0^L \tilde{z}(\xi) \cdot \tilde{s}(x - \xi) \cdot d\xi \quad (3.13)$$

For more information regarding the principle of periodic convolution, refer [SE14]. To conclude, the filter mean line obtained using Gaussian filter has distortions at the ends of the profile as a result of the open profiles, and is not the case when it is applied to closed profiles. The roughness profile is obtained as the difference between primary profile and filter mean line obtained by the Gaussian filter (waviness profile). The problem of end effect is reduced by shortening the roughness profile in terms of the primary profile for a length of λ_c or $\lambda_c/2$ at both ends of the profile. Its a common practice to discard the start and end area of the filtered profile depending on manufacturers. In the proposed reference software the cut out area can be selected as per user convenience. The recommended cutoff values are 0.25 mm, 0.8 mm and 2.5 mm. The valid sampling distances are described in ISO 4288 [DIN98d]. However ISO 16610-28 specifies the area to be discarded. Sensitivity to deep valleys (porous surface) results the mean line to be pulled down towards the valley, this drawback led to a special filter standardised in ISO 13565-1.

3.2 Spline filter

A spline filter is a digital filter, it is defined by filter equations and not by weighting functions. For n datapoints (x_k, z_k) , a function $w(x)$ must be found that solves the following optimisation problem with $k=1, \dots, n$

$$\min_{w(x)} \left(\int_{x_1}^{x_n} \left(\frac{d^2 w(x)}{dx^2} \right)^2 dx \right), w(x_k) = z_k, k = 1, \dots, n \quad (3.14)$$

Solving this problem leads to cubic polynomial segments that pass continuously through the second derivative and directly through all the datapoints (x_k, z_k) . As

a result of the mechanical beam bending model, the second derivative of $w(x)$ is zero at the start ($x = x_1$) and end ($x = x_n$). This model gives rise to a mathematical variation error, where the spring c occurs as the Langrange parameter λ

$$\min_{w(x)} \left(\int_{x_1}^{x_n} \left(\frac{d^2 w(x)}{dx^2} \right)^2 dx + \lambda \sum_{k=1}^n (z_k - w(x_k))^2 \right) \quad (3.15)$$

The mathematical problem described in equation 3.15 is expanded such that the mechanical tension of the spline can be controlled by a special tension parameter β . The tension parameter plays a vital role for a spline filter. With a spline filter as in equation 3.15, negative values arise in the weighting function. This effect can be reduced by applying the tension parameter β [See13].

$$\min_{w(x)} \left((1-\beta)\alpha^4 \int_{x_1}^{x_n} \left(\frac{d^2 w(x)}{dx^2} \right)^2 dx + \beta\alpha^2 \int_{x_1}^{x_n} \left(\frac{dw(x)}{dx} \right)^2 dx + \sum_{k=1}^n (z_k - w(x_k))^2 \right) \quad (3.16)$$

Spline filter equations

Applying the mathematical variation problem given in equation 3.16 to equidistant sampled profile, leads to the linear spline filter described in ISO 16610-22. It is obtained by replacing the continuous derivatives with differential quotients. And the result is

$$J = (\beta\alpha^2) ((w_2 - w_1)^2 + (w_n - w_{n-1})^2) + (z_1 - w_1)^2 + (z_n - w_n)^2 + \sum_{k=2}^{n-1} ((1-\beta)\alpha^4 (w_{k+1} - 2w_k + w_{k-1})^2 + (\beta\alpha^2) (w_k - w_{k-1})^2 + (z_k - w_k)^2) \quad (3.17)$$

Spline filters are classified into non-periodic profile filter and periodic profile filter. Non-periodic spline filter is used to filter open profiles and the periodic spline filter to filter the closed profiles. Equation 3.17 demonstrates the filter behavior for open profiles.

Filter equation for the non-periodic spline filter (open profile):

$$[1 + \beta\alpha^2 P + (1-\beta)\alpha^4 Q] w = z \quad (3.18)$$

The regression function is spanned by the matrix

$$X_k = \begin{bmatrix} 1 & x_{1,k} & x_{1,k}^2 \\ \vdots & \vdots & \vdots \\ 1 & x_{n,k} & x_{n,k}^2 \end{bmatrix} \quad (3.27)$$

where,

$$x_{l,k} = (l - k) \cdot \Delta x,$$

$$l = 1, \dots, n$$

The space variant weighting function, S_k is given by:

$$S_k = \begin{bmatrix} s_{1,k} \cdot \delta_1 & 0 & \dots & 0 \\ 0 & s_{2,k} \cdot \delta_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & s_{n,k} \cdot \delta_n \end{bmatrix} \quad (3.28)$$

with the Gaussian function

$$s_{l,k} = \frac{1}{\gamma \cdot \lambda c} \cdot \exp\left(-\pi \left(\frac{x_{l,k}}{\gamma \cdot \lambda c}\right)^2\right), \quad l=1, \dots, n \quad (3.29)$$

and the parameter

$$\gamma = \sqrt{\frac{-1 - W\left(-\frac{1}{2 \cdot \exp(1)}\right)}{\pi}} \approx 0,7309 \quad (3.30)$$

The additional weights

$$\delta_l = \begin{cases} \left(1 - \left(\frac{z_l - w_l}{c}\right)^2\right)^2 & \text{for } |z_l - w_l| \leq c \\ 0 & \text{for } |z_l - w_l| > c \end{cases} \quad (3.31)$$

where,

$$l = 1, \dots, n$$

are derived from the biweight influence function with the parameter

$$c = \frac{3}{\sqrt{2} \cdot \operatorname{erf}^{-1}(0,5)} \cdot \operatorname{median}|z - w| \quad (3.32)$$

$$c \approx 4,4478 \cdot \operatorname{median}|z - w| \quad (3.33)$$

where,

$W(X)$ = Lambert W function

$\text{erf}^{-1}(x)$ = inverse error function

n = number of values in the profile

k = index of the profile ordinate $k=1, \dots, n$

z = vector of dimension n of the profile values before filtering

w = vector of dimension n of the profile values of the filter reference line

w_k = value of filter mean line at position k

λ_c = cutoff wavelength of the profile filter

Δx = sampling interval

c is equal to three times R_q of the surface roughness for Gaussian distributed profiles and is said to be the default case in ISO 16610-31 [Int10b]. Vector w gives the profile values of the long wave component (mean line) and the r gives the profile values of short wave component obtained by the difference vector $r = z - w$.

Filter equation of the robust Gaussian regression filter for closed profiles:

$$\tilde{w}_k = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \cdot (\tilde{X}_k^T \cdot \tilde{S}_k \cdot \tilde{X}_k)^{-1} \cdot \tilde{X}_k^T \cdot \tilde{S}_k \cdot \tilde{z} \quad (3.34)$$

The regression function is spanned by the matrix

$$\tilde{X}_k = \begin{bmatrix} 1 & \tilde{x}_{1,k} & \tilde{x}_{1,k}^2 \\ \vdots & \vdots & \vdots \\ 1 & \tilde{x}_{n,k} & \tilde{x}_{n,k}^2 \end{bmatrix} \quad (3.35)$$

with

$$\tilde{x}_{l,k} = ((l - k + \frac{n}{2}) \bmod n - \frac{n}{2}) \cdot \Delta x,$$

$$l = 1, \dots, n$$

The space variant weighting function, \tilde{S}_k is given by:

$$\tilde{S}_k = \begin{bmatrix} \tilde{s}_{1,k} \cdot \tilde{\delta}_1 & 0 & \dots & 0 \\ 0 & \tilde{s}_{2,k} \cdot \tilde{\delta}_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \tilde{s}_{n,k} \cdot \tilde{\delta}_n \end{bmatrix} \quad (3.36)$$

with the Gaussian function

$$\tilde{s}_{l,k} = \frac{1}{\gamma \cdot \lambda_c} \cdot \exp\left(-\pi \left(\frac{\tilde{x}_{l,k}}{\gamma \cdot \lambda_c}\right)^2\right), \quad l=1, \dots, n \quad (3.37)$$

and the parameter

$$\gamma = \sqrt{\frac{-1 - W\left(-\frac{1}{2 \cdot \exp(1)}\right)}{\pi}} \approx 0,7309 \quad (3.38)$$

The additional weights

$$\tilde{\delta}_l = \begin{cases} \left(1 - \left(\frac{\tilde{z}_l - \tilde{w}_l}{c}\right)^2\right)^2 & \text{for } |\tilde{z}_l - \tilde{w}_l| \leq c \\ 0 & \text{for } |\tilde{z}_l - \tilde{w}_l| > c \end{cases} \quad (3.39)$$

where,

$$l = 1, \dots, n$$

are derived from the biweight influence function with the parameter

$$\tilde{c} = \frac{3}{\sqrt{2} \cdot \operatorname{erf}^{-1}(0,5)} \cdot \operatorname{median}|\tilde{z} - \tilde{w}| \quad (3.40)$$

$$\tilde{c} \approx 4,4478 \cdot \operatorname{median}|\tilde{z} - \tilde{w}| \quad (3.41)$$

where,

$W(X)$ = Lambert W function

$\operatorname{erf}^{-1}(x)$ = inverse error function

n = number of values in the profile

k = index of the profile ordinate $k=1, \dots, n$

\tilde{z} = vector of dimension n of the profile values before filtering

\tilde{w} = vector of dimension n of the profile values of the filter reference line

\tilde{w}_k = value of filter mean line at position k

λ_c = cutoff wavelength of the profile filter

Δx = sampling interval

\tilde{c} is equal to three times R_q of the surface roughness for Gaussian distributed profiles. Vector \tilde{w} gives the profile values of the long wave component (mean line) and the \tilde{r} gives the profile values of short wave component obtained by the difference vector $\tilde{r} = \tilde{z} - \tilde{w}$.

For surfaces with sharp peaks and deep valleys at the boundaries, the robustness can be increased by setting $p = 0$. In this case, the nominal form is eliminated by prefiltering technique [Int10b].

The filter equation for $p = 0$:

$$w_k = (X_k^T \cdot S_k \cdot X_k)^{-1} \cdot X_k^T \cdot S_k \cdot z \quad (3.42)$$

$$w_k = \left(\sum_{l=1}^n s_{l,k} \cdot \delta_l \right)^{-1} \cdot \sum_{l=1}^n (s_{l,k} \cdot \delta_l \cdot z_l) \quad (3.43)$$

where,

$$X_k = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}, \quad \gamma = \sqrt{\frac{\ln 2}{\pi}} \quad (3.44)$$

The Gaussian regression filter of degree 0 and degree 1 are identical, however for surface roughness measurement it is recommended to use filter of degree 1 according to author Seewig [See05].

3.4 Sorting algorithm

It is an approach to evaluate Abbott curve parameters. According to this algorithm, we consider spline interpolation of datapoints. These datapoints are sorted in descending order. At each of these datapoints, the zero crossings are detected and the material ratios are calculated according to DIN EN ISO 4287. Based on the material ratios calculated, a smooth S-shaped curve as shown in right side of figure 3.4 is obtained.

Let $\Delta x = l/N$ be the sampling distance, where l is the given length and N is the number of samples. Each sampled profile value is expressed as

$$z_i = z(x_i) \quad (3.45)$$

with $x_i = i\Delta x$ and $i = 0, 1, 2, \dots, N-1$

The material ratio curve is given by:

$$\left(\frac{k}{N} 100\%, c_k \right), k = 1, 2, \dots, N \quad (3.46)$$

where,

N = number of samples

c_k = the uniformly sampled profile values sorted in descending order

At each level, starting from the highest peak to the lowest valley, the percentage of the total profile to be truncated is calculated and plotted as a curve. At the highest

peak the percentage of the profile ($\% R_{mr}$) is relatively low. As it approaches the level of dominant valleys, however the percentage of the profile approaches one hundred percent. From figure 3.4 its clear that the Abbott-Firestone curve obtained using the sorting algorithm gives a smooth S-shaped curve. From this smooth curve, the surface parameters according to DIN EN ISO 13565 are calculated. These parameters calculated give summary of information contained in the Abbott-Firestone curve.

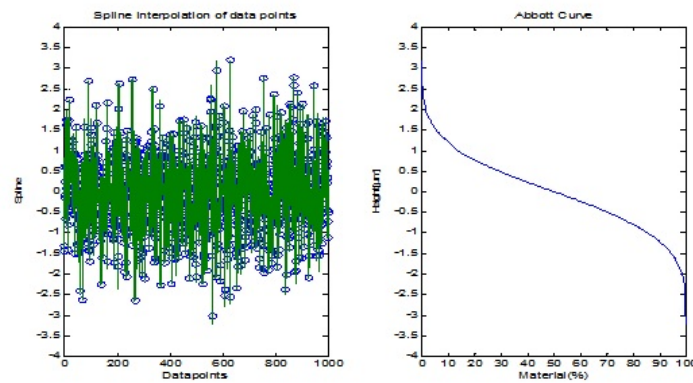


Figure 3.4: Abbott curve using sorting algorithm for random datapoints

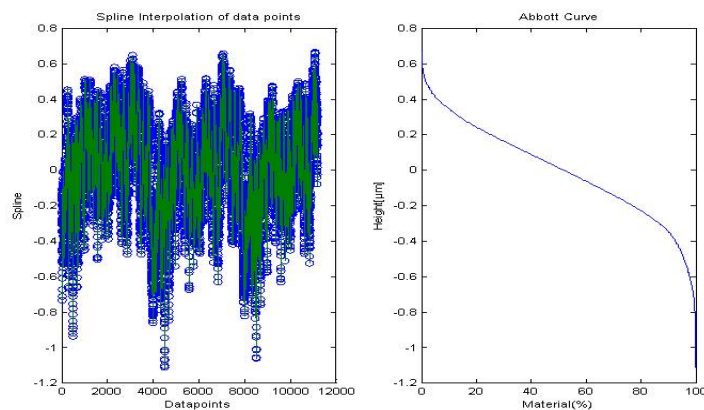


Figure 3.5: Abbott curve of a reference measurement with 11200 points

Evaluations were carried out over several datapoints to check the behavior of the curve. Figure 3.5 shows the response of a reference measurement with 11200 datapoints. Depending on the number of datapoints and its distribution the curve gets steeper and smoother. A Straight line, approximately 40% divides the curve into three segments. From these segments the peak and valley area are calculated. Once the area is known, the R_k parameters can be evaluated. The R_k parameters are used to describe the curve and its characteristics. The parameters R_{pk} and R_{vk} are each calculated as the height of the right angle triangle which is constructed to

have the same area as the “peak area” or “valley area”. The right angle triangle corresponding to the “peak area” has M_{r1} as its base, and that corresponding to “valley area” has the difference between 100% and M_{r2} as its base. The R_k parameters should only be calculated if the material ratio curve is “S” shaped as shown in figure 3.4 and 3.5. Experience has shown that this is always the case of lapped, ground or honed surfaces [Pri15]. Table 3.1 lists the available R_k parameters.

Table 3.1: The R_k parameters

R_k	Core roughness depth
R_{pk}	Reduces peak height
R_{vk}	Reduced valley depth
M_{r1}	Material ratio relative to peaks
M_{r2}	Material ratio relative to valleys

4 Measurement uncertainty

A measurement indicates something about the quality of surface being measured. For example the wear of an object, the load it can sustain, friction and so on. Every measurement give rise to some kind of uncertainty. No matter how careful the measurement is taken, it is always subjected to some kind of uncertainty. They can arise from the measuring instrument, environment, surface itself or from any other source. These uncertainties cannot be avoided. It must not be confused with the errors, while error is the difference in the measured value and the actual value of the object being measured [Bel01]. Errors are to be fixed when known. The result of a measurement is complete only when it is stated with its associated uncertainty. For each measurement the expression of the result and data evaluation must be clearly defined. One best way to express uncertainty is to repeat measurements several time to get the best estimate by statistical calculation (i.e. by mean and standard deviation).

In surface metrology, the uncertainty of roughness parameters are used as a quantitative measure of accuracy. They are dependent on the manufacturing process and has strong effect on an input parameter on determining the roughness parameters. Repeating a measurement several time and plotting the frequency with which a value x occurs as a function of x gives a bell shaped curve which can be approximated in many cases by normal distribution. To obtain an accurate value it is important to calculate the mean and standard deviation for a set of infinite values. The mean gives an estimate to the accurate value and the standard deviation tells the uncertainty of a measurement. The uncertainty in any measurement can either be random or systematic. Random errors are due to random results when repeating the measurement several time. Taking the mean of the results would result in a better estimate of accurate value. Whereas, systematic errors are errors that should be detected and eliminated by applying suitable corrections.

There are two ways to estimate uncertainties, Type A and Type B evaluations. Type A evaluations are uncertainty estimates using statistics. i.e. by repeated readings. Whereas, Type B evaluations are uncertainty estimates from experiences of measurement, from specifications and calibration certificates [Bel01].

4.1 Uncertainty

Chapter 2 discussed about various kinds of measurement instruments used to obtain reference datapoints, how they are filtered and how parameters are evaluated

according to ISO standards. But it is extremely rare to find the measurement uncertainty in parameters evaluated. This chapter does not focus in depth about the uncertainty analysis but puts an insight about uncertainty and how it can be reduced or eliminated to obtain true value of parameters.

In surface measurement, profiles are usually measured over an evaluation length of 4mm . It is then filtered with long cutoff wavelength (λ_c) and short cutoff wavelength (λ_s). The nominal filter cutoff λ_c is 0.8mm and λ_s is $2.5\mu\text{m}$. These measurement conditions are default in roughness measurements. To simplify the arithmetic, it is assumed that the roughness values are uncorrelated by profile filtering. The surface parameters are then evaluated. The parameters R_a , R_q , R_t and R_z are inspected according to DIN EN ISO 4287 and the parameters R_k , R_{pk} and R_{vk} are inspected according to DIN EN ISO 13565-2.

The profile used in chapter 2 (see figure 2.4) is measured from the workpiece KNT 4070/03 (S3) roughness measurement standard using stylus instrument T8000 at the laboratory of Measurement and Sensor Technology (MTS), T.U. Kaiserslautern. Figure 2.4 shows traces on a super fine surface with 3000 datapoints. Most cases the surface to be measured is tilted relative to the measuring device, hence the underlying geometry must be removed. Before any parameter can be evaluated, a mean line has to be fitted to eliminate the geometry from the surface. The most commonly used default method is the least square method.

The equation of a straight line is given by:

$$z = a + bx \quad (4.1)$$

where,

a = intercept

b = gradient

The uncertainties in a and b propagates through the parameters evaluated hereafter, hence its really important to consider the uncertainties right from this point. The uncertainties in a and b are [Lea02]:

$$u(a) = \sqrt{\frac{S_{xx}}{SS_{xx} - S_x^2}} \quad (4.2)$$

$$u(b) = \sqrt{\frac{S}{SS_{xx} - S_x^2}} \quad (4.3)$$

Once the least square fit is eliminated from the data set, we obtain

$$z' = z_i - (a + bx_i) \quad (4.4)$$

An uncertainty in z' is given by

$$U^2(z') = U^2(z) + U^2(a) + x^2 U^2(b) + b^2 U^2(x) \quad (4.5)$$

Once the basic conditions like fitting and filtering is applied, the roughness parameters like average roughness, root mean square roughness, maximum height of the profile, maximum peak height and valley depth are evaluated followed by the Abbott curve parameters. The next step is to consider the uncertainties in the parameters evaluated such that the uncertainties in the parameters are clearly defined. A clear definition of parameter play a vital role in quality control.

4.2 Uncertainty of roughness parameters

Roughness average R_a : R_a is defined as the arithmetic average of the absolute values of the profile z measured from the mean line.

$$R_a = \frac{1}{n} \sum_{i=1}^n |r_k| \quad (4.6)$$

$$U^2(R_a) = \frac{1}{n} \sum_{i=1}^n U^2(r_k) \quad (4.7)$$

Root mean square roughness R_q : R_q is defined as the root mean square average of the absolute values of the profile z measured from the mean line.

$$R_q = \sqrt{\frac{1}{n} \sum_{i=1}^n r_k^2} \quad (4.8)$$

$$U^2(R_q) = \frac{1}{n} \frac{1}{R_q} \sum_{i=1}^n z(i)^2 U^2(r_k) \quad (4.9)$$

Maximum height R_t : R_t is defined as the distance between the highest peak and the deepest valley of the surface profile z within the evaluation length.

$$R_t = \max(r_k) - \min(r_k) \quad (4.10)$$

R_t cannot be realized with the simple deviation propagation method. It reacts sensitively to outliers and is specially used to detect cracks in the boundary layer [See96]. The uncertainty of roughness parameters by author J. Seewig elaborates how roughness parameters dependent on the manufacturing process can be described with probability density functions (refer [See96]).

Abbott curve parameters R_{pk} , R_{vk} : When considering the uncertainty in Abbott curve parameters (see 3.4), it is important to keep in mind that sorting the values in descending order leads to high uncertainty around the corners of the curve thereby giving rise to uncertainty in parameters R_{pk} and R_{vk} substantially. The standard deviation of the parameters R_{pk} and R_{vk} is as shown in equations 4.11 and 4.12 [See96].

$$\sigma_{Rpk} = \frac{2}{n \cdot M_{r1}} \sqrt{\sum_{k=1}^{n \cdot M_{r1}} \sigma_k^2 + 2 \sum_{k=1}^{n \cdot M_{r1} - 1} \sum_{j=k+1}^{n \cdot M_{r1}} \sigma_{k,j}^2} \quad (4.11)$$

$$\sigma_{Rvk} = \frac{2}{1 - n \cdot M_{r2}} \sqrt{\sum_{k=n \cdot M_{r2}}^n \sigma_k^2 + 2 \sum_{k=n \cdot M_{r2}}^{n-1} \sum_{j=k+1}^n \sigma_{k,j}^2} \quad (4.12)$$

As already discussed, this chapter is only an insight to uncertainty in measurement and does not discuss the analysis in detail. Detailed study of uncertainty in measurement is to be considered in the future.

5 Reference software

The motive behind developing such a reference software for surface characterisation is to avail it to all the clients around the globe for free of cost. This chapter is divided into three main sections. Section 5.1 of this chapter describes the patterns and frameworks used in developing the reference software, section 5.2 details the actual implementation of the reference software and the last section (section 5.3) describes the Java and C++ interoperations.

5.1 Patterns and frameworks

5.1.1 Web-applications

This sub-section describes the general structure and principles of a web application. Web applications can be conceptually divided into two parts, the presentation layer and the business logic. The presentation layer on a browser based application relates to the look of the application. On the other hand the business logic determines the behavior of the application and is implemented in the Java programming language in this work. Figure 5.1 shows a general model of a web application, where the server stores persistent data in the database.

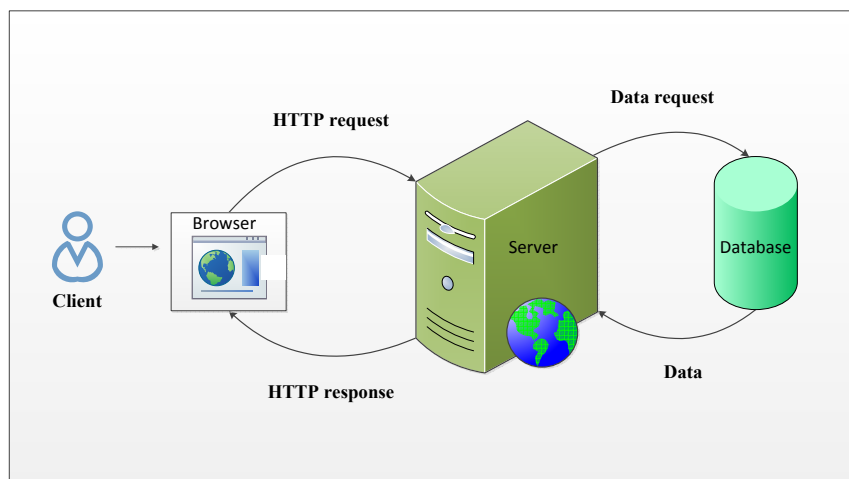


Figure 5.1: Web application model

Client-Server application: In a web application there could be several programs running at the same time. The codes at the client side (browser) responds to the

user and the codes at the server side responds to HTTP (Hypertext Transport Protocol) requests. Client side scripts are written in scripting languages like JavaScript, CSS and the pages HTML elements. On the other hand any language or framework that responds to the HTTP request can run a server. The HTTP protocol is used for communicating between the client and the server. The commonly used HTTP methods are the GET, POST and DELETE methods.

5.1.2 Programming languages

Java: Java is a high-level programming language developed by James Gosling and colleagues at Sun Microsystems Inc. in 1991. It is an object-oriented language with lot of features that make the language well suited for web development. The Java Development Kit (JDK) is a program development environment. It includes runtime environment and tools required to compile, debug and run applications written in Java. Java development kit version 1.7 (32 bit) is used in this research work with Eclipse as IDE (Integrated Development Environment) for Java EE developers (32 bit).

Java Native Interface (JNI): Java Native Interface enables the integration of Java code with native code written in C or C++. JNI directs a Java program to call a function in C or C++ and allows a C or C++ program to call a method in a Java program. It defines a mapping between the Java code types and the native code types. Figure 5.2 shows the interoperation of Java code with C++ using JNI.

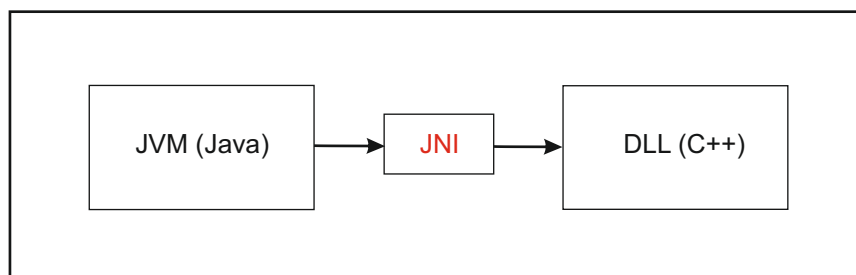


Figure 5.2: Interoperation using JNI

HTML, CSS: HTML is a markup language designed for defining static webpage content. An HTML file is placed on a server and when requested by a user, it is read into the users browser and displayed as a webpage. It is considered to be the building block for creating websites. Things that involve the presentation of a website are described by CSS (Cascading Style Sheets). A CSS file separates the webpage HTML content from its style. [Duc11]

JavaScript: JavaScript is an object-oriented scripting language. It supports parsing results from the server and bounds the resulting data on to the webpage by updating an elements property, or by changing elements CSS properties. They are generally used to validate data over webpage.

Client side JavaScript provides objects to control a browser. It can be connected to the objects of its environment to provide control over them.

Server side JavaScript provides objects relevant to run JavaScript on a server. It allows an application to communicate with a database and performs file manipulations on a server. [Fla11]

JSON: JSON stands for JavaScript Object Notation. It is a data interchange format. For AJAX applications writing, fetching and parsing of JSON data is easy and efficient when compared to that of XML, hence most developers rely on JSON. It can structure data with a string, number, Boolean, array, object or a null value. It is commonly used to read data from a web server and display it on a webpage.

5.1.3 Software patterns and frameworks

This sub-section explains the software patterns and frameworks used to structure the software. Software patterns are the general concepts for software development.

AJAX: AJAX stands for "Asynchronous JavaScript and XML". Ajax is initiated by JavaScript and is handled asynchronously in the background. Once the Ajax interaction is complete, the changes are updated by JavaScript without reloading the entire webpage. Hence AJAX is commonly used to create interactive web applications. The technologies that help build an AJAX web application include HTML, CSS, DOM and JavaScript. DOM stands for "Document Object Model". It is an object-oriented representation of HTML documents that provides support for modifying documents.

JQuery: JQuery is a framework which helps to design web applications. It work easily with the JSON format. It has an AJAX method to fetch data from the server without reloading the entire webpage. JQuery helps in generating JSON formatted data, communicating with the server and to parse JSON data. It abstracts the browser JavaScript functionality to make it easier to write cross browser compatible JavaScript code.

MVVM: The MVVM (Model View ViewModel) design pattern separates the presentation layer from the business logic. It splits the user interface code into Model, View and ViewModel. In this work, it is used for the implementation of the client (browser). *Model* - Model represents data formatted in JSON received from or sent to the

server.

View - Views are the visual representation of the data (HTML, CSS). It displays the data or receives the user input and forwards it to the ViewModel.

ViewModel - ViewModel wraps the data from the Model and makes it available for being presented or modified by the View. Its role is to control the Views interaction with the application. [And12]

KnockoutJS: It is a framework used for the development of advanced User Interface (UI) on webpages. It is a JavaScript library designed to develop MVVM applications in HTML, CSS and JavaScript. It helps with the interaction between the View and the ViewModel through data binding. A block diagram of the Model-View-ViewModel design pattern using KnockoutJS is shown in figure 5.3.

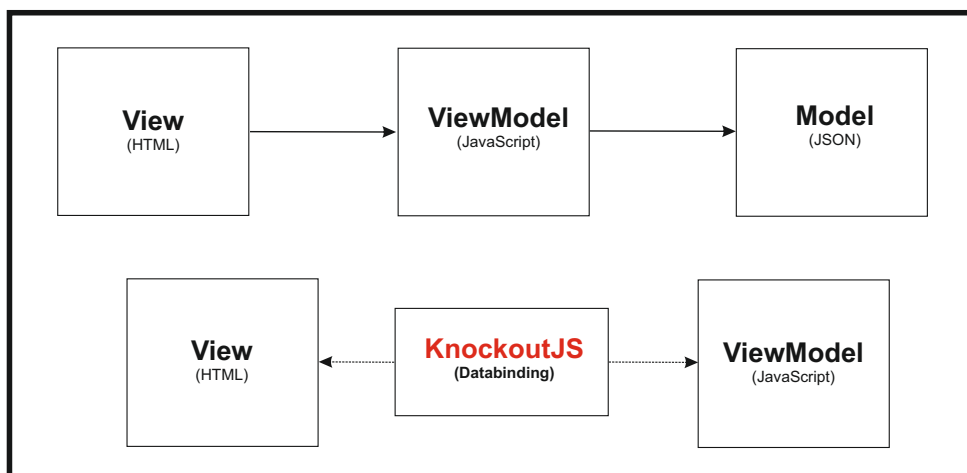


Figure 5.3: Model-View-ViewModel using KnockoutJS

MVC: The business layer of a web application is generally implemented on the server side using the MVC (Model View Controller) design pattern. This pattern has a lot in common with the MVVM design pattern, they both have a View and Model concept. At the server side development the Views represent the whole webpage with HTML, CSS and JavaScript. So it also represents elements of the MVVM pattern. The server side responds to the HTTP requests and organizes the flow of the application in a manner desired by the application.

Model - The data and functions which the application handles, load and store are managed by the Model.

View - View renders and generates a webpage using HTML, CSS or JavaScript. The Views are then sent to the web browsers as HTTP response.

Controller - Controller responds to the HTTP requests made by the client side.

Since there can be several requests made by the users, the requests are processed by a router. The router analyses the request and forwards the request to the appropriate action controlled by the Controller. Depending on the action, the Controller fetches the data from the Model and forwards it to the View.

Play Framework: It is a Java framework which can be used to implement a web application with the use of the MVC design pattern on the server. The Play framework 2.04 is used in this research work to build web application with Java. It is based on a lightweight, web friendly architecture. Plays "new" command is used to create a new project, and loads it with series of files and directories required for the application. It detects the changes made to the Java source file and automatically reloads them at runtime. Once the web application is developed using Play framework, it is important to deploy the application as a Windows service. Additional information can be found in appendix B.3. Installation and documentation of play framework is available on their official website [Bor07].

JFreeChart: JFreeChart framework is used to display charts as image in the application. It is a free Java chart library that can be used for charting (plotting). It is important to place the required JAR (Java Archive File) files into the projects library path. i.e. *lib* folder of the project. Special features of this framework are exporting the charts as image (PNG or JPEG format), exporting the charts as PDF files and other features like zooming and panning [Gil13].

FOP: The FOP (Formatting Objects Processor) framework is used to generate PDF documents. It is a Java application that reads a formatting object tree as input and renders the resulting pages as output (PDF). It relies on the use of XML and XSL to create a PDF document. This framework is managed by the Apache Software Foundation [Tau11].

5.2 Implementation

This section describes the implementation of the reference software, which was carried out with the use of the patterns and frameworks described in section 5.1. The whole structure of the web application is described with the focus on the communication between the separated parts of the web application. This will be followed by the explanation of the user interface implemented by the web client (see section 5.2.1). Section 5.2.2 details through the organisation of the server side implementation. The last part of the chapter (see section 5.3) explains the interoperation of Java with the C++ implemented algorithms.

The architecture divides the web application model into a client side architecture and a server side architecture. The client side architecture is running on a web browser as a website and the server side runs on a web server. The communication between the two is via HTTP protocol and is based on AJAX and JSON with the help of the JQuery framework. The server side communication is the Java code running on the JVM (Java Virtual Machine) as organized by the MVC design pattern. The implementation of the algorithms and the IO interfaces used for this reference software is mainly encapsulated in DLLs programmed in C++. The communication between the Java parts of the web server and the DLLs (Dynamic Link Library) is implemented with the JNI framework. In the client side, the communication between the user interface (View) and the JavaScript parts of the web (ViewModel) are based on the MVVM pattern with KnockoutJS framework. Figure 5.4 gives a short overview about the communication between the client and server in the software.

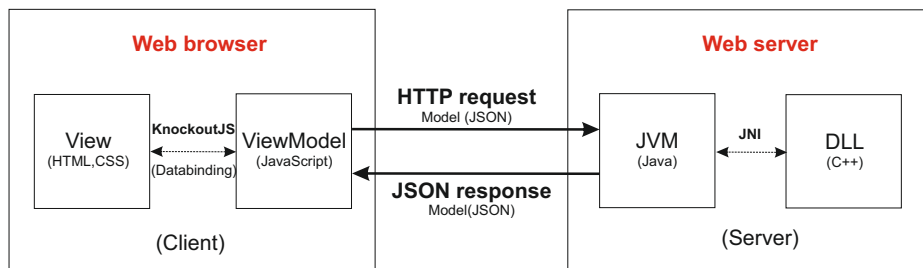


Figure 5.4: Communication between the client and server

5.2.1 Client implementation

The client side web development focus on to the structural design of the user interface and its implementation. The client of the reference software contains all the HTML, CSS and JavaScripts which are required for the user interface. It contains a main page, which holds all parts of the user interface and is fetched as soon as the user visits the reference software. As we proceed further with the evaluation process, the website is updated with JavaScripts and will not be replaced. Therefore the data is exchanged between the client and the server by implementing the AJAX concept. The architecture of the client is designed using the MVVM pattern to keep the implementation of the AJAX concept maintainable for the web developer. The main page of the reference software is as shown in figure 5.5. It can be accessed via "mtspub.mv.uni-kl.de:9000". On the main page, the user has the possibility to register and create a user ID and password for future login or can access the webpage as a guest user. As a guest user, the data used for evaluation will be automatically

deleted at the end of process or after 48 hours depending upon the default settings. As a registered user, one can login anytime and access the data files stored.

Structure of the user interface The user interface is structured in evaluation steps as shown in the progress bar of figure 5.5. The user can view the current step in process and the steps to be performed ahead. The user can at any time return to the previous step to alter the profile settings. Once the user proceed to the next step, the current process is highlighted in blue, the previous steps are highlighted in green and the step to be followed next in yellow. The evaluation steps carried out are described below.

Step 1: To begin with, the client first creates or loads a series of surface measurements. Once the user has entered a name for a new series of measurements, a new measurement series is created. Alternatively, user can also load a saved measurement series. Once the measurement series is created or loaded, the client is automatically directed to the next step.

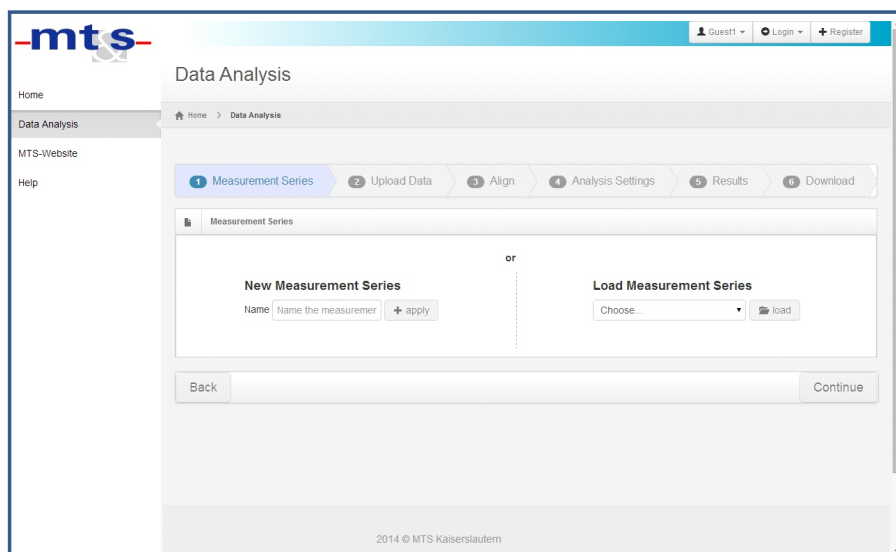


Figure 5.5: Step 1 - Main page of reference software

Step 2: Here the client has the possibility to upload the data files for further analysis. This step opens a file selection dialog box, where multiple data files can be selected. If the file extension of the selected data files are of same format, the files will be uploaded immediately. However, if there are files with different file formats, then the user must first select the appropriate file format. The file formats used in the proposed software are .smd, .x3p, .sdf, .asc, and .prf files. Once the files are uploaded, the client can proceed to the next step which is alignment of the measurement data. The first uploaded file will be used as default for further evaluation.

However, the client can choose a different data file from the measurement file list (at the upper right corner of the page) for evaluation, based on which the intermediate results are presented. Step 2 is shown in figure 5.6.

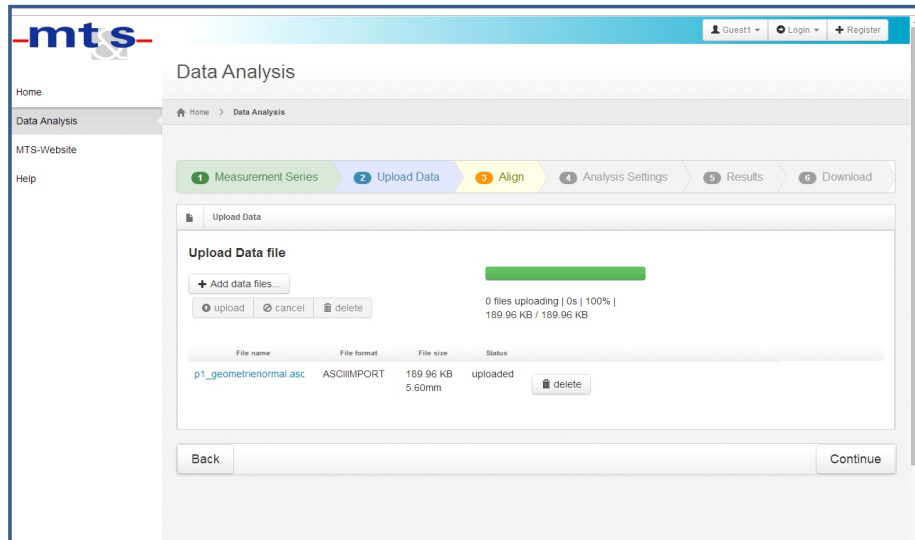


Figure 5.6: Step 2 - Creating or loading new measurement series

Step 3: Alignment of the profile using compensating elements like straight line fit and circular fit are displayed in a selection box. Straight line fit of the profile minimizes the sum of the squares of the distance in y-axis using Least Square method, minimizes the sum of the square distance along x-axis and y-axis using Total Least Square method and circular fit using Gaussian iteration method. Straight line fit using least square method is applied as default in the software. The alignment using the compensation element is confirmed by pressing a button after each selection. The profile before and after alignment is then displayed as charts on the right side of the screen. If alignment of the profile is not necessary, the user can proceed to the next step without selecting any compensation element. Step 3 is shown in figure 5.7.

Step 4: This step deals with the analysis setting of the profile. The client can choose the filter of choice and the sampling length L_C . There is possibility to select one of the given standard sampling length or to enter a value of choice. The value entered will be verified to satisfy the ISO standards and accordingly accepted or rejected with a warning. The profile can then be L_S filtered or ignored based on users choice. Once the filter settings are set, the profile charts are updated based on the changes made by the client. The chart shows the primary profile (P-profile) in blue and the waviness profile (W-profile) in red at the top followed by the roughness profile (R-profile) at

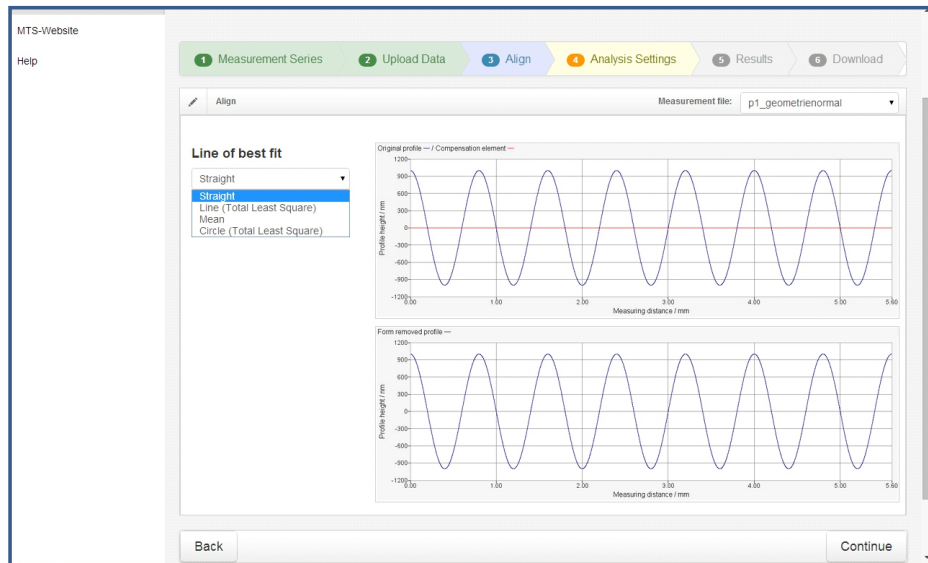


Figure 5.7: Step 3 - Aligning profile with compensation element

the bottom. For R-profile the client can control the graphical settings. i.e. the client can select the R-profile for the entire evaluation length or for a number of sampling lengths. The default number of sampling length is set to five, i.e. the evaluation length is equal to five times the sampling length. Where sampling length is equal to the cutoff wavelength L_C . Figure 5.8 refers to step 4 of the software analysis. After the client has set the profile filter according to the ISO standards (ISO 16610 series and ISO 13565-1), the surface parameters are ready to be evaluated.

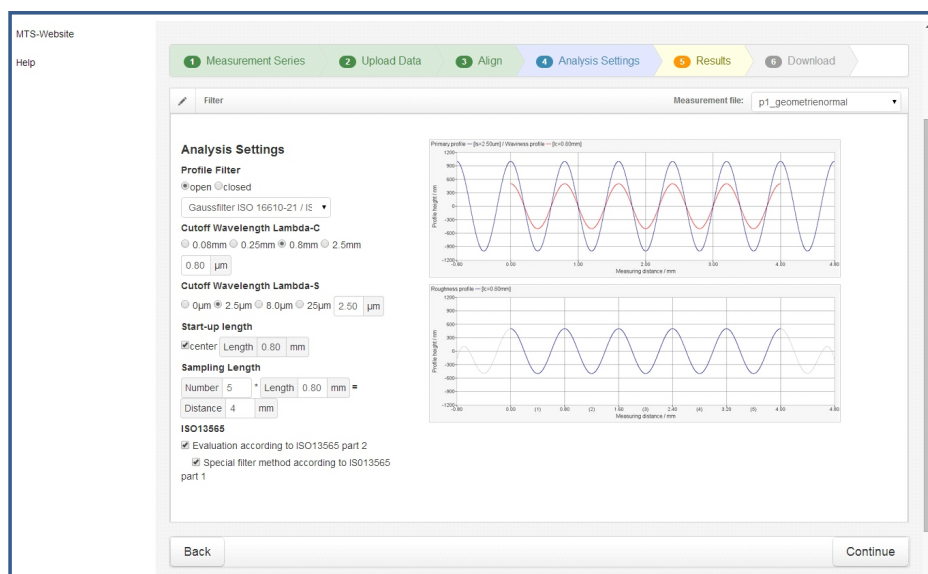


Figure 5.8: Step 4 - Filter analysis setting (Gaussian filter, $L_C=0.8\text{mm}$, $L_S=2.5\mu\text{m}$)

Step 5: This step displays the P, W and R-profiles on the right and the material ratio curves (Abbott curve) for the W and R-profile on the left side of the chart. The settings made by the user for evaluating the profile are displayed above the profiles for reference. Figure 5.9 displays the profiles of the software analysis. The parameters of the profile evaluated according to ISO 4287 and the roughness parameters accordingly to ISO 13565 part 2 are calculated and listed in table format below the profiles. The parameters calculated in this step is displayed in figure 5.10 and figure 5.11. Figure 5.10 refers to the amplitude, hybrid and spacing parameters evaluated according to ISO 4287 and figure 5.11 refers to the Abbott curve parameters evaluated according to ISO 13565-2 in the reference software.

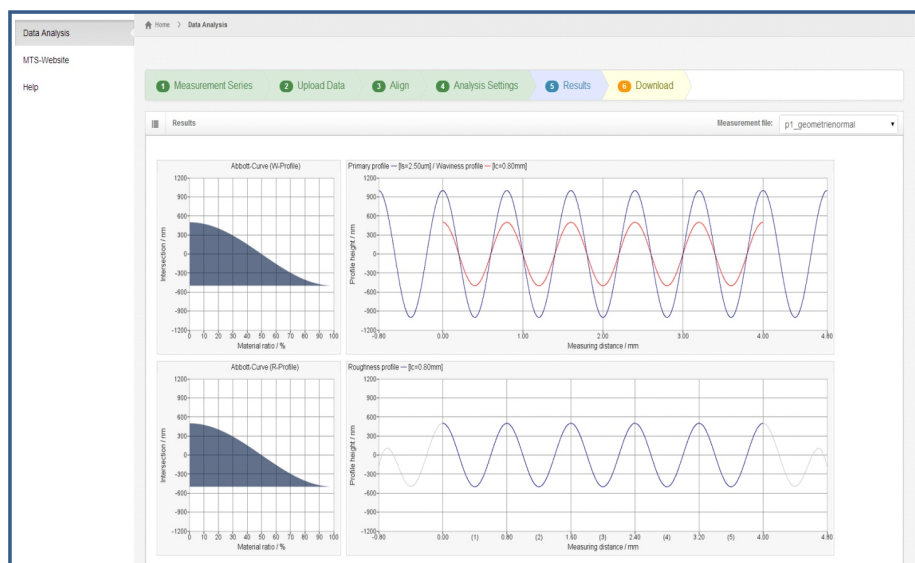


Figure 5.9: Step 5 - (a) Material ratio curve

Step 6: The last step of the evaluation process is to download the analysis results. Figure 5.12 displays the results page of the software. The client can select the results to be downloaded as PDF-file or save the individual profiles into specific file formats. It is also possible to store the results into a compressed file.

Use of the MVVM pattern The implementation of the user interface rely on the MVVM pattern described in the introduction of this chapter. Section 5.2.1 discussed the user interface as seen by the user. In the MVVM pattern the user can see the structure and the design as described by the Views. The Models are the data exchanged between the server and the client in JSON format. A detailed description of Model is not written in this chapter as they are not the major part to the understanding of the application. The ViewModels of the client side are split into three

ISO 4287				
Name	P-Profil	W-Profil	R-Profil	
Length of reference distance	0.80mm	0.80mm	4.00mm	
Number of reference distance	5	5	1	
Evaluation area	0.80mm x 4.60mm	0.80mm x 4.80mm	0.80mm x 4.60mm	
Name	R-Profil	W-Profil	P-Profil	Norm
Mean height of profile elements	Rc 1.00um			ISO4287
Total height of profile	Rt 1.00um	Wt 1.00um	Pt 2.00um	ISO4287
Root mean square slope of the assessed profile	Rdq 0.28%			ISO4287
Root mean square deviation of the assessed profile	Rq 0.35um	Wq 0.35um	Pq 0.71um	ISO4287
Arithmetical mean deviation of the assessed profile	Ra 0.32um	Wa 0.32um	Pa 0.64um	ISO4287
Peak count number	Rpc NaN			ISO4287
Maximum profile valley depth	Rv 0.50um	Wv 0.50um	Pv 1.00um	ISO4287
Maximum height of profile	Rz 1.00um	Wz 1.00um	Pz 2.00um	ISO4287
Maximum profile peak height	Rp 0.50um	Wp 0.50um	Pp 1.00um	ISO4287
Skewness of the assessed profile	Rsk -0.00	Wsk -0.00	Psk -0.00	ISO4287
Kurtosis of the assessed profile	Rku 1.50	Wku 1.50	Pku 1.50	ISO4287
Mean width of profile elements	Rsm 0.80mm	Wsm 0.80mm	Psm 0.80mm	ISO4287
Maximum peak to valley depth	Rmax 1.00um	Wmax 1.00um	Pmax 2.00um	DIN4768
Root mean square wavelength of profile	Rlq 0.80mm	Wlq 0.80mm	Plq 0.80mm	DIN4768
Arithmetic mean slope of the profile	Rda 0.25%	Wda 0.25%	Pda 0.50%	DIN4768
Arithmetic mean wavelength of profile	Rla 0.80mm	Wla 0.80mm	Pla 0.80mm	DIN4768
Maximum profile peak height	Rpmax 0.50um	Wpmax 0.50um	Ppmax 1.00um	-
Maximum profile valley depth	Rvmax 0.50um	Wvmax 0.50um	Pvmax 1.00um	-

Figure 5.10: Step 5 - (b) ISO 4287 parameter evaluation

groups, they are the Interface-ViewModels, App-Data-ViewModels and the User-Data-ViewModel. These three ViewModel groups help to structure the JavaScript code further. The purpose of the ViewModel groups are explained below.

1. Interface-ViewModels: The Interface-ViewModels contain the state and behavior of the user interface. They are used to implement the user interaction. Therefore they also hold data required to keep track of the user progress through steps of the evaluation process. This data is not saved in the server.

2. App-Data-ViewModels: App-Data-ViewModels handle data related to the web application. They are retrieved from the web server when loading a page. For example the types of fits and filters that are available for the user to perform the evaluation process. The data is only fetched from the server but cannot be modified by the user.

3. User-Data-ViewModels: It manages the measurement data uploaded and settings made by the user during the evaluation. The data of the user managed by this ViewModels can be stored in the server and the stored data is retrieved from the server when required in the application. Therefore the User-Data-ViewModels use the AJAX concept to communicate with the server. When the user makes changes in the web application or adds new data, the data is first created in User-Data-ViewModels, it then forwards its data in JSON format for validation to the server. The server checks the data and stores it accordingly. The User-Data-ViewModels updates with the data of server responses.

ISO 13565-2

Name	R-Profile
Evaluation length	4.00mm
Evaluation area	0.80mm x 4.80mm
Number of classes	5759
Class width	0.00um
Number of classes in the secant	2304
Core area search	Moving line fit
Outlier suppression	0.05um
Approximation of Abbott curve	constant

Name	R-Profile
Total peak height	Rpk: Infinity
Core roughness depth	Rk: Infinity
Material component relative to peaks	Mr1: Infinity
Material component relative to valleys	Mr2: Infinity
Reduced peak height	Rpk: Infinity
Reduced valley depths	Rvk: Infinity
Total valley depth	Rvko: 0.00um
Profile peak area	A1: Infinity
Profile valley area	A2: Infinity
Position of the regression line	t: 0.53um
Slope of the regression line	m: -0.01um/%

Back Continue

2014 © MTS Kaiserslautern

Figure 5.11: Step 5 - (c) ISO 13565 part 2 parameter evaluation

5.2.2 Server implementation

This section explains the implementation of the server side components. First an introductory overview is given and then the use of the MVC pattern is discussed to describe the structure of the server implementation. The server side development of the web application is done using Java programming language. In order to build a robust and user friendly software, It is necessary to develop the server side architecture model according to the MVC design pattern. Usually a web framework is used to organize the interaction of the server with the browser, hence Play Framework based on MVC design pattern is used. Other frameworks like the FOP framework and JFreeChart framework are also used for specific functions like generating PDF files and charting. There are several algorithms used in this web application, some are partly implemented in Java and some as DLLs. Hence its necessary to have a strong architecture to organize access to these algorithms and it is designed such that the extensions and access to these algorithms are made easy to handle [Wie13].

Use of the Model-View-Controller pattern The Model-View-Controller pattern allows the separation of the user interface (View), the application logic (Controller) and the data (Models). The Models and Views of the server are not representing the same parts of the software as the Models and Views of the client. They are virtual concepts, which are partly overlaid by each other in this web application. The server has to deliver the client implementation to the web browser. The client implementation is in this context the View of the MVC pattern used by the server.

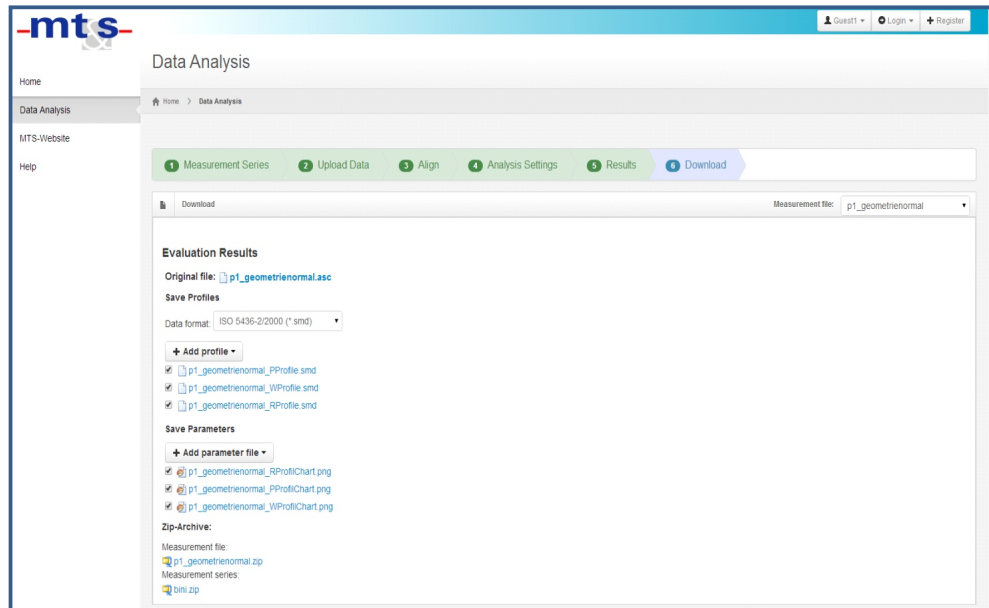


Figure 5.12: Step 6 - Results

The View of the server can contain a View of the client. In a normal context of data exchange in an AJAX application, the View of the server structures data with the JSON. In the client MVVM pattern this JSON data is the Model and for the server the JSON formatted data is the View. It is the part of the server visible to the user. This can be a bit confusing as both the client and server has a Model and View concept. The Models of the server are the data objects exchanged with the database (structured with Java classes).

1. Controller: The play framework used in the server side development has a global class in which the global settings for the web application are set. The configurations are performed in this function, which relates to the algorithms in the software. The user interacts to the application by means of controller. It handles all requests made by clients and forwards it to the appropriate handler. The controller is addressed via HTTP request to send and receive data to and from the server. An HTTP service call like GET, POST and DELETE can also be used in addition to the request path. Based on the HTTP requests, the controller handles the execution of the algorithm and accordingly manipulates the data in the model. The requests are forwarded to the controller by a router, which triggers one action at a time to the controller. It forwards the request to the associated controller. The controller gets the input from the client and instructs the Model and View to perform accordingly. If a wrong controller is invoked, the Model directs the controller to send an error message to the client. The controller takes the View representation and maps it to the Model

and sends the response back to the View i.e. for every request from client, the controller retrieves the necessary Model object and renders it to the client. Here we come across two kinds of Controller: application specific Controller and user specific Controller. Both have their own functionality.

Application specific Controller This controller deal with data that are same for all users i.e. data that does not change in the web application. The controllers that manage the applications are listed below.

CompensationElementController: This controller returns information about the compensation elements available as an array. In this software application, compensation elements like straight line fit using least square and total least square and circular fit using Gaussian iteration are used.

FilterController: This controller provides information of available filters. An HTTP request to this controller retrieves all information about available filters and returns them as JSON formatted array to View (client).

FileFormatController: This controller provides information about different formats, which the uploaded data files can have along with its extensions.

ProfileExportFormatController: This controller retrieves information of the formats in which the profiles can be downloaded.

ResultExportController: This controller manages the file formats in which the results can be downloaded.

User specific Controller This controller organizes the user related data. They are accessible only by the authorized user. The controllers that manage the user related data are listed below:

UserController: Every user will have an account created, regardless he is a guest or a registered user. For non registered user, a guest account is created and is automatically deleted at the end of the process or after 48 hours depending on the default setting. A registered user can login any time and manage the saved files. When loading the website, a user is automatically registered as a guest. When a user registers an account, the email address and the password are sent to the controller via POST request. The controller returns the created data as a response. When a registered user logs in, the server sends a response with login information or a failure message if the credentials do not meet the requirements.

The user related data after the user is logged in is managed by the controllers below. The data belonging to a specific user is accessible via path `"/users/{userid}"`. The

"*userid*" parameter specifies the logged in user. When the user signs out, the main page of the website is reloaded and the user is no longer allowed to access his data.

MeasurementSeriesController: This controller is responsible for managing all the data files in the measurement series. It allows a registered user to create a series of measurements. A GET request to this controller sends an array of measurement series of the user. A new measurement series can be made by a POST request at the url ["/users/{userid}/measurementSeries/{seriesid}"](#). Deleting the files is also made possible using DELETE request. The "*seriesid*" parameter specifies the measurement series.

ProfileFileController: This controller handles the uploading and managing of files. A GET request to this url ["/users/{userid}/measurementSeries/{seriesid}/files/{fileid}"](#) provides information about all the files in the measurement series. In each measurement series, one or more files can be added. All files uploaded must have the same length, hence the first uploaded file determines the length of the profiles in the measurement series. Once the files are uploaded, the length of the profiles are checked and the profiles with odd lengths are rejected. The "*fileid*" parameter specifies the profiles accessible in the series.

AnalysisSettingsController: This controller handles the settings for the evaluation of the files. The analysis settings are applied for each measurement series. A GET request to this url ["/users/{userid}/measurementSeries/{seriesid}/settings"](#) provides the settings of one specific measurement series. The settings can be modified by sending a POST request. So the settings can also be altered and adjusted by the user accordingly.

ProfileController: This controller is created to handle the data files of the measurement series profile. After analysis setting, information about the profile and an image to view the profile is generated and retrieved via this controller. A GET call to this url ["/users/{userid}/measurementSeries/{seriesid}/files/{fileid}/profile/{profiletype}/{filename}"](#) passes information about the profile such as the maximum and minimum value of X and Y axis and the datapoints of the profile. The profile is determined by the profile type parameter "RProfile", "WProfile", "PProfile", "OriginalProfile" and "CompensatedProfile". Through a GET request an image file is downloaded, which represents the profile. This image file is used for display in the webpage. There are different representation for profiles to be accessed i.e. P and W-profiles have a combined representation and they can be accessed with the profile type parameter

"PWProfile". The filename parameter can be any file name with the file extension ".png". The image files are not created until a query is made to the controller.

AbbottProfileController: The material ratio curves are managed by this controller. A GET request to this url ["/users/{userid}/measurementSeries/{seriesid}/files/{fileid}/abbott/{profiletype}/{filename}"](#) returns the material ratio curve of a profile. The profile is determined by the profile type parameter such as "PProfile", "WProfile" and "RProfile". The image file of the material ratio curve is created on request and displayed on the webpage.

ISO4287CharacteristicValuesController: This controller handles the evaluation results according to DIN EN ISO 4287. The results are retrieved by the controller only after analysis settings. For each data file, P-profile, W-profile and R-profile are generated and the evaluation results for each profile according to DIN EN ISO 4287 are returned to the user. The controller is specified by a request to this url ["/users/{userid}/measurementSeries/{seriesid}/files/{fileid}/iso4287/{profiletype}"](#).

ISO13565CharacteristicValuesController: A GET call to this controller ["/users/{userid}/measurementSeries/{seriesid}/files/{fileid}/iso13565"](#) handles the evaluation of parameters according to DIN EN ISO 13565 part 2 on the R-profiles. Since the parameters can be evaluated only after the R-profile, the profile type parameter is not used.

ResultExportController: The profiles generated during the process may be downloaded in different file formats. A GET request to this address ["/users/{userid}/measurementSeries/{seriesid}/files/{fileid}/profile/{profiletype}.../format/{profileexportid}/{filename}"](#) will be forwarded to this controller. The controller responds to the request with the desired file format. The format in which the file is created is specified by the "profileexportid" parameter.

ResultFileController: The evaluation results can be downloaded in various formats for any data file in the measurement series. A GET request at this address ["/users/{userid}/measurementSeries/{seriesid}/resultfiles/..{resultformatid}/{filename}"](#) is forwarded to this controller. The controller creates the result file in the desired format and sends it to the user. The format in which the file is created is specified by the "resultformatid" parameter.

ArchiveController: This controller serves by downloading a collection of files as a zip file. The file can contain both the profiles and the results. All specified files are generated and compressed into a zip file and returned to the user on request. The url to this controller is ["/users/{userid}/measurementSeries/{seriesid}/archive/{filename}"](#)

The controllers mentioned above organise the web application in a better way. It enables a clear and neat communication between the user and the server.

2. Model: Models are used to hold data of the web application and to store them into a database. The models used in the web application are listed below with its properties.

UserModel: contains the data of the user account, i.e. for guest as well as for registered users. If it is a guest account, it is given by *"isGuest"* parameter.

MeasurementSeriesModel: contains the data of a measurement series and it is connected to a unique user. It can contain certain number of files, whose data is encapsulated by *ProfileFileModel*. The analysis settings are assigned to the measurement series which is represented by the *AnalysisSettingsModel*. Objects of this class contain the settings for compensation elements, filters, cutoff wavelength, and to provide information if ISO 13565 is to be applied or not. The sampling length setting is managed by *SamplinglengthSettingModel*. They are created for P, W, R and SR-Profiles.

ProfileModel: holds the data of the profile, i.e. the name of the file is stored in *"FileName"*, the name of the profile (Roughness Profile) in *"Name"*, short name of the profile (R-Profile) in *"ShortName"*, and the datapoints of the profile are managed through *"Data"* as an array. The total number of measured values are stored in *"ItemsX"*. *"XMin"*, *"XMax"*, *"YMin"*, *"YMax"* are the minimum and maximum values of the X-axis and Y-axis. *"XUnit"* indicates the unit in which the values of X-axis are stored and the unit of the measured values are given in *"YUnit"*. *"ProfileType"* tells about the profile type (example: RProfile).

The evaluation results according to DIN EN ISO 4287 are stored in *ISO4287 AnalysisModel*, similarly the results according to DIN EN ISO 13565 are stored in *ISO13565_2AnalysisModel*. Since these evaluations are to be deeply integrated into the application, there are individual classes written for each of these standard evaluation. The analysis function in the class is started only when the profile and filter settings are defined. This function evaluates the profile with the settings given and stores the parameters in *"characteristicValues"*. The characteristic values are managed by the *CharacteristicValueModel* objects as list. These objects contain the name of the characteristic value (Average Roughness), a short name (Ra), description (if any), standard (ISO4287), position (10) at which the characteristic value is to be displayed in the list and value holds the result with a formatted unit. Evaluation according to ISO4287 is generally used for the P, W and R profiles and the evaluation according to ISO 13565_2 is used only for R profiles.

A model may also interact with the database to fetch information requested by the client.

3. View: The views in the web application are mainly used for display. There is a main view page that contains all the HTML, CSS and JavaScript elements that are required for the application. Other views are used to format the object (JSON) which the controller uses to format its response to the clients.

5.3 Java and C++ interoperation

Java interoperation with a native DLL was initially done by Java Native Access (JNA). JNA is very useful and easy to use interface. Since most of the algorithms used in this software are implemented in C++ as DLL, this section explains the integration of the existing algorithms in the DLL into the Java application. The JNA framework consists of a "jna.jar" file that is referenced into the projects build path. Basically a Java interface is created to wrap the DLL functions to be invoked. Depending on the conventions that the DLL expects, the interface needs to inherit from "com.sun.jna.library". JNA takes care of all run-time aspects. It is important to create a Java interface (JNA-Interface) with method definition that matches the DLLs functions. Therefore there should be one Java interface available for each DLL. In Java a class must be written that implements the appropriate Java interface. Java interoperation with DLLs using JNA was found to be error prone while using structures in the Java interface. Hence Java Native Interface (JNI) wrapper was developed to avoid the definition of structures in Java interface. JNA was successful to some extent but slowly Java threw exceptions related to memory mapping. It is very important to check if the DLLs used are compatible with the processors architecture. It is also important to check if the JVM is 32 bit or 64 bit and compile all DLLs called in the Java code against the same bit format. All the DLLs are supposed to have the same platform. Table 5.1 refers to the JNI wrapper and its corresponding DLL used in the software. The DLLs used in this software are listed in the last column of table 5.1. In sub-section 5.3.1, description of DLL in C++ is tabulated and the interoperation of DLL in Java is explained in detail. Additional information about the use of JNI with native code may be found in appendix B.4.

5.3.1 Integration

As discussed in section 5.3, most of the algorithms used in the software are implemented in C++ as DLL. This sub-section explains the integration of the existing algorithms from DLL into Java application.

Table 5.1: Software interoperation structure

JNI Wrapper	DLL
ImpLibJNIWrapper	implib.dll
FftLibJNIWrapper	FftLib.dll
Iso4287JNIWrapper	iso4287.dll
Iso13565JNIWrapper	iso13565.dll

ImportInterface(): This function (see table 5.2) loads a data file and stores the result in a structure "*TNativeSurface*" (see table 5.3). It refers to the implib.dll. It provides an interface to load data files of different formats. The first parameter of the function accepts the file path, the second is the length of the first parameter, the third parameter specifies the file format to be loaded. The fifth parameter is the output parameter (*TNativeSurface *nativesurface*), it returns an object containing the profile information of the loaded file. The remaining parameters are constant and are not relevant for the application.

Table 5.2: ImportInterface()

Function	ImportInterface()
DLL	implib.dll
Return value	bool
Input parameter	char *filepath, unsigned filepathlen, LONGLONG *prgid, char prgtyp, TNativeSurface *nativesurface, unsigned ap-handle, unsigned progressbar, char *sendmessage, char *receivemessage, bool showdialog
Output parameter	TNativeSurface *nativesurface

TNativeSurface: This Structure stores the result from ImportInterface(). The information of the file loaded is returned using this structure type. The measured datapoints are stored in "*data*" as an array. The total number of datapoints is given in "*itemsx*", "*xmin*" and "*xmax*" specify the first and last value of x-axis. "*filename*" specifies the name of the file uploaded. The other values are not important for this work.

ProfileFilterInterface(): This function (see table 5.4) performs low pass filtering of the profile with user defined cutoff wavelength. The type of filter to be used is determined by an filter ID. The filter ID is the first parameter, the second and third parameters are void pointers which receive a structure type from "*TNativeProfile*".

Table 5.3: TNativeSurface

Structure Type	TNativeSurface
DLL	impLib.dll
Data elements	double **data; long itemsx; long itemsy; unsigned char **nv; long nvcounts; double xmin; double xmax; double ymin; double ymax; char *filename; char *dataobject; char *comment; bool valid;

The profile to be filtered is passed through the input parameter. The resulting filtered profile is obtained in the output parameter. The parameter *"lc"* passes the user defined cutoff wavelength. The remaining parameters are not important. This function refers to the fftLib.dll.

Table 5.4: ProfileFilterInterface()

Function	ProfileFilterInterface()
DLL	fftLib.dll
Return value	void
Input parameter	LONGLONG *APrgId, void *input, void *output, double lc, HANDLE AAppHandle, HANDLE AProgressBar, char *ASendMessage, char **AReceiveMessage
Output parameter	void *output

TNativeProfile: The profile to be filtered and the filtered profile are passed to or from ProfileFilterInterface() as TNativeProfile (see table 5.5). It is similar to TNativeSurface.

Table 5.5: TNativeProfile

Structure Type	TNativeProfile
DLL	fftLib.dll
Data elements	double *data; long ItemsX; unsigned char *nv; long NVCounts; double XMin, XMax; char *FileName; char *DataObject; char *Comment; bool Valid;

GetCounts(): This function (see table 5.6) returns the number of available filters or the loadable data formats. The value corresponds to the number of structural objects that are retrieved with GetItem(). This function with respect to fftLib.dll returns

the number of filters provided by the DLL and GetCounts() with respect to impLib.dll returns the number of file formats that can be read.

Table 5.6: GetCounts()

Function	GetCounts()
DLL	implib.dll and fftLib.dll
Return value	long
Input parameter	void

GetItem(): This function (see table 5.7) gives information about the filter (fftLib.dll) or data formats (impLib.dll). The parameter *"index"* refers to an element in the register, to which the information is returned. It returns a structure of type TProcLib containing the information described in table 5.8.

Table 5.7: GetItem()

Function	GetItem()
DLL	implib.dll and fftLib.dll
Return value	TProcLib*
Input parameter	long index

TProcLib: *"ResName"* provides a descriptive name for a file format or a filter. *"PrgId"* is used to identify the program. Filter also contain an ID called FilterIDs, if the last bit of the ID is (1) then the filter is open else (0) for closed. *"ResString"* contains information about file extensions. It refers to impLib.dll and fftLib.dll.

Table 5.8: TProcLib

Structure Type	TProcLib
DLL	implib.dll and fftLib.dll
Data elements	char *ResName; LONGLONG PrgId; char *ResString;

fnISO4287(): This function (see table 5.9) calculates the parameters of the profile as International standard ISO 4287. The setting for sampling length are defined by the parameters *"nple"*, *"nle"*, and *"noffset"*. *"nple"* is the number of samples per sampling length, *"nle"* is the number of sampling length and *"noffset"* is the length of the delay line that goes to the first sampling length. The profile is passed as an array by the *"profile"* parameter. The distance of the measured values is given by *"dx"*.

Table 5.9: fncISO4287()

Function	fncISO4287()
DLL	iso4287.dll
Return value	long
Input parameter	long nple, long nle, long noffset, double *profile, double dx, STRISO4287 **ptr2iso4287
Output parameter	STRISO4287 **ptr2iso4287

Table 5.10: STRISO4287

Structure Type	STRISO4287
DLL	iso4287.dll
Data elements	long fileversion; long nple; long nle; long noffset; double xt; double xa; double *xai; double xq; double xz; double *xzi; double xp; double *xpi; double xv; double *xvi; double xsk; double *xski; double xku; double *xkui; double xsm; double xc; double xdq; double xpc; double xmax_nstd; double xiq_nstd; double xda_nstd; double xla_nstd; double xpmax_nstd; double xvmax_nstd; long nabbott; double *material; double *intersection;

The last parameter "*STRISO4287*" is used to display the parameters as structure type (see table 5.10). It refers to the iso4287.dll.

fncISO13565(): This function (see table 5.11) calculates the parameters of the profile according to International standard ISO 13565. The setting for measuring distance is defined by the parameters "*np*" and "*noffset*". "*np*" is the number of measured values of the evaluation length. The length of delay line to the first value of the measurement path is given in the parameter "*noffset*". The profile is passed as an array by the "*profile*" parameter. The parameter "*STRISO13565_2*" is used to display the parameters as Structure type (see table 5.12). It refers to the iso13565.dll.

Table 5.11: fncISO13565_2()

Function	fncISO13565_2()
DLL	iso13565.dll
Return value	long
Input parameter	long np, long noffset, double *profile, STRISO13565_2 **ptriso13565_2
Output parameter	STRISO13565_2 **ptr2iso13565_2

Table 5.12: STRISO13565_2

Structure Type	STRISO13565_2
DLL	iso13565.dll
Data elements	long np; long noffset; double rk; double mr1; double mr2; double rpk; double rvk; double rpkx; double rvkx; double a1; double a2; long nabbott; double *material; double *in- tersection; double t; double m; long r1; long r2;

6 Soft gauges

There are several industrial measuring instruments available in industry. They are computer based and have their own software to measure and characterise the profiles. ISO 5436-2 discusses two kinds of surface measurement standards: Type F1 and Type F2. Data files are usually used as input to the software under test and the results are compared with the certified results provided with the type F1 software measurement standard. Where Type F1 measurement standard can be generated mathematically, for which parameters are calculated analytically and independently [Lea13]. The mathematically generated function can be passed as an input to the software under test to check its correctness. If the software gets through the acceptance test, it is considered as type F2 software.

The parameters calculated by industrial measuring instruments and software packages are compared with the reference software. The parameters calculated by the reference software is also compared to internet based software likes the National Institute of Standards and Technology (NIST) and Physikalisch -Technische Bundesanstalt (PTB) for its robustness. The end results are interesting and is tabulated in table 6.1, 6.2, and 6.3. Small differences may be noticed, which may be due to difference in the implementation of the mathematical equations. The user can import external data files into the reference software for profile parameter calculation. The comparison was made on the surface profiles using the default filter i.e. the Gaussian filter with a cutoff wavelength 0.8mm. The profile parameters used in the investigation are R_a , R_q , R_z , R_p , R_v , R_t , R_{sk} , R_{ku} , and R_{sm} .

The proposed reference software is advanced and updated to the International standard methods. Existing software use the least square straight line fit (LS) to remove the influence of tilt caused by the mounting table and Gaussian filter (ISO 11562) to filter the profile. The proposed reference software helps to fit in reference line or circle using total least square (TLS) method which was recently added into the ISO standard for removing form and curvature from the surface. It also filters the profiles according to ISO 13565-1 (special filter) and ISO 16610 series. It provides the client with various options to choose the appropriate filter depending upon their surface type. Height, spacing and Abbott curve parameters can be evaluated according to ISO 4287 and ISO 13565-2 standards. All the parameters described in table 2.2 are evaluated after form removal. The Abbott curve parameters are evaluated on the roughness profiles. The basic flowchart of the analysis developed in software is shown in figure 6.1. It is assumed that the datapoints applied are equally spaced.

The default settings used in the analysis are shown in red. Figure 6.1 specifies various methods used for evaluation and the parameters that are calculated. The robust spline filter was recently removed from the ISO 16610 series. To make the software user friendly, additional features like user login, data security, alert messages and exporting results as pdf, image, zip files are included into the software.

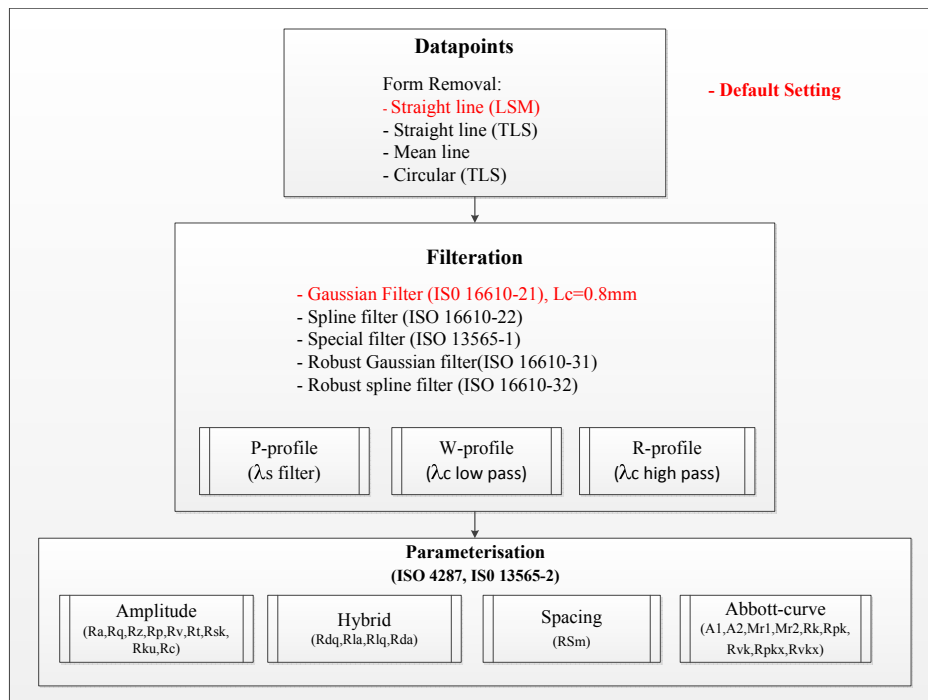


Figure 6.1: Flowchart of the soft gauge

6.1 Comparison with existing software

This section compares the proposed reference software with internet based surface metrology algorithm testing system by NIST and the reference software by PTB, Braunschweig. The comparison of the surface texture parameters are tabulated in tables 6.1, 6.2 and 6.3. It is clear from the table that the proposed software evaluates more parameters when compared to that of others. The additional parameters are added in blue at the bottom part of tables 6.1, 6.2 and 6.3. Table 2.2 lists the parameters that are evaluated as per sampling lengths and evaluation lengths. It is noticed that most of the parameters are evaluated over the sampling length and an average of it is taken to improve reliability. The roughness and waviness profiles are evaluated over several successive sampling lengths and for primary profile the evaluation length is equal to the sampling length. According to ISO 4287, five sampling lengths are used as default for evaluation. It is possible to change the number of sampling lengths in the software, provided that it satisfies the ISO standard applied. Table 6.1,

6.2 and 6.3 shows comparison of primary, waviness and roughness parameters with respect to other software available.

6.1.1 Primary profile parameters

Table 6.1: P-profile parameters (profile:Gn_56, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)

P-profile	NIST	MTS	PTB	Norm
P_a	0.43072 μm	0.43031 μm	0.43038 μm	ISO 4287
P_q	0.49188 μm	0.49150 μm	0.49168 μm	ISO 4287
P_z	1.54281 μm	1.51458 μm	1.54281 μm	ISO 4287
P_p	0.77553 μm	0.77552 μm	0.77553 μm	ISO 4287
P_v	0.76728 μm	0.73905 μm	0.76728 μm	ISO 4287
P_t	1.54281 μm	1.51458 μm	1.54281 μm	ISO 4287
P_c	1.48838 μm	1.48801 μm	1.48804 μm	ISO 4287
P_{sk}	-0.0027	0.02555	-0.003	ISO 4287
P_{ku}	1.68023	1.68094	1.682	ISO 4287
P_{dq}	0.03252	0.03251	0.0329	ISO 4287
P_{sm}	100.02533 μm	100.051 μm	100.03 μm	ISO 4287

Additional parameters evaluated using MTS reference software are listed below.

P-profile	MTS	Norm
P_{Ia}	0.09029 mm	DIN 4768
P_{Iq}	0.09496 mm	DIN 4768
P_{da}	2.99437%	DIN 4768
P_{max}	1.51458 μm	DIN 4768
P_{pmax}	0.77552 μm	-
P_{vmax}	0.73905 μm	-

6.1.2 Waviness profile parameters

Table 6.2: W-profile parameters (profile:Gn_56, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)

W-profile	NIST	MTS	PTB	Norm
W_a	0.0092 μm	0.00919 μm	0.0092 μm	ISO 4287
W_q	0.00983 μm	0.01002 μm	0.01003 μm	ISO 4287
W_z	0.01184 μm	0.01184 μm	0.01184 μm	ISO 4287
W_p	0.00978 μm	0.00977 μm	0.00978 μm	ISO 4287
W_v	0.00206 μm	0.00206 μm	0.0076 μm	ISO 4287
W_t	0.03465 μm	0.03465 μm	0.03465 μm	ISO 4287
W_c	-	-	0.00 μm	ISO 4287
W_{sk}	0.60898	0.85362	0.609	ISO 4287
W_{ku}	1.34846	1.65204	1.348	ISO 4287
W_{dq}	2.0E-5	0.00245	0.0000 μm	ISO 4287
W_{sm}	-	NaN	0.00 μm	ISO 4287

Additional parameters evaluated using MTS reference software are listed below.

W-profile	MTS	Norm
W_{Ia}	2.99966 mm	DIN 4768
W_{Iq}	2.56230 mm	DIN 4768
W_{da}	0.00192%	DIN 4768
W_{max}	0.02002 μm	DIN 4768
W_{pmax}	0.01880 μm	-
W_{vmax}	0.01584 μm	-

6.1.3 Roughness profile parameters

Table 6.3: R-profile parameters (profile:Gn_56, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)

R-profile	NIST	MTS	PTB	Norm
R_a	0.43029 μm	0.43028 μm	0.43032 μm	ISO 4287
R_q	0.49143 μm	0.49143 μm	0.49146 μm	ISO 4287
R_z	1.50158 μm	1.50158 μm	1.50158 μm	ISO 4287
R_p	0.76237 μm	0.76237 μm	0.76237 μm	ISO 4287
R_v	0.73921 μm	0.73920 μm	0.73921 μm	ISO 4287
R_t	1.51092 μm	1.51091 μm	1.51092 μm	ISO 4287
R_c	1.48811 μm	1.48800 μm	1.48791 μm	ISO 4287
R_{sk}	-0.00585	-0.00583	-0.006	ISO 4287
R_{ku}	1.68022	1.68024	1.680	ISO 4287
R_{dq}	0.0325	3.25179 %	0.0325	ISO 4287
R_{sm}	100.01699 μm	0.10004 mm	100.04 μm	ISO 4287

Additional parameters evaluated using MTS reference software are listed below.

R-profile	MTS	Norm
R_{pc}	99.95052	ISO 4287
R_{Ia}	0.09028 mm	DIN 4768
R_{Iq}	0.09495 mm	DIN 4768
R_{da}	2.99435%	DIN 4768
R_{max}	1.50638 μm	DIN 4768
R_{pmax}	0.76587 μm	-
R_{vmax}	0.74504 μm	-

The skewness and kurtosis values are strongly influenced by the peaks and valleys in the profile. If it displays a high value, that is likely due to the evaluation of skewness across the evaluation length of R_q . Negative skew value indicates good bearing properties. Table 6.4 and 6.5 compares the roughness parameters of another profile with PTB software (updated version: status Feb'2018) for reference.

After thorough evaluation and comparison of the software with the existing internet based software, It is encouraging to notice that the proposed software satisfies the desired requirements like:

1. Unambiguous definition of parameters,

Table 6.4: R-profile parameters (profile:normal, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)

R-profile	MTS	PTB
R_a	0.18657 μm	186.57 nm
R_q	0.23142 μm	231.42 nm
R_z	1.23203 μm	1232.05 nm
R_p	0.50793 μm	507.98 nm
R_v	0.72410 μm	724.07 nm
R_t	1.42598 μm	1426.07 nm
R_c	0.72264 μm	540.43 +/- 215.57 nm
R_{sk}	-0.26782	-0.2678
R_{ku}	2.79029	2.7905
R_{da}	5.37267 %	5.38178 %
R_{dq}	6.85868 %	6.86612 %
R_{la}	0.02181 mm	21.8064 μm
R_{lq}	0.02120 mm	21.1908 μm
R_{sm}	0.06423 mm	39.16 +/- 28.51 μm
R_{pmax}	0.55257 μm	552.72 nm
R_{vmax}	0.87340 μm	873.35 nm

Table 6.5: Abbott curve parameters (profile:normal, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)

R-profile	MTS	PTB
R_k	0.65026 μm	650.47 nm
R_{pk}	0.13443 μm	134.27 nm
R_{vk}	0.25107 μm	250.94 nm
$Mr1$	6.97760 %	6.97 %
$Mr2$	89.83698 %	89.84 %
$A1$	0.46900 $\mu\text{m}^*\%$	468.2
$A2$	1.27586 $\mu\text{m}^*\%$	1274.5

2. Algorithms with speed and accuracy,
3. Robust,
4. Evaluation according to ISO standard,
5. User friendly.

7 Summary and future work

7.1 Summary and conclusion

Roughness plays a vital role in determining how a real workpiece behaves with its environment. It is a good predictor of the performance of mechanical components, but it is difficult and expensive to control it in manufacturing. The evaluation of surface roughness on technical functional areas and the comparison of roughness parameters with tolerances is an integral part of industrial quality assurance. The aim of the thesis is to develop user friendly software which makes the evaluation methods for verification of surface characteristics easier for clients. Most of the software used in industrial measuring instruments may give variations or differences in the parameters calculated. Such variations can be verified using the proposed reference software.

Characterisation of surface roughness plays a vital role when dealing with issues like friction, lubrication and wear. In this thesis work, the concept of roughness and most of the parameters used to characterise it have been presented. The most commonly used technique for assessing surface characteristics is through the use of stylus instruments. Today, rapid increase in the number of 2-D parameters are not sufficient to understand the functional behavior of the surface. Only limited information on height and character of the surface profile are provided, hence it is really important to provide three dimensional nature of the surface. Surface characteristics are determined in terms of height and spacing asperities. The height parameters are evaluated relative to a reference line. This line follows the geometrical shape of the surface such that the profile can be measured as deviations from the reference line. Assessing a profile from a surface can at times be misleading, the profile may fail to consider the important features of the surface by skipping predominant peaks and valleys. Parameters related to the functional behavior of the surface may not be reliably evaluated in two dimensions. Fundamental problems associated with surface roughness require information of the surface in three dimension. Analysis of the surface based on 3-D measurement represents a more realistic approach to understand the surface topography. However, the conclusion is that there is fundamental limitations in using 2-D analysis to describe a 3-D surface topography [Sto00].

The importance of three dimensional analysis and the limitations of two dimensional analysis has to be considered in the near future for complete assessment of the surface topography. The three dimensional analysis is said to rapidly grow in industry over the decades thereby reducing the analysis time and cost. Web based reference

software for characterisation of surface roughness in three dimensional dataset is to be implemented in the near future. The techniques of roughness measurement and characterisation may be considered on various grounds like cost, accuracy, resolution, easiness and so on. In any case, the choice purely depends on the user and the application. It is also important to consider the uncertainty in parameters in the future, so that the parameters are stated with its associated uncertainty. Coming on to the software, it would be nice to have a new look and feel with stable application deployment followed by active administration and security access into the website (SSL certificate).

Bibliography

- [Ahn04] Ahn, Sung J.: *Least squares orthogonal distance fitting of curves and surfaces in space*. Berlin and New York : Springer, 2004. – ISBN 9783540239666
- [And12] Anderson, Chris (Hrsg.): *Pro Business Applications with Silverlight 5*. Apress, 2012. – ISBN 1430235004
- [ARR99] Ahn, Sung J. ; Rauh, Wolfgang ; Recknagel, Matthias: *Geometric Fitting of Line, Plane, Circle, Sphere, and Ellipse*. 25.-26.01.1999 (ABW-Workshop 6)
- [ARW01] Ahn, Sung J. ; Rauh, Wolfgang ; Warnecke, Hans-Jürgen: Least-squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola, and parabola. In: *Pattern Recognition* 34 (2001), Nr. 12, S. 2283–2303. [http://dx.doi.org/10.1016/S0031-3203\(00\)00152-7](http://dx.doi.org/10.1016/S0031-3203(00)00152-7). – DOI 10.1016/S0031-3203(00)00152-7. – ISSN 00313203
- [Bel01] Bell, Stephanie (Hrsg.): *A beginner's guide to uncertainty of measurement*. NPL, Teddington, UK : Measurement Good Practice Guide No. 11, March 2001, 2001. ISSN 1368–6550
- [BH92] Bodschwinna, Horst ; Hillmann, Walter: *Oberflächenmesstechnik mit Tastschnittgeräten in der industriellen Praxis*. 1. Aufl. Berlin and Köln : Beuth, 1992 (Beuth-Kommentare). – ISBN 3410126570
- [Bhu99] Bhushan, Bharat: *Principles and Applications of Tribology*, 1999. – ISBN 978-0-471-59407-9, S. 122–131
- [Bor07] Bort, Guillaume: *Play documentation*. <https://www.playframework.com/documentation/2.3.x/Home>, 2007. – [Online; accessed 19-January-2013]
- [DIN98a] DIN Deutsches Institut für Normung e.V.: *Geometrische Produktspezifikation (GPS); Oberflächenbeschaffenheit: Tastschnittverfahren; Meßtechnische Eigenschaften von phasenkorrekten Filtern*. DIN EN ISO 11562: 1998-09. Berlin, 1998
- [DIN98b] DIN Deutsches Institut für Normung e.V.: *Geometrische Produktspezifikation (GPS); Oberflächenbeschaffenheit: Tastschnittverfahren; Oberflächen mit plateauartigen funktionsrelevanten Eigenschaften; Teil 1: Fil-*

- terung und allgemeine Meßbedingungen.* DIN EN ISO 13565-1: 1998-04. Berlin, 1998
- [DIN98c] DIN Deutsches Institut für Normung e.V.: *Geometrische Produktspezifikation (GPS); Oberflächenbeschaffenheit: Tastschnittverfahren; Oberflächen mit plateauartigen funktionsrelevanten Eigenschaften; Teil 2: Beschreibung der Höhe mittels linearer Darstellung der Materialanteilkurve.* DIN EN ISO 13565-2: 1998-04. Berlin, 1998
- [DIN98d] DIN Deutsches Institut für Normung e.V.: *Geometrische Produktspezifikation (GPS); Oberflächenbeschaffenheit: Tastschnittverfahren; Regeln und Verfahren für die Beurteilung der Oberflächenbeschaffenheit.* DIN EN ISO 4288: 1998-04. Berlin, 1998
- [DIN09] DIN Deutsches Institut für Normung e.V.: *Geometrische Produktspezifikation (GPS); Oberflächenbeschaffenheit: Tastschnittverfahren; Benennung, Definitionen und Kenngrößen der Oberflächenbeschaffenheit.* DIN EN ISO 4287: 2009-11. Berlin, 2009
- [DIN12] DIN Deutsches Institut für Normung e.V.: *Geometrische Produktspezifikationen (GPS) - Filterung - Teil 21: Lineare Profilfilter: Gauß-Filter (ISO 16610-21:2011); Deutsche Fassung FprEN 16610-21:2012.* E DIN EN ISO 16610-21:2012. Berlin, 2012
- [Duc11] Duckett, Jon (Hrsg.): *HTML & CSS: Design and Build Web Sites.* John Wiley & Sons, Inc, 2011. – ISBN 978–1–118–00818–8
- [Fla11] Flanagan, David (Hrsg.): *JavaScript: The Definitive Guide, 6th Edition.* O'Reilly, 2011. – ISBN 978–0–596–80552–4
- [Gil13] Gilbert, David: *Welcome To JFreeChart.* <http://www.jfree.org/jfreechart/>, 2013. – [Online; accessed 9-March-2014]
- [Int06] International standard: *Geometrical product specifications (GPS) - Filtration- Part 22: Linear profile filters: Spline filters.* ISO/TS 16610-22:2006. Berlin, 2006
- [Int10a] International standard: *Geometrical product specifications (GPS) - Filtration - Part 28: Profile filters: End effects.* Berlin, 2010
- [Int10b] International standard: *Geometrical product specifications (GPS) - Filtration - Part 31: Robust profile filters: Gaussian regression filters.* Berlin, June 2010

- [Jor04] Jordan, Dr. Hans.-Joachim.: *Highly accurate 2D and 3D surface metrology using confocal white-light sensors with chromatic distance coding*. 2004
- [Jor08] Jordan, H.-J.: *Optical chromatic confocal probes*. 2008
- [JS08] J. Seewig, T. Hercke R. V. H.-J. Jordan J. H.-J. Jordan: *Optical High Speed Twist Characterisation*. 2008
- [KMN⁺] K.Kadrigama ; M.M.Noor ; N.M.Zuki.N.M ; Rahman, M.M. ; M.R.R.Rejab ; Daud, R. ; Abou-El-Hossein, K.A.: Optimization of Surface Roughness in End Milling on Mould Aluminium Alloys (AA6061-T6) Using Responses Surface Method and Radian Basis Function Network
- [Kry96] Krystek, Michael: *Form filtering by splines*. Measurement
- [Lea02] Leach, R. K. (Hrsg.): *Calibration, traceability and uncertainty issues in surface texture metrology*. NPL, Teddington, UK, 2002. ISSN 1361–4088
- [Lea13] Leach, R. K. (Hrsg.): *Characterisation of areal surface texture*. Berlin and New York : Springer, 2013. – ISBN 9783642364570
- [MBM97] M.Alauddin ; Baradie, M.A.EL ; M.S.J.Hashmi: Prediction of tool life in end milling by response surface methodology. In: *Journal of Materials Processing Technology* 71 (1997), S. 456–465
- [MGC01] Mainsah, E. ; Greenwood, J.A. ; Chetwynd, D.G: *Metrology and Properties of Engineering Surfaces*. Klumer Academic Publishers., 2001. – 13–33 S. – ISBN 0–412–80640–1
- [MR09] Muralikrishnan, Bala ; Raja, J.: *Computational surface and roundness metrology*. London : Springer, 2009. – ISBN 978–1–84800–297–5
- [NPL11] NPL: *Softgauges for Surface Topography*. <http://resource.npl.co.uk/softgauges/default.htm>, 2011. – [Online; accessed 6-April-2015]
- [Pri15] *Private Communication*. 2015
- [SE14] Seewig, Jörg ; Eifler, Matthias: Periodic Gaussian filter according to ISO 16610-21 for closed profiles. In: *Precision Engineering* 38 (2014), Nr. 2, S. 439–442. <http://dx.doi.org/10.1016/j.precisioneng.2013.09.004>. – DOI 10.1016/j.precisioneng.2013.09.004. – ISSN 01416359
- [See96] Seewig, Jörg: *The Uncertainty of Roughness Parameters*. 2013, pages 291- 296

- [See05] Seewig, Jörg: Linear and robust Gaussian regression filters. In: *Journal of Physics: Conference Series* 13 (2005), Nr. 1, S. 254–257
- [See13] Seewig, J: Characterisation of areal surface texture. (2013), S. 67–106. ISBN 9783642364570
- [Sta12] *Play framework 2-0 application as a windows service*. <http://stackoverflow.com/questions/12120489/how-do-i-run-a-play-framework-2-0-application-as-a-windows-service>. Version: 2012. – last viewed Jan, 2014
- [Sto00] Stout, Ken: *Development of methods for the characterisation of roughness in three dimensions*. Rev. repr. London : Penton Press, 2000. – ISBN 1857180232
- [Tau11] Tauber, James: *Apache: Quick Start Guide*. <http://xmlgraphics.apache.org/fop/quickstartguide.html>, 2011. – [Online; accessed 2-December-2013]
- [TG04] T. Goto, K. Umeda S. Kadowaki K. Y. J. Miyakura M. J. Miyakura: *A Robust Spline Filter on the basis of L2-norm*
- [Tho82] Thomas, Tom R.: *Rough surfaces*. London and New York : Longman, op. 1982. – ISBN 0582468167
- [VDI08] *Roughness measurement - Roughness measurement using contact (stylus) instruments - Profile method - Setup, measurement conditions, procedure*. Düsseldorf : Beuth-Verlag, 2008
- [Wei99] Weisstein, Eric W.: *Least Squares Fitting–Polynomial*. <http://mathworld.wolfram.com/LeastSquaresFittingPolynomial.html>, 1999. – [Online; accessed 26-April-2015]
- [Whi94] Whitehouse, D. J.: *Handbook of surface metrology*. Bristol and Philadelphia : Institute of Physics Pub., 1994. – ISBN 0750300396
- [Wie13] Wiegel, Timo: *Entwicklung einer webbasierten Software zur Auswertung von Oberflächenrauheit*. Kaiserslautern, TU Kaiserslautern, Diss., 03.2013

List of Figures

0.1	Work flow of the software	III
0.2	Blockdiagramm der Arbeit	V
1.1	Overall view of the analysis	4
2.1	Type F1 measurement standard	9
2.2	Type F2 measurement standard	11
2.3	Primary profile after mean line fitting	13
2.4	Primary profile after least square fitting	15
2.5	Primary profile after total least square fitting	16
2.6	Circular view of the surface profile	19
2.7	Circular fit of a surface profile (cylinder) measured using T8000	20
2.8	2RC Weighting Function	22
2.9	Gaussian filtering, $L_c=0.8\text{mm}$	24
2.10	Weighting function of spline filter	26
2.11	Spline filtering, $L_c=0.8\text{mm}$	27
2.12	Transmission characteristic of Gaussian and spline filter	28
2.13	Gaussian regression filter	29
2.14	Relationship between the mechanical profile and the primary surface profile [Pri15]	30
2.15	Measuring chain to obtain P-profile, W-profile and R-profile [Pri15]	31
2.16	Surface profile	33
2.17	Peak height of profile, from DIN EN ISO 4287	34
2.18	Valley depth of profile, from DIN EN ISO 4287	34
2.19	Maximum height of profile, from DIN EN ISO 4287	35
2.20	Mean width of profile, from DIN EN ISO 4287	36
2.21	Abbott curve of a profile at a cutting depth C [Mahr]	38
2.22	Abbott curve with R_k parameters [Mahr]	39
3.1	Weighting function of a Gaussian profile filter	41
3.2	Long wave transmission characteristics of the Gaussian filter	42
3.3	Short wave transmission characteristics of the Gaussian filter	43
3.4	Abbott curve using sorting algorithm for random datapoints	53
3.5	Abbott curve of a reference measurement with 11200 points	53
5.1	Web application model	59
5.2	Interoperation using JNI	60
5.3	Model-View-ViewModel using KnockoutJS	62

5.4	Communication between the client and server	64
5.5	Step 1 - Main page of reference software	65
5.6	Step 2 - Creating or loading new measurement series	66
5.7	Step 3 - Aligning profile with compensation element	67
5.8	Step 4 - Filter analysis setting (Gaussian filter, $L_c=0.8\text{mm}$, $L_s=2.5\mu\text{m}$)	67
5.9	Step 5 - (a) Material ratio curve	68
5.10	Step 5 - (b) ISO 4287 parameter evaluation	69
5.11	Step 5 - (c) ISO 13565 part 2 parameter evaluation	70
5.12	Step 6 - Results	71
6.1	Flowchart of the soft gauge	83

List of Tables

2.1	Type F1 (Profile: sine wave, filter: Gaussian , Lc = 0.8 mm)	10
2.2	Surface parameters, ISO 4287:1997	32
2.3	Additional surface parameters	32
3.1	The R_k parameters	54
5.1	Software interoperation structure	77
5.2	ImportInterface()	77
5.3	TNativeSurface	78
5.4	ProfileFilterInterface()	78
5.5	TNativeProfile	78
5.6	GetCounts()	79
5.7	GetItem()	79
5.8	TProcLib	79
5.9	fnISO4287()	80
5.10	STRISO4287	80
5.11	fnISO13565_2()	81
5.12	STRISO13565_2	81
6.1	P-profile parameters (profile:Gn_56, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)	84
6.2	W-profile parameters (profile:Gn_56, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)	85
6.3	R-profile parameters (profile:Gn_56, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)	86
6.4	R-profile parameters (profile:normal, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)	87
6.5	Abbott curve parameters (profile:normal, filter:Gaussian, Lc=0.8mm, Ls=2.5 μm)	87

A Glossary

MTS	Measurement and Sensor Technology
DIN	Deutsche Institut für Normung
ISO	International Organisation for Standardisation
TLS	Total Least Square
BS	British Standard
NIST	National Institute of Standards and Technology
PTB	Physikalisch Technische Bundesanstalt
NPL	National Physical Laboratory
PSD	Power spectral density
SDF	Surface Data File
LSM	Least Squares Method
SVD	Singular Value Decomposition
RMS	Root Mean Square
HTTP	Hypertext Transport Protocol
CSS	Cascading Style Sheets
HTML	Hyper Text Markup Language
JDK	Java Development Kit
IDE	Integrated Development Environment
JNI	Java Native Interface
JVM	Java Virtual Machine
JSON	JavaScript Object Notation
AJAX	Asynchronous JavaScript and XML
XML	Extensible Markup Language
DOM	Document Object Model
MVVM	Model View ViewModel
UI	User Interface

- MVC** Model View Controller
- JAR** Java Archive File
- PNG** Portable Network Graphics
- JPEG** Joint Photographic Experts Group
- PDF** Portable Document Format
- FOP** Formatting Objects Processor
- XSL** EXtensible Stylesheet Language
- DLL** Dynamic Link Library
- PDF** Portable Document Format
- JNA** Java Native Access
- AWGN** Additive White Gaussian Noise

B Appendix

B.1 M-script for total least square fit

```
(X,Y) = datapoints;
```

```
%Mean/ Average
```

```
N=length(Y);
```

```
Xc=mean(X)
```

```
Yc=mean(Y)
```

```
for i=1:N
```

```
    x(i)=X(i)-Xc;
```

```
    y(i)=Y(i)-Yc;
```

```
    sxy(i)=x(i)*y(i);
```

```
end
```

```
sx=(x.^2);Mxx=sum(sx);
```

```
sy=(y.^2);Myy=sum(sy);
```

```
Mxy=sum(sxy);
```

```
%Matrix
```

```
H(1,1:2) = [Myy -Mxy];
```

```
H(2,1:2) = [-Mxy Mxx];
```

```
%SVD
```

```
[U, Wh, Vh] = svd(H, 0)
```

```
Wh=diag(Wh); %Singular values (Wh); Orthogonal vectors (Vh)
```

```
distance=min(Wh); %Minimum of the singular value
```

```
mini=find(Wh==distance);
```

```
R(1,1)=Vh(1,mini); %min column of Vh=R
```

```
R(2,1)=Vh(2,mini)
```

```
%Error variance
```

```
sigma_line=sqrt(distance/N)
```

```
%Straight line
```

```
for i=1:N
```

```

Xnew1(i) = (((Y(i) - Yc) * R(1)) / R(2)) + Xc;
Ynew1(i) = (((Xnew1(i) - Xc) * R(2)) / R(1)) + Yc;
end

```

B.2 M-script for circular fit

```

(xx,yy) = datapoints;
N=length(xx);
for i=1:N
    x(2*i-1)=xx(i);
    x(2*i+0)=yy(i);
end
xlen= length(x)/2;

%Mean/ Average
Xc=mean(xx)
Yc=mean(yy)
R=0;
for i=1:xlen
    R=R+((x((i*2)-1)-Xc)*(x((i * 2)-1) - Xc)+
        (x((i*2)+0)-Yc)*(x((i * 2)+0)-Yc));
end

R=sqrt(R/xlen)
Ri=R;
xci=Xc;
yci=Yc;

th = 0:pi/N:2*pi;
xunit=Ri*cos(th)+xci;
yunit=Ri*sin(th)+yci;

for i=1:xlen
    dx=x((2* i)-1)-Xc;
    dy=x((2* i)+0)-Yc;

    %Calculate square norm
    sqr_norm=(dx*dx)+(dy*dy);
    norm=sqrt(sqr_norm);

```

```

%Calculate orthogonal error distance
Xerr((2 * i) - 1) = (norm - R) * (x((2 * i) - 1) - Xc) / norm;
Xerr((2 * i) + 0) = (norm - R) * (x((2 * i) + 0) - Yc) / norm;
Rnorm = R / norm;

%Matrix
dda(1, 1:3) = [0 1 0];
dda(2, 1:3) = [0 0 1];

%Matrix
dX0(1, 1:3) = [dx / norm 0 0];
dX0(2, 1:3) = [dy / norm 0 0];

%Matrix
ID(1, 1:3) = [0((1 - dx * dx / sqr_norm) * Rnorm)((0 - dx * dy / sqr_norm)
              * Rnorm)];
ID(2, 1:3) = [0((0 - dx * dy / sqr_norm) * Rnorm)((1 - dy * dy / sqr_norm)
              * Rnorm)];

%Jacobian matrix
for k = 1:2
    for j = 1:3
        J((2 * i) + (k - 2), j) = dda(k, j) + dX0(k, j) - ID(k, j);
    end
end

end

for i = 1:N
    Ox(i) = Xerr((2 * i) - 1);
    Oy(i) = Xerr((2 * i) + 0);
end

JXFa(1) = 0;
JXFa(2) = 0;
JXFa(3) = 0;
for j = 1:2 * xlen

```

```

    JXFa(1)=JXFa(1)+J(j,1)*Xerr(j);
    JXFa(2)=JXFa(2)+J(j,2)*Xerr(j);
    JXFa(3)=JXFa(3)+J(j,3)*Xerr(j);
end

%Singular value decomposition(SVD) of Jacobian matrix
[u, s, v]=svd(J, 0)

XFa(1)=0;
XFa(2)=0;
XFa(3)=0;
for j=1:2*xlen
    XFa(1)=XFa(1)+J(j,1)* Xerr(j);
    XFa(2)=XFa(2)+J(j,2)* Xerr(j);
    XFa(3)=XFa(3)+J(j,3)* Xerr(j);
end

delta_R=v(1,1)*XFa(1)/s(1,1)+v(1,2)*XFa(2)/s(2,2)+v(1,3)
        *XFa(3)/s(3,3);
delta_Xc=v(2,1)*XFa(1)/s(1,1)+v(2,2)*XFa(2)/s(2,2)+v(2,3)
        *XFa(3)/s(3,3);
delta_Yc=v(3,1)*XFa(1)/s(1,1)+v(3,2)*XFa(2)/s(2,2)+v(3,3)
        *XFa(3)/s(3,3);

step=1.3;
R=R+step*delta_R;
Xc=Xc+step*delta_Xc;
Yc=Yc+step*delta_Yc;

for i=1:xlen
    %Calculate orthogonal error distance with new R,Xc,Yc
    dx=x((2*i)-1)-Xc;
    dy=x((2*i)+0)-Yc;

    %Calculate square norm
    sqr_norm=(dx*dx)+(dy*dy);
    norm=sqrt(sqr_norm);

```



```

Xerr((2*i)-1)=(norm-R)*(x((2*i)-1)-Xc)/norm;
Xerr((2*i)+0)=(norm-R)*(x((2*i)+0)-Yc)/norm;
end

newdata=length(x)
newdata(1,1:2)=[((Xerr(1)/abs(Xerr(1)))*sqrt((Xerr(1)^2)
+(Xerr(2)^2))) 0]
for i=1:xlen*2
    nearestCirclePt1(2)=x((2*i)-1)+Xerr((2*i)-1)
    nearestCirclePt2(2)=x((2*i)+3)+Xerr((2*i)+3)

    nearestCirclePt1(1)=x((2*i)-0)+Xerr((2*i)-0)
    nearestCirclePt2(1)=x((2*i)+2)+Xerr((2*i)+2)

    nearestCirclePt1Angle=abs(atan2(nearestCirclePt1(2)-Yc,
    nearestCirclePt1(1)-Xc)-pi)
    nearestCirclePt2Angle=abs(atan2(nearestCirclePt2(2)-Yc,
    nearestCirclePt2(1)-Xc)-pi)

    Xd=(nearestCirclePt1Angle-nearestCirclePt2Angle)*R
    newdata(i+1,2)=newdata(i,2)+Xd;
    newdata(i+1,1)=(Xerr((2*i)-1)/abs(Xerr((2*i)-1)))*
    sqrt(Xerr((2*i)-1)^2+Xerr((2*i)-0)^2)
end

for i=1:xlen
    X_new(i)=newdata(i+1,1);
    Y_new(i)=newdata(i+1,2);
end

variance=0;
for k=1:xlen
    variance=variance+(Xerr(k)*Xerr(k));
end
sigma=sqrt(variance)

```

B.3 Play framework deployment settings

Once the web application is developed using Play framework, it is important to deploy the application as a Windows service. The procedure for Windows service setup are as follows [Sta12]:

1. To start with, create folder *C:\myapp*
2. Go to play application folder (example: ReferenceSoftware) in command line and type *play dist*
3. A folder is created "Name-SNAPSHOT" in the application folder, which is to be added into the folder *C:\myapp*
4. Download YAJSW and extract it to *C:\myapp*
5. In *C:\myapp\Name-SNAPSHOT* add a new file called *start.bat* and fill it with command like :

```
java -cp "C:\myapp\Name-SNAPSHOT\lib\*" -DapplyEvolutions.default=true  
play.core.server.NettyServer
```
6. Save the file and launch it. The application must start and work as usual over localhost:9000. Note the servers PID which is displayed on the command line, which is to be used in the next step.
7. Using command prompt go to *C:\myapp\yajsw\bat* and type
genConfig.bat < PID from previous step >
8. This creates *C:\myapp\yajsw\conf\wrapper.conf*, edit and save this file as
wrapper.nts.service.name=My App Name
wrapper.nts.service.displayname=My App Name
9. Terminate the *start.bat* script, which was launched at step 6 and in *C:\myapp\yajsw\bat* launch *runConsole.bat*. If the application starts, terminate the file and proceed to step 10.
10. Install service launch (Run as administrator)
installService.bat
11. Start service launch
startService.bat

The above steps allow the application to run as a window service. To stop the window service, launch *stopService.bat*.

B.4 Java Native Interface (JNI)

The steps involved in it are as follows:

1. Java source code is created with native methods, i.e. declares a native method with native keyword and an empty method body. At the native language side, provides an implementation for the native method.
2. Java source code is compiled to obtain the class files (**javac**).
3. Generates C/C++ header for the native methods (**javah**).
4. C/C++ source code is written for the native methods. The native method function definition in the implementation must match the function prototype generated in the header file.
5. Source code is compiled with the right header files.
6. Linker is used to create a dynamic linked library file.
7. Finally the java code is executed with the dynamic library loaded at the beginning of the code.
8. Once the code is executed .jar files are created and saved in the projects library folder.

Bini A. George

M.Sc.

Objective

- To obtain total growth professional and personal.
- To secure a challenging job benefiting my qualifications and skills.

Occupation

2014 - present **Smiths detection**, Wiesbaden, Germany.
Title: Software Engineer

Education

- 2009 - 2017 **PhD in the Institute of Measurement and Sensor Technology (MTS)**, TU Kaiserslautern, Germany.
Research Area: Surface Metrology
- 2007-2009 **M.Sc in Electrical Engineering**, TU Kaiserslautern, Germany, Grade: 2.0.
Master in Digital Systems
- 2002-2006 **Bachelor degree in Electronics and communication Engineering**, Sambhram Institute of Technology, Bangalore-India, Grade: 67.45%.
Thesis title: Pressure monitoring and data recording system

PhD thesis

Title *"Web-Based Reference Software For Characterisation Of Surface Roughness"*
Doctoral Advisor Prof.Dr.-Ing J. Seewig

Summary Roughness plays an important role in determining how a real object interacts with its environment. Its a good predictor of the performance of mechanical components, but its difficult and expensive to control in manufacturing. The evaluation of surface roughness on technical functional areas and the comparison of roughness parameters with tolerances is an integral part of industrial quality assurance. The aim of the project is to develop user friendly software which makes the evaluation methods for verification of surface characteristics easier for clients. The algorithms used in the software are based on the International ISO standards. Most of the software used in industrial measuring instruments may give variations or differences in the parameters calculated. Such variations can be verified using the proposed reference software.

Master thesis

Title *Untersuchung zu Steuerungskonzepten zur Synchronisation elektrischer Antriebe.*
Supervisors Prof.Dr.-Ing N. Wehn

Description Project on automobile seat adjustment-motor control system. Involves several hardware and software control concepts. Task includes building circuits for various testing cases that would help control and synchronize the motors, System integration, PCB design using EAGLE (Decentralised motor controller, size 33mm*68mm with SMD components).

Experience

Research Assistant, T.U Kaiserslautern, Germany, (2009-2010)

- Research in the field of Surface Metrology
- Detection of chattermarks in nano meter range
- Measurements using industrial instruments (NanoFocus, Mahr MMQ, ISIS sensor)
- Algorithms for characterisation of surface roughness (MATLAB)
- Web application using Java Servlet, Apache Tomcat Server, JNA
- Web based reference software (JAVA), with Eclipse as IDE

Student Assistant, KEIPER GmbH, Germany, (2007-2009)

- PCB design (EAGLE)
- Circuit testing
- System integration

Skills

Tools	MATLAB, Maple, CorelDraw, Eagle PCB design
OS	MS Windows, Linux
Applications	Latex, MS Office
Programming	Java, C++
Web	HTML, JavaScript, CSS, Java Servlets, PrimeFaces
Languages	Working knowledge of English, German(B2), Kannada, Hindi and speaking awareness of Malayalam and Tamil.

Strength

- Ability to work both independently and collectively to achieve objective
- Communication skills enthusiastic and positive

Perspective in life

- Perseverance is the essence of life. I draw inspiration from the fact that continuous drops of water can create an impression on the surface of a rock. It is not force but sheer perseverance!

Publications

Robust evaluation of intensity curves measured by confocal microscopies- Seewig, Jörg; Raid, Indek; Wiehr, Christian; George, Bini A.; Proc. SPIE 8788, (May 13, 2013).