

Politecnico di Torino

Corso di Laurea Magistrale in Ingegneria Meccanica

Tesi di Laurea Magistrale



“Statistical processing and analysis of surface
topography data for a machine-based evaluation
of the measurement uncertainty”

Relatore

prof. Gianfranco Genta

Corelatore

prof. Danilo Quagliotti

Candidato

Matteo Gilardi

Anno Accademico 2019/2020

ABSTRACT

Surface measurements in industrial activities are required principally to check tolerances or to characterise the surface functionality of components. There are many engineering application in which surface metrology carry out an important role (e.g. electronics, information technology, energy, optics, tribology, biology, biomimetics, etc.), but focusing only on autonomous manufacturing processes, in a perspective of what is called Industry 4.0, the areas of interest are the in-line quality check and the analysis of additive manufacturing processes with the aim of understanding and governing them. To achieve these, it is required that the measurement is acquired and processed quickly, hence, recently the optical instruments have been adopted more and more. Measurement to be reliable has to be coupled with its uncertainty; however rarely areal surface measurements presents this value. Indeed, a standard infrastructure for the traceability of areal surface measurement is still missing. This is probably due to both the complexity of the measurand and the optical instruments, whose interaction with the component is still not completely known. Hence, this project aims to cope with the complexity of such measurands by exploring applicability and limitations of methods for an autonomous, machine-based, statistical evaluation of the measurement uncertainty. First of all, the surfaces are processed in order to remove the form and manage possible non-measured points and/or spikes. The non-measured points are replaced with the values of the fitting surface, extracted during the form removal stage, of the input data. The spikes, instead, are managed with a threshold method that limits the peaks and the valleys at three times the S_q value (Root Mean Square Height of the scale-limited surface). Spikes are not intentionally managed with statistical methods because they are principally due to the instrument systematic behaviour for the most. By contrast, outliers are due to the operator for the most and are rare, and not systematic. Carried out these preliminary operations, the measurements are subjected to the statistical analysis, which considers the repeated measurements of

each pixel. Both the Chauvenet's criterion and the modified interquartile range method have been implemented to detect and manage the outliers. However, after an analysis of the result, we decided to proceed without managing them. In fact, without filtering the measurements, outliers are mixed with measurement noise, therefore, the classical methods cannot model this condition. Finally, a possible systematic effect in the measurements of the workpiece is detected through the linear regression taking into account the correlation domain of each pixel. Once correcting this effect, the uncertainty related to the measurements of the workpiece is estimated. As far as the uncertainty is concerned, to achieve the complete uncertainty estimate, the measurements of the calibrated artefacts are required in order to take into account the effect of the instrument. However, because of the pandemic of COVID-19, we could not carry out these measurements but the model equation, inspired to both the ISO 14253-2 and the ISO 15530-3, is provided. This equation, applying the law of uncertainty propagation, allows the estimate of the expanded uncertainty of the measurement.

ACKNOWLEDGMENTS

The author acknowledges Ing. Maculotti Giacomo for the support in both the implementation of the algorithms and in writing this written work.

The Computational resources were provided by HPC@POLITO, a project of Academic Computing within the Department of Control and Computer Engineering at the Politecnico di Torino (<http://hpc.polito.it>).

TABLE OF CONTENTS

ABSTRACT.....	i
AKNOWLEDGMENTS.....	iii
LIST OF ABBREVIATIONS	vi
LIST OF FIGURES	vii
LIST OF TABLES	x
1. INTRODUCTION	1
1.1. State of the art.....	1
1.2. Cluster computing	9
1.3. Structure of the thesis	11
2. SURFACE METROLOGY.....	13
2.1. Introduction to surface metrology	13
2.2. Surface topography characterization	14
2.3. Instrument used for measurements.....	26
2.3.1. Coherence scanning interferometry (CSI)	26
2.3.2. Imaging Confocal Microscopes (ICM).....	29
3. METHODOLOGY	31
3.1. Software environment definition.....	31
3.2. Preliminary management of topography	33
3.2.1. Plane correction	33
3.2.2. Areal field parameters.....	36
3.2.3. Spikes and voids management.....	37
3.3. Statistical analysis	40
3.3.1. Outliers' identification and management.....	40
3.3.2. Normality tests.....	43

3.3.3.	Correction of possible systematic behaviour	47
3.4.	Uncertainty evaluation	54
3.5.	Data reporting and visualization.....	62
4.	RESULTS	67
5.	DISCUSSION	71
6.	CONCLUSIONS	73
	ANNEX A - MATLAB Code.....	75
	BIBLIOGRAPHY	103

LIST OF ABBREVIATIONS

AM	Additive Manufacturing
AW	Amplitude-Wavelength
CASPER	Cluster Appliance for Scientific Parallel Execution and Rendering
CMM	Coordinate Measuring Machine
COVID	Coronavirus Disease
CSI	Coherence Scanning Interferometry
DTU	Technical University of Denmark
HPC	High Performance Computing
ICM	Imaging Confocal Microscope
IQR	Interquartile range
ISO	International Organization for Standardization
JCGM	Joint Committee for Guides in Metrology
MRSes	Metrological Reference Surfaces
NPP	Normal Probability Plot
PSI	Phase Shift Interferometry
SEM	Scanning Electron Microscope
SSE	Error Sum of Squares
SSR	Regression Sum of Squares
SST	Total Sum of Squares
VIM	International Vocabulary of Metrology

LIST OF FIGURES

Figure 1.1.1 – Example of a structured surface.	2
Figure 1.1.2 – Schematic representation of the spray cooling processes.	3
Figure 1.1.3 – Example of freeform surface.	4
Figure 1.1.4 – Examples of “bio-inspired” functional surfaces.	5
Figure 1.1.5 - A water drop on a tilted superhydrophobic surface.	5
Figure 1.1.6 – Examples of some complex geometry realizable through additive manufacturing.	7
Figure 1.2.1 – Schematic differences between serial and parallel computing.	9
Figure 1.2.2 - Technical specification of CSPER system of HPC cluster at Politecnico of Turin.	10
Figure 2.2.1 - Causes of deviation in a surface.	15
Figure 2.2.2 – Filters wavelengths representation according to ISO 4287.	16
Figure 2.2.3 – Representation of the different lengths utilized for profile analysis.	17
Figure 2.2.4 – The difference in information between the areal and the profile method.	18
Figure 2.2.5 – Relationship between the different areal filters.	21
Figure 2.2.6 – Filtration scheme of areal surface measurements.	22
Figure 2.2.7 – AW space plot for three common instruments.	24
Figure 2.3.1 – Imaging interferometer for areal surface profiling.	27
Figure 2.3.2 – CSI signal for a single pixel.	28
Figure 2.3.3 – Geometry of an interference microscope suitable for CSI.	29
Figure 2.3.4 – Basic setup of a laser scanning microscope.	30
Figure 3.1.1 – Computational time differences between serial and parallel computing.	32
Figure 3.2.1 – Least-squares fit result.	34
Figure 3.2.2 – How the same surface looks before and after the plane correction.	35
Figure 3.2.3 – Voids identification on the surface.	38

Figure 3.2.4 – Spikes identification on the surface.....	39
Figure 3.3.1 – Boxplot.	41
Figure 3.3.2 – Scatter plot of the repeated measurements of a pixel before and after the outliers' management with both Chauvenet's criterion and modified IQR method.	42
Figure 3.3.3 – χ^2 distribution at different degrees of freedom (ν).	44
Figure 3.3.4 – χ^2 results for a map of pixels at a confidence level $P = 95\%$	45
Figure 3.3.5 – NPP of a Hyper- and Hypo-normal distribution.	47
Figure 3.3.6 – Representation of a data arrangement in order of measurement collection and a linear regression.....	48
Figure 3.3.7 – Possible values of Pearson's coefficient ρ	51
Figure 3.3.8 – Examples of different correlation lengths for some pixels in the grid, represented by a darker colour.....	52
Figure 3.3.9 – Focus on the components of the coefficient of determination.	53
Figure 3.4.1 – Traceability infrastructure.	55
Figure 3.4.2 – Different probability distribution type for the analysis of type B standard uncertainty.....	58
Figure 3.5.1 – Representation of how the surface has been defined in the software.	62
Figure 3.5.2 – Voids identification on the surface.....	63
Figure 3.5.3 – Spikes (a) and outliers (b) identification on the surface.....	63
Figure 3.5.4 – Results representation of the coefficient of determination of Zygo measurements.....	64
Figure 3.5.5 – Result representation of the significance test of Zygo measurements.	64
Figure 3.5.6 – Uncertainty band representation from two points of view of Zygo measurements.....	65
Figure 4.1.1 - Result representation of the significance test of Olympus measurements.....	68
Figure 4.1.2 – Results representation of the coefficient of determination for Olympus measurements.....	68

Figure 4.1.3 – Uncertainty band representation from two points of view of Olympus measurements.....	69
-----------------------------------------------------------------------------------------------------	----

LIST OF TABLES

Table 1 – Profile method ISO standards.	19
Table 2 – List of ISO 25178 parts.....	20
Table 3 – Similarity requirements for the workpiece and the calibrated standard [35].....	59
Table 4 – Surface characterization results.	67

1. INTRODUCTION

1.1. State of the art

Over the years, as knowledge of classical physical phenomena has been acquired, the micro and nano length scales attracted increasing attention in trying to study and govern what occurs at such level. Nowadays, the knowledge of surface phenomena is fundamental to understand and control many fields at micro and nano level, such as electronics, information technology, energy, optics, tribology, biology, biomimetics, etc. In this perspective, surface metrology plays a significant role, which requires indeed traceable measurement processes.

Although modern surface topography measuring instruments are widely developed, the achievement of the traceability and the assessment of the measurement uncertainty are still lacking of prevailing reliable guidelines to deal with areal topographic measurements [1]. Hence, this project aims to cope with the complexity of such measurands by exploring applicability and limitations of methods not commonly acknowledged in the field, thus paving the way for an autonomous, machine-based, statistical evaluation of the measurement uncertainty, which could benefit of new as well as already existing models for instruments' performance estimate (e.g., the metrological characteristics [2]).

Speaking of surfaces, two broad types can be introduced:

- The structured surfaces, defined as surfaces with a deterministic pattern, made of geometric features usually with high aspect ratio, designed to give a specific function, see figure 1.1.1.
- The engineered surfaces, where the manufacturing process is optimized to generate variation in the geometry and/or in the near-surface material properties to give a specific function [3].

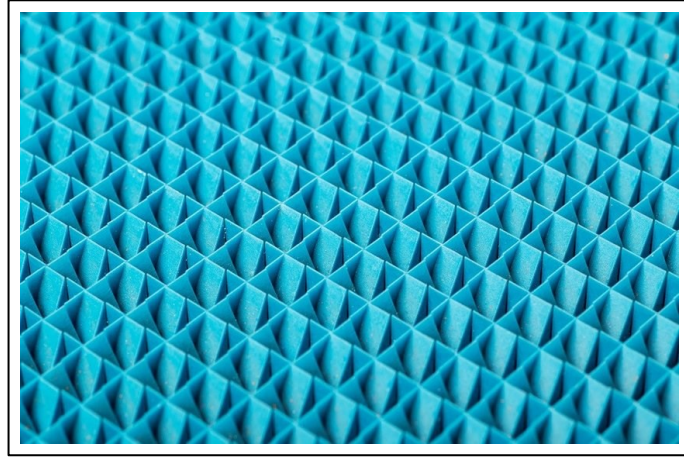


Figure 1.1.1 – Example of a structured surface.

As stated by Brinksmeier et al. [4], the structured surfaces recently evolved into multiscale structured surfaces, which consist of at least two hierarchical structures of intentionally generated features forming a pattern (with repeating or periodic elements) whose dimensions (lateral and vertical) differ considerably. The aim is to achieve combined functionalities and/or distinct functions that supplement each other. Such surfaces are also known as “bio-inspired” functional surfaces because numerous applications have their origin of inspiration in the natural world, and rely on a multitude of hierarchically stacked structures. An accurate and effective hierarchical-information-based characterization method for multiscale structured surfaces has been provided by Cheung et al. [5].

Structured surfaces play a fundamental role in the energetic field, especially in spray cooling and boiling systems. The spray cooling technology, whose process is shown in figure 1.1.2, lets the dissipation of high thermal flux with a rather low coolant mass flux, hence, increasing the efficiency of the whole process. Among the factors that influences the spray heat transfer, there also is the surface roughness. After many pieces of research, it has been noticed that controlling the surface texture may increase the heat transfer coefficient value. Many authors already remarked on the complexity of the phenomenon, and the need for further research. This technology is applied, for instance, in the aerospace field, supercomputing cooling systems, in metal quenching, in medical treatments, etc. [6]. Moreover, the boiling heat transfer is exploited in domestic and industrial applications, in which

thermal power transfer is required. Surface topography affects the boiling process, in fact, many boiling surfaces appropriate designs have been studied and then developed in order to facilitate the heat transfer.

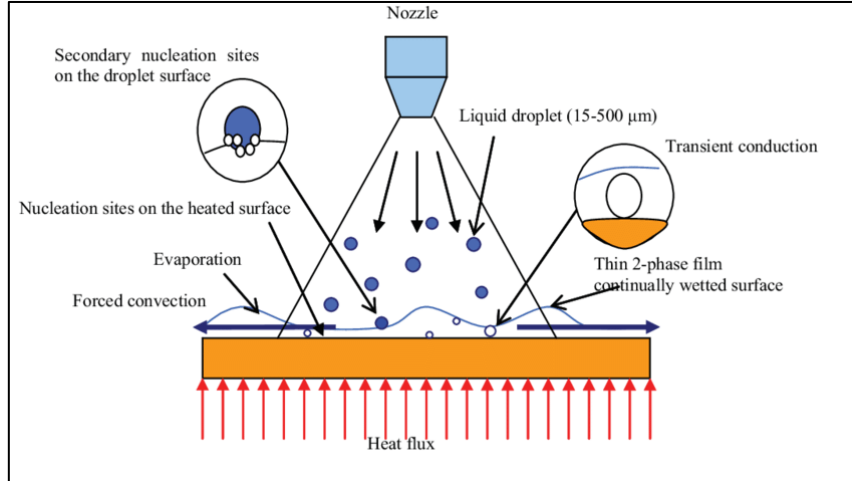


Figure 1.1.2 – Schematic representation of the spray cooling processes [7].

Concerning the optical field, the micro-optics are examples of application area in which the control of surface texture holds a principal role. The increasing diffusion of micro-optics is due to the shift from refractive to diffractive optical elements to fulfil miniaturizing requirements promoted by some industries, such as communications, computing, consumer electronics, automotive and aerospace [6]. The increasing request for high-quality optical components induced the development of new polymers, which could also be used for replacing conventional glass, thus achieving high efficiency, low cost and automation. Many optical lenses are multi-structured surfaces, currently made of special resins with characteristics such as low density, impact resistance and low cost [8].

The surface analysis of optical element is usually performed on very small portions of the overall component. Therefore, such parts have geometrical form, always almost planar, which is irrespective of the shape of the overall geometry. However, in modern optical systems it is quite common to work with asphere and freeform surfaces (figure 1.1.3). While uncertainty in any structured surfaces can be evaluated using instruments' metrological characteristics, as described by

MacAulay et al. [9], for asphere and freeform surfaces well-known artefacts are required in order to calibrate the measurement instruments. At this regard, Fortmeier et al. [10] proposed the design of artefacts named metrological reference surfaces (MRSes), with characteristic features that would allow to decompose freeform features into measurable quantities.

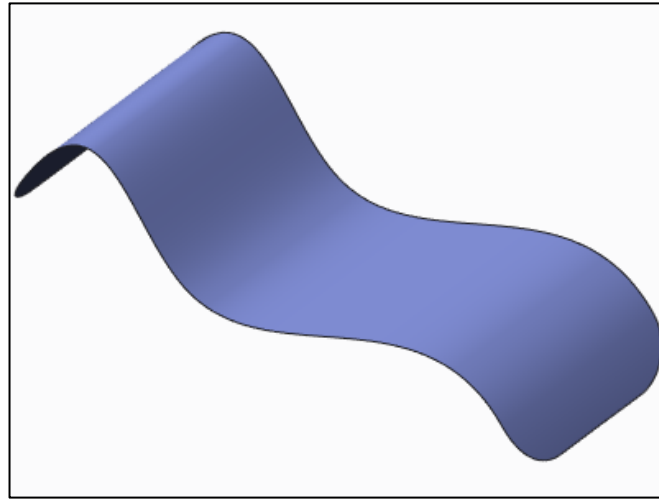


Figure 1.1.3 – Example of freeform surface.

Another typical function for surface texture is the wettability. Many generated surfaces were inspired to the lotus effect for controlling the wetting of roof tiles, paint coatings, fabrics and other surfaces that require to be dry and clean. This technology is not only related to cleanliness but is also related to safety, in fact, current studies are oriented in resistance to fingerprints [6]. The lotus effect is the capability of certain materials to self-clean, and it results in superhydrophobicity. The micro- and nano-scopic architecture on a surface, acting on wetting, minimizes the droplet's adhesion to that surface, as described in figure 1.1.4, so that dirt particles are picked up by water droplets. Currently, the focus area in wettability studies is a technology for manufacturing lotus-liked structured surfaces that can switch wettability between superhydrophobicity and superhydrophilicity [11].

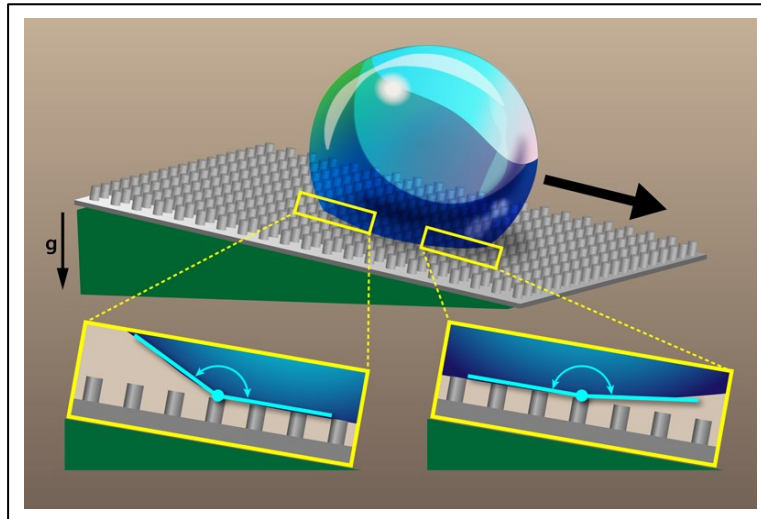


Figure 1.1.4 - A water drop on a tilted superhydrophobic surface [12].

Many application areas of structured surfaces with multiscale features are the same of structured surfaces but the application of multiscale features amplify the properties. The most common properties of interest are the adjustment of adhesive properties (gecko feet), the control of the surface wetting state (lotus leaf), improvement of tribological behaviour (snake or shark skin), and the use of optical functions such as anti-reflection (moth eye) or color effects (butterfly wings), see figure 1.1.5 [4].

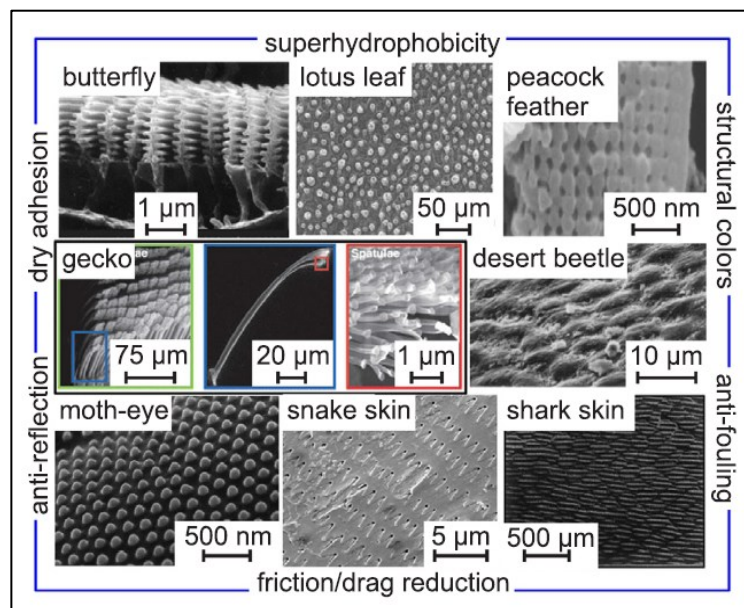


Figure 1.1.5 – Examples of “bio-inspired” functional surfaces [4].

The control of surface apparent colour and optical reflectivity is allowed through plasmonic colours, which are based on the resonant interactions between light and metallic nanostructures. Because of the increasing interest in the engineering of plasmonic colours, Kristensen et al. [13], once described their basic properties, present recent nanofabrication developments in the area. Yang et al. [14], instead, introduce a new production method based on elliptical vibration texturing to efficiently colour metallic surfaces with periodic micro/nano-gratings.

Nowadays, a large choice of optical measurement equipment offers a number of advantages for areal topography measurements. Nevertheless, such instruments are to cope not only with the challenges of the before mentioned functionalities, but also with the digitalization and Big Data analytics of Industry 4.0. Central is also to support the continuous development of processes, such as Additive Manufacturing techniques (AM), where the exploitation of the full potentiality requires an adequate assessment of the measurement uncertainty, which indeed takes into account the 3D and free-shape natures of AM processes [15–16]. The main advantage of AM is the capability to realize complex geometry and internal features that could not be obtained through conventional manufacturing, as can be seen in figure 1.1.6. In addition, the reduced time-to-market, due to a lower set-up, tooling and assembly time, is convenient for the affordability of the manufacturing process. By contrast, the current limitations are due to the difficulties in dimensional controlling and surface integrity of specific surfaces. Particularly, surfaces are characterized by high roughness caused by the layer-by-layer nature of the process. Additive Manufacturing Technologies would be capable of increasing productivity if the surface quality and overall quality are improved.



Figure 1.1.4 – Examples of some complex geometry realizable through additive manufacturing.

To analyse surface conditions and to report feedback to the machine for closed-loop control a reliable measurement system is necessary, hence, Syam et al. are investigating an in-process surface detection approach for polymer AM parts [17]. The surface topography metrology in AM processes can be applied for both the compliance verification to specific surface texture requirements and the acquirement of information about physical phenomena occurring during an AM process. At the moment, such research is at an early stage of development, and a systematic approach is still under definition. The stylus-based contact instrument is still the measurement instrument most frequently chosen. Recently, computer tomography and optical instruments are preferred in surface metrology. For what concerns optical instruments, the most utilized in the field of AM are the focus variation microscopy, the confocal microscopy and the coherence scanning interferometry, being the latter less utilized because not very suitable for the highly irregular AM surfaces [18]. AM is becoming an important resource to conventional manufacturing technology and, thus, the measurement and the characterization of surface texture may assume a leading role in further growth.

Over the last few years, many methods to produce engineered surfaces have been developed in order to allow surface functionality. These methods can be categorized according to their physical principles as follow [6]:

- Adding material: the desired functions are obtained through the addition of material, creating small areas of relief.
- Removing material: the definition of specific characteristics is achieved by the removal of material of the surface, creating small depressions.
- Moving material: the change in the surface structure is caused by plastic deformation and redistribution of material from some parts of the surface to others.
- Self-forming: wear-resistant regions are formed on a surface in order that texture develops through the wear of the surface; this happens because these areas wear more rapidly than the surrounding material.

The potential of structured and engineered surfaces is high, but they must face the complexity of the relationships between surface properties, surface engineering technologies and issues related to the metrology.

To conclude, with the increasing application of the surface metrology in many fields (e.g., the ones mentioned above), the necessity of a quick but reliable estimate of the measurement uncertainty is crucial. Therefore, this project aims to establish an architecture of algorithms, in which advanced statistical processing and uncertainty evaluation are to be applied to surface topography measurements and translated into algorithms suitable for a machine implementation. In doing so we also ask ourselves whether the classical statistical methods are able to model the problem and how pixels interact with each other. Moreover, we are interested in understanding whether a separation between areal-parameter computation and pixel by pixel evaluation can be feasible.

1.2. Cluster computing

Due to the increasing computational time, caused by both the complexity of the calculi and the size of data, cluster computing together with parallel computing has recently become a hot topic of research among academic and industry. Clusters are based on communication between nodes; hence the design of a fast and low latency network is a must [19]. A node is a hardware device that communicates with the other devices in the network (e.g., physical servers). Clusters are an excellent platform for high data set processing and storing. This technology aims to break up a complex ‘job’ (i.e., a complex set of computational operations) into much more simple and manageable jobs that will be solved in parallel (i.e., concurrent computation) in the cluster’s nodes, see figure 1.2.1.

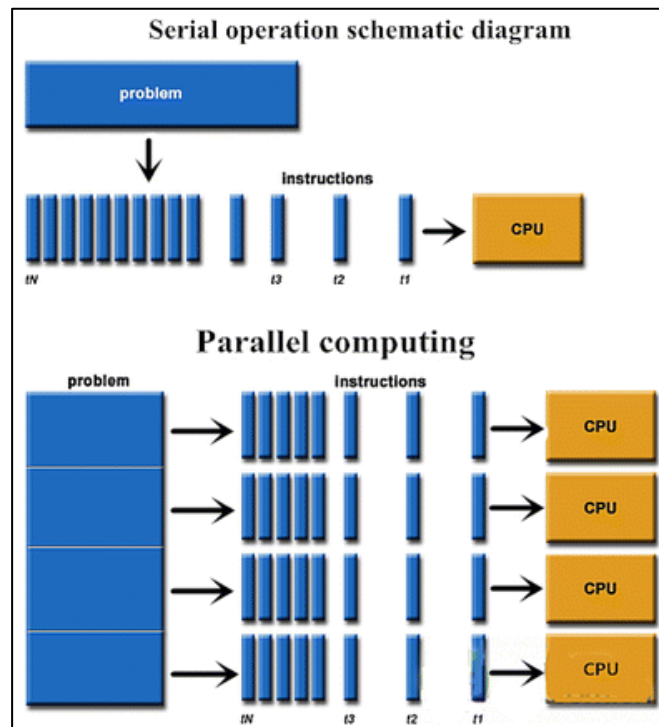


Figure 1.2.1 – Schematic differences between serial and parallel computing [20].

Nowadays, clusters have become fundamental in many applications in both scientific and commercial areas. The usage of clusters allows to pursue the supercomputer computational power in scientific applications, such as earthquakes

or hurricanes predictions, complex crystallographic and microtomographic structural problems, protein dynamics and biocatalysis, relativistic quantum chemistry of actinides, virtual materials design and processing including crash simulations, and global climate modelling [19]. For industry applications, as the Industry 4.0 is becoming a reality, specific computational systems for the management of BigData are requested. The technology of BigData needs to analyze vast amounts of production data generated in the process of quality assurance and completion state control in a smart factory [21].

For this project analyses, the cluster HPC (High Performance Computing) of the Politecnico of Turin has been used. The system utilized was CASPER, an acronym of Cluster Appliance for Scientific Parallel Execution and Rendering, which is made of 16 nodes and all the technical specifications are listed in figure 1.2.2 [22]. DTU has also cluster computing availability. Complete information can be found on the dedicated webpage [23].

Architecture	Linux Infiniband-DDR MIMD Distributed Shared-Memory Cluster
Node Interconnect	Infiniband DDR 20 Gb/s
Service Network	Gigabit Ethernet 1 Gb/s
CPU Model	2x AMD Opteron 6276/6376 (Bulldozer) 2.3 GHz (turbo 3.0 GHz) 16 cores
Sustained performance	4.360 TFLOPS (last update: 2014)
Peak performance	5.658 TFLOPS (last update: 2014)
Green500 Index	422.31 MFLOPS/W
Power Consumption	3.6 kW
Computing Cores	512
Number of Nodes	16
Total RAM Memory	2 TB DDR3 REGISTERED ECC
OS	Centos 7.6 - OpenHPC 1.3.8.1
Scheduler	SLURM 18.08.8

Figure 1.2.2 - Technical specification of CASPER system of HPC cluster at Politecnico of Turin [22].

1.3. Structure of the thesis

Chapter 1 describes what is the state of the art in the surfaces field and briefly introduces the cluster computing technique. In the description of the state of the art, many industrial applications related to the control and analysis of surfaces are listed. Concerning cluster computing, in its section are also introduced the systems available for the computational analysis of this project. The second chapter includes a picture of the main aspects of surface metrology, considering both the profile and areal method, and of the measurement instruments that have been employed in this project. Chapter 3 opens with some information about how the software has been set and how a parallel computation is required to reduce computational time. Then, all the aspects treated in the software are here described. In the section named 'Preliminary management of topography', the plane correction, the calculation of some areal field parameters and the management of spikes and non-measured points (voids) are presented. The section called 'Statistical analysis', instead, illustrates the methods to detect outliers and connected problems. Moreover, it introduces the normality tests and the linear regression to search for possible systematic behaviour. Finally, still in chapter 3, a method for the estimation of the measurement uncertainty is illustrated and the data are reported and visualized in the section of the same name. It follows the chapter concerning results, discussion and conclusions.

2. SURFACE METROLOGY

2.1. Introduction to surface metrology

From an engineering point of view, surface metrology is the science that measures the difference of a workpiece from its designed shape. Because of the presentation of products to the customer gives the impression of quality, surfaces should meet both functional and aesthetic requirements. In engineering, the quality of a surface is often measured in terms of performances. Knowledge of surface topography is an essential ingredient to prevent component failures, which is often related to both the material characteristics and surface topography [24]. The surface topography, as well as the material features, of a workpiece, can affect how the part appears and feels, how two mating parts slide together or how fluids interact with the piece. Indeed, the characterization of functional surfaces, and of their quality, is essential for the inspection of micro- and nano-products in the manufacturing industry. In particular, the surface roughness is fundamental in determining the performance of a workpiece in many advanced engineering fields, such as electronic, information technology, energy, optics, tribology, biology, etc. Once understood how surface texture affects the performance, it is possible to design them with specific geometric characteristics. Moreover, surface metrology in an automated manufacturing perspective is becoming a powerful tool for quality controls. Controls can be done either in-line or off-line with both their advantages and disadvantages. The most important advantage in in-line inspection is the lower time request for the sample analysis which leads to an increased throughput; this has been achieved by deleting intermediate steps such as sample loading and unloading from the main assembly line for inspection. In addition, statistical analysis can be applied in order to detect in advance any possible component failure. For example, Yao et al. [25] describe and integrate a system for in-line inspection of nanoscale features into current semiconductor manufacturing processes.

2.2. Surface topography characterization

A surface may be defined as the boundary that separates an object from the surrounding medium. The surface, therefore, refers to a physical boundary while the topography refers to its description [24]. Depending on the measurement method a surface can be mechanical or electromagnetic. The mechanical surface in ISO 14406 is described as the *boundary of the erosion, by a sphere of radius r , of the locus of the centre of an ideal tactile sphere, also with radius r , rolled over the skin model of a workpiece* [26]. As far as the electromagnetic surface is concerned, in ISO 25178-2 is defined as the *surface obtained by the electromagnetic interaction with the skin model of a workpiece* [27].

The nature of surfaces, their characterization and their measurement cannot be separated because they are intertwined. A surface can be seen as the result of three main contribution (figure 2.2.1), that can be identified in terms of deviation with respect to an ideal flat plane:

1. Roughness, which is an irregularity often caused by the manufacturing process.
2. Waviness, which is a longer wavelength irregularity may be caused by improper manufacture.
3. Form, which is the underlying shape of a part that fits to a measured surface and may be caused by errors in slideways, in rotating members of the machine, or in thermal distortion.

Often the first two are gathered under the general expression of surface texture.

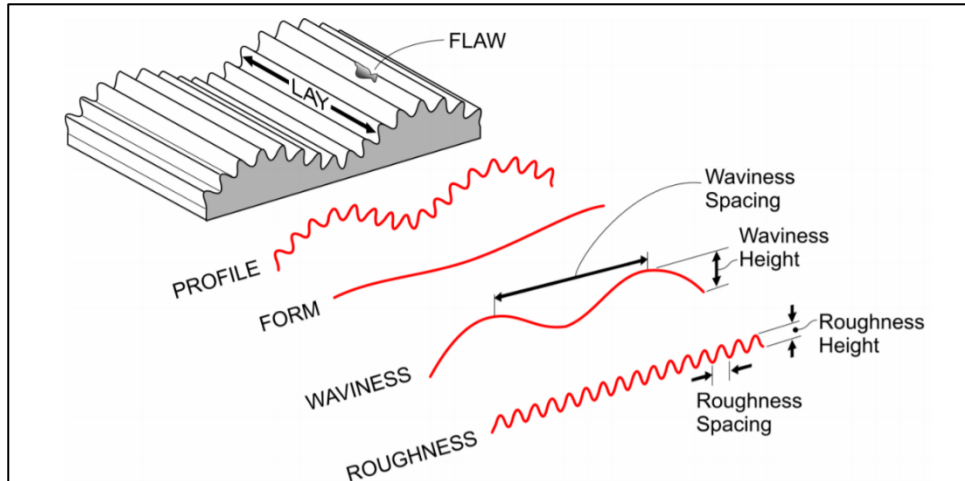


Figure 2.2.1 - Causes of deviation in a surface .

In the last century, the surface texture was extracted by the trace of a contacting stylus across the surface measuring the vertical motion of the stylus as it traversed the surface. Optical instruments, with the benefit of being non-contact, hence more flexible, with a higher resolution and potentially faster, closely followed the development of stylus instruments. However, it was only the development of high-speed computing that makes them usable at a large scale [15]. Profile and areal measurement are respectively the measurements of a line across the surface and of an area on the surface. The profile can be represented mathematically as a height function with lateral displacement, $z(x)$, whilst the areal can be described as a height function with displacement across a plane, $z(x,y)$. The profile is usually obtained by stylus instruments, however, many optical instruments allow the extraction of a profile directly (scanned) or through software from an areal map [28].

Texture and form can be described as a surface contributions in the spatial frequencies domain, hence, they can be characterized by wavelengths. These components can be detected through an appropriate filter, which allows the selection of interesting information and, at the same time, the rejection of irrelevant, to target proper characterisation. Filtering methods are based on the scale in the displacement direction, or rather in terms of wavelength and spatial frequencies.

By convention, filters are referred to frequencies. A low-pass filter rejects short wavelengths, by contrast, a high-pass filter excludes longer wavelengths. A band-pass filter selects a restricted range of wavelengths included between high regions and low regions with the combination of a low-pass and a high-pass filter. The cut-off of a filter is, by convention, the wavelength at which the transmission is 50 %, as described in ISO 16610-60 [29]. For profile characterization, ISO 4287 defines three filters used for measuring roughness, waviness and primary profiles [30]:

- λ_s : filter which separates roughness from shorter wavelength components present in a surface (figure 2.2.2);
- λ_c : filter which defines where the interception between roughness and waviness components takes place (figure 2.2.2);
- λ_f : filter which separates waviness from longer wavelength components present in a surface (figure 2.2.2).

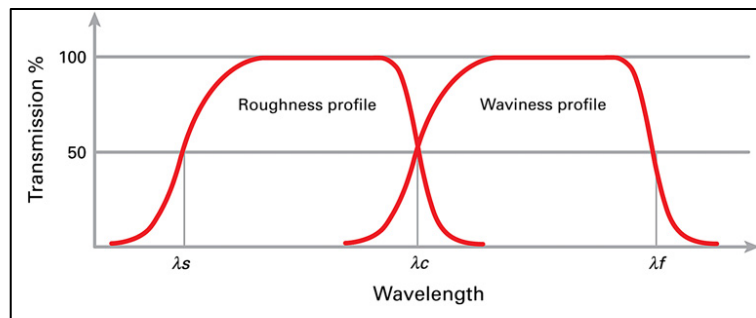


Figure 2.2.2 – Filters wavelengths representation according to ISO 4287 [28].

The ISO 3274 differentiates the definition of profile [31]. Firstly, the traced profile is the imprinting centre of a stylus whose geometrical form and dimensions are ideal (conical, with spherical tip) and the tracing force is nominal, as it traverses the surface. Then, whilst the reference profile is defined as the mark on which the probe imprints within the intersection plane along with the guide, the total profile is the digital form of the profile and it is characterized by a digital step both horizontally and vertically. Finally, the primary profile can be defined as the total profile after the application of the short-wavelength filter λ_s . Roughness profile and waviness profile can be extracted from the primary profile respectively suppressing the long-

wave component using a long-wavelength (high-pass) filter, λ_f , and applying a band-pass filter, λ_c . Surface texture parameter, be it profile or areal, have been defined to give a quantitative value to the surface texture and have to be calculated once the form has been removed from the measurement data.

The profile parameters computation needs also the definition of the sampling, the evaluation and the total traverse length, which are presented in figure 2.2.3. The evaluation length is the total length of the surface whose profile is under evaluation. The sampling length, normally utilized for the roughness and waviness profile characterization, is the length of each segment in which the evaluation length is divided into. As ISO 4287 advocates, the roughness profile should be evaluated on five sampling length, whilst no indication is provided for the waviness profile [30].

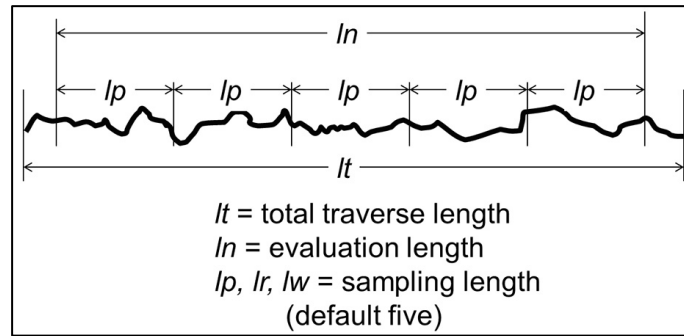


Figure 2.2.3 – Representation of the different lengths utilized for profile analysis.

The extraction process of parameters should be conducted in each successive sampling length and the resulting values averaged over all the sampling lengths in the evaluation length. Finally, the total traverse length is the total length measured by the instrument.

Profile parameters are listed in ISO 4287 [30] and in ISO 4288 [32], they are named according to the type of surface profile from which they are calculated, for example, P-parameters (P_a , P_{sk} , etc.) are calculated on the primary profile, R-parameters (R_a , R_{sk} , etc.) on the roughness profile and W-parameters (W_a , W_{sk} , etc.) on the waviness profile. Profile parameters are composed of nine height parameters, one

spacing parameter, one hybrid parameter and a number of curves designed to give functional information.

The evaluation of the profile alone has some limitation with surface measurement and characterization. In fact, the exact nature of a topographic feature is often indeterminable and it cannot be a priori established, because of an overall view is missing, see figure 2.2.4 [28]. In addition, a profile measurement will have less statistical significance than an areal measurement, simply because an areal map is a closer representation of the “real surface” [33].

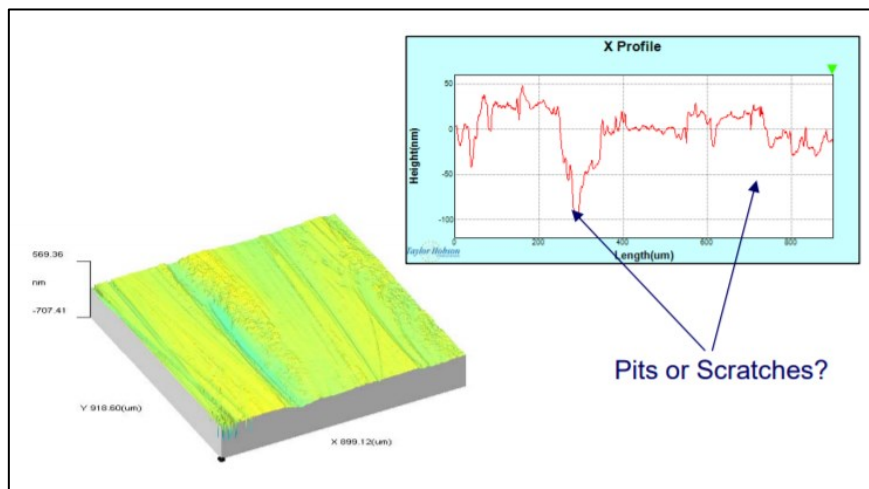


Figure 2.2.4 – The difference in information between the areal and the profile method [34].

Table 1 – Profile method ISO standards.

<i>Profile method ISO standards</i>	
ISO 1302	Indication of surface texture in technical product documentation
ISO 3274	Nominal characteristics of contact (stylus) instruments
ISO 4287	Terms, definitions and surface texture parameters
ISO 4288	Rules and procedures for the assessment of surface texture
ISO 5436-1	Measurement standards – Material measures
ISO 5436-2	Software measurement standards
ISO 12085	Motif parameters
ISO 12179	Calibration of contact (stylus) instruments
ISO 13565-1	Surfaces having stratified functional properties – Filtering and general measurement conditions
ISO 13565-2	Surfaces having stratified functional properties – Height characterization using the linear material ratio curve
ISO 13565-3	Surfaces having stratified functional properties – Height characterization using the material probability curve
ISO 16610-21	Filtration—Linear profile filter: Gaussian filters

The areal specification standards are still under definition; all the areal standards are part of ISO 25178 and a list of its parts is reported in Table 2. The final scope is to have the profile standards as a subset of the areal standards, hence, once the areal standard will be concluded the profile standards will be re-published with a new numbering scheme that is consistent with that of the areal standards and with some corrections [28]. The current standards of the profile are listed in Table 1.

Table 2 – List of ISO 25178 parts.

<i>ISO 25178</i> Geometrical product specification (GPS) - Surface texture: Areal	
Part 1	Indication of surface texture
Part 2	Terms, definitions and surface texture parameters
Part 3	Specification operators
Part 4	Comparison rules
Part 5	Verification operators
Part 6	Classification of methods for measuring surface texture
Part 70	Material measures
Part 71	Software measurement standards
Part 72	Software measurement standards - XML file format
Part 73	Software measurement standards - defects
Part 600	Metrological characteristics of surface texture instruments
Part 601	Nominal characteristics of contact (stylus) instruments
Part 602	Nominal characteristics of non-contact (confocal chromatic probe) instruments
Part 603	Nominal characteristics of non-contact (phase-shifting interferometric microscopy) instruments
Part 604	Nominal characteristics of non-contact (coherence scanning interferometry) instruments
Part 605	Nominal characteristics of non-contact (point autofocus) instruments
Part 606	Nominal characteristics of non-contact (variable focus) instruments
Part 701	Calibration and measurement standards for contact (stylus) instruments
Part 702	Calibration and measurement standards for non-contact (confocal chromatic probe) instruments
Part 703	Calibration and measurement standards for non-contact (phase-shifting interferometric microscopy) instruments
Part 704	Calibration and measurement standards for non-contact (coherence scanning interferometry) instruments
Part 705	Calibration and measurement standards for non-contact (point autofocus) instruments
Part 706	Calibration and measurement standards for non-contact (variable focus) instruments

Areal surface characterization depends on the type of scale-limited surface used to calculate the parameter, therefore, parameters are not named according to the type of surface, but are referred to the applied filter, see figure 2.2.5.

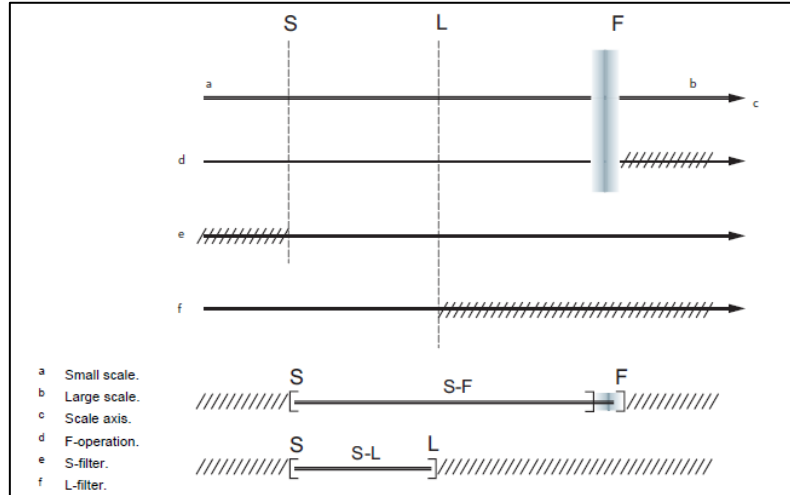


Figure 2.2.5 – Relationship between the different areal filters [27].

ISO 25178-2 defines two filters, the S-filter and the L-filter [27]. Before the filters definition, the primary surface has to be introduced, in the standards is described as the surface represented through a specified primary mathematical model with specified nesting index, which is an extension of the notion of the original cut-off wavelength [27]. The S-filter is defined as a filter that separates small-scale lateral components, such as measurement noise or functionally irrelevant small features, from the measured surface. The primary surface can be also defined as an S-filtered surface. By contrast, the L-filter removes large-scale lateral components of the measured surface. Finally, ISO 25178-2 defines also an F-operator which removes the nominal form [27]. An S-F surface results from the primary surface using an S-filter to remove the form, whilst the S-L surface is a result of an applying L-filter on an S-F surface. Both the S-F and S-L surface are defined as scale-limited surfaces. In figure 2.2.6 is presented a scheme of the filtration process.

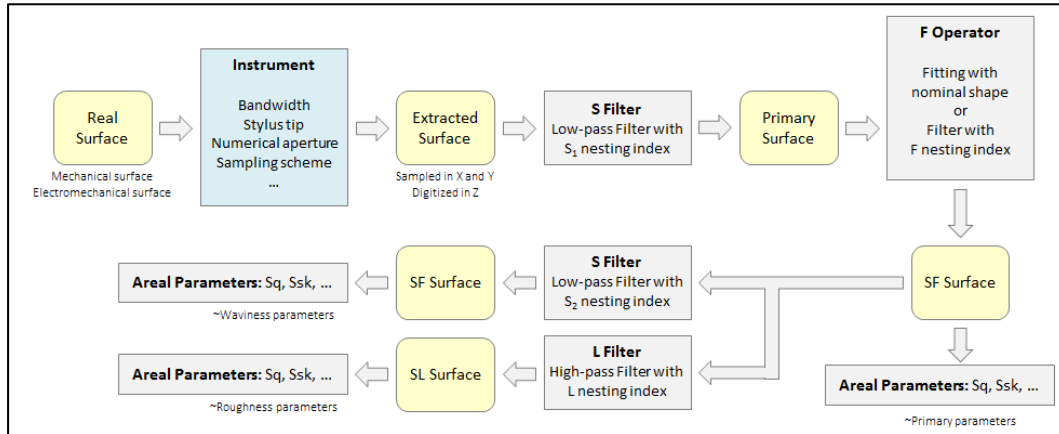


Figure 2.2.6 – Filtration scheme of areal surface measurements.

Areal parameters can be divided into two main classes [35] and they are defined and listed in ISO 25178-2 [24]:

- **Field parameters:** every data point of the scale-limited surface is taken into account to extract these parameters. ISO 25178-2 defines the field parameters; these can be either S- or V-parameters: the prefix V specifies that the parameter is referred to material volume or void volume, whilst the S-parameters are defined as parameters that are not V-parameters. The field parameters are composed of amplitude, spacing, hybrid, fractal, functions and related parameters, and one miscellaneous parameter. Once the F-operation is applied, these parameters are calculated on the measured surface without cutting the surface into small sub-areas that depend on the sampling length.
- **Feature parameters:** only a subset of predefined topological features is considered from the scale-limited surface. The feature parameters are based on the fact that we do not usually see field parameter value on a surface but features, such as hills and valleys. Feature characterization does not have specific feature parameters defined but has instead a toolbox of pattern-recognition techniques that can be used to characterize specified features on a scale-limited surface.

- The feature characterization process, as defined in ISO 25178-2, is composed of five stages [27]:
 1. selection of the type of texture feature;
 2. segmentation;
 3. determining significant features;
 4. selection of feature attributes;
 5. quantification of feature attribute statistics.

To conclude, the profile method is relatively simple to apply and well-established, even though it is less statistically robust and representative of the surface. Therefore, as high-speed computing is increasingly and as international standards are establishing, areal methods of analysis are becoming more commonplace. Using a profile filter is different than using an areal filter, therefore, a profile analysed according to the methods detailed in the profile standards is not mathematically equivalent to profile extracted from an S-L surface or an S-F surface. In fact, even with similar filter types and cut-off, the filtration is the largest difference between the two methods [28].

ISO 25178-6 classifies the method for measuring surface texture, particularly it defines three methods for surface texture measuring instruments [36]:

- Line profiling method: the surface irregularities are represented in a 2D graph or profile as measurement data, which may be mathematically described by a height function, $z(x)$. Examples of instruments that are able to measure line profile include stylus instruments, phase-shifting interferometry, circular interferometric profiling and optical differential profiling.
- Areal topography method: the surface is reproduced through a topographical image, which can be mathematically expressed by a height function, $z(x, y)$. The surface reconstruction can be also achieved by juxtaposing a set of parallel profiles. Examples of instruments that are able to measure areal topography are stylus instruments, phase-shifting interferometry, coherence scanning interferometry, confocal microscopy, confocal chromatic microscopy, structured light projection, focus variation

microscopy, digital holography microscopy, angle-resolved scanning electron microscopy SEM, SEM stereoscopy, scanning tunnelling microscopy, atomic force microscopy, optical differential and point-autofocus profiling.

- Area-integrating method: a representative area of a surface is measured through the production of numerical results that depend on area-integrated properties of the surface texture. Examples of instruments for the achievement of this method include those that use the technique of total integrated scatter.

Amplitude-wavelength (AW) space is a good way to show the area of operation of the instrument (figure 2.2.7). The measurement capabilities of the instruments are constrained by a number of factors, such as range, resolution, tip geometry, lateral wavelength limit. Through modelling and parameterising these constraints, relationships between these parameters can be derived. In order to have a better representation, the relationships are defined by inequalities. Amplitude and wavelength are two constraints whose inequalities can be plotted to construct a polygon, which shape defines the operating region of the instrument.

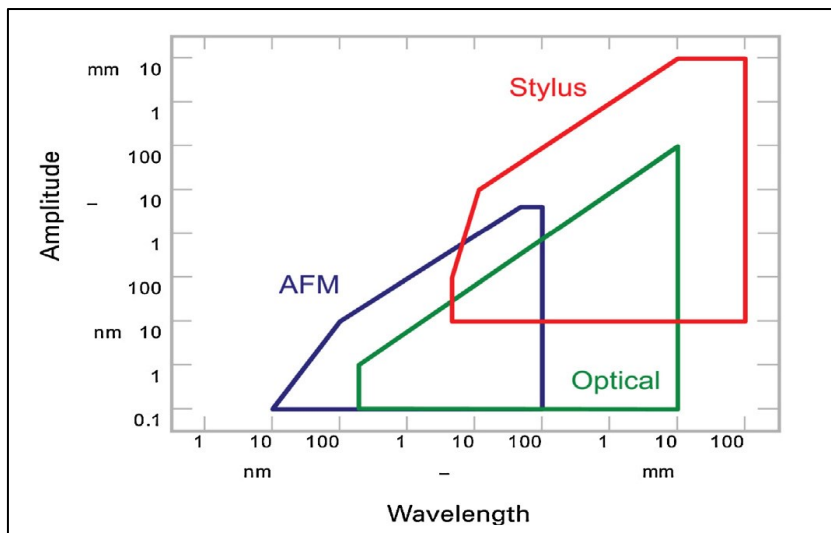


Figure 2.2.7 – AW space plot for three common instruments [28].

Currently, stylus instruments are still widely employed in the surface metrology area. Because of the complexity of modelling the beam-to-surface interaction, stylus instruments tend to be adopted as primary instruments. The measurement is straightforwardly performed through the displacement of a stylus and the transducer conversion of vertical movement into an electrical signal. Usually, the contact with the surface occurs through a diamond tip with a carefully manufactured shape, included in the stylus. The most relevant issue is caused by the finite shape of the tip, which not being able to penetrate into some valleys provides a distorted or filtered measure. When stylus instruments are used to perform areal measurements the measuring time considerably increase and, furthermore, the precision and accuracy along the x-axis direction are incomparable with the ones along the lateral direction (y-axis).

The optical instruments are becoming more common, commercially; in fact, over the year many different types of instruments that can measure surface texture have been developed. All the instrument types are discussed in detail elsewhere [33]. Thanks to the non-contact characteristic of optical instruments, the risk of damaging the surface is absent and the measurement time is lower. Although the non-contact nature of optical instruments is an advantage over stylus instruments, it is still not such a trivial matter how an electromagnetic field interacts with the surface [16]. Such lack of knowledge makes the estimate of measurement uncertainty rather difficult.

To provide users with confidence in the obtained results, and enable for comparison of results, performances and application in quality control, the measurement has to be coupled with its uncertainty. Nowadays, the uncertainty estimation of surface texture measurement is still up for discussion. The difficulty in uncertainty estimation is probably due to the complexity of the measurand and the measurement process. In the uncertainty analysis both the contribution to the uncertainty due to the measuring instrument and those due to the specimens must be considered. Concerning the contribution to uncertainty from areal instruments, the available knowledge is rather limited. Now, the concepts of calibration and adjustment are introduced in order to avoid any confusion between the two terms. Calibration is

defined in VIM [37] as the *operation that, under specified conditions, in a first step establishes a relation between the quantity values with measurement uncertainties provided by measurement standards and corresponding indications with associated measurement uncertainties and, in a second step, uses this information to establish a relation for obtaining a measurement result from an indication*. Adjustment, instead, is a *set of operations carried out on a measuring system so that it provides prescribed indications corresponding to given values of a quantity to be measured* [37]. The calibration of an instrument can be achieved either resorting to some calibration artefacts, referred to as material measures, listed in ISO 25178-70 [38], for the calibration of the metrological characteristics or inspiring to the method for determining the measurement uncertainty of coordinate measuring machines (CMM), described in ISO 15530-3 [39].

2.3. Instrument used for measurements

In this thesis, optical measurements were performed by Zygo NewView 9000 coherence scanning interferometry (CSI) (magnification 50x, numerical aperture 0.55, vertical resolution 3.5 nm, pixel size 170.90 nm \times 170.90 nm) and by Olympus Lext OLS 4100 confocal microscope (magnification 50x: numerical aperture 0.95, vertical resolution 10 nm, pixel size 63.08 nm \times 63.08 nm).

2.3.1. Coherence scanning interferometry (CSI)

Coherence Scanning Interferometers (CSI) and Phase Shift Interferometers (PSI) are the two areal surface measuring instruments based on interferometry. CSI have been more recently introduced in the market with respect to PSI. They both are optical areal-topography microscopes that relies on interferometry to measure the surface topography. Interferometry is based on superposition, that is, the idea that two waves with the same frequency and phase will add constructively whilst two waves with the same frequency but that are out of phase by 180 ° will destructively cancel each other [40]. The result is a set of dark and light bands known as fringes. In these systems, a light beam is split by a splitting mirror in two separate paths,

one of these includes the reference surface while the other the surface under investigation. Then, the two separated beams reunify, and the resulting wave is reflected to a digital camera that captures the light intensity over multiple image points simultaneously. A schematic representation of an interferometer is shown in figure 2.3.1. The light intensity depends sensitively on the difference in path length of the two rays [33]. Considering that destructive interferences decrease the intensity of the sensed signal, it is possible to deconvolute the interference fringes and, by reconstructing the path differences of the light from the reference mirror and the measurand surface, to measure topographical difference of the latter with respect to the former. CSIs additionally cater for intensity shift, this is achieved by additionally scanning on a vertical range, which makes CSI capable of dealing with rougher and unpolished surfaces [41].

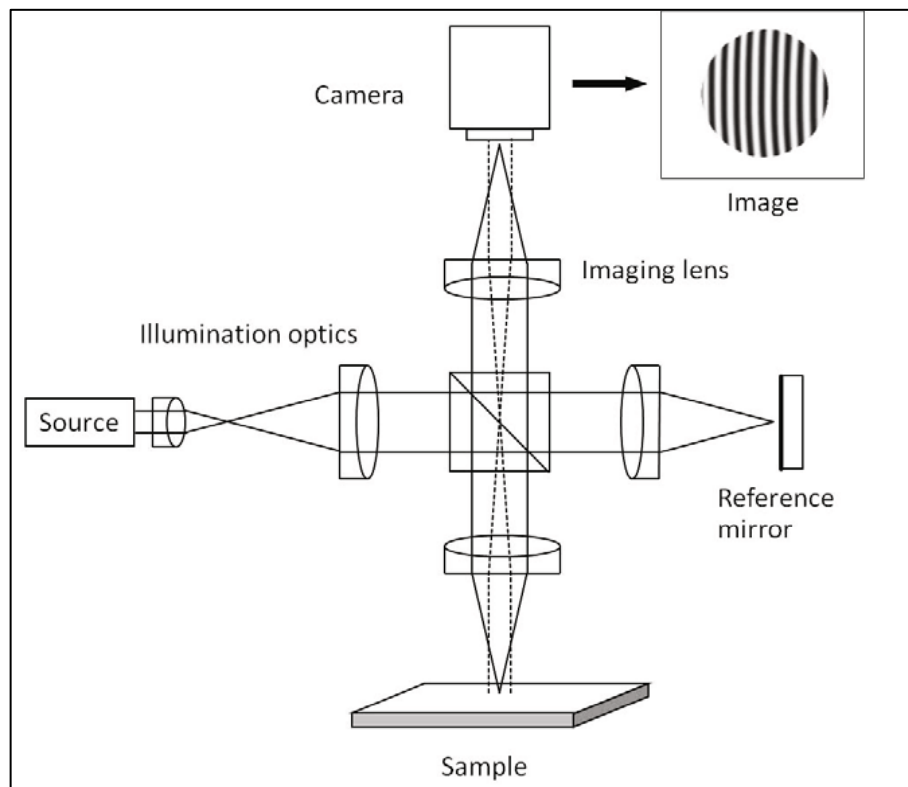


Figure 2.3.1 – Imaging interferometer for areal surface profiling [33].

CSI is based on the coherence envelope and phase of interference fringes, see figure 2.3.2. The envelope of the fringes represents the coherence function. Most CSI instrumentation firstly uses the modulation envelope to estimate fringe order, then the surface is extrapolated from the phase of the underlying fringe pattern [42]. The light source utilized, typically either a monochromatic source or a white light source, influences interferometry, therefore, different aspects of the optical designs of the interferometers are designed [40]. In order to avoid ambiguity in outlining fringe order, white light is used rather than monochromatic light because it has a shorter coherence length [42].

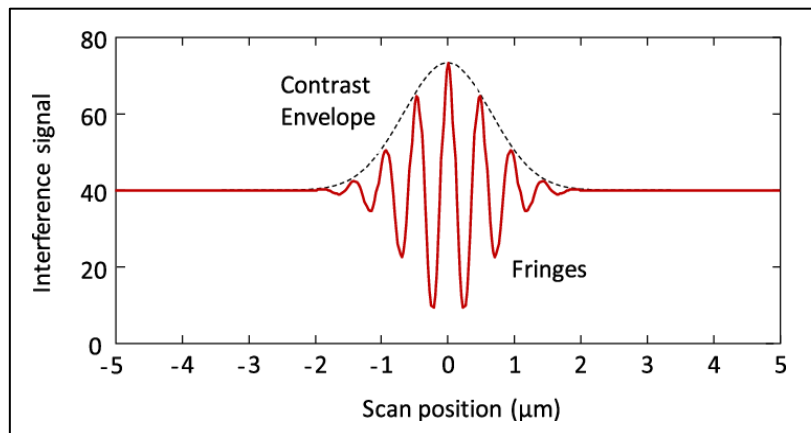


Figure 2.3.2 – CSI signal for a single pixel [43].

To enable for the measurement of a real surface, which typically has heights that vary, a mechanical scanner provides a smooth, continuous scan of the interference objective in the z-direction. During the scan, a computer records intensity data for each image point or pixel in successive camera frames. CSIs are standardised ISO 25178-604 [44], best practices can be found in NPL good practice guide 108 [42] and NPL good practice guide 116 [40].

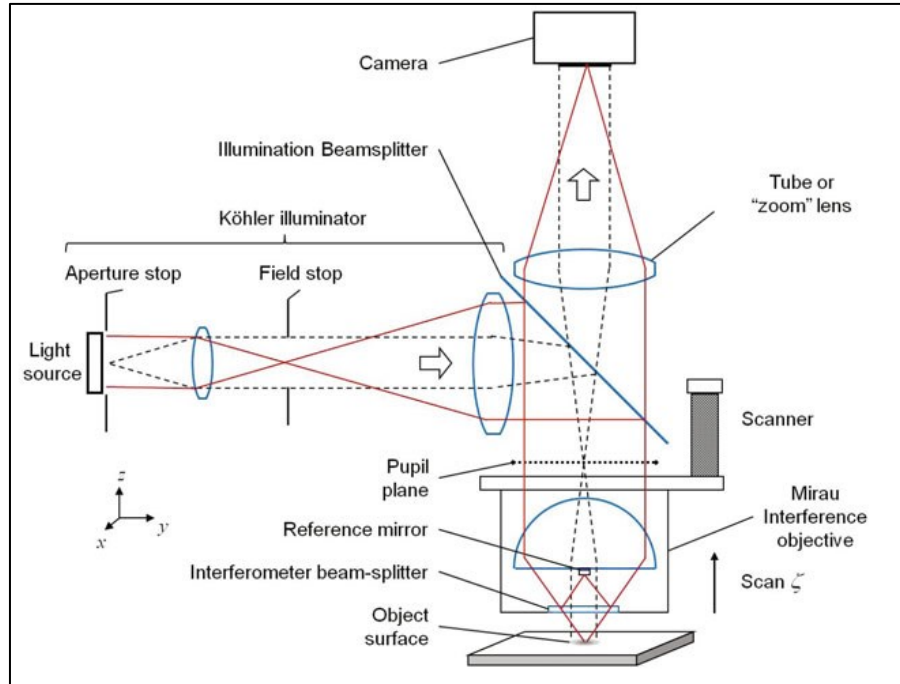


Figure 2.3.3 – Geometry of an interference microscope suitable for CSI [33].

2.3.2. Imaging Confocal Microscopes (ICM)

A confocal microscope provides optically sectioned images of the sample under examination. To get an optically sectioned image, restricting the illuminated regions on the sample by means of a structured illumination pattern is essential. The reflected light must be captured through a second pattern identical to the illumination pattern, in this way, the light that comes from regions out of the focal plane of the microscope's objective is blocked. In order to provide a complete optically sectioned image an in-plane scanning, which can be achieved mechanically or optically, of the patterns is required [33]. The simplest configuration of confocal microscopy is a laser scanning microscope, see figure 2.3.4. Light from a laser passes an illumination pinhole and then is captured by a detection pinhole after the reflection on the sample surface. Behind the detection pinhole is situated a photodetector that records the signal [45].

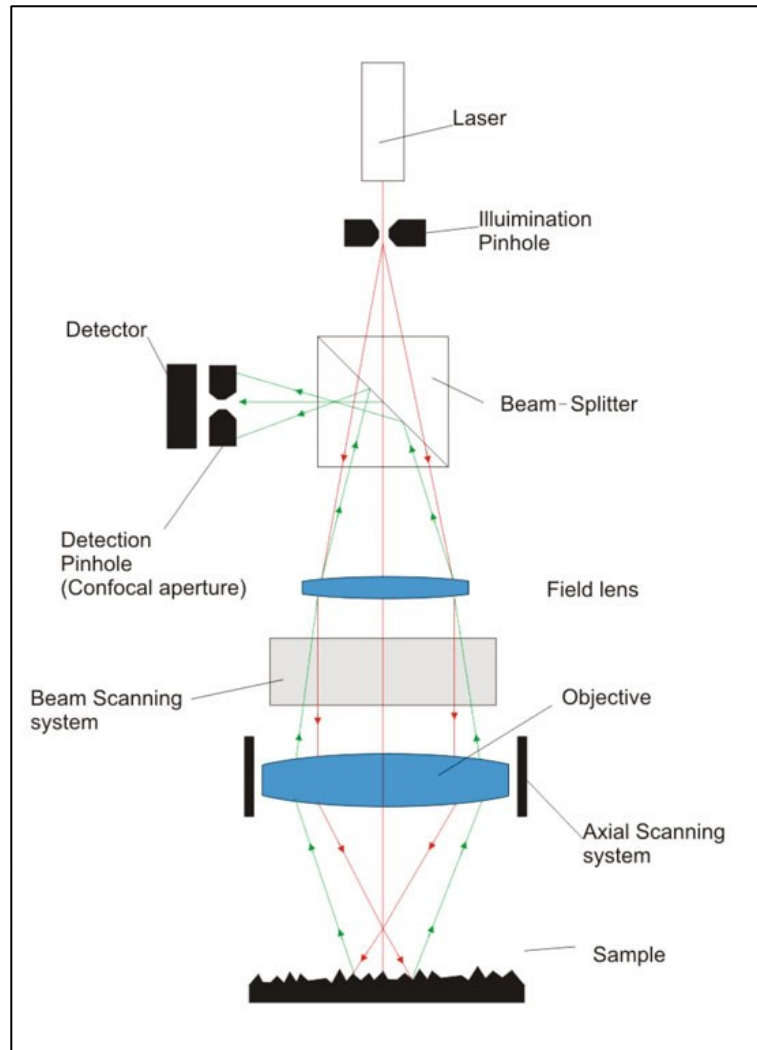


Figure 2.3.4 – Basic setup of a laser scanning microscope [33].

By scanning the ray onto the surface point by point, a confocal image is recovered from the focal plane of the objective. The maximum signal of the axial response is reached when the surface is exactly located on the focal plane of the microscope's objective. According to the 3D surface shape, different pixels will have the axial response maximum placed on different z axis positions. Hence, to reconstruct the 3D surface, the z axis position of the maximum of the axial response for each pixel is located [33]. ICMs are standardized in ISO 25178-607 [46], best practices can be found in NPL good practice guide 128 [45].

3. METHODOLOGY

The project aims to establish an architecture of algorithms, in which advanced statistical processing and uncertainty evaluation are to be applied to surface topography measurements and translated into algorithms suitable for a machine implementation. Received a number of micrographs in input, the algorithms are to provide a statistical analysis according to the frequentist method, an uncertainty evaluation, and a graphical representation [37–47]. In fact, to provide users with confidence in the obtained results, and enable for comparison of results, performances and application in quality control, the measurement has to be coupled with its uncertainty. Because of the data numerousness, a graphical representation is useful for the users to analyse the results and, depending on the display scale, it may provide qualitative or quantitative results. The architecture has been entirely developed in MATLAB® environment and in the case of some measurements, which required a large amount of memory, has been necessary to lean on cluster computing for the code running. To conclude, the purpose of this project is to provide the main structure of a larger research work that will require specific studies for some of the topics treated, such as the right method for the outliers detection or the most suitable normality test in case of pixels correlation.

3.1. Software environment definition

The software architecture has been designed in order to import surface measurements performed by different instruments. Currently, the files formats supported for the analysis are currently three:

- .asc: this file extension is the result of a pre-processing through the software SPIP™ of the measurements acquired by the Olympus Lext OLS 4100, whose file have format .lxt. As how to manage the files .lxt in MATLAB® is unknown, the conversion through SPIP has been necessary to get a file extension manageable in MATLAB®;

- .datx: this is the native file format of scanned surfaces through Zygo NewView 9000;
- .txt: this file extension has been implemented in order to have a useful file format for other instruments whose files type is still not supported by the code. In fact, it is enough to convert or copy the measurement data from a specific file extension to a text file and set the most suitable text reading setting in MATLAB.

The .asc and .txt file formats enable more in general the management of a wide range of measurement proprietary file, provided the preliminary conversion through a commercial software, e.g. SPIPTM, MountainsMap[®]

To reduce the computational times it has resorted, where possible, to parallel computing. Parallel computing allows to significantly speed-up complex computational problems by the simultaneous execution of processes, see figure 3.1.1. Therefore, conventional for-loops have been switched, when it has been possible, in parfor-loops. The higher efficiency of parallel computing can be observed in the case of the computer cluster where the greater number of cores affects the computing performance.

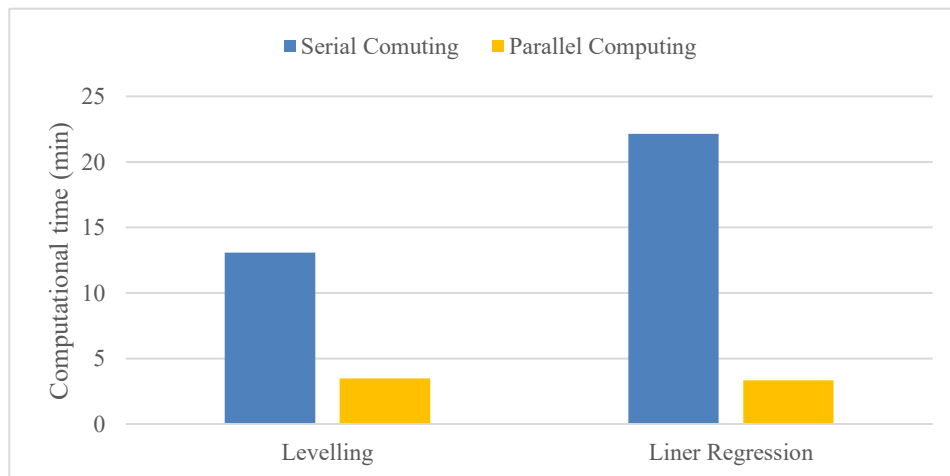


Figure 3.1.1 – Computational time differences between serial and parallel computing.

3.2. Preliminary management of topography

3.2.1. Plane correction

Specifications of surface texture are made on a flat surface, regardless of the original geometrical form, so a plane correction typically is the first operation applied to the measured surface. ISO 25178-2 defines the F-operation as the process that removes form from the primary surface. Form removal is characterized by three procedure [35]:

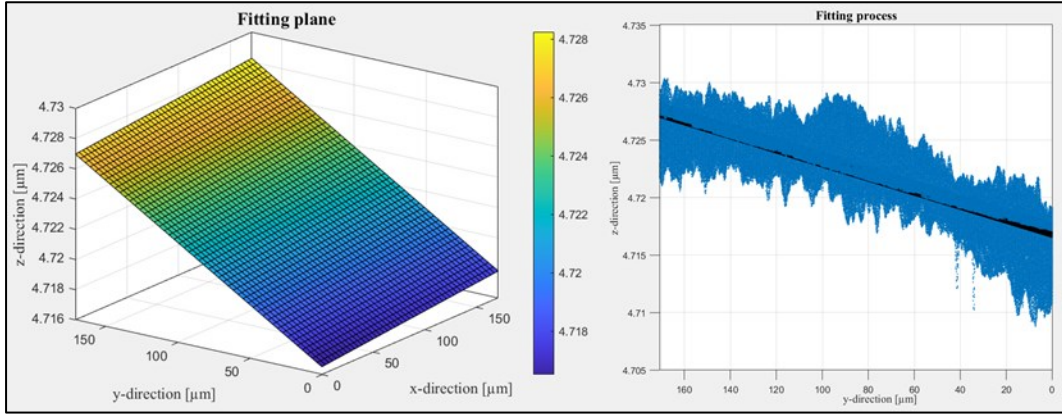
- the selection of surface model, e.g. geometric element, polynomial surface, tensor product spline, to reproduce the form;
- the choice of criterion to be applied for the form fitting of the measured data;
- the definition of how the form subtraction of the fitted model surface is achieved to the measured data.

In this project, the fitting criterion applied on the measured surface is the least-squares method. A plane can be uniquely identified by a point (x_0, y_0, z_0) in the plane and by the direction cosines (a, b, c) of the normal to the plane (such that, $a^2 + b^2 + c^2 = 1$). Thus, any plane can be identified by six parameters, which are not independent of each other. Now, supposing of having a plane, we want to quantify how well the plane fits the data points. In order to do that, we consider the distance of each pixel from the plane as the error of the fit. The distance of a generic point from the plane is:

$$d_i = a(x_i - x_0) + b(y_i - y_0) + c(z_i - z_0) \quad (1)$$

According to which side of the plane the point lies, the distance d_i can be either positive or negative, hence, for the measurement of the goodness of the fit, the sum of the square distances is usually utilized. The square distances sum is:

$$E = \sum_{i=1}^n d_i^2 \quad (2)$$



Fitting Plane

(b) How the fitting plane fits the data

Figure 3.2.1 – Least-squares fit result.

To find the best-fitting plane, the value of the parameters have to be extracted in order to minimize this summation, hence, the term least-squares method [48]. Figure 3.2.1 shows the fitting plane extracted through the least-squares methods and how it fits the data. The selection of the surface model is achieved through the definition of a functionally defined surface, which describes the height of the surface as a function of x and y :

$$z = f(x, y, p) \quad (3)$$

For example, a plane is defined as:

$$z = p_{00} + p_{10}x + p_{01}y \quad (4)$$

A portion of sphere instead is represented by:

$$z = p_{00} + p_{10}x + p_{01}y + p_{20}x^2 + p_{11}xy + p_{02}y^2 \quad (5)$$

Once the coefficients p_{ij} have been extracted according to the least square method, the fitting surface is subtracted to the measured surface in order to remove the form, see figure 3.2.2.

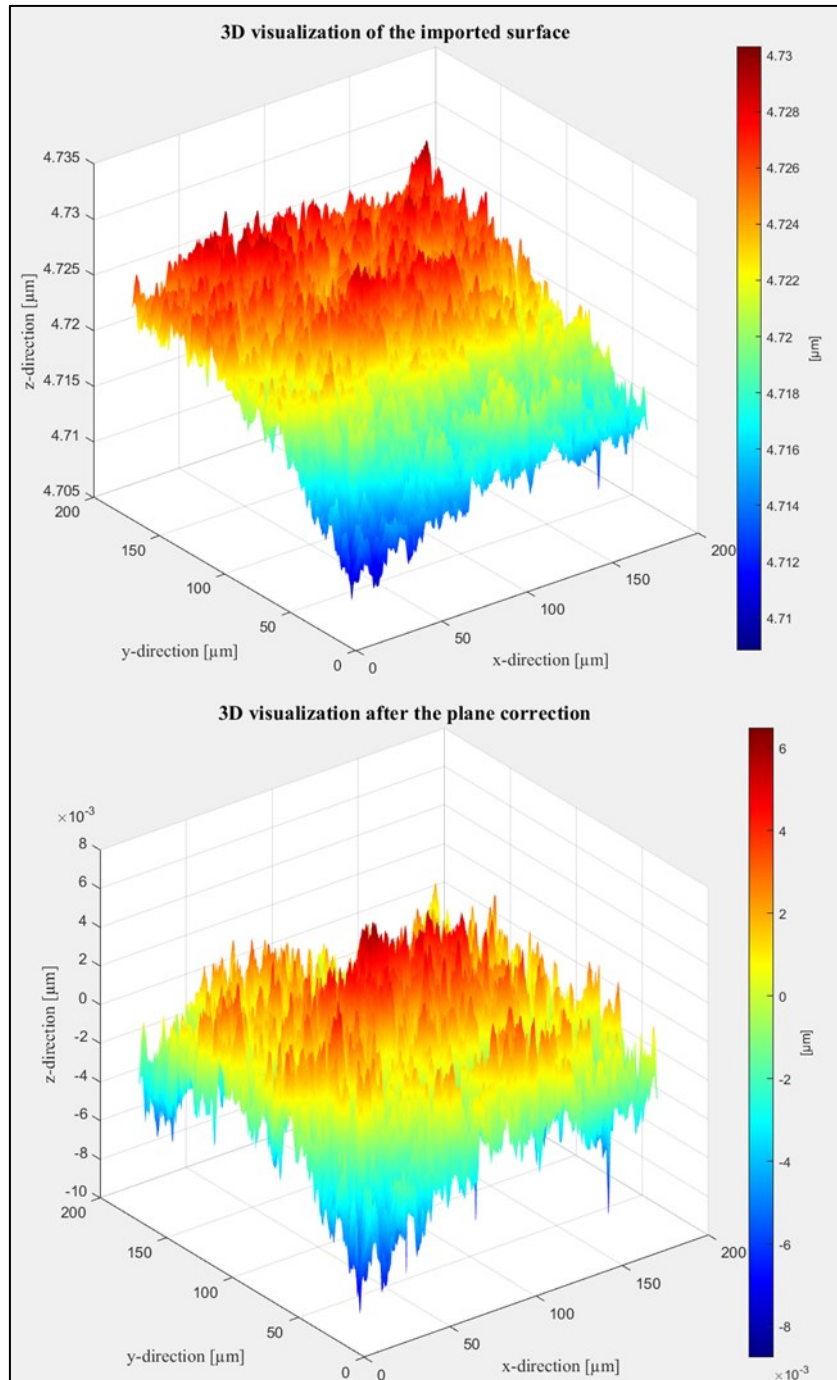


Figure 3.2.2 – How the same surface looks before and after the plane correction.

3.2.2. Areal field parameters

While profile parameters are named accordingly with the type of surface profile from which they are calculated, for example, R-parameters are based on roughness profile and W-parameters on waviness profile, surface parameters are just named S-parameters or V-parameters. ISO 25178 part 2 defines the V-parameters as material or void volume field of feature parameters and S-parameters as field or features parameters that are not V-parameters. A field parameter is defined from all the points on a surface, whilst a feature parameter from a subset of predefined topographic features from the surface [35]. In surface texture parameter equations, the height function, $z(x, y)$ must be centred, this means that the plane correction is already executed. In ISO 25178-2 [27], the parameters are always defined for the continuous case, i.e. expressed with integrals, although in practice measured surfaces are always sampled and digitised. Because of the surface measured is discrete the integrals are turned into summations. In the analysis of parameter uncertainty, also the effect due to the digitization, that is the digitizing step that cannot be infinitesimally small, must be considered. In fact, to provide a significant result the size of the pixels should be sufficiently small. According to the norm and turning the integrals into summations the following parameters have been taken into account in the project:

- Arithmetic Mean Height of the scale-limited surface:

$$S_a = \frac{1}{n_x n_y} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} |z(x_i, y_j)| \quad (6)$$

- Root Mean Square Height of the scale-limited surface:

$$S_q = \sqrt{\frac{1}{n_x n_y} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} z^2(x_i, y_j)} \quad (7)$$

- Root Mean Square Gradient of the scale-limited surface:

$$S_{dq} = \sqrt{\frac{1}{n_x n_y} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left(\frac{\partial z(x_i, y_j)}{\partial x} \right)^2 + \left(\frac{\partial z(x_i, y_j)}{\partial y} \right)^2} \quad (8)$$

n_x and n_y are respectively the number of points along the x and y axis.

3.2.3. Spikes and voids management

When a photodetector is involved in optical measurement instruments, the detector can receive the light intensity only within two limits. The first one is due to the low intensity of the signal that involves the impossibility to detect a z-axis position. On the other hand, if the signal has a high intensity the detector may become saturated, therefore, some signal is detected but not enough to allow a correct calculation. The result is that the device leads to an erroneous calculation of the z-axis position, which may be much higher or lower than the surrounding area [33]. These unusual values can be named outliers. In surface metrology, an outlier is defined as the local portion in a data set which is not typical for the integral feature [49]. To distinguish it from the statistical definition of outlier [50], i.e. an inconsistent value in a sample, it will be called spike. Usually, spikes are due to the instrument systematic behaviour for the most, whilst outliers are caused by the operator, thus, should be rare and not systematic events. The outliers from a statistical point of view will be treated in chapter 3.3.1 considering the repeated measurements of each pixel.

Non-measured points, or voids, can result from many different causes, such as defects due to local slope, material transparency, scanning limits, etc.. They could be treated by replacing them with a plausible height value or leaving them unmeasured [51]. In this project has been chosen to replace them with the values of the fitting surface obtained during the plane correction.

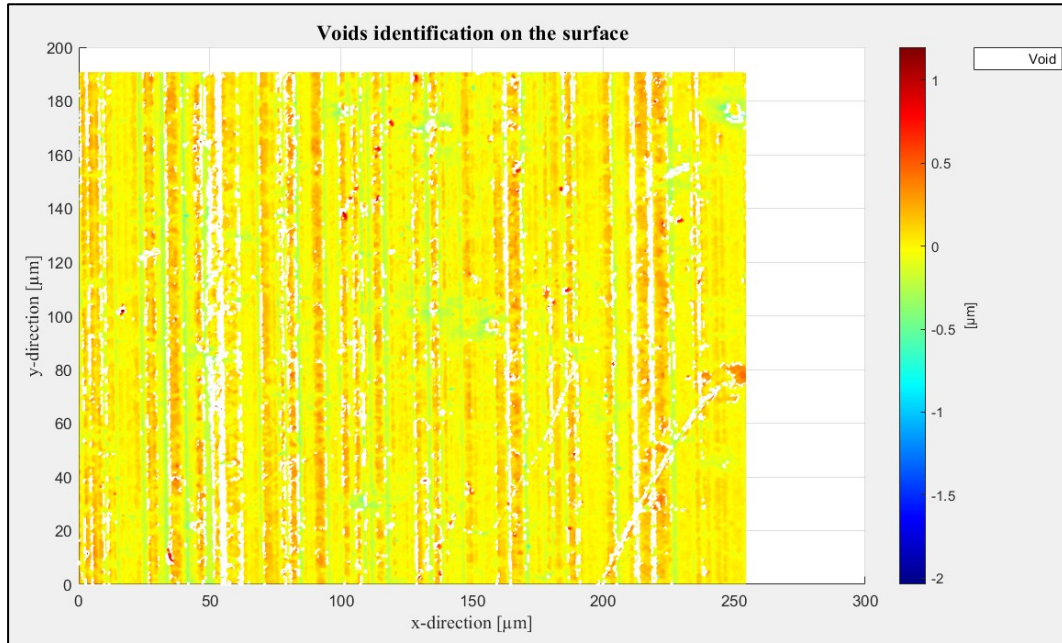


Figure 3.2.3 – Voids identification on the surface.

As far as spikes are concerned, they may be caused by spurious light reflections or other optical aberrations, the material nature, the closeness of the local slope to the limits, etc. [33]. No filter technique is applied, therefore, for the spikes detection and their management, the peaks and valleys have been limited at three times the S_q value [52]. In presence of many spikes should be better repeat the measurement. In literature there are many different methods to detect and then manage spikes. Wang et al. [51], present and compare ten statistical methods to identifies spikes on a surface. Moreover, Lou et al. [53] introduce a numerical filtration technique that, at the same time, allows both the reduction of the high-frequency measurement noise and the suppression of the outliers.

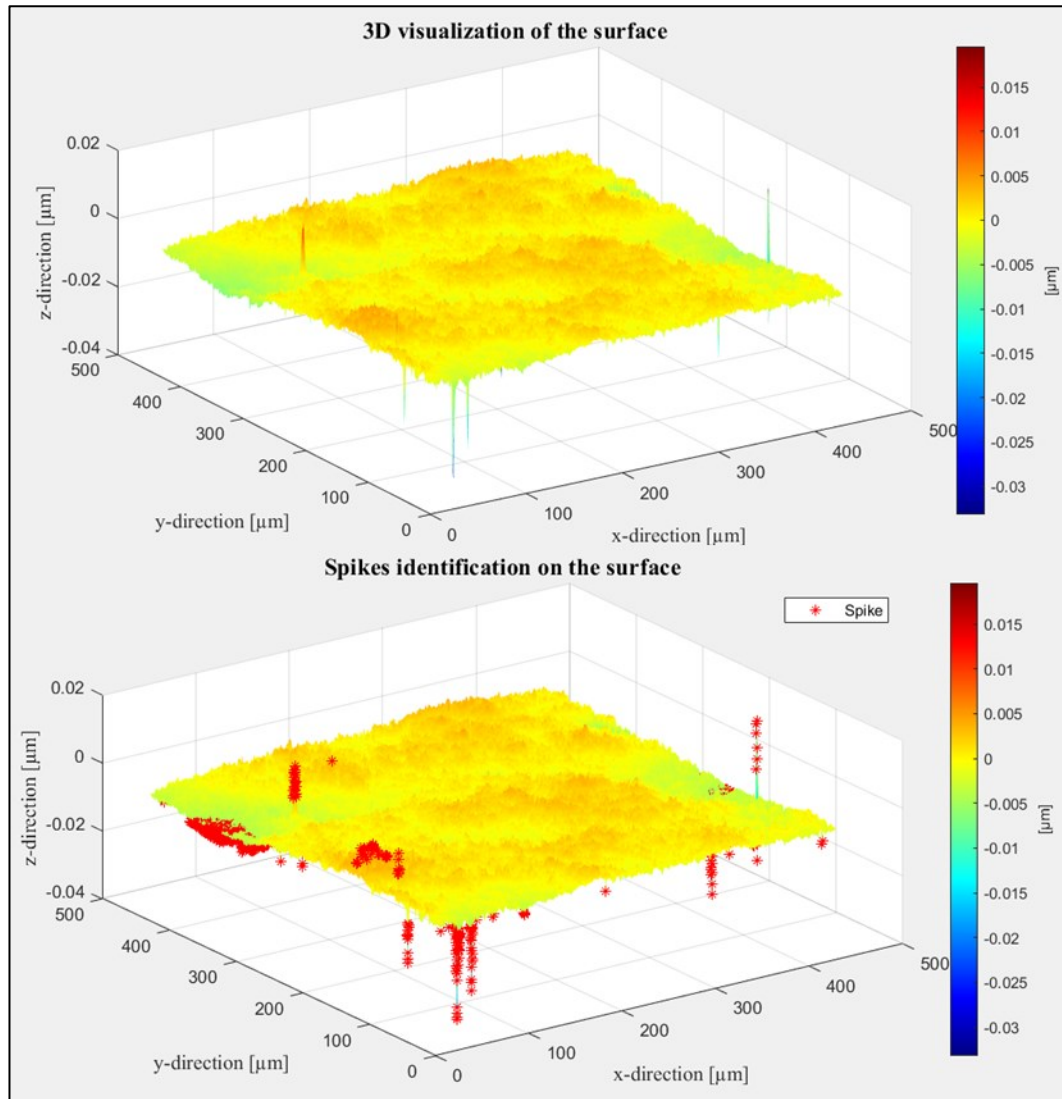


Figure 3.2.4 – Spikes identification on the surface.

3.3. Statistical analysis

3.3.1. Outliers' identification and management

This section discusses two different methods of outliers' identification, the Chauvenet's criterion and the modified interquartile range (IQR) method. An outlier can be defined as a fact that is very different from others so that it cannot be used to draw general conclusions [54]. Some scientists believe that discarding a measurement just because the value seems anomalous is not justified. The exclusion should be taken into account only when the cause of the presence of outliers is known. The exclusion principles are based on the idea that accidents are rare. On the other hand, it is not necessarily true that all the rare occurrences are accidents [55].

Chauvenet's criterion

Chauvenet assigned a conventional probability (P) of 50 % to the occurrence of having one outlier among n tests carried out. In other words, Chauvenet expected a measurement accident to happen once out of two repetitions of the complete experiment, independently from the type of experiment and the number of tests considered [56]. Chauvenet approximated the probability p of having an accident in any single measurement as $p \approx P/n$ and assumed as reference the normal distribution centred on the mean. The probability of accident p is symmetrically divided between the lower tail and the upper tail. Thus, the probability that a value is below the lower limit or above the upper limit is:

$$P(x < x_{li}) = P(x > x_{ls}) = \frac{p}{2} = \frac{50\%}{2n} \quad (9)$$

From which the exclusion boundaries x_{li} and x_{ls} can be acquired.

Modified IQR method

The interquartile range is defined as the difference between the 75th and the 25th percentiles of the sample data, which are respectively the third (Q_3) and first (Q_1) quartiles for a sorted data set [50].

$$IQR = Q_3 - Q_1 \quad (10)$$

The information is usually used to draw a boxplot (figure 3.3.1), the central region between Q_1 and Q_3 is represented by a box, in which also the median (Q_2) is marked. Two lines, called whiskers, are marked off from either side. The lower whisker limit, or lower inner fence, is computed as $Q_1 - 1.5IQR$, and the upper whisker limit, or upper inner fence, is $Q_3 + 1.5IQR$; any value beyond either limit is considered an outlier, regardless of the set size. Simplicity and low sensitivity to distortion due to outliers (robustness) are the main advantages; furthermore, it can be applied with success even when precious little information is available about data distribution [56].

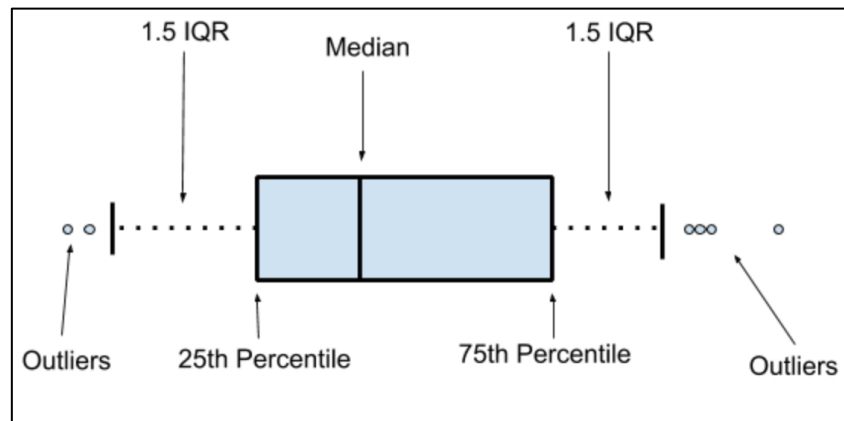


Figure 3.3.1 – Boxplot.

However, this method seems to be too severe, therefore, a modified method, which introduces the sample size n , is proposed. The whisker limits are modified to $Q_1 - 1.5IQR[1 + 0.1 \log(n/10)]$ and $Q_3 + 1.5IQR[1 + 0.1 \log(n/10)]$ [56]. The main advantages of the IQR method are thus retained while taking into account the sample size.

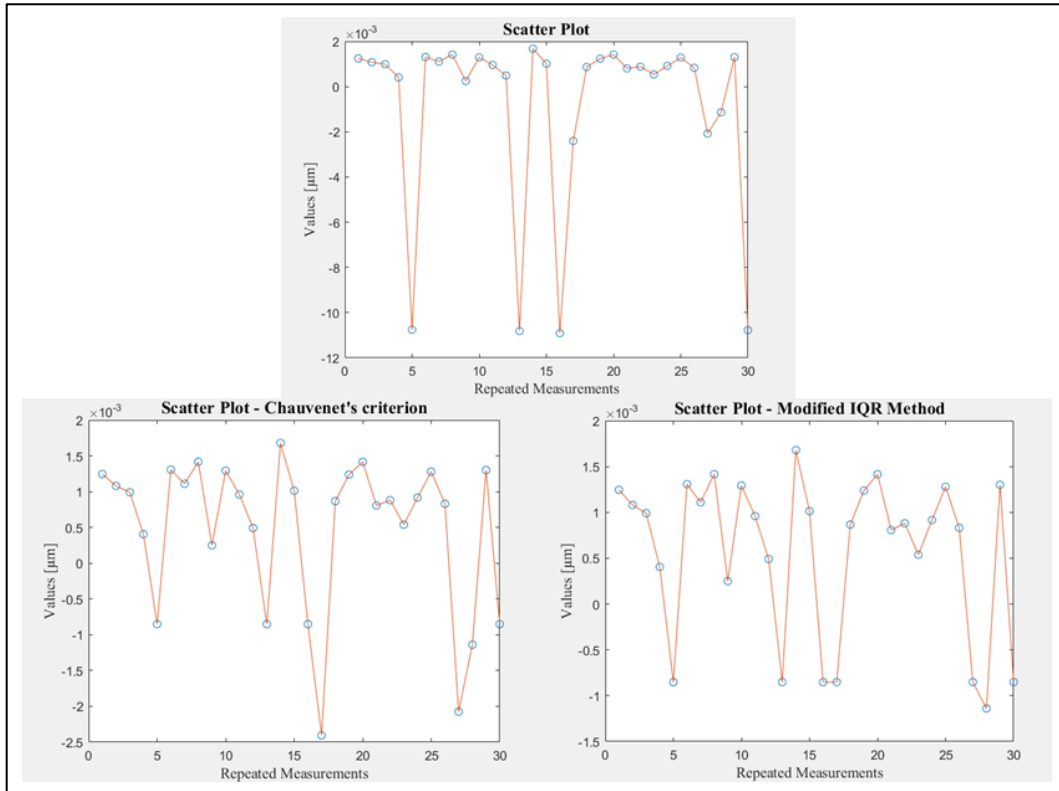


Figure 3.3.2 – Scatter plot of the repeated measurements of a pixel before and after the outliers' management with both Chauvenet's criterion and modified IQR method.

Both methods have been successfully implemented as can be seen in figure 3.3.2. In this project, the analysis has been conducted detecting outliers among the repeated measurements of each pixel. Then the identified outliers are replaced with the mean of the non-outliers' values; the process is executed taking into account a single pixel each time. Once the results of both the Chauvenet's criterion and the modified IQR method have been acquired for the pixel by pixel analysis of surface measurement, has been clearly noted that the number of outliers was too high. The statistical analysis works but evidently the model does not fit the problem. Here, surfaces which have a discrete nature that maps something continuous, are analysed, hence, in the transfer from continuous to discrete domain there is a loss of information. This loss influences the methods adopted by classical physics compromising their validity. Moreover, the existence of measurement noise probably disturbs the measurements and modifies their results. The measurement noise can be characterized by different contributions of noise, such as white noise

and flicker noise of the electronics, which are defined by their frequencies. The different nature of noise components requires different filtering techniques. In this project, we do not have applied any filter, hence, outliers and noise are mixed. However, measurement noise and outliers have different statistical distributions which are incompatible, hence, the classical methods for the outliers' identification may be inapplicable. Therefore, for this project aim, we decided to proceed without managing the outliers; in fact, this aspect deserves to be thoroughly studied in further researches.

3.3.2. Normality tests

Through the analysis of a data statistical distribution, the presence of some systematic effects can be detected; in fact, while a data normal distribution is usually connected to random effects, a non-normal distribution is typically caused by systematic factors. In this project two ways to analyze the normality of the repeated measurements of each pixel have been implemented: the chi-squared test and the normal probability plot (NPP).

Chi-squared test (χ^2 -test)

It is a numerical method which allows or not to reject the hypothesis of normality at a defined confidence level P . It resorts to Pearson's χ^2 distribution (figure 3.3.3) to analyse the fitting grade of the frequencies of the observed values to the theoretical frequencies of a normal distribution.

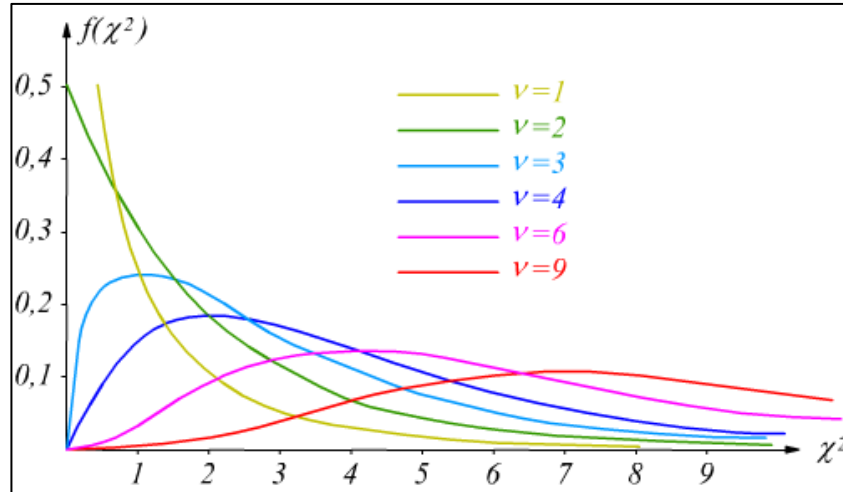


Figure 3.3.3 – χ^2 distribution at different degrees of freedom (ν).

Now the variable W can be introduced to carry out the test:

$$W = \sum_{j=1}^n \frac{(fa_j - fat_j)^2}{fat_j} \quad (11)$$

It is possible to demonstrate that W has a Pearson's χ^2 distribution with $n - 3$ degrees of freedom (ν). The constraint is three because, in the comparison between the normal distribution and the histogram of the experimental distribution, the equality of the mean, the standard deviation and the total area (unitary) have been imposed [55]. In equations (11) fa is the absolute experimental frequency that can be evaluated by counting the number of data in each class. The classes, also called categories or bins, are the zones in which the axis is subdivided. fat , on the other hand, is the theoretical absolute frequency, which can be achieved as the product between the theoretical relative frequency f_{rt} and the number of data n . Therefore, the theoretical relative frequencies are calculated as the differences of the cumulative distributions at the class limits. Once accomplished the summation, W has to be compared with the theoretical limits of χ^2 distribution at both the degrees of freedom and the required confidence level P . If W is a value included between the theoretical limits, the hypothesis of the normal distribution cannot be rejected. On the other hand, when W is outside the theoretical limits, the hypothesis is then rejected. The adoption of a bilateral hypothesis test allows to reject the null

hypothesis also when the distribution looks too close to the normal distribution, in other words when W is too small.

The χ^2 test has been implemented, as done in the outliers' analysis, taking into account the repeated measurements of each pixel and the results are plotted in a 2D graph that considers the map of pixels, see figure 3.3.4.

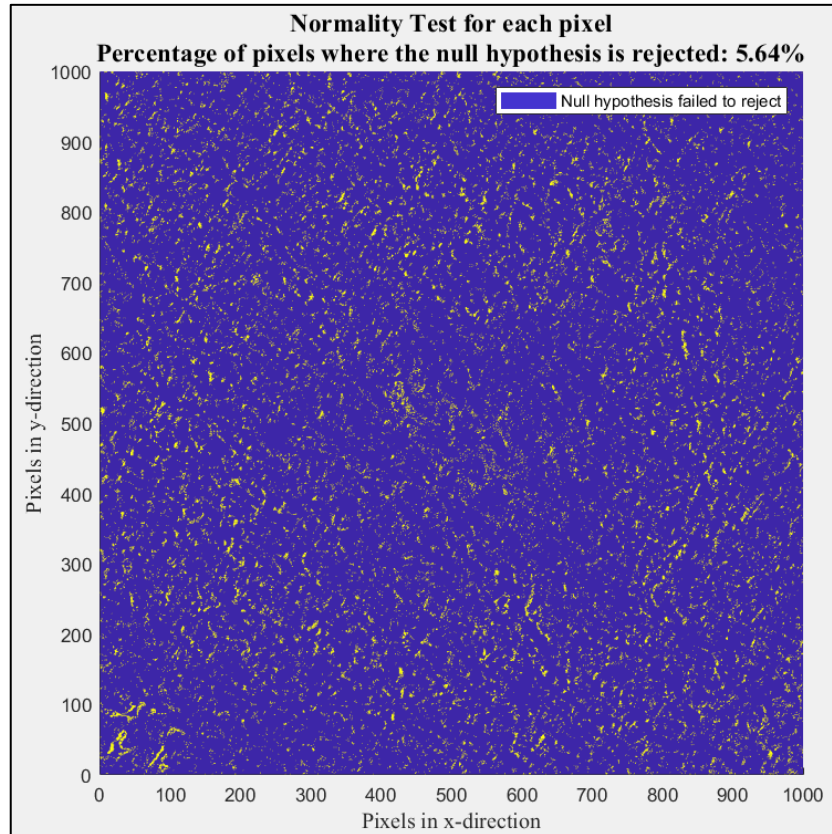


Figure 3.3.4 – χ^2 results for a map of pixels at a confidence level $P = 95 \%$.

Normal probability plot (NPP)

The NPP is a graphical method for estimating whether or not a data set is approximately normally distributed. The vertical axis of the NPP is the values of the cumulative relative frequencies. The relative frequencies can be obtained through the sequence number of each data, sorted in increasing order. Considering the first of n data, there is a step from $0/n$ to $1/n$, therefore, the central value of

the interval, that is equal to $0.5/n$, should be assumed as the value of the cumulative relative frequency corresponding to the first datum. Similarly, in correspondence with the successive data there is always a step, hence, the cumulative relative frequency can be generally assumed as follow [55]:

$$frc_i = \frac{(i - 0.5)}{n} \quad (12)$$

The hypothesis of a normal distribution cannot be rejected when the trend of the experimental data is similar to a straight line. However, the trend evaluation is rather arbitrary, hence, there is no confidence level. The main advantage is the accessibility ease to the test results, but, in addition, in case of rejection of normality hypothesis, some information on the causes of non-normality may be derived, for example, random errors appear as local irregularities with respect to a perfect rectilinear trend. A filtering process, which has preferentially eliminated the data farther from the mean value, may lead to a hyper-normal distribution, that is a bell curve characterized by a higher central part than the normal distribution, and, therefore, lower tails. The corresponding NPP has a greater slope in the central part and lower in the tails (figure 3.3.5 – (a)). The outliers' management, acting on the farther values from the mean value, distorts the data distribution and does not allow an exact uncertainty evaluation. Therefore, the outliers' management must be conducted only when the cause of their appearance is known. However, in the production field a hyper-normal distribution is the cause of a selection process. On the other hand, the presence of either discontinuities or tendencies is a typical cause of a hypo-normal distribution. Discontinuities and tendencies are rather different from each other, hence, the NPP has to be used as a sign for further analysis. The bell curve of this distribution has a lower central part than the normal distribution, and, therefore, higher tails. Consequently, the corresponding NPP has a lower slope in the central part and greater in the tails (figure 3.3.5 – (b)).

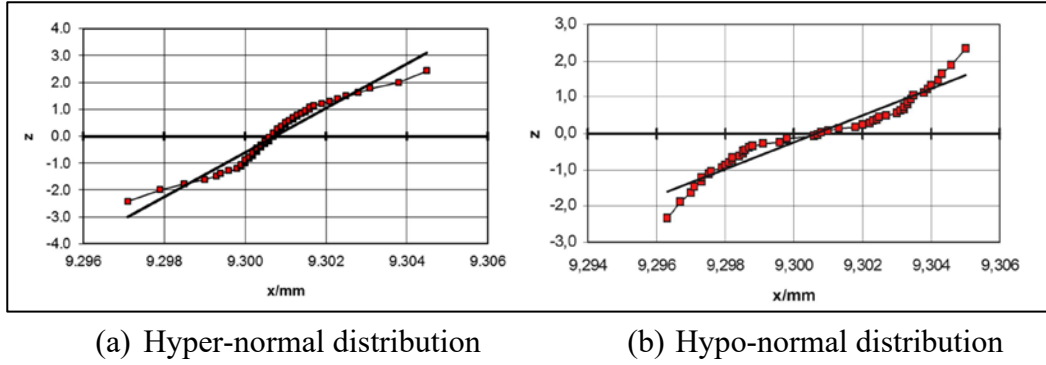


Figure 3.3.5 – NPP of a Hyper- and Hypo-normal distribution [55].

The two normality tests provide different answers, but not incompatible, or rather, considering the respective advantages and disadvantages, we used both to test the normality distribution of the data obtained from the repeated measurements of each pixel. Because the number of pixels is rather high, it was decided to first apply the χ^2 test and then analyse through the NPP the pixels where the normality hypothesis has been rejected. Regardless, the NPP has been implemented, hence, nothing prevents to use it to examine also other pixels.

3.3.3. Correction of possible systematic behaviour

In general, a measurement is affected by an error which is composed of two components. One is defined as a random component that is unpredictable or stochastic, while the other is a systematic component [47]. Examples of systematic errors are the measurement temperature, the drift caused by the temperature in the electronic devices and some effects due to errors in the optics (e.g., lens or reference mirror aberration). When the measurement temperature is sensibly different from the calibration temperature, it can affect the measurement results, hence, its effect has to be considered and corrected. Regarding the electronic equipment, during a measuring session it warms up causing a drift in the measurements results that depends on time. Both the components cannot be eliminated but can be reduced, the random one by increasing the number of observation and the systematic applying a proper correction. Both the χ^2 test and the NPP provides information about the potential presence of systematic behaviour. Examining the data trend, it is possible to detect only some systematic factors. The NPP with hypo-normal

distribution can show a possible drift occurrence, that is a systematic tendency of the results obtained in subsequent times to vary in the same sense, hence, a monotonic trend as a function of time. The estimate of a possible drift may be executed by arranging data in order of measurement collection, as shown in figure 3.3.6, whenever the time corresponding to each surveying is unknown, and applying a linear regression.

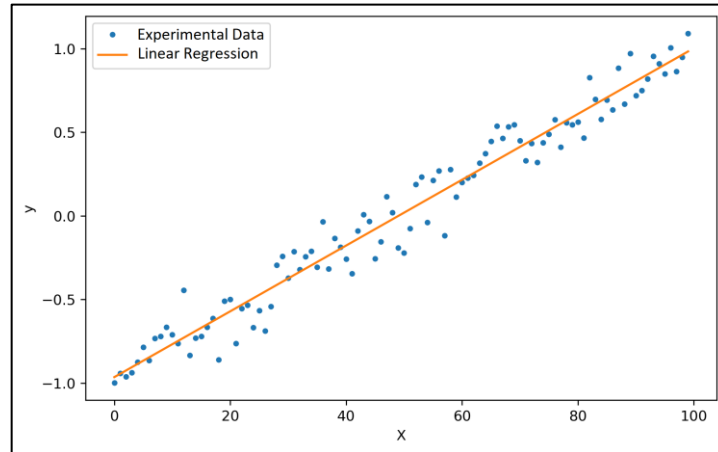


Figure 3.3.6 – Representation of a data arrangement in order of measurement collection and a linear regression.

Before proceeding with the linear regression, it is fundamental to define which is the dependent variable (Y) and, instead, which is the independent variable (X). In this project, we have assumed as the independent variable the time and as the dependent variable the measurement along the z -axis of each pixel.

To achieve a linear regression the steps to follow are three:

- provide a graphical representation of the experimental data in order to make an assumption of the mathematical model which best fits the data trend;
- calculate the parameters of the mathematical model through the least-squared method;
- analyse the residuals, that is the differences between the experimental data and the estimated data obtained from the mathematical model, to check whether the initial assumptions are correct.

If the physics of the phenomenon under study is unknown or results from the combination of several factors, the choice of the mathematical method does not occur through an a priori mathematical procedure, but it is entrusted to the experience of whom analyse the data. Therefore, data representation is rigorously necessary to give acceptable hypotheses. The mathematical model proposes only the shape of the line that describes the experimental data trend, hence, the grade of the polynomial associated with the mathematical model has to be defined. In this project, a first-order polynomial has been assumed to describe data trend of each pixel because analysing the data trend of each pixel is time costly and the first-order polynomial usually fits rather good the data. The polynomial has the following expression [55]:

$$Y = a_0 + a_1X \quad (13)$$

The model parameters a_0 and a_1 are respectively the intercepts and the slope of the straight line that represents the trend of the experimental data.

Once the mathematical model has been assumed, the next step is the extraction of the model's parameters through the least-squared method, easily applicable when the mathematical function is linear with respect to the parameters. All linear functions can be expressed in the following form [55]:

$$Y = G(X_1, \dots, X_j, \dots, X_q) \quad (14)$$

In which the variables X_j are the various contributions of the selected function, in our case we only have one contribution, hence, $X_1 = X$. Switching from the relationships between variables to the relationships between experimental data, the experimental error ε is totally attributed to the variable Y .

$$y = G(x_1, \dots, x_j, \dots, x_q) + \varepsilon \quad (15)$$

Here, conventionally, the error in the predictor x_j is assumed to be negligible; actually, the error is partially due to part to the experimental errors on the data x_j , mostly to the errors on the corresponding values y of the dependent variable Y ; to

cater for this, more refined regression techniques could be adopted, i.e. error-in-variable regression also called orthogonal-least square regression.

At this point, it is possible to linearize the relationship between the experimental data creating an appropriate system of equation for each pixel, in which the experimental values x_{ij} and y_i are known, whereas the parameters a are unknown [55].

$$\begin{Bmatrix} y_1 \\ . \\ y_i \\ . \\ y_n \end{Bmatrix} = \begin{bmatrix} 1 & x_{11} & \dots & x_{1j} & \dots & x_{1q} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & x_{i1} & \dots & x_{ij} & \dots & x_{iq} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & x_{n1} & \dots & x_{nj} & \dots & x_{nq} \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ \dots \\ a_j \\ \dots \\ a_q \end{Bmatrix} + \begin{Bmatrix} \varepsilon_1 \\ . \\ \varepsilon_i \\ . \\ \varepsilon_n \end{Bmatrix} \quad (16)$$

According with the assumption of a first-order model, j is equal to one. The system can be also express in a matrix form [55]:

$$\{y\} = [1, x]\{a\} + \{\varepsilon\} \quad (17)$$

Because of pixels may be correlated among them [57], to achieve a correct result it has been indispensable solve the system with the least-squared method through a function which considered also the correlation among pixels. Hence, the regression curve minimizes the sum of the squares of the deviations ε_i between experimental values y_i of the dependent variable and the corresponding values of the regression curve.

The challenge of this approach is the definition of the covariance matrix from which extract the correlation length of each pixel. Indeed, because of the number of data, it requires a large computer memory that often is far from available. Once defined the covariance matrix, the haracteristic length of the correlation domain has been obtained considering the number of pixels nearby the pixel under test for which the absolute value of the Pearson's coefficient (ρ) is bigger than 0.5.

The Pearson's coefficient is defined as folloes:

$$\rho_{Q,R} = \frac{cov(Q,R)}{\sigma_Q \cdot \sigma_R} \quad (18)$$

in which, $cov(Q,R)$ is the covariance and σ is the standard deviation respectively of Q and R. ρ can assume the values in figure 3.3.7 depending on the sign of the correlation and whether there is a correlation.

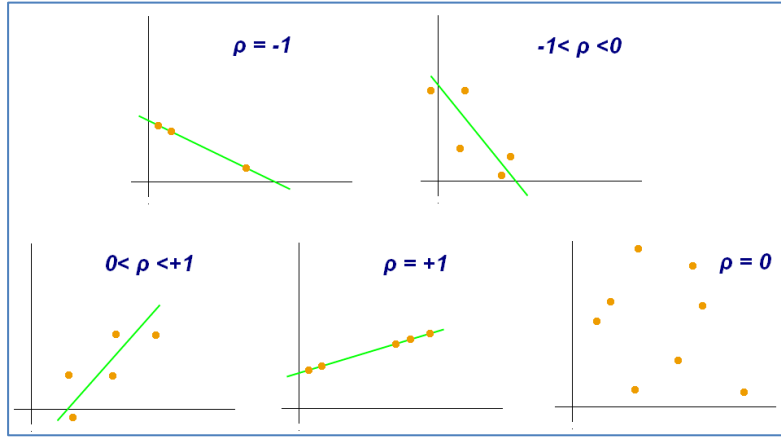


Figure 3.3.7 – Possible values of Pearson's coefficient ρ .

Knowing the correlation length of each pixel (figure 3.3.8) a multivariate linear regression has been implemented to extract the regression coefficients. In order to apply a multivariate linear regression, the vectors in (17) have become matrices to consider the series of multivariate measurements. Hence, the system is now in the following form:

$$[y] = [1, x][a] + [\varepsilon] \quad (19)$$

in which $[y]$ is a n -by- d matrix, $[a]$ is a 2-by- d and $[\varepsilon]$ is a n -by- d matrix; n and d are respectively the number of observations and the number of correlated pixels.

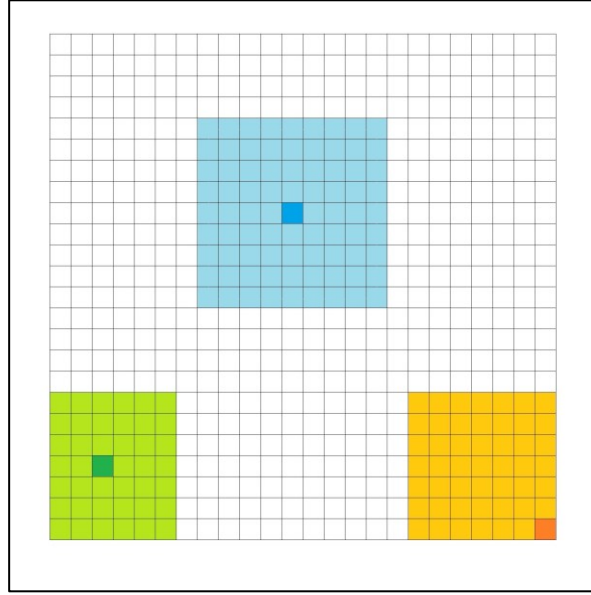


Figure 3.3.8 – Examples of different correlation domains for some pixels of the grid, represented by a darker colour.

Finally, the system has been solved through a feasible generalized least-squares method, in which the generalized least-squares estimate is the covariance-weighted least-square estimate with a known covariance matrix of the pixel under test and its correlated pixels. Given the covariance matrix C_0 the solution of the systems has been obtained minimizing the following summation for each variable:

$$\sum_{i=1}^n (y_i - x_i a) C_0 (y_i - x_i a) \quad (20)$$

The adequacy of the estimated model must be determined. The residuals allow assessing the goodness of the model fit. In fact, if the systematic effect is detected and then corrected, then the residuals present only the random component of the error, hence, the distribution of the residuals should be normal. Therefore, the χ^2 test and the NPP can be used to verify whether the assumed model fit correctly the examined data. There are other methods for the analyses of the goodness of the adapted model, such as the coefficient of determination (R^2) or the significance tests of the regression coefficients.

The coefficient of determination can be defined with the following expression:

$$R^2 = \frac{\sum_i (\hat{y}_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2} \quad (21)$$

in which \hat{y}_i is the predicted data through the mathematical model, y_i is the experimental data and \bar{y} is the mean of the experimental data. The numerator of the fraction is also known as the regression sum of squares (SSR), whereas the denominator is also known as the total sum of squares (SST). Finally, the error or residuals sum of squares (SSE) is:

$$SSE = SST - SSR = \sum_i (y_i - \hat{y}_i)^2 \quad (22)$$

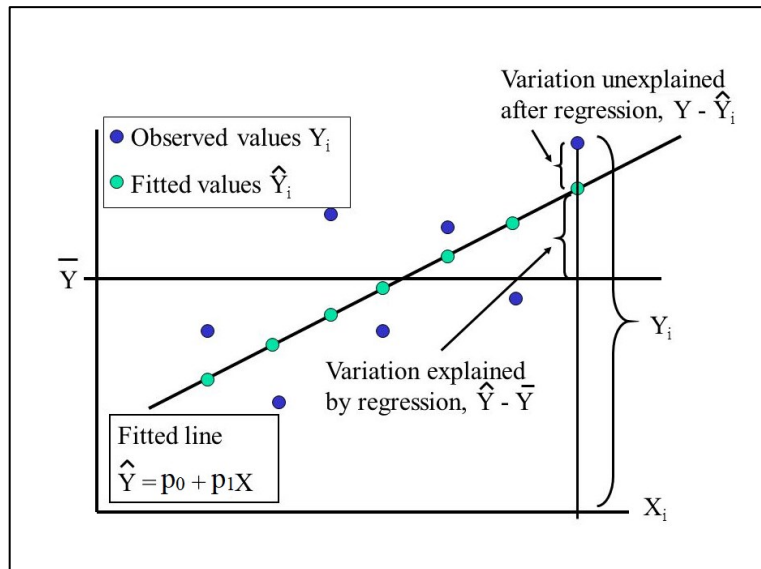


Figure 3.3.9 – Focus on the components of the coefficient of determination.

As can be seen in figure 3.3.9, in order that the model exactly fit the observed values, the coefficient of determination must be equal to unity. Therefore, R^2 should be as higher as possible, in fact, a value of 0.70 suggests that 70% of the experimental data is predicted by the model.

The significance test starts with the definition of the null hypothesis, which, in this circumstance, is verified when the prediction of the regression coefficient is equal to zero. If both the coefficients are null and the regression is correctly carried out, then the distribution of the coefficient is normal and centred in zero. Because of the number of degrees of freedom is finite, the normal distribution is approximate with a t-student distribution. Now, it is possible to extract from the inverse of the t-student the quantile with respect to the assigned probability and the degrees of freedom. In this case, the degrees of freedom are provided by the difference between the number of repeated measurement and the constraints, that is the number of coefficients detected through the linear regression. With the quantile, it is possible to define the confidence interval multiplying the quantile for the standard deviation of the coefficient. Then, the ratio between the coefficient and its confidence interval is the parameter to consider in the analysis of the goodness of fit of the model. Therefore, in order that the regression has been correctly accomplished, this parameter has to be higher than one.

3.4. Uncertainty evaluation

The evaluation of measurement uncertainty, besides being an aim of this project, is core for the thorough characterization of the surface topography measurement area, which is the final purpose of the discussed measurements. In fact, a measurement is only an estimate value of the measurand and to be meaningful has to be coupled with its uncertainty [47]. Currently, regarding the surface measurements conducted by optical instruments, a standard to extract the measurement uncertainty is still missing. Therefore, this is the reason why in literature there are many ways to estimate it and this multitude is confusing. The lack of a standard can be traced back to both the fact that this is a relatively new science and the complexity of the measurand. The originality of this project is that it has the ambition to define the traceability of a surface topography measurement through the estimate of the measurement uncertainty of each pixel. Working with a pixel's map complicates the analyses. From a technological point of view, the limit is the management of an

enormous amount of data due to both the high resolution of some optical instruments and the needed of repeated measurements. This amount of data may not be managed with the everyday systems, but sometimes could be necessary to turn to the cluster computing, as it has been done in this project.

The traceability is a property of the result of a measurement that consists in the relationship to established references, usually national or international standards, through a documented unbroken chain of comparison, all having an established uncertainty [37]. The effect of the measurement instruments can be determined by using either calibrated artefacts or traceable instruments. Whereas for contact instruments a traceable infrastructure is present, for the optical instruments a traceable infrastructure is still under study.

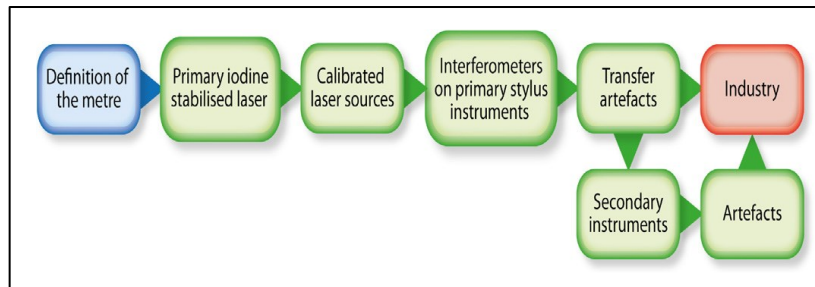


Figure 3.4.1 – Traceability infrastructure.

As can be seen from figure 3.4.1, secondary instruments are calibrated through calibrated artefacts, which have been in turn calibrated by the primary instruments. These artefacts are then used to extract the metrological characteristics of the instruments. To calibrate the measurement instruments, it is also possible to use the ISO 15530-3 [39], which defines the method for the uncertainty evaluation of coordinate measuring machines, still resorting to calibrated workpieces.

At this point, it is useful to provide a definition of measurement uncertainty, that is *non-negative parameter, associated with the result of a measurement, that characterizes the dispersion of the values that could reasonably be attributed to the measurand* [47]. The term ‘reasonably’ that appears in the definition of uncertainty can be expressed by the confidence level in a statistical interpretation of the uncertainty. Hence, the measurement uncertainty defines a confidence interval, i.e.

a band, within which the result of a measurement can be detected with a specified probability level. The uncertainty results from different contributions combined according to the law of variance propagation which can be divided into two components, called A and B, on the basis of their evaluation method. Both the types are based on probability distributions; hence, the uncertainty components can be evaluated through variances or standard deviations [47].

A mathematical model of the measurement is required to evaluate the uncertainty of a measurement through the law of propagation of uncertainty. In general, a measurand Y is obtained from n other quantities through a function f :

$$Y = f(X_1, X_2, \dots, X_n) \quad (23)$$

Using input estimates, such as x_1, x_2, \dots, x_n , for the value of the n quantities X_1, X_2, \dots, X_n , the measurand Y can be estimated by y , which is the result of the measurement, hence:

$$y = f(x_1, x_2, \dots, x_n) \quad (24)$$

A measurement result is usually achieved after a series of observations that can be under either repeatability or reproducibility conditions. The repeatability requires that all the observations are conducted after a short period of time under the same conditions, such as the same measurement procedure, the same observer, etc.. By contrast, the reproducibility implies that the measurement process is the same but one or more condition change during the observation procedure and the period between the measurement can be longer. The estimated standard deviation related to the measurement result y is named combined standard uncertainty $u_c(y)$ and it is determined from the standard uncertainty $u(x_i)$ of each input estimate x_i . When the input quantities are independent, the combined standard uncertainty $u_c(y)$ is given by the law of propagation of uncertainty, that is [47]:

$$u_c(y) = \sqrt{\sum_{i=1}^n \left(\frac{\partial f}{\partial x_i} \right)^2 u^2(x_i)} \quad (25)$$

The partial derivatives $\frac{\partial f}{\partial x_i}$ are called sensitivity coefficients c_i and describe how the measurement results y varies with changes in the values of the input estimates. Finally, the expanded uncertainty U , which is the one we refer to when we simply say uncertainty, is obtained by multiplying the combined standard uncertainty $u_c(y)$ by a coverage factor k :

$$U = ku_c(y) \quad (26)$$

Thus, the result of measurement is then expressed by:

$$Y = y \pm U \quad (27)$$

The coverage factor k usually is either 2 or 3 corresponding to a particular level of confidence p , such as approximately 95 % or 99 %.

Type A contribution is evaluated from a probability density function that derives from an experimentally observed frequency distribution, while a Type B contribution is extracted from an assumed probability density function based on the degree of belief that an event will occur. Therefore, in an analysis of type B standard uncertainty, the lack of repeated measurements leads to evaluate the standard uncertainty by scientific judgement based on all the available information on the possible variability of the input quantity. It is important to say that a Type B evaluation can be as reliable as a Type A evaluation, when:

- the standard uncertainty is taken from a manufacturer's specification, calibration certificate, handbook, or other source;
- on the basis of experience or literature, a probability distribution can be assumed, where the estimated value and its uncertainty are respectively the expectation value of the distribution and the squared root of its variance. The analysis needs the knowledge of the parameter a . The different available distributions are presented in figure 3.4.2.

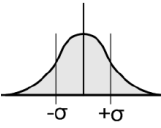
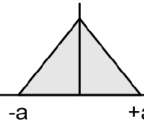
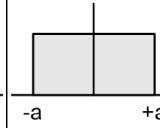
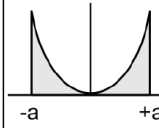

Type	Normal	Triangular	Rectangular	U-shaped	Step
Diagram					
Equivalent standard deviation	$s = \sigma$	$s = \frac{a}{\sqrt{6}} \approx 0,4 \cdot a$	$s = \frac{a}{\sqrt{3}} \approx 0,6 \cdot a$	$s = \frac{a}{\sqrt{2}} \approx 0,7 \cdot a$	$s = \frac{a}{2\sqrt{3}} \approx 0,3 \cdot a$
Use	When type A evaluations can be shown to follow this distribution	When "hard" limits can be estimated easier than σ , e.g.: <ul style="list-style-type: none"> Noise Vibration 	When only the variation limits are known, e.g.: <ul style="list-style-type: none"> Calibration certificates Specification by the manufacturer 	For cyclic influences, e.g.: <ul style="list-style-type: none"> Temperature variation 	When resolution is limited, e.g.: <ul style="list-style-type: none"> Digital readout Verniers Scale read by operator

Figure 3.4.2 – Different probability distribution type for the analysis of type B standard uncertainty.

The model equation is inspired to both the ISO 14253-2 [57] and the ISO 15530-3 [39]. Therefore, we combined the substitution method described in [39] and the correction of the systematic behaviour presented in [58]. The substitution method consists in providing a further correction of the systematic effect measuring both the workpiece and the calibrated standard. It is extremely important that some similarity conditions, listed in table 3, are respected.

Table 3 – Similarity requirements for the workpiece and the calibrated standard [39].

Subject	Requirements	
Dimensional characteristics	Dimensions	Identical within: <ul style="list-style-type: none"> • 10 % beyond 250 mm • 25 mm below 250 mm
	Angles	Identical within $\pm 5^\circ$
Form error and surface texture	Similar due to functional properties	
Material (e.g. thermal expansion, elasticity, hardness)	Similar due to functional properties	
Measuring strategy	Identical	
Probe configuration	Identical	

The evaluation of measurement uncertainty is performed carrying out a sequence of measurements. The differences between the results obtained through the measurement and the calibrated values are used to extract the measurement uncertainty. The uncertainty contribution must take into account the measurement procedure, the calibration process of the calibrated workpieces and the variation of the components under analysis. The effect of any variation in environment condition is considered the correction of the systematic effect performed with respect of JCGM 100 [47]. In order to have enough data to estimate the uncertainty, at least 20 measurements of the calibrated workpiece and at least 10 measurements of the component.

Therefore, the model equation for the estimate of the measurement uncertainty is achieved through some operations. In order to establish traceability and evaluate accuracy, assuming that the correction of the systematic behaviour of the measurement with respect to the calibrated artefact is performed, according to [47] by a first-order model, the results of the regression can be compared to the calibration certificate as follows:

$$y^{trac} = Y^{ref} - X_{ave,cal} \quad (28)$$

In which $X_{ave,cal}$ is the calibrated value in the calibration certificate. Y^{ref} is the measurements of the calibrated workpiece corrected by the linear regression performed to remove the systematic error component, and is expressed by the following equation:

$$Y^{ref} = p \cdot X_{gauge}^{ref} + h \pm rep_{ref} \quad (29)$$

where p and h are respectively the slope and the intercept of the regression straight line, and rep_{ref} can be, based on the measuring conditions, either the repeatability or reproducibility of the measurement. If the fit model of the linear regression would not be first-order, then the equation that expresses Y^{ref} would be changed according to the order of the regression model.

Hence, by substituting (29) in (28), y^{trac} can also be described through the following relationship:

$$y^{trac} = p \cdot X_{gauge}^{ref} + h \pm rep_{ref} - X_{ave,cal} \quad (30)$$

In doing this, we want to correct what has been established as traceability in the calibration certificate with the measurements of the calibrated artefact. As far as uncertainty is concerned, at this stage, it is propagated as follows:

$$u_{trac}^2 = u_{ref}^2 + u_{ave,cal}^2 \quad (31)$$

where, $u_{ave,cal}$ is obtained from the ratio between the expanded uncertainty of the artefact, and where u_{ref} is the measurement uncertainty of the measured calibrated sample.

Considering the component under test, the systematic effects are removed through a linear regression according to [47], with a first-order fit model. The linear regression of the measurements of the component, is expressed by the following relationship:

$$Y^{manuf} = a_1 \cdot X_{opt}^{manuf} + a_0 + \varepsilon \quad (32)$$

in which Y^{manuf} is the measured value, a_1 and a_0 are respectively the slope and the intercept of the regression straight line, and ε is the random error.

Finally, the measurements of the component are corrected through y^{trac} , obtaining in this way the following model equation:

$$\begin{aligned} Y^{meas} &= Y^{manuf} - y^{trac} = \\ &= a_1 \cdot X_{opt}^{manuf} + a_0 + \varepsilon - (p \cdot X_{gauge}^{ref} + h \pm rep_{ref} - X_{ave,cal}) \end{aligned} \quad (33)$$

and the combined measurement uncertainty is extracted as follows:

$$u_{meas}^2 = u_{manuf}^2 + u_{trac}^2 \quad (34)$$

Because of the pandemic of COVID-19, I was not allowed to travel and, besides, the laboratories were inaccessible for a long time, consequently, these affected considerably the project. Not being able to access any laboratory, I could not perform any measurements. Therefore, some micrograph measurements were provided to me by my supervisors, but the measurements of the calibrated artefacts were not available. Hence, we were only able to estimate partially the measurement uncertainty, that is the component obtainable from the repeated measurements of the workpiece. Moreover, we only assumed the order of the model of the regression for the measurements of the calibrated artefact because not being in possession of them we were not able to check the validity of the model. In fact, currently, a complete estimate of the measurement uncertainty requires the measurements of the calibrated artefacts to consider the influence of the optical instrument.

The combined standard uncertainty related to the measurements of the workpiece has been achieved through the law of propagation of uncertainty (equation (25)), adjusting it appropriately to our input quantities of the model. Hence, the combined standard uncertainty is given by:

$$u_{manuf}^2 = (X_{opt}^{manuf} \cdot u_{a1})^2 + u_{a0}^2 + u_{\varepsilon}^2 \quad (35)$$

in which u_{a1} and u_{a0} are the uncertainty contributions of the coefficients of the regression and u_{ε} is the mean squared error of the residuals of the regression. The analysis has been conducted for each pixel.

3.5. Data reporting and visualization

The visualization of the surface has been carried out taking into account the pixel's size, hence, the representation is not an interpolation of the measurement along the z-direction but is a sort of a step sequence of each pixel. As can be seen in figure 3.5.1 the pixel has been considered as a finite entity defined by the point on the lower left.

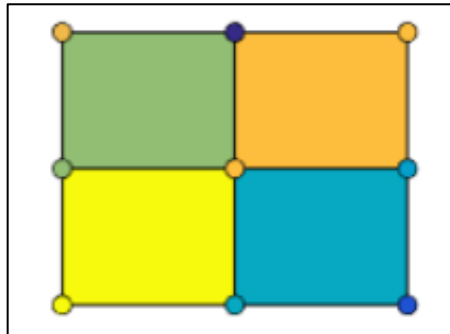


Figure 3.5.1 – Representation of how the surface has been defined in the software.

If the measurement contains non-measured points, they are identified through a white point on the surface, see figure 3.5.2, in order that the representation seems as much realistic as possible.

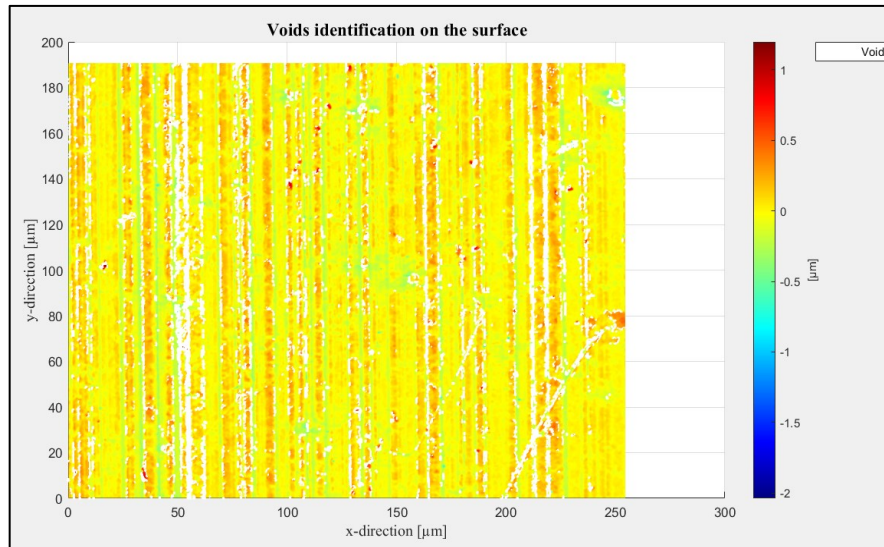
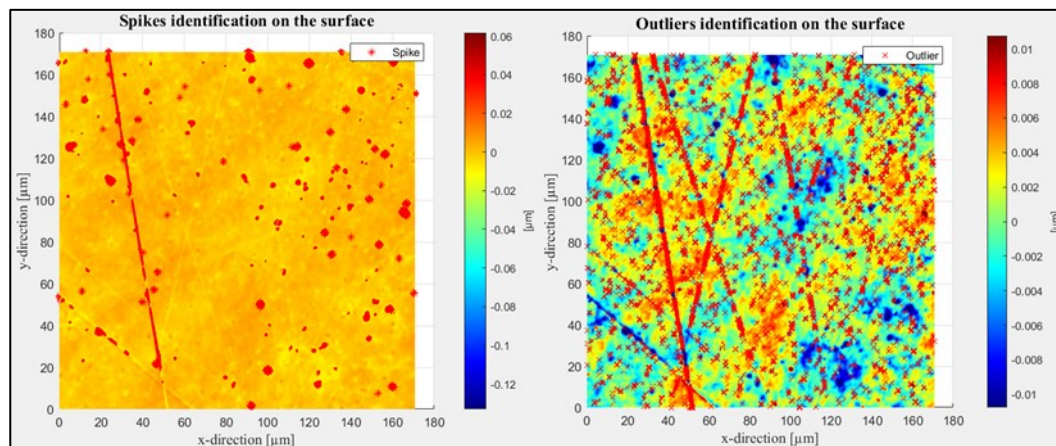


Figure 3.5.2 – Voids identification on the surface.

By contrast, as shows figure 3.5.3, spikes and outliers are highlighted with a red cross in the coordinates that identify the pixel which results a spike or an outlier. This choice has been made to provide a quickly estimate of the number of the spikes or the outliers in order to understand whether the measurement can be valid, or it is better to repeat it.



(a) Spikes

(b) Outliers

Figure 3.5.3 – Spikes (a) and outliers (b) identification on the surface.

Because of the numerousness of data, the adequacy test results of the linear regression estimated model have been provided through a graphical representation. The graphical representation, indeed, allows a qualitative but also quantitative analysis of the results, depending on the display scale. Consequently, while the results of the coefficient of determination are presented in figure 3.5.4, the results of the significance test for both the coefficients estimated by the linear regression are shown in figure 3.5.5.

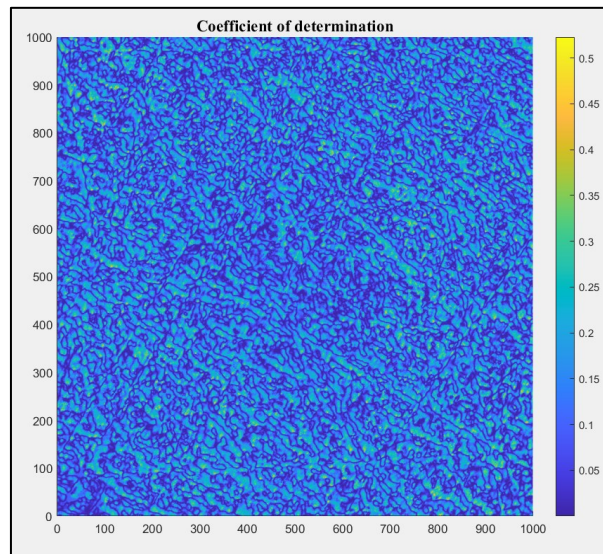


Figure 3.5.4 – Results representation of the coefficient of determination of Zygo measurements.

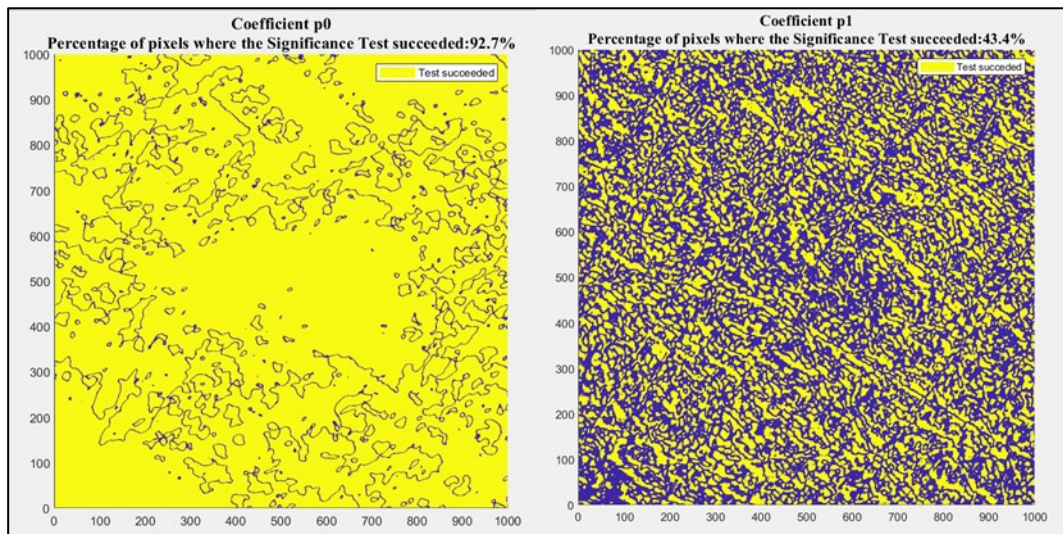


Figure 3.5.5 – Result representation of the significance test of Zygo measurements.

As far as uncertainty is concerned, as mentioned in the previous section, we were not able to perform some calibrated artefact measurements, hence, we could only estimate the uncertainty contribution related to the measurement of the workpiece. Therefore, the uncertainty band representation is only due to this contribution. In order to have a clear representation of the uncertainty band of each pixel we represent it through two surfaces respectively above and under the mean surface (figure 3.5.6).

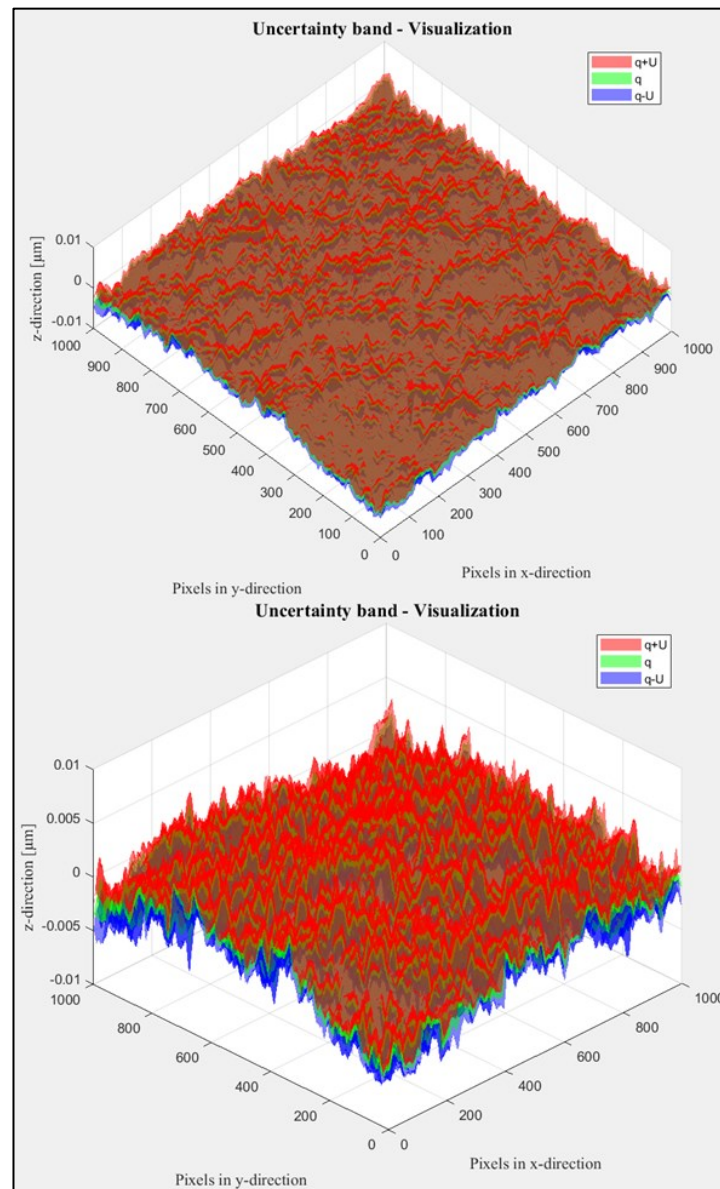


Figure 3.5.6 – Uncertainty band representation from two points of view of Zygo measurements.

In software development, we made some choices about the graphical representation of the results that can be modified by the final user on the basis of the purpose of the analysis. In fact, surface measurements are carried out in order to either check the tolerances or characterize the functionality of the component. After the definition of the measurement uncertainty pixel by pixel and its trend, it may be useful to extract some surface average to carry out the analysis of functionality characterization.

4. RESULTS

In this chapter, the results obtained through the algorithms developed in this project are presented and analysed. The results have been achieved processing two measurement series, one consisting of 30 repeated measurements carried out by the Zygo NewView 9000 and the other of 75 repeated measurements carried out by the Olympus Lext OLS 4100.

Firstly, the surface has been characterized by the extraction of parameters such as S_a and S_q . The two parameters have been calculated both after the plane correction and after the correction of any systematic behaviour, the results are presented in table 4.

Table 4 – Surface characterization results.

After the plane correction		
	Zygo	Olympus
S_a / nm	1.44 ± 0.05	650.87 ± 4.17
S_q / nm	1.84 ± 0.06	810.73 ± 5.79
After the correction of systematic behaviour		
S_a / nm	1.43 ± 0.06	649.11 ± 2.91
S_q / nm	1.82 ± 0.08	805.68 ± 3.72

The other purpose of this project was to research a possible systematic behaviour considering the repeated measurements of each pixel, which has been obtained resorting to the cluster computing. As far as the measurements carried out through the Zygo NewView 9000 are concerned, as can be seen in figure 3.5.4, in many cases there is not a systematic trend. Such a result is also confirmed by the scatter plot of a pixel in figure 3.3.2 and by the fact that in many pixels the slope coefficient of the regression straight line is not significant, see figure 3.5.5.

On the other hand, the regression results of the measurements performed by the Olympus Lext OLS 4100 show systematic behaviour in some region of the pixels' map. Indeed, the coefficient of determination is rather high in some regions, see

figure 4.1.1, hence, in these regions the regression model fits really good the trend of the measurements. Moreover, also the regression coefficients suggest the presence of a systematic effect, being significant for the most in the pixels' map, see figure 4.1.2.

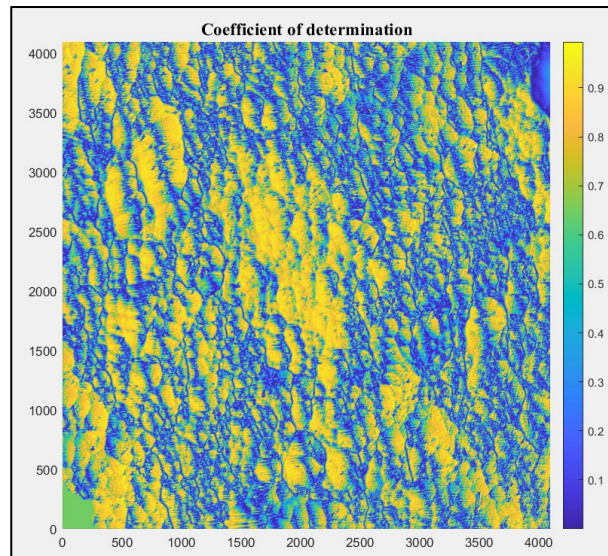


Figure 4.1.1 – Results representation of the coefficient of determination for Olympus measurements.

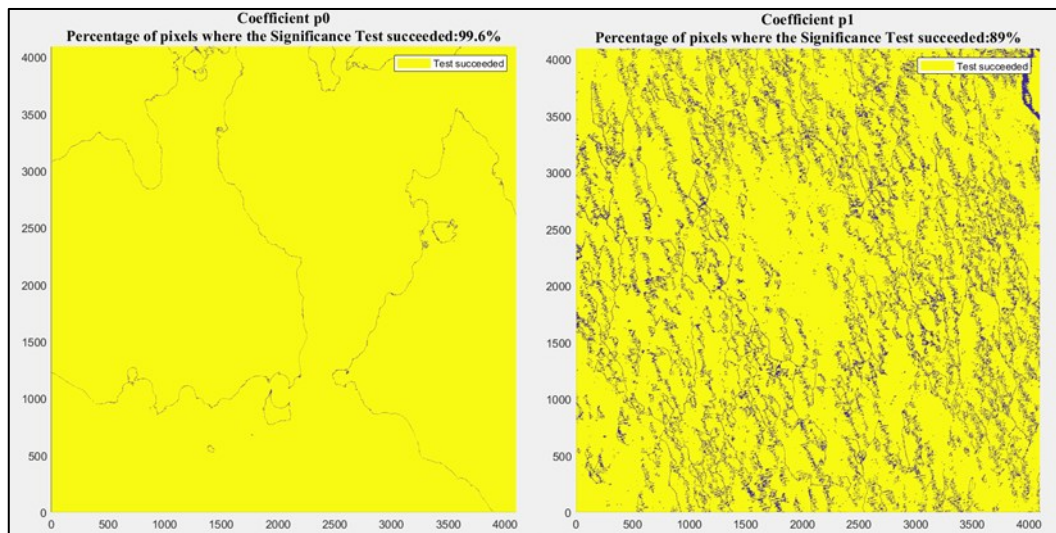


Figure 4.1.2 - Result representation of the significance test of Olympus measurements.

Finally, as mentioned before, only the estimate of the uncertainty component related to the measurement of the workpiece has been obtained. The uncertainty band of the measurements performed by the Zygo NewView 9000 is shown in figure 3.5.6, while the uncertainty band extract from the measurements carried out through the Olympus Lext OLS 4100 is represented in figure 4.1.3.

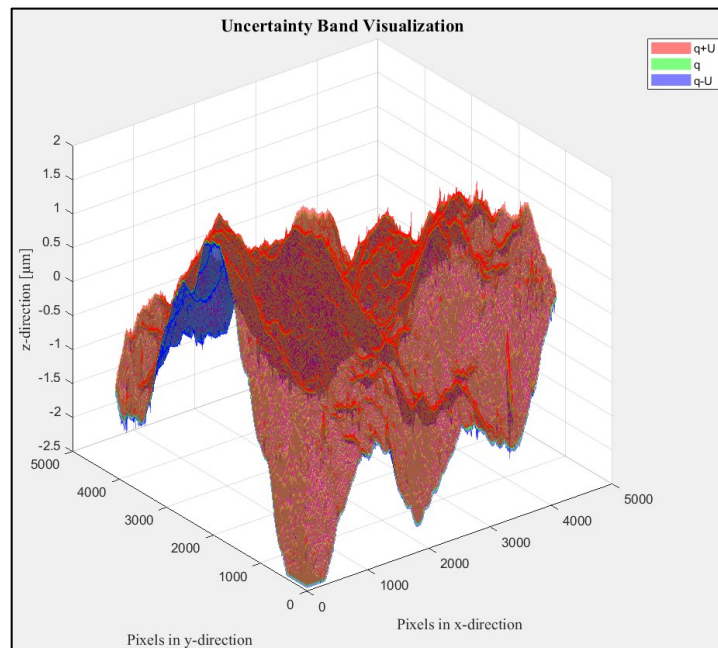


Figure 4.1.3 – Uncertainty band representation from two points of view of Olympus measurements.

5. DISCUSSION

The uncertainty component associated with the measurements of the workpiece is provided through the model equation and the implementation of the law of uncertainty propagation for a possible machine-based evaluation of measurement uncertainty. Not having the measurements of the calibrated artefacts, the evaluation of the measurement uncertainty is incomplete. Indeed, passing through the calibrated artefacts is currently the only way to estimate the uncertainty contribution of the measurement instrument. However, measuring the artefacts and taking into account the model equation provided in this project, the expanded uncertainty can be easily estimated.

As we can understand by the results of this project, the classical method to identify the outliers cannot model the situations in which outliers are mixed with noise. In the micrographs, the presence of low-frequency noise mixed with high-frequency noise hinders the possibility of eliminating it. Moreover, the real measurand is difficult to identify because pixels are correlated and, even considering the correlation among pixels we still neglect the dynamicity of the pixel. Indeed, the pixel is an element with memory, due to the optical sensor, hence, it should be considered as a dynamic element. Therefore, we expect that the correlation will be different carrying out different measuring session in the same measuring conditions. Moreover, the correlation length has been roughly extracted to provide a first analysis of the problem, but because of its complexity, it required further specific researches. In fact, the method to extract this parameter should also consider how the correlation domains of each pixel interact with each other.

A way of identifying the outliers could be to apply the classical method (e.g. Chauvenet's criterion) to a certain surface average that could lower the effect of the noise. In that case, a whole micrograph potentially becomes an outlier. Hence, the sharp separation between areal-parameter computation and pixel by pixel evaluation did not work. Both are required together in the investigation, averaging

when necessary using height parameters of hybrid parameters or whatelse can be related to the surface characterization.

6. CONCLUSIONS

In this project, an architecture of algorithms capable of processing some statistical analysis and evaluating the measurement uncertainty to surface topography measurements has been established. The study has been conducted considering each pixel of the grid and its repeated measurements.

The greatest limit of this approach is due to the numerousness of the data which makes the analysis very complex from a computational point of view. Indeed, cluster computing has been crucial to extract the results presented in chapter 4, which would not be obtained with an ordinary computer.

After some preliminary operation on the surface, the implementation of the statistical analysis allowed some considerations regard the classical method for the outliers' management in the presence of both measurement noise and correlation among pixels. In fact, Chauvenet's criterion and the modified interquartile method have been successfully implemented but they are not able to model the phenomenon. Finally, once the measurements have been corrected of systematic effect, the measurement uncertainty has been estimated through the model equation provided in section 3.4.

To conclude, the implemented architecture allows us to comprehend better the complex subject of areal surface measurements detecting the criticalities of some methods and inspiring feature researches.

ANNEX A - MATLAB Code

```

1 %% Statistical processing and analysis of surface topography data
  for a machine-based evaluation of the measurement uncertainty
2 %%% Matteo Gilardi %%%
3
4 clear
5 clc
6 close all
7
8 %% Data Import
9 lp=1;
10 ns=0;
11 lCount=0;
12 dCount=0;
13 aCount=0;
14 tCount=0;
15 nLoop=0;
16
17 while lp
18
19   [nFile,fPath,~]=uigetfile({'*.*'; '*.lect'; '*.datx'; '*.asc'; '*.txt'},
   ...
20   'Select Source File/Files (.lect, .datx, .asc, .
   txt)', 'MultiSelect', 'on');
21   nLoop=nLoop+1;
22
23   if isequal(nFile,0)
24     disp('The operation was canceled by the user.' );
25     lp=0;
26   else
27     if ~iscell(nFile), nFile={nFile}; end
28
29     if nLoop==1
30       C = cell(length(nFile),1);
31       fileInput=struct('lect',C,'datx',C,'asc',C,'txt',C);
32       S=struct('XIn',[],'YIn',[],'ZIn',[],'Zpc',[],'Zv',[],'Z',[],...
33       'parameters',[],'Ix',[],'Iy',[]);
34       data=struct('lect',S,'datx',S,'asc',S,'txt',S);
35     end
36
37     for q=1:length(nFile)
38       inFile=nFile{q};
39       fFilePath=fullfile(fPath,inFile);
40       [~,~,extT]=fileparts(fFilePath);
41
42       switch extT
43       case '.lect'

```

```

44 lCount=lCount+1;
45 fileInput(lCount).ltext=fFilePath;
46 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
47 %%%% Add here function for opening .ltext file %%%%
48 %%%% lCount to control the files to be opened %%%%
49 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
50 case '.datx'
51 dCount=dCount+1;
52 fileInput(dCount).datx=fFilePath;
53 % dCount to control the files to be opened
54 [data.datx.XIn,data.datx.YIn,data.datx.ZIn(:, :, dCount),
data.datx.Zv(:, :, dCount), ...
55 data.datx.parameters,data.datx.Ix,data.datx.Iy] =
datx_import(fFilePath);
56 data.datx.Zv=logical(data.datx.Zv);
57 case '.asc'
58 aCount=aCount+1;
59 fileInput(aCount).asc=fFilePath;
60 % aCount to control the files to be opened
61 [data.asc.XIn,data.asc.YIn,data.asc.ZIn(:, :, aCount),data.
asc.Zv(:, :, aCount), ...
62 data.asc.parameters,data.asc.Ix,data.asc.Iy] =
asc_import(fFilePath);
63 data.asc.Zv=logical(data.asc.Zv);
64 case '.txt'
65 tCount=tCount+1;
66 fileInput(tCount).txt=fFilePath;
67 % tCount to control the files to be opened
68 [data.txt.XIn,data.txt.YIn,data.txt.ZIn(:, :, tCount),data.
txt.Zv(:, :, tCount), ...
69 data.txt.parameters,data.txt.Ix,data.txt.Iy] =
txt_import(fFilePath);
70 data.txt.Zv=logical(data.txt.Zv);
71 otherwise
72 ns=1;
73 clear fileInput
74 end
75 end
76 if ns, fprintf('File(s) type not supported. \n'); ns=0; end
77
78 if lp
79 reply=input('Opening more file(s)? Y/N (enter --> Y):','s');
80 if ~isempty(reply) && (reply=='N' || reply=='n')
81 lp=0;
82 end
83 end
84 end
85 end
86
87 %% Delete not used variables and definition of the data structure
88 if lCount~=0, fprintf('Opening %d .ltext file(s). \n',
lCount);data=data.ltext;end

```

MGT 2

```

89 if dCount~=0,fprintf('Opening %d .datx file(s). \n',
dCount);data=data.datx;end
90 if aCount~=0,fprintf('Opening %d .asc file(s). \n',
aCount);data=data.asc;end
91 if tCount~=0,fprintf('Opening %d .text file(s). \n',
tCount);data=data.txt;end
92 if nLoop~=0, fprintf('Performed %d iterations. \n', nLoop); end
93 clear q extT ns fPath indx fFilePath C S inFile lp reply nLoop
lCount dCount
aCount tCount fileInput
94
95 %% Imported-Data Visualization
96 t = '3D visualization of the imported surface';
97 visualization(data.XIn,data.YIn,data.ZIn,nFile,t)
98 clear t
99
100 %% Plane Correction
101 compTime=struct('planeCorr',[],'outMan',[],'linReg',[]);
102 reply=input('Applying the Plane Correction? Y/N (enter -->
Y):','s');
103 if ~isempty(reply) && (reply=='N' || reply=='n')
104 disp('The plane correction operation was canceled by the user.');
```

```

105 data.Zpc=data.ZIn;
106 else
107 [indx,tf] = pc_dialogbox();
108 [data.Zpc,compTime.planeCorr] =
plane_correction(indx,tf,data.XIn,data.YIn,
data.ZIn);
109 end
110 clear reply indx tf
111
112 %% Data Visualization after the plane correction
113 t = '3D visualization after the plane correction' ;
114 visualization(data.XIn,data.YIn,data.Zpc,nFile,t)
115 clear t
116
117 %% Voids management after the plane correction
118 data.Zpc(data.Zv)=0;
119
120 %% Roughness parameters before outliers management
121 S=struct('Sa',[],'Sq',[],'Sdq',[]);
122 rougPar=struct('BeforeOutMan',S,'AfterOutMan',S,'AfterLinReg',S);
123
[rougPar.BeforeOutMan.Sa,rougPar.BeforeOutMan.Sq,rougPar.BeforeOutMan
.Sdq] = ...
124 roughness_parameters(data.Zpc,data.parameters,data.Ix,data.Iy);
125 clear S
126
127 %% Spikes and Voids detection
128 spikes=struct('Xs',[],'Ys',[],'Zs',[]);
129 voids=struct('Xv',[],'Yv',[],'Zv',[]);
130 [spikes.Xs,spikes.Ys,spikes.Zs,voids.Xv,voids.Yv,voids.Zv] = ...

```

MGT 3

```

131 sv_detection(data.Zpc,data.Zv,rougPar.BeforeOutMan.Sq,data.Ix,data.Iy
);
132 clear S
133
134 %% Spikes and Voids identification on the surface
135 sv_visualization(nFile,data,spikes,voids)
136
137 %% Spikes management
138 [data.Zpc] = s_management(data.Zpc,rougPar.BeforeOutMan.Sq);
139
140 %% Data Visualization after the spikes management
141 t = '3D visualization after the spikes management' ;
142 visualization(data.XIn,data.YIn,data.Zpc,nFile,t)
143 clear t
144
145 %% Outliers Management
146 outliers=struct('Xol',[],'Yol',[],'Zol',[]);
147 reply=input('Managing the Outliers? Y/N (enter --> Y):' , 's');
148 if ~isempty(reply) && (reply=='N' || reply=='n')
149 disp('The Outliers Management operation was canceled by the
user.' );
150 else
151 [data.Zpc,zol,compTime.outMan] = chauvenet(data.Zpc);
152 % [data.Zpc,zol,compTime.outMan] = modified_IQR(data.Zpc);
153 [outliers.Xol,outliers.Yol,outliers.Zol] =
ol_identification(data.Zpc,data.
Ix,data.Iy,zol);
154 ol_visualization(nFile,data,outliers);
155 end
156 clear reply zol
157
158 %% Statistical Indexes
159 statIndx=struct('M',[],'Me',[],'V',[],'S',[]);
160 [statIndx.M,statIndx.Me,statIndx.V,statIndx.S] =
stat_indx(data.Zpc);
161
162 %% Data Visualization after the outliers management
163 t = '3D visualization after the outliers management' ;
164 visualization(data.XIn,data.YIn,data.Zpc,nFile,t)
165 clear t
166
167 %% Roughness parameters after outliers management
168
[rougPar.AfterOutMan.Sa,rougPar.AfterOutMan.Sq,rougPar.AfterOutMan.Sd
q] = ...
169 roughness_parameters(data.Zpc,data.parameters,data.Ix,data.Iy);
170
171 %% Normality Test
172 NormTest_results=struct('data',[],'residuals',[]);
173 reply=input('Proceeding with the Normality Test for each pixel
and then for the Residuals? Y/N (enter --> Y):' , 's');

```

MGT 4

```

174 if ~isempty(reply) && (reply=='N' || reply=='n')
175 disp('The Normality Test operation was canceled by the user. ');
176 else
177 t='Normality Test for each pixel';
178 [NormTest_results.data] = chi_squared_test(data.Zpc,t);
179 % Normal Probability Plot
180 npp(data.Zpc,NormTest_results.data);
181 end
182
183 %% Research of possible systematic behaviours
184 % data.Zpc=data.Zpc-statIndx.M;
185
186 % Upper triangular block covariance matrix
187 covUBM=covaMat_sampling_fcn(Zpc);
188
189 % Correlation length
190 [row,col,~]=size(covUBM);
191 N=row*col;
192 corr_dist=zeros(1,N);
193 parfor k=1:N
194 [corr_dist(k),~]=correlation_distance(covUBM,0.5,k);
195 end
196 corr_dist=(reshape(corr_dist,row,col))';
197
198 % Regression
199 [~,~,pg]=size(Zpc);
200 t=-floor(pg/2):1:floor(pg/2)-1;
201 [Regression] = cov_linReg(data.Zpc,corr_dist,t);
202
203 clear k row col pg N
204
205 %% Significance Test
206 t = 'Residuals Normality Test';
207 NormTest_results.residuals = significance_test(Regression.Res,
...
208 Regression.RobustnessPar,t,reply);
209 clear t reply
210
211 %% Data Visualization after the correction of systematic
behaviour
212 t = '3D visualization after the correction of systematic
behaviour' ;
213 visualization(data.XIn,data.YIn,Regression.Corrected,nFile,t)
214 clear t
215
216 %% Roughness parameters after the correction of systematic
behaviour
217
[rougPar.AfterLinReg.Sa,rougPar.AfterLinReg.Sq,rougPar.AfterLinReg.Sd
q] = ...
218
roughness_parameters(Regression.Corrected,data.parameters,data.Ix,dat
a.Iy);

```

```

219
220 %% Uncertainty evaluation
221 uncertainty=struct('q',[],'U',[]);
222 [uncertainty.q,uncertainty.U] =
uncertainty_evaluation(Regression);
223 clear S measInst
224
225 %% Uncertainty Visualization
226 U_visualization(uncertainty);

```

MGT 6

```

1 function [XIn,YIn,ZIn,Zv,parameters,Ix,Iy] = datx_import(pathfile)
2
3 scala1 = 1e-3; % From nm to μm
4 scala2 = 1e+9; % From m to nm
5
6 Ix = (h5readatt(pathfile,'/Measurement/Attributes',...
7 'Surface Data Context.Lateral Resolution:Value' ))*scala2;
8 Iy = Ix;
9 parameters = zeros(7,1);
10 % parameters(2) = x_pixels
11 % parameters(3) = y_pixels
12 % parameters(4) = x_length
13 % parameters(5) = y_length
14 parameters(2) = h5readatt(pathfile,'/Measurement/Attributes',...
15 'Data Context.Data Attributes.Camera Width:Value' );
16 parameters(3) = h5readatt(pathfile,'/Measurement/Attributes',...
17 'Data Context.Data Attributes.Camera Height:Value' );
18 parameters(4) = Ix*parameters(2);
19 parameters(5) = Iy*parameters(3);
20
21 ZIn = (h5read(pathfile,'/Measurement/Surface'));
22
23 % Correct the "No Data" values to NaN: typically a large number,
1.7977e+308
24 zv = h5readatt(pathfile,'/Measurement/Surface','No Data');
25 Zv = ZIn == zv;
26 ZIn(Zv) = NaN;
27 ZIn = ZIn*scala1;
28
29 % Definition of the grid in x-direction and y-direction.
30 x = [0:Ix:parameters(4)-1]*scala1;
31 y = [0:Iy:parameters(5)-1]*scala1;
32 % XIn (In as input) is a matrix where each row is a copy of x, and
YIn
33 % is a matrix where each column is a copy of y. The grid
represented

```

datx_import 1

```

34 % by the coordinates X and Y has length(y) rows and length(x)
columns.
35 [XIn,YIn] = meshgrid(x,y);

```

datx_import 2

```

1 %% Extraction of image parameters from the file
2
3 function [XIn,YIn,ZIn,Zv,parameters,Ix,Iy] = asc_import(pathfile)
4
5 % column1: string (%s)
6 % column2: string (%s)
7 % column3: string (%s)
8 % column4: double (%f)
9 formatSpec = '%s%s%s%f';
10 headerLines = 3;
11 lines = 10;
12 fileID = fopen(pathfile,'r');
13
14 if fileID == -1,disp('Error, check file name and path');
15 else
16 dataArray = textscan(fileID, formatSpec,lines, 'HeaderLines', ...
17 headerLines, 'Delimiter', ' ');
18 % It reads file data using the format of the data fields
('formatSpec')
19 % for the number of 'lines'. In this specific case we have some
20 % header lines, and the delimiter is a white-space.
21 fclose(fileID);
22 parameters = dataArray{:,4};
23 end
24
25 % parameters(1) = forcecurve
26 % parameters(2) = x_pixels
27 % parameters(3) = y_pixels
28 % parameters(4) = x_length
29 % parameters(5) = y_length
30 % parameters(6) = x_offset
31 % parameters(7) = y_offset
32
33 Ix = parameters(4)/parameters(2); % Pixel's size, expressed in nm.
34 Iy = parameters(5)/parameters(3);
35
36 scala = 1e-3; % From nm to µm.
37 % Definition of the grid in x-direction and y-direction.
38 x = [0.5*Ix:Ix:parameters(4)]*scala;
39 y = [0.5*Iy:Iy:parameters(5)]*scala;
40 % XIn (In as input) is a matrix where each row is a copy of x, and
YIn

```

asc_import 1

```

41 % is a matrix where each column is a copy of y. The grid
    represented
42 % by the coordinates X and Y has length(y) rows and length(x)
    columns.
43 [XIn,YIn] = meshgrid(x,y);
44
45 % Creation of a datastore from the collection of data specified by
    pathfile.
46 ttlds = tabularTextDatastore(pathfile);
47 % Reading of all data in datastore; if all the data in the
    datastore
48 % does not fit in memory, then readall returns an error.
49 table = readall(ttlds);
50
51 array = table2array(table);
52 array(:,parameters(2)+1) = [];
53 % Definition of the matrix with all the measurement data.
54 ZIn = array(1:1:parameters(3),1:1:parameters(2)).*scala;
55
56 %% Voids detection
57 % Whether in the image voids are present, their position is
    indicated
58 % in a logical table at the bottom of the input file.
59 if length(array) > parameters(3)
60 % Definition of the matrix with all the voids.
61 Zv = array(parameters(3)+1:1:2*parameters(3), ...
62 1:1:parameters(2));
63 else
64 Zv=zeros(size(ZIn));
65 end
66
67 %% Value for Voids.
68 Zv = logical(Zv);
69 % The voids in the ZIn matrix are treated 'as not a number'
70 ZIn(Zv) = NaN;

```

asc_import 2

```

1 function [XIn,YIn,ZIn,Zv,parameters,Ix,Iy] = txt_import(pathfile)
2
3 formatSpec='%f%f%f';
4 fileID = fopen(pathfile,'r');
5 if fileID == -1,disp('Error, check file name and path');
6 else
7 % Reading of file data using the format of the data fields
8 % ('formatSpec') for the number of 'lines'.
9 % In this specific case the delimiter is a white-space.
10 dataArray=textscan(fileID, formatSpec, ...

```

txt_import 1


```

11 'Delimiter', ' ', 'EmptyValue', NaN, 'ReturnOnError', false);
12 fclose(fileID);
13 end
14
15 parameters = zeros(7,1);
16 scala = 1e-3;
17 % parameters(2) = x_pixels
18 % parameters(3) = y_pixels
19 % parameters(4) = x_length
20 % parameters(5) = y_length
21 parameters(2)=sum(dataArray{1,2} == dataArray{1,2}(1,1));
22 parameters(3)=sum(dataArray{1,1} == dataArray{1,1}(1,1));
23 Ix=dataArray{1,1}(2,1)-dataArray{1,1}(1,1); % Pixel's size
expressed in nm.
24 Iy=dataArray{1,2}(1+parameters(2),1)-dataArray{1,2}(1,1);
25 parameters(4)=Ix * parameters(2);
26 parameters(5)=Iy * parameters(3);
27 parameters(6)=min(dataArray{1,1});
28 parameters(7)=min(dataArray{1,2});
29
30 % The input data are reshaped in order to have XIn (In as input)
31 % as a matrix where each row contains the coordinates along
32 % the x-direction, and YIn as a matrix where each column contains
33 % the coordinates along the y-direction.
34 XIn =
((reshape(dataArray{1,1},[parameters(2),parameters(3)]))').*scala;
35 YIn =
((reshape(dataArray{1,2},[parameters(2),parameters(3)]))').*scala;
36 % Off-set removal
37 XIn = XIn-(parameters(6)*scala);
38 YIn = YIn-(parameters(7)*scala);
39 ZIn =
((reshape(dataArray{1,3},[parameters(2),parameters(3)]))').*scala;
40 Zv = isnan(ZIn);

```

txt_import 2

```

1 %% 3-D visualization
2
3 function [] = visualization(XIn,YIn,ZIn,file,t)
4
5 [~,~,dimPage]=size(ZIn);
6 for q=1:dimPage
7 figure('Name',file{q},'NumberTitle','off');
8 surf(XIn,YIn,ZIn(:,:,q),'LineStyle','none','FaceColor','flat')
9 title(t,'FontSize',14,'FontName','Times New Roman')
10 colormap jet; c = colorbar; c.Label.String = '[ $\mu$ m]';
11 xlabel('x-direction [ $\mu$ m]','FontSize',12,'FontName','Times New
Roman');
12 ylabel('y-direction [ $\mu$ m]','FontSize',12,'FontName','Times New
Roman');
13 zlabel('z-direction [ $\mu$ m]','FontSize',12,'FontName','Times New
Roman');
14 end

```

visualization 1

```

1 %% Plane correction dialog box
2
3 function [indx,tf] = pc_dialogbox()
4
5 % This cell array contains the names of the various methods for the
plane correction.
6 list = {'Surface leveling','Surface curvature removal'};
7
8 % It creates a dialog box that allows the user to select one
9 % (the 'SelectionMode' is set on 'single') item from the specified
list.
10 % The function returns two output arguments:
11 % indx, which contains information about which items the user
selected,
12 % and tf, which indicates whether the user made a selection.
13 [indx,tf] = listdlg('ListString',list,'SelectionMode','single',...
14 'ListSize',[350,150],'Name','Plane
correction','OKString','Apply',...
15 'PromptString',{'Select a file.',...
16 'Only one file can be selected at a time.' , ''});
17
18 fprintf('User selected: %s.\n',list{indx});

```

pc_dialogbox 1

```

1 %% Plane Correction
2
3 function [Zpc,ct] = plane_correction(indx,tf,XIn,YIn,ZIn)
4
5 if tf == 1
6     switch indx
7     case 1
8         [Zpc,ct] = surface_leveling(XIn,YIn,ZIn);
9     case 2
10        [Zpc,ct] = surface_currem(XIn,YIn,ZIn);
11    end
12 else
13     disp('The Plane Correction operation was canceled by the user.' );
14 end

```

plane_correction 1

```

1 %% Surface leveling
2
3 function [Zpc,ct] = surface_leveling(XIn,YIn,ZIn)
4
5 tic
6 [~,~,dimPage]=size(ZIn);
7 Zsf = zeros(size(ZIn));
8
9 parfor q=1:dimPage
10    [XOut,YOut,ZOut]=prepareSurfaceData(XIn,YIn,ZIn(:, :, q));
11    % Creation of a surface fit to the data in vectors XOut, Yout and
12    % ZOut,
13    % conveniently processed with 'prepareSurfaceData' function.
14    % Model type to fit is a Polynomial surface, model names are
15    % 'polyij',
16    % where i is the degree in x and j is the degree in y.
17    sf=fit([XOut,YOut],ZOut,'poly11');
18    Zsf(:, :, q)=sf(XIn,YIn);
19 end
20
21 % The plane correction is achieved subtracting the fit surface to
22 % the measured
23 % surface.
24 Zpc=ZIn-Zsf;
25 ct=toc;

```

surface_leveling 1

```

1 %% Surface curvature removal
2
3 function [Zpc,ct] = surface_currem(XIn,YIn,ZIn)
4
5 tic
6 [~,~,dimPage]=size(ZIn);
7 Zsf = zeros(size(ZIn));
8
9 parfor q=1:dimPage
10 [XOut,YOut,ZOut]=prepareSurfaceData(XIn,YIn,ZIn(:, :, q));
11 sf=fit([XOut,YOut],ZOut, 'poly22');
12 % Creation of a surface fit to the data in vectors XOut, Yout and
    ZOut,
13 % conveniently processed with 'prepareSurfaceData' function.
14 % Model type to fit is a Polynomial surface, model names are
    'polyij',
15 % where i is the degree in x and j is the degree in y.
16 Zsf(:, :, q)=sf(XIn,YIn);
17 end
18
19 % The plane correction is achieved subtracting the fit surface to
    the measured
    surface.
20 Zpc=ZIn-Zsf;
21 ct=toc;

```

surface_currem 1

```

1 %% Roughness parameters
2
3 % This function calculates the areal field parameters on the basis
    of the ISO
    25178-2:2012.
4
5 function [Sa,Sq,Sdq] = roughness_parameters(Zpc,parameters,Ix,Iy)
6
7 [~,~,dimPage]=size(Zpc);
8 % Sa=[ $\mu\text{m}$ ]
9 Sa=zeros(1,dimPage);
10 % Sq=[ $\mu\text{m}$ ]
11 Sq=zeros(1,dimPage);
12 % Sdq=[rad]
13 Sdq=zeros(1,dimPage);
14
15 x_pixels=parameters(2);
16 y_pixels=parameters(3);
17
18 parfor q = 1:dimPage
19 Sa(q)=sum(sum(abs(Zpc(:, :, q))))/(x_pixels*y_pixels);

```

roughness_parameters 1

```

20 Sq(q)=sqrt(sum(sum(Zpc(:, :, q).*Zpc(:, :, q)))/(x_pixels*y_pixels));
21
22 % Zpc has to be expressed in nm because Ix and Iy are expressed in
nm.
23 [zdx,zdy]=gradient(Zpc(:, :, q)*1e3,Ix,Iy);
24 Sdq(q)=sqrt(sum(zdx.^2+zdy.^2, 'all')/(x_pixels*y_pixels));
25 end

```

roughness_parameters 2

```

1 %% Spikes and Voids detection
2
3 function [Xs,Ys,Zs,Xv,Yv,Zv] = sv_detection(Zpc,Zv,Sq,Ix,Iy)
4
5 [~,~,dimPage]=size(Zpc);
6 Xs=cell(1,dimPage);
7 Ys=cell(1,dimPage);
8 Zs=cell(1,dimPage);
9 Xv=cell(1,dimPage);
10 Yv=cell(1,dimPage);
11 parfor q = 1:dimPage
12 %Spikes
13 Zs{1,q}=abs(Zpc(:, :, q)) > 3*Sq(1,q);
14 Zs{1,q}=Zpc(:, :, q).*Zs{1,q};
15 %Find indices and values of nonzero elements.
16 [Ys{1,q},Xs{1,q},Zs{1,q}] = find(Zs{1,q});
17 Xs{1,q} = ((Xs{1,q}-1).*Ix)*1e-3;
18 Ys{1,q} = ((Ys{1,q}-1).*Iy)*1e-3;
19 %Voids
20 [Yv{1,q},Xv{1,q}] = find(Zv(:, :, q));
21 Xv{1,q} = ((Xv{1,q}-1).*Ix)*1e-3;
22 Yv{1,q} = ((Yv{1,q}-1).*Iy)*1e-3;
23 end
24
25 Zv = cell(1,dimPage);
26 parfor q = 1:dimPage
27 Zv{1,q} = zeros(length(Xv{1,q}),1);
28 end

```

sv_detection 1

```

1 %% Spikes and Voids identification on the surface
2
3 function [] = sv_visualization(file,data,spikes,voids)
4
5 [~,~,dimPage]=size(data.Zpc);
6 for q=1:dimPage
7 figure('Name',file{q},'NumberTitle','off');
8 surf(data.XIn,data.YIn,data.Zpc(:,:,
q),'LineStyle','none','FaceColor','flat')
9 title('Spikes and Voids identification on the surface' , 'FontSize',
14, 'FontName','Times New Roman')
10 colormap jet; c = colorbar; c.Label.String = '[ $\mu$ m]';
11 hold on
12 % Displays cross at the locations specified by the vectors Xs, Ys,
and Zs to show the spikes.
13 s1=scatter3(spikes.Xs{1,q},spikes.Ys{1,q},spikes.Zs{1,
q}, '*', 'MarkerEdgeColor','red');
14 hold on
15 % Displays circles at the locations specified by the vectors Xv,
Yv, and Zv
to show the voids.
16 s2=scatter3(voids.Xv{1,q},voids.Yv{1,q},voids.Zv{1,q},5,
'w','filled');
17 legend([s1 s2], 'Spike', 'Void', 'Location', 'northeast');
18 xlabel('x-direction [ $\mu$ m]', 'FontSize',12, 'FontName', 'Times New
Roman');
19 ylabel('y-direction [ $\mu$ m]', 'FontSize',12, 'FontName', 'Times New
Roman');
20 zlabel('z-direction [ $\mu$ m]', 'FontSize',12, 'FontName', 'Times New
Roman');
21 end

```

sv_visualization 1

```

1 %% Spikes management with a tresholding method
2
3 function [Zpc] = s_management(Zpc,Sq)
4
5 [dimRow,dimCol,~] = size(Zpc);
6 Sq = permute(Sq,[1 3 2]);
7 Sq = repmat(Sq,[dimRow dimCol 1]);
8 % Upper limit
9 Zs = Zpc>3*Sq;
10 Zs = logical(Zs);
11 Zpc(Zs) = 3*Sq(Zs);
12 % Lower limit
13 Zs = Zpc<-3*Sq;
14 Zs = logical(Zs);
15 Zpc(Zs) = -3*Sq(Zs);

```

s_management 1

```

1 %% Chauvenet's criterion for each pixel
2
3 function [Zpc,zol,ct] = chauvenet(Zpc)
4
5 tic
6 [dimRow,dimCol,dimPage] = size(Zpc);
7 % Mean calculation along dimension 3 of the Zpc matrix, in this way
8 % the mean of each pixel of the Zpc value for the n images is
9 % obtained.
10 M = mean(Zpc,3);
11 % Standard deviation extraction along the dimension 3 of the Zpc
12 % matrix.
13 S = std(Zpc,0,3);
14 % Inverse cdf for the normal distribution with mean M and standard
15 % deviation S. xli and xls are the exclusion boundaries.
16 xli = norminv(0.25/dimPage,M,S);
17 xls = norminv(1-(0.25/dimPage),M,S);
18 % Chauvenet's criterion is applied to detect the outliers.
19 zol = Zpc > xls | Zpc < xli;
20 % This matrix represents which are the pixel where an outlier is
21 % detected.
22 % The presence of an outlier is identified by the value 1.
23 n_ol = sum(zol,3);
24
25 figure('Name','Number of outliers per pixel-Chauvenet Criterion'
26 ,...
27 'NumberTitle','off');
28 surf(1:dimRow,1:dimCol,n_ol,'LineStyle','none','FaceColor',...
29 'flat','EdgeColor','flat');
30 colormap jet
31 title(['Total number of Outliers: ',num2str(sum(n_ol,'all'))],...
32 'FontSize',14,'FontName','Times New Roman')
33 xlabel('pixels in x-direction','FontSize',12,'FontName','Times New
34 Roman');
35 ylabel('pixels in y-direction','FontSize',12,'FontName','Times New
36 Roman');
37
38 % Identification of the non-outliers values using Chauvenet's
39 % criterion.
40 Znol = Zpc < xls | Zpc > xli;
41 % The mean is performed taking into account only the non-outliers
42 % values.
43 M = mean(reshape(Zpc(Znol),size(Zpc)),3);
44 % The matrix with the mean value for each pixel is replicated
45 % along
46 % the pages to have a matrix with the same size of Zpc.
47 M = repmat(M,[1 1 dimPage]);
48 % The outliers are replaced with the mean value.
49 %The process is executed taking into account a single pixel each
50 % time.
51 Zpc(zol) = M(zol);
52 ct=toc;

```

chauvenet 1

```

1 %% Modified IQR method for each pixel
2
3 function [Zpc,Zol,ct] = modified_IQR(Zpc)
4
5 tic
6 [dimRow,dimCol,dimPage] = size(Zpc);
7 % Probabilities
8 p = [0.25 0.75];
9 % 'Whisker' for the modified IQR method
10 w = 1.5*(1+0.1*log(dimPage/10));
11 % Quantiles of the elements in data vector or array Zpc along the
dimension 3
12 % for the probabilities p.
13 q = quantile(Zpc,p,3);
14 % Interquartile range along the dimension 3 of Zpc.
15 Y = iqr(Zpc,3);
16
17 % Definition of the modified IQR method to detect the outlines.
18 Zol = Zpc < (q(:,:,1) - w.*Y) | ...
19 Zpc > (q(:,:,2) + w.*Y);
20 n_ol = sum(Zol,3);
21 % Identifies the non-outliers values using the modified IQR
method.
22 Znol = Zpc > (q(:,:,1) - w.*Y) | ...
23 Zpc < (q(:,:,2) + w.*Y);
24 % The mean is performed taking into account only the non-outliers
values.
25 M = mean(reshape(Zpc(Znol),size(Zpc)),3);
26
27 figure('Name','Number of outliers per pixel-Modified IQR Method'
,...
28 'NumberTitle','off');
29 surf(1:dimRow,1:dimCol,n_ol,'LineStyle','none','FaceColor',...
30 'flat','EdgeColor','flat');
31 colormap jet
32 title(['Total number of Outliers: ',num2str(sum(n_ol,'all'))],...
33 'FontSize',14,'FontName','Times New Roman')
34 xlabel('pixels in x-direction','FontSize',12,'FontName','Times New
Roman');
35 ylabel('pixels in y-direction','FontSize',12,'FontName','Times New
Roman');
36
37
38 % The matrix with the mean value for each pixel is replicated
along
39 % the pages to have a matrix with the same size of Zpc.
40 M = repmat(M,[1 1 dimPage]);
41 % The outliers are replaced with the mean value.
42 %The process is executed taking into account a single pixel each
time.
43 Zpc(Zol) = M(Zol);
44 ct=toc;

```

modified_IQR 1


```

1 %% Outliers Identification
2
3 function [Xol,Yol,Zol] = ol_identification(Zpc,Ix,Iy,zol)
4
5 [~,~,dimPage] = size(Zpc);
6 Xol = cell(1,dimPage);
7 Yol = cell(1,dimPage);
8 Zol = cell(1,dimPage);
9 zol = zol.*Zpc;
10 parfor k = 1:dimPage
11 [Yol{1,k},Xol{1,k},Zol{1,k}] = find(zol(:, :, k));
12 Xol{1,k} = ((Xol{1,k}-1).*Ix)*1e-3;
13 Yol{1,k} = ((Yol{1,k}-1).*Iy)*1e-3;
14 end

```

ol_identification 1

```

1 %% Outliers identification on the surface
2
3 function [] = ol_visualization(file,data,outliers)
4
5 [~,~,dimPage]=size(data.Zpc);
6 for q=1:dimPage
7 figure('Name',file{q},'NumberTitle','off');
8 surf(data.XIn,data.YIn,data.Zpc(:, :,
q),'LineStyle','none','FaceColor','flat');
9 colormap jet; title('Outliers identification in the surface',...
10 'FontSize',14,'FontName','Times New Roman')
11 hold on
12 % A cross at the locations specified by the vectors Xol, Yol, and
Zol
13 % shows the outliers.
14 s1 = scatter3(outliers.Xol{1,q},outliers.Yol{1,q}, ...
15 outliers.Zol{1,q}, 'x', 'MarkerEdgeColor','red');
16 legend(s1, 'Outlier', 'location', 'northeast');
17 c = colorbar; c.Label.String = '[μm]';
18 xlabel('x-direction [μm]', 'FontSize',12,'FontName','Times New
Roman');
19 ylabel('y-direction [μm]', 'FontSize',12,'FontName','Times New
Roman');
20 zlabel('z-direction [μm]', 'FontSize',12,'FontName','Times New
Roman');
21 end

```

ol_visualization 1

```

1 %% Statistical Indexes
2
3 function [M,Me,V,S] = stat_indx(Zpc)
4
5 % Mean, Median and Variance along dimension 3 of the Zpc matrix
6 % in order to have the mean of the repeated measurements.
7 M = mean(Zpc,3);
8 Me = median(Zpc,3);
9 V = var(Zpc,0,3); % When the second element is 0, V is normalized
by the number
of observations-1.
10 S = sqrt(V); % Standard deviation.

```

stat_indx 1

```

1 %% Chi squared method for normality test
2
3 function [h] = chi_squared_test(Zpc,t)
4
5 [dimRow,dimCol,dimPage] = size(Zpc);
6 % Definition of the matrix with the Normal Distribution test result
for each
pixel.
7 h = zeros(dimRow,dimCol);
8 for i = 1:dimRow
9 parfor j = 1:size(Zpc,2)
10 % Reshape is used to get the column vector required by the
function.
11 % The column vector contains the repeated measurements for a
pixel.
12 h(i,j) = chi2gof(reshape(Zpc(i,j,:),dimPage,1), 'Alpha',0.05);
13 % If h=1, this indicates the rejection of the null hypothesis at
the
Alpha significance level.
14 % If h=0, this indicates a failure to reject the null hypothesis
at the
Alpha significance level.
15 end
16 end
17
18 s = sum(h,'all')/(dimRow*dimCol)*100;
19 format shortg
20 s = round(s,3,'significant');
21 figure('Name','Normality Test','NumberTitle','off');
22 title({t,['Percentage of pixels where the null hypothesis is
rejected: ' ,...
23 num2str(s),'%']},'FontSize',14,'FontName','Times New Roman')

```

chi_squared_test 1

```

24 surface(h,'LineStyle','none','FaceColor','flat')
25 axis square, axis equal, axis xy, axis([0 size(h,2) 0 size(h,1)])
26 xlabel('Pixels in x-direction','FontSize',12,'FontName','Times New Roman');
27 ylabel('Pixels in y-direction','FontSize',12,'FontName','Times New Roman');
28 legend('Null hypothesis failed to reject');

```

chi_squared_test 2

```

1 %% Normal Probability Plot
2
3 function [] = npp(Zpc,h)
4
5 [~,~,dimPage]=size(Zpc);
6
7 [x,y]=find(h);
8 pixel = cell(length(x),1);
9 for q=1:length(x)
10 pixel(q)={['Pixel:',num2str(x(q))',' ',num2str(y(q))',' ']};
11 end
12
13 [pixelIndx,pt] =
listdlg('ListString',pixel,'ListSize',[500,150],...
14 'Name','Normal Probability Plot-Pixels where the null hypothesis
is rejected','OKString','Apply',...
15 'PromptString',{'Multiselection is on.'},...
16 'Pixel:(x,y) ---> the x index represent the row of the pixels
grid, the y index the column.'});
17 if pt == 0, disp('The pixel(s) selection process was canceled by
the user.');
```

```

18
19 for q=1:length(pixelIndx)
20 figure('Name',['Pixel:
',num2str(x(pixelIndx(q))),',' ',num2str(y(pixelIndx
(q))),',' '],...
21 'NumberTitle','off')
22
23 normplot(reshape(Zpc(x(pixelIndx(q)),y(pixelIndx(q)),:),1,dimPage));
23 end

```

npp 1

```

1 %% Extraction of the upper triangular block covariance matrix
2
3 function [covUBM]=covaMat_sampling_fcn(ZIn)
4
5
6 %% Import dataset
7 % the indeces are adjusted for surface data organized in each page
8 [dimR,dimC,dimP] = size(ZIn);
9 dsR=ones(dimP,dimR*dimC);
10
11 % the reshape of the matrix is done for each page separately,
which becomes
12 % a single row in the matrix dsR (data set reshaped)
13 parfor q=1:dimP
14 dsR(q,:)=reshape(ZIn(:,:,q)',[1,dimR*dimC]);
15 end
16 clear q
17
18 w=1:dimR*dimC; % w=1:total point of the matrix
19 shortMatR=dsR(:,w);
20
21 % [dimRow,dimCol] = size(shortMatR);
22 % covBM=cov(shortMatR); % full covariance matrix
23 % [mutualcorr,delays]=xcorr(shortMatR); % cross-correlation
24 % [Mcorr,idxM]=max(mutualcorr);
25 % [mcorr,idxm]=min(mutualcorr);
26 % [MMcorr,idxM]=max(Mcorr);
27 % [mmcorr,idxm]=min(mcorr);
28 % delayM=idxM(idxM); % index delay of maximum
29 % delaym=idxm(idxm); % index delay of maximum
30
31 %% Initiate block matrices
32 % fprintf('Type the number of diagonal blocks (submultiple of the
\ndimension of column in the covariance matrix %d): ',dimRow*dimCol);
33 % nSplit=input(' ');
34 nSplit = max([dimR,dimC]); % number of diagonal elements
35
36 % Calculation of a upper triangular block covariance matrix
37 covUBM=covaMatUT_cluster(shortMatR,nSplit); % versione cluster
38
39 % upper triangular block covariance matrix
40 iLimUMB=min(covUBM,[],'all');
41 % iLimUMB=-1;
42 ILimUMB=max(covUBM,[],'all');
43 % ILimUMB=1;
44
45 for q=1:nBlock
46 f=figure(q+2);
47 imC=imagesc(covUBM(:,:,q),[iLimUMB ILimUMB]);
48 colormap(gray)
49 %colormap(bone)
50 UMB=imC.CData;
51 %imC.CDataMapping='direct';

```

covaMat_sampling_fcn 1

```

52 imC.CDataMapping='scaled';
53 % savefig(f, strcat('Image-like_representation_upper triangular
block
covariance matrix', num2str(q)));
54 close(f);
55 end
56 clear q

```

covaMat_sampling_fcn 2

```

1 %% Calculation of a upper triangular block covariance matrix
2
3 function covfUBM=covaMatUT_cluster(dsfR,nfSplit)
4
5 [~,dimfCol]=size(dsfR);
6 nfSel=dimfCol/nfSplit; % number of elements in a block
matrix
7 nfBdia=dimfCol/nfSel; % number block matrices on the
diagonal
8 nfBlock=0.5*nfSplit*(nfSplit+1); % number of block matrix
9 covfUBM=nan(nfSel,nfSel,nfBlock); % block matrices arranged in
pages (upper triangular side)
10
11 %nfp=0; % track of the number of pages
in cBM
12 %cfi=0; % selection of block matrices on
lines parallel to the block diagonal
13 % treat separately the diagonal line of blocks
14 % parfor v=1:nfBdia
15 % for p=1:nfSel
16 % for q=p:nfSel % build the blocks on the
diagonal
17 % tmpfC=cov(dsfR(:,p+(v-1)*nfSel),dsfR(:,q+(v-1)
*nfSel));
18 % covfUBM(p,q,v)=tmpfC(1,2);
19 % end
20 % end
21 % end
22 %nfp=nfBdia; % matrix pages update
23 clear v p q
24 jjj=1:1:nfSplit;delta=nfSplit-(jjj-1);delta0=[0 delta];
25 end_diag=cumsum(delta0);end_diag=end_diag(2:end);
26 kk=[0 0:1:nfSplit-2];
27 TCOV=mat2cell(covfUBM,nfSel,nfSel,[nfSplit diff(end_diag)]);
28 % A=zero(nfSel,nfSel,nfBdia-2+1);
29 parfor u=1:nfSplit % lines of blocks
30 A=TCOV{u};
31 if u==1
32 for v=1:nfBdia

```

covaMatUT_cluster 1

```

33 for p=1:nfSel
34 for q=p:nfSel % build the blocks on the
diagonal
35 tmpfC=cov(dsfr(:,p+(v-1)*nfSel),dsfr(:,q+(v-1)*nfSel));
36 A(p,q,v)=tmpfC(1,2);
37 end
38 end
39 TCOV{u}=A;
40 end
41 else
42 for v=1:nfBdia-u+1
43
44 for p=1:nfSel
45 for q=1:nfSel % build the upper blocks
46 tempC=cov(readMatrix1(dsfr,p,v,nfSel),readMatrix2(dsfr,
q,v,nfSel,kk(u)));
47 A(p,q,v)=tempC(1,2);
48
49 end
50 end
51 TCOV{u}=A;
52 end
53 end
54 end
55 covfUBM=cell2mat(TCOV);
56
57 end
58
59 function [outputM]=readMatrix1(inputM,p,v,nfSel)
60 outputM=inputM(:,p+(v-1)*nfSel);
61 end
62
63 function [outputM]=readMatrix2(inputM,q,v,nfSel,cfi)
64 outputM=inputM(:,q+(v+cfi)*nfSel);
65 end
66
67 % function [outputM]=readMatrixDiag(inputM,)

```

covaMatUT_cluster 2

```

1 %% Determination of correlation distance
2 % I: covariance matrix arranged as (p,q,v)
3 % the measured surface is p*q, v=max{p,q}
4 % the covariance is uppertriangular
5 % rearrange the covariance of each pixel in a single array
6 % the correlation distance is function of a threshold s
7 % i counts the interested pixel and ranges from 1 to p*q
8
9 function [corr_dist,corr_pxl]=correlation_distance(covUT,s,i)
10 [p,q,~]=size(covUT);
11 var_pxl=zeros(q,p);
12
13 r=floor(i/(p));c=i-r*p;
14 if c==0
15 r=r-1; c=p;
16 end
17 jjj=1:1:p;delta=p-(jjj-1);delta0=[1 delta];k=r+1;
18 start_diag=cumsum(delta0);start_diag=start_diag(1:end-1);
19 clear jjj delta delta0
20 corr_pxl=zeros(p,q);
21 % main symmetric diagonal
22 if r>0
23 corr_pxl_vert=zeros(r,q);
24 end
25
26 corr_pxl(r+1,1:c)=(covUT(1:c,c,start_diag(1)+r))';
27 corr_pxl(r+1,1+c:end)=covUT(c,1+c:end,start_diag(1)+r);
28
29 % variance diagonal
30 parfor jj=1:p
31 var_pxl(:,jj)=diag(covUT(:, :, jj));
32 end
33 var_pxl=var_pxl';
34 deltaneg=0:-1:-p+1;
35 % other diagonals
36 % negj=-1.*(1:1:p)+k;
37 if r==floor(p/2) && rem(p,2)~=0
38 parfor j=2:p-r
39
40
41 % if j<=r+1 % diagonals with double entrances
42
43 corr_pxl_vert(j-1,:)=(covUT(:,c,start_diag(j)+r+deltaneg
(j)))';%% vertical
44 corr_pxl(j+r,:)=covUT(c,:,start_diag(j)+r); %% horizontal
45 % else % diagonals with single entrances (r+j-1)*q+1
46 % corr_pxl(j+r,:)=covUT(c,:,start_diag(j)+r);%% horizontal
47 %
48 % end
49
50 end
51 else
52 if r<floor(p/2) || (r==floor(p/2) && rem(p,2)==0) % extra block

```

correlation_distance 1

```

With horizontal reading
53 parfor j=2:p-r
54
55
56 if j<=r+1 % diagonals with double entrances reading
57
58 corr_pxl_vert(j-1,:)=(covUT(:,c,start_diag(j)+r+deltaneg(j)))';%%
vertical
59 corr_pxl(j+r,:)=covUT(c,:,start_diag(j)+r); %% horizontal
60 else % diagonals with single entrances (r+j-1)*q+1
61 corr_pxl(j+r,:)=covUT(c,:,start_diag(j)+r); %% horizontal
62 end
63
64 end
65 elseif r>floor(p/2) % extra block with vertical reading
66 parfor j=2:r+1 %%
67
68
69 if j<=p-r % diagonals with double entrances reading
70
71 corr_pxl_vert(j-1,:)=(covUT(:,c,start_diag(j)+r+deltaneg
(j)))';%% vertical
72 corr_pxl(j+r,:)=covUT(c,:,start_diag(j)+r); %% horizontal
73 else % diagonals with single entrances (r+j-1)*q+1
74 corr_pxl_vert(j-1,:)=(covUT(:,c,start_diag(j)+r+deltaneg
(j)))';%% vertical
75
76 end
77
78 end
79 end
80 end
81 % Compacting
82 if r>0
83 corr_pxl_vert=flip(corr_pxl_vert,1);
84 corr_pxl(1:r,:)=corr_pxl_vert; %reshape(corr_pxl_vert',1,[]);
85 clear corr_pxl_vert
86 end
87
88 % var_pxl=reshape(var_pxl',1,[]);
89 % Calculating
90 maxcov=(sqrt(var_pxl).*sqrt(var_pxl(r+1,c))); %%%%%%%%%
91 Pearson_pxl=abs(round(corr_pxl,8)./round(maxcov,8));
92 % Pearson_pxl_sq=Pearson_pxl.^2;
93 logic=Pearson_pxl<s;%logic_sq=Pearson_pxl_sq<0.5;
94 if i==1 || i==1000*1000 || i==1000*500 || i==500*500 || i==250*250
95 f=figure;surf(Pearson_pxl);
96 savefig(f, strcat('Pearson_pxl', num2str(i)));
97 close(f);
98 end
99
corr_distxn=find(logic(r+1,1:c),1);corr_distxp=find(logic(r+1,c+1:end
),1)+c;

```

correlation_distance 2


```

100
corr_distyn=find(logic(1:r+1,c),1);corr_distyp=find(logic(r+1+1:end,c),1);
101 corr_dist=max([corr_distxn,corr_distxp,corr_distyn,corr_distyp]);
102 if isempty(corr_dist)
103 corr_dist=p*q-min([r+1,c]);
104 end
105
106 end
107

```

correlation_distance 3

```

1 %% Perform regression based on Covariance of pixels correlated
2 % B is cell array containing the output of the regression.
3 % Regression is based on mvregress.
4
5 function [B] = cov_linReg(Zpc,l_matrix,x)
6
7 [dimRow,dimCol,~]=size(Zpc);
8
9 B.Par=zeros(dimRow,dimCol,2);
10 B.CovPar=cell(dimRow,dimCol);
11 B.Res=zeros(size(Zpc));
12 B.Corrected=zeros(size(Zpc));
13 B.Rsq = zeros(dimRow,dimCol);
14 B.rmse=zeros(dimRow,dimCol);
15 B.RobustnessPar=zeros(dimRow,dimCol,2);
16 B.se=zeros(dimRow,dimCol,2);
17
18 for w=1:dimRow
19 for k=1:dimCol
20 y=crop_corr_domain(Zpc,w,k,l_matrix(w,k));
21 [nR,nC,~]=size(y);
22 yt=reshape(permute(y,[2,1,3]),1,nR*nC,[]);
23 % Arranged in [observation-by-variables].
24 % the output is a column vector w by k.
25 Y=permute(yt,[3,2,1]);
26 [n,d]=size(Y);
27 X = cell(n,1);
28 for ii = 1:n
29 X{ii} = [eye(d) x(ii)*eye(d)];
30 end
31 try
32 chol(cov(Y));
33 catch ME
34
35 if strcmp(ME.message,'Matrix must be positive definite.')

```

cov_linReg 1

```

36 % V=diag(diag(cov(Y)));
37 [beta,~,E,SE] = mvregress(X,
Y, 'algorithm', 'cqls', 'covtype', 'full', 'varformat', 'beta');
38
39 else
40 % V=cov(Y);
41 [beta,~,E,SE] = mvregress(X,Y, 'algorithm', 'cqls', 'covar0', cov
(Y), 'covtype', 'full', 'varformat', 'beta');
42
43 end
44 end
45 B.Par(w,k,1)=beta(w*k); % Intercept
46 B.Par(w,k,2)=beta(w*k+d); % Slope
47
B.CovPar{w,k}=[SE(w*k,w*k),SE(w*k,w*k+d);SE(w*k,w*k+d),SE(w*k+d,w*k+d
)];
48 B.Res(w,k,:)=E(:,w*k);
49 B.Corrected(w,k,:)=mean(Y(:,w*k))+E(:,w*k);
50 B.Rsq(w,k)=1-var(E(:,w*k))/var(Y(:,w*k));
51 B.rmse(w,k)=std(E(:,w*k));
52 B.chitest=chi2gof(E(:,w*k));
53 U=tinv(0.975,n-2).*reshape(sqrt(diag(B.CovPar{w,k})),1,2);
54 B.RobustnessPar(w,k,1)=(abs(B.Par(w,k,1))./U(1))>1;
55 B.RobustnessPar(w,k,2)=(abs(B.Par(w,k,2))./U(2))>1;
56 B.se(w,k,1)=B.CovPar{w,k}(1,1);
57 B.se(w,k,2)=B.CovPar{w,k}(2,2);
58 end
59 end
60 end
61
62 function [A]=crop_corr_domain(Zpc,i,j,l)
63 [dimRow,dimCol,~]=size(Zpc);
64 if i <= l+1
65 % for j=1:dimCol
66 if j <= l+1
67 A = Zpc(1:(i+1),j:(j+1),:);
68 elseif j>= dimCol-1
69 A = Zpc(1:(i+1),(j-1):dimCol,:);
70 else
71 A = Zpc(1:(i+1),(j-1):(j+1),:);
72 end
73 % end
74 elseif i >= dimRow-1
75 % for j=1:dimCol
76 if j <= l+1
77 A = Zpc((i-1):dimRow,j:(j+1),:);
78 elseif j>= dimCol-1
79 A = Zpc((i-1):dimRow,(j-1):dimCol,:);
80 else
81 A = Zpc((i-1):dimRow,(j-1):(j+1),:);
82 end
83 % end

```

cov_linReg 2

```

84 else
85 % for j=1:dimCol
86 if j <= 1+1
87 A = Zpc((i-1):(i+1),j:(j+1),:);
88 elseif j>= dimCol-1
89 A = Zpc((i-1):(i+1),(j-1):dimCol,:);
90 else
91 A = Zpc((i-1):(i+1),(j-1):(j+1),:);
92 end
93 % end
94 end
95 end

```

cov_linReg 3

```

1 %% Significance Test
2
3 function [hr] = significance_test(residuals,RobustnessPar,t,reply)
4
5 if ~isempty(reply) && (reply=='N' || reply=='n')
6 disp('The Normality Test for the Residuals operation was canceled
by the user. ');
7 hr = [];
8 else
9 hr = chi_squared_test(residuals,t);
10 end
11
12 [dimRow,dimCol,~]=size(residuals);
13
14 s0 = sum(RobustnessPar(:, :, 1), 'all')/(dimRow*dimCol)*100;
15 format shortg
16 s0 = round(s0,3, 'significant');
17 s1 = sum(RobustnessPar(:, :, 2), 'all')/(dimRow*dimCol)*100;
18 format shortg
19 s1 = round(s1,3, 'significant');
20 figure('Name','Significance Test','NumberTitle','off');
21 title({'Coefficient p0', ['Percentage of pixels where the
Significance Test succeeded:', num2str(s0), '%']}, ...
22 'FontSize',14, 'FontName', 'Times New Roman')
23
24 surface(RobustnessPar(:, :, 1), 'LineStyle', 'none', 'FaceColor', 'flat'),
colorbar
25 axis square, axis equal, axis xy, axis([0 dimCol 0 dimRow])
26 figure('Name','Significance Test','NumberTitle','off');
27 title({'Coefficient p1', ['Percentage of pixels where the
Significance Test succeeded:', num2str(s1), '%']}, ...
28 'FontSize',14, 'FontName', 'Times New Roman')
29 axis square, axis equal, axis xy, axis([0 dimCol 0 dimRow])

```

significance_test 1

```

1 %% Uncertainty evaluation
2
3 function [q,U] = uncertainty_evaluation(Regression)
4
5 Z=Regression.Corrected;
6
7 [~,~,dimPage]=size(Z);
8 time = (-(dimPage/2):(dimPage/2)-1)';
9 time = permute(time,[3,2,1]);
10 u_p0 = sqrt(Regression.se(:, :,1));
11 u_p1 = sqrt(Regression.se(:, :,2));
12
13 % Combined standard uncertainty
14 uc = sqrt((time.*u_p1).^2+(u_p0).^2+(Regression.rmse).^2));
15 % Coverage factor (level of confidence ---> 95 %)
16 k = 2;
17 % Expanded uncertainty
18 U = k.*uc;
19
20 q = mean(Z,3);

```

uncertainty_evaluation 1

```

1 %% Uncertainty visualization
2
3 function [] = U_visualization(uncertainty)
4
5 [dimRow,dimCol,dimPage]=size(uncertainty.U);
6 X=(1:dimCol)'; Y=(1:dimRow)';
7
8 for q=1:dimPage
9 figure('Name',['Uncertainty band - Visualization - ',num2str
(q)],'NumberTitle','off');
10 s1=surf(X,Y,uncertainty.q,'FaceColor','g','FaceAlpha',
0.5,'EdgeColor','none');
11 hold on, s2=surf(X,Y,uncertainty.q-uncertainty.U(:, :,
q),'FaceColor','b','FaceAlpha',0.5,'EdgeColor','none');
12 hold on, s3=surf(X,Y,uncertainty.q+uncertainty.U(:, :,
q),'FaceColor','r','FaceAlpha',0.5,'EdgeColor','none');
13 title('Uncertainty band - Visualization','FontSize',14,'FontName',
'Times New Roman')
14 xlabel('Pixels in x-direction','FontSize',12,'FontName','Times New
Roman');
15 ylabel('Pixels in y-direction','FontSize',12,'FontName','Times New
Roman');
16 zlabel('z-direction [μm]','FontSize',12,'FontName','Times
NewRoman');
17 legend([s3 s1 s2], 'q+U', 'q', 'q-U', 'Location', 'northeast');
18 end

```

U_visualization 1

BIBLIOGRAPHY

- [1] Tosello G, Haitjema H, Leach R K, Quagliotti D, Gasparin S and Hansen H N 2016 An international comparison of surface texture parameters on polymer artefacts using optical instruments CIRP Ann. – Manuf. Technol. 65, pp. 529–532
- [2] ISO 25178-600:2019 Geometrical product specification (GPS)—Surface texture: Areal – Part 600: Nominal characteristics of non-contact (variable focus) instruments. ISO DIS 25178 part 600 (Geneva: International Organization for Standardization)
- [3] Evans C J and Bryan J B 1999 “Structured”, “Textured” or “Engineered” Surfaces CIRP Annals 48:2, pp. 541–556
- [4] Brinksmeier E, Karpuschewski B, Yan J and Schönemanna L 2020 Manufacturing of multiscale structured surfaces CIRP Annals – Manufacturing Technology 00, pp. 1–23
- [5] Cheung C F, Liu M, Leach R, Feng X and Zhao C 2018 Hierarchical-information-based characterization of multiscale structured surfaces CIRP Annals –Manufacturing Technology 67, pp. 539–542
- [6] Bruzzone A A G, Costa H L, Lonardo P M and Lucca D A 2008 Advances in engineered surfaces for functional performance CIRP Annals – Manufacturing Technology 57, pp. 750–769
- [7] Zhibin Y, Rui Z, Fei D, Teck N W, Kok C T, Kok F C, Poh K C and Yong S C 2011 Spray Cooling Two Phase Flow, Phase Change and Numerical Modeling, pp. 285-310
- [8] Liu J, Chen X, Lin Z and Diao S 2017 Multiobjective Optimization of Injection Molding Process Parameters for the Precision Manufacturing of Plastic Optical Lens Mathematical Problems in Engineering Volume 2017, Article ID 2834013

- [9] MacAulay G D and Giusca C L 2016 Assessment of uncertainty in structured surfaces using metrological characteristics CIRP Annals – Manufacturing Technology 65, pp. 533–536
- [10] Fortmeier I, Schulz M and Meeß R 2019 Traceability of form measurements of freeform surfaces: metrological reference surfaces Optical Engineering 58:9
- [11] Wang S and Jiang L 2007 Definition of Superhydrophobic States Advanced Materials 19:21, pp. 3423-3424
- [12] Courbin L 2016 Rethinking Superhydrophobicity
- [13] Kristensen A, Yang J K W, Bozhevolnyi S I, Link S, Nordlander P, Halas N J and Mortensen N A 2016 Plasmonic colour generation Nature Reviews: Materials 2:1
- [14] Yang Y, Panb Y and Guoa P 2017 Structural coloration of metallic surfaces with micro/nano-structures induced by elliptical vibration texturing Applied Surface Science 402, pp. 400–409
- [15] Whitehouse D 2011 Handbook of Surface and Nanometrology, Second Edition (CRC Press)
- [16] Leach R K, Giusca C L, Haitjema H, Evans C and Jiang X 2015 Calibration and verification of areal surface texture measuring instruments CIRP Annals – Manufacturing Technology 64, pp. 797–813
- [17] Syam W P, Leach R, Rybalcenko K, Gaio A and Crabtree J 2018 In-process measurement of the surface quality for a novel finishing process for polymer additive manufacturing Procedia Cirp 75, pp. 108–113
- [18] Townsend A, Senin N, Blunta L, Leachb R K and Taylord J S 2016 Surface texture metrology for metal additive manufacturing: a review Precision Engineering 46, pp. 34 – 47
- [19] Buyya R, Jin H and Cortes T 2002 Cluster computing Future Generation Computer Systems 18:3, pp. 5–8

- [20] Blaise Barney, Introduction to Parallel Computing
- [21] Zakoldaev D A, Shukalov A V and Zharinov I O 2019 Application of the BigData technology for processing manufacturing data at smart factories of the Industry 4.0 IOP Conf. Series: Materials Science and Engineering 560
- [22] <https://hpc.polito.it> (Last accessed: 29/09/2020)
- [23] <https://www.hpc.dtu.dk/> (Last accessed: 29/09/2020)
- [24] Mainsah E, Greenwood J A, Chetwynd D G 2001 Metrology and Properties of Engineering Surfaces (Springer-Science+Business Media, B.Y.)
- [25] Yao T F, Duenner A and Cullinan M 2017 In-line metrology of nanoscale features in semiconductor manufacturing systems Precision Engineering 47, pp. 147–157
- [26] ISO 14406:2011 Geometrical product specification (GPS) – Extraction. ISO 14406 (Geneva: International Organization for Standardization)
- [27] ISO 25178-2:2012 Geometrical product specification (GPS) - Surface texture: Areal – Part 2: Terms, definitions and surface texture parameters. ISO 25178 part 2 (Geneva: International Organization for Standardization)
- [28] Leach R 2014 Fundamental Principles of Engineering Nanometrology, Second Edition (Elsevier)
- [29] ISO 16610-60:2015 Geometrical product specification (GPS) – Filtration – Part 60: Linear areal filters – Basic concepts. ISO 16610 part 60 (Geneva: International Organization for Standardization)
- [30] ISO 4287:2009 Geometrical product specification (GPS) - Surface texture: Profile method - Terms, definitions and surface texture parameters
- [31] ISO 3274:1996 Geometrical product specification (GPS) - Surface texture: Profile method – Nominal characteristics of contact (stylus) instruments

- [32] ISO 4288:2000 Geometrical product specifications (GPS) - Surface texture: Profile method - Rules and procedures for the assessment of surface texture
- [33] Leach R K 2011 Optical Measurement of Surface Topography (Springer)
- [34] Brown, L. and Blunt, Liam (2008) Surface Metrology for the Automotive Industry Inaugural Automotive Researchers' Conference
- [35] Leach R 2013 Characterisation of Areal Surface Texture (Springer)
- [36] ISO 25178-6:2010 Geometrical product specifications (GPS) – Surface texture: Areal – Part 6: Classification of methods for measuring surface texture
- [37] JCGM 200:2012 International vocabulary of metrology—Basic and general concepts and associated terms (VIM), Joint Committee for Guides in Metrology (Sèvres: Bureau International des Poids et Mesures, BIPM).
- [38] ISO 25178-70:2014 Geometrical product specification (GPS)—Surface texture: Areal – Part 70: Material measures. ISO 25178 part 70 (Geneva: International Organization for Standardization)
- [39] ISO 15530-3:2011 Geometrical product specification (GPS)—Coordinate measuring machines (CMM): Technique for determining the uncertainty of measurement - Part 3: Use of calibrated workpieces or measurement (Geneva: International Organization for Standardization)
- [40] Petzing J, Coupland J and Leach R 2010 Measurement Good Practice Guide No. 116: The Measurement of Rough Surface Topography using Coherence Scanning Interferometry
- [41] Lee B S and Strand T C 1990 Profilometry with a coherence scanning microscope Applied Optics 29:26, pp. 3784-3788
- [42] Leach R K, Brown L, Jiang X, Blunt R, Conroy M, Mauger D 2008 Measurement Good Practice Guide No. 108: Guide to the Measurement of Smooth Surface Topography using Coherence Scanning Interferometry

- [43] Peter J de Groot 2019 A review of selected topics in interferometric optical metrology Reports on Progress in Physics 82
- [44] ISO 25178-604:2013 Geometrical product specification (GPS)—Surface texture: Areal – Part 604: Nominal characteristics of non-contact (coherence scanning interferometry) instruments. ISO 25178 part 604 (Geneva: International Organization for Standardization)
- [45] Giusca C L, Leach R K 2012 Measurement Good Practice Guide No. 128: Calibration of the metrological characteristics of imaging confocal microscopes (ICMs) (NPL)
- [46] ISO 25178-607:2018 Geometrical product specification (GPS)—Surface texture: Areal – Part 607: Nominal characteristics of non-contact (confocal microscopy) instruments. ISO 25178 part 607 (Geneva: International Organization for Standardization)
- [47] JCGM 100:2008 Evaluation of measurement data—Guide to the expression of uncertainty in measurement, Joint Committee for Guides in Metrology (Sèvres: Bureau International des Poids et Mesures, BIPM)
- [48] Forbes A B 1991 Least-squares best-fit geometric elements NPL Report DITC 140/89, National Physical Laboratory
- [49] ISO 16610-1:2015 Geometrical product specification (GPS) – Filtration - Part 1: Overview and basic concepts. ISO 16610 part 1 (Geneva: International Organization for Standardization)
- [50] ISO 16269-4:2010 Statistical interpretation of data - Part 4: Detection and treatment of outliers. ISO 16269 part 4 (Geneva: International Organization for Standardization)
- [51] Wang C, Caja J, Gómez E 2018 Comparison of methods for outlier identification in surface characterization Measurement 117, pp. 312-325

- [52] Giusca C L, Leach R K, Helary F, Gutauskas T and Nimishakavi L 2012 Calibration of the scales of areal surface topography-measuring instruments: part 1. Measurement noise and residual flatness Measurement science and technology 23, 10 pp.
- [53] Lou S, Tang D, Zeng W, Zhang T, Gao F, Muhamedsalih H, Jiang X and Scott P J 2020 Application of Clustering Filter for Noise and Outlier Suppression in Optical Measurement of Structured Surfaces IEEE Transactions on instrumentation and measurement 69:9
- [54] <https://dictionary.cambridge.org> (Last accessed: 29/09/2020)
- [55] Barbato G, Germak A, Genta G 2013 Measurements for Decision Making (Società Editrice Esculapio)
- [56] Barbato G, Barini E M, Genta G, Levi R 2011: Features and performance of some outlier detection methods Journal of Applied Statistics 38:10, pp. 2133-2149
- [57] Haitjema H 2015 Uncertainty in measurement of surface topography Surface Topography: Metrology and Properties 3
- [58] ISO 14253-2:2011 Geometrical product specifications (GPS) - Inspection by measurement of workpieces and measuring equipment - Part 2: Guidance for the estimation of uncertainty in GPS measurement, in calibration of measuring equipment and in product verification