

# **Inline quality control system for a reliable Additive Manufacturing layer-by-layer inspection**

Teză destinată obținerii  
titlului științific de doctor inginer  
la  
Universitatea Politehnica Timișoara  
în domeniul Inginerie Mecanică  
de către

**Dipl.ing. Bogdana Simina Fulga**

Conducător științific: prof.univ.dr.ing. Arjana Davidescu  
Referenți științifici: prof.dr.ing. dr.h.c.mult. Alexander Verl  
prof.dr.ing. dr.h.c. Alin Olimpiu Albu-Schäffer  
prof.univ.dr.ing. Ivan Bogdanov

Ziua susținerii tezei: 15 decembrie 2016

Seriile Teze de doctorat ale UPT sunt:

- |   |  |
|---|--|
| 1. Automatică                               | 9. Inginerie Mecanică                      |
| 2. Chimie                                   | 10. Știința Calculatoarelor                |
| 3. Energetică                               | 11. Știința și Ingineria Materialelor      |
| 4. Ingineria Chimică                        | 12. Ingineria sistemelor                   |
| 5. Inginerie Civilă                         | 13. Inginerie energetică                   |
| 6. Inginerie Electrică                      | 14. Calculatoare și tehnologia informației |
| 7. Inginerie Electronică și Telecomunicații | 15. Ingineria materialelor                 |
| 8. Inginerie Industrială                    | 16. Inginerie și Management                |

Universitatea Politehnica Timișoara a inițiat seriile de mai sus în scopul diseminării expertizei, cunoștințelor și rezultatelor cercetărilor întreprinse în cadrul Școlii doctorale a universității. Seriile conțin, potrivit H.B.Ex.S Nr. 14 / 14.07.2006, tezele de doctorat susținute în universitate începând cu 1 octombrie 2006.

Copyright © Editura Politehnica – Timișoara, 2016

Această publicație este supusă prevederilor legii dreptului de autor. Multiplicarea acestei publicații, în mod integral sau în parte, traducerea, tipărirea, reutilizarea ilustrațiilor, expunerea, radiodifuzarea, reproducerea pe microfilme sau în orice altă formă este permisă numai cu respectarea prevederilor Legii române a dreptului de autor în vigoare și permisiunea pentru utilizare obținută în scris din partea Universității Politehnica Timișoara. Toate încălcările acestor drepturi vor fi penalizate potrivit Legii române a drepturilor de autor.

România, 300159 Timișoara, Bd. Republicii 9,  
Tel./fax 0256 403823  
e-mail: editura@edipol.upt.ro

## Acknowledgements

This Doctoral Thesis is the result of my research work at the Mechatronics Department of the Polytechnic University of Timisoara as well as at the Machine Vision and Signal Processing Department of Fraunhofer Institute for Manufacturing Engineering and Automation IPA.

First and foremost I would like to express my special appreciation and thanks to my doctoral supervisor, Professor Arjana Davidescu, for her support, guidance and help throughout my Ph.D related research and writing process of this thesis. I am very grateful and thankful to Professor Alexander Verl who has guided my research work at Fraunhofer IPA.

I would also like to thank Professor Alin Olimpiu Albu-Schäffer and Professor Ivan Bogdanov who have made me the honor of accepting to be part of the dissertation defence committee. Many thanks to the doctoral advisory committee, to Professor George Savii, to Associate Professor Carmen Sticlaru and to Senior Lecturer Rodica Militaru, respectively for their complete and competent guidance throughout all these years.

I appreciatively acknowledge the support of my colleagues from the Machine Vision and Signal Processing Department of Fraunhofer IPA under the supervision of Mr. Markus Hüttel.

A special thought of gratitude goes to Mrs. Ira Effenberger who supported me during each stage of my research work and has always been close to me with both technical and personal advice.

I highly appreciate the way Mr. Ralf Becker incentivized me to undertake research and work perseverently in this field.

From the deep of my heart I thank my mother Monica Fulga, who has always been close to me watching every step of my way with infinite love and dedication. I thank her for the knowledge she has shared with me and for having taught me to love and respect work and people.

Last but not least, I would like to say how deeply grateful I feel to my loving, supportive and encouraging fiancé Michael Beising, not only for proofreading my thesis, but also for his valuable advices and his patience which play a highly important role in the process of completing my doctoral thesis. Thank you.

Timișoara, December 2016

Bogdana Simina Fulga

To my high school physics teacher Ioan Dumitru Valeriu A.  
To my grandmother.  
To you as a reader.

Fulga, Bogdana Simina

**Inline quality control system for a reliable Additive Manufacturing *layer-by-layer inspection***

Teze de doctorat ale UPT, Seria X, Nr. YY, Editura Politehnica, 2016, 159 pagini, 110 figuri, 14 tabele.

Keywords: additive manufacturing, inline quality control, product quality, selective laser sintering

Summary

Additive Manufacturing (AM) is one of the genuine hopes for the forth industrial revolution since digital data is controlling the whole layered production process. However, to be the driving force behind a new industrial revolution the quality, reliability and reproducibility of the AM parts must be 100% assured.

In the actual context of the state of the art in the field of Quality Control (QC) for AM Powder Bed Fusion (PBF) processes, taking into consideration the deficiencies of the actual approaches, first an inline QC system's tasks identification approach was developed. Based on it, a comprehensive concept for an inline QC system was elaborated. Furthermore, the development and implementation of the system components were carried through.

The results of the conducted research work, closing a gap in the state of the art, culminate into a novel and innovative inline QC control system, based on Industrial Image Processing, having a high degree of applicability in the AM PBF processes.

## CONTENTS

<b>TABLE LIST .....</b>	<b>8</b>
<b>FIGURE LIST .....</b>	<b>9</b>
<b>TERMS LIST .....</b>	<b>13</b>
<b>1. INTRODUCTION .....</b>	<b>15</b>
1.1. Problem statement .....	15
1.2. Thesis overview.....	16
<b>2. BACKGROUND .....</b>	<b>19</b>
2.1. Additive Manufacturing .....	19
2.1.1. History.....	20
2.1.2. The need for additive production .....	20
2.1.3. Process chain.....	23
2.1.4. Process categories and techniques .....	24
2.1.5. Selective Laser Sintering: from one-off production to batch production.....	36
2.2. Industrial Image Processing .....	37
2.2.1. Industrial Image Processing for inline Quality Control .....	39
2.2.2. Inline IIP enabler of SLS batch production.....	41
<b>3. STATE OF THE ART REVIEW ON QUALITY ASSURANCE OF AM PBF PROCESSES .....</b>	<b>45</b>
3.1. Quality assurance chain.....	45
3.2. Quality assurance by means of industrial computer tomography .....	46
3.3. Science and technology needs in metrology .....	47
3.4. Control schemes.....	48
3.5. In-process measurements .....	49
3.6. Conclusions .....	50
<b>4. TASKS FOR INLINE QC AND IN-SITU OPTIMISATION OF AM PBF PROCESSES .....</b>	<b>51</b>
4.1. Overview .....	51
4.2. Quality control in AM PBF processes.....	52
4.3. Identification of inline defects and failures .....	53
4.3.1. Equipment .....	54
4.3.2. Feedstock .....	56
4.3.3. Production.....	56

4.3.4. Part.....	58
4.3.5. Results .....	58
4.4. Tasks prioritisation .....	60
<b>5. OVERALL CONCEPT FOR AN INLINE QC SYSTEM IN THE AM PBF PRODUCTION .....</b>	<b>63</b>
5.1. Hardware platform.....	64
5.1.1. Appropriate technologies and sensors for data and signal acquisition ..	64
5.1.2. Hardware architecture design: concept and interfaces .....	67
5.2. Software platform.....	69
5.2.1. Concept .....	69
5.2.2. Modular Software Framework .....	70
<b>6. INLINE QC SYSTEM COMPONENTS .....</b>	<b>75</b>
6.1. Experimental hardware platform.....	75
6.1.1. Technologies and sensors for data acquisition .....	76
6.1.2. Hardware architecture .....	77
6.1.3. Hardware setup .....	78
6.2. Software engine .....	79
6.2.1. Advanced IIP routine for data acquisition during the AM PBF manufacturing processes.....	80
6.2.2. Advanced IIP routine and dedicated algorithms for data evaluation during the AM PBF manufacturing processes .....	81
6.2.3. Quality protocol .....	103
6.3. Graphical user interface.....	103
<b>7. RESULTS .....</b>	<b>107</b>
7.1. Inline QC system for a reliable AM .....	107
7.1.1. Advanced IIP SW routine for the inline layer-by-layer inspection .....	107
7.1.2. Hardware and software integration .....	109
7.2. Verification of the results.....	110
7.2.1. CT as reference technology for overall results: i.O. vs. n.i.O.....	111
7.2.2. Accuracy of the inline QC system .....	114
<b>8. CONCLUSIONS, DISTINCT CONTRIBUTIONS AND FUTURE RESEARCH WORK .....</b>	<b>119</b>
8.1. Conclusions .....	119
8.2. Distinct contributions and future research work .....	120
<b>ANNEXES .....</b>	<b>123</b>
Annex 1: Hardware components .....	123

Annex 2: Software - Framework .....	131
Annex 3: Software - IIP .....	139
<b>REFERENCES .....</b>	<b>155</b>

## TABLE LIST

Table 2.1. Evaluation of AM techniques [5].	37
Table 2.2. Classification of digital image enhancement algorithms in common use.	39
Table 3.1. Exemplary - 3D-evaluation results for the quality criteria of AM PBF [32].	46
Table 4.1. Equipment – identified inline defects and failures, using the example of SLS.	54
Table 4.2. Feedstock – identified inline defects and failures, using the example of SLS.	56
Table 4.3. Production – identified inline defects and failures, using the example of SLS.	57
Table 4.4. Part – identified future tasks for an inline QC system.	58
Table 4.5. Overall identified SLS inline defects and failures.	59
Table 4.6. Ranking of the top inline defects and failures to be identified by the QC system.	61
Table 5.1. Ranked tasks during the AM PBF production steps: basis for a firm foundation of a hardware and software platform for an inline QC system and a future in-situ optimisation.	63
Table 5.2. Technologies and sensors necessary for solving the overall tasks – exemplary.	64
Table 5.3. Inline Quality Control System Modules - exemplary for a SLS machine.	65
Table 6.1. IQCSM 2 overall tasks.	76
Table 6.2. Experimental HW setup and components.	78



## FIGURE LIST

Figure 2.1. Examples to AM.....	19
Figure 2.2. François Willème experiments. ....	20
Figure 2.3. Individualised products with complex geometries and integrated functionalities require an additive production.....	21
Figure 2.4. 2-Finger angular grippers from Schunk (left) and IPA (right).....	21
Figure 2.5. Additively manufactured bench.....	22
Figure 2.6. Simulation and CAD model of a jawbone. ....	22
Figure 2.7. Additively manufactured prosthesis.....	22
Figure 2.8. AM - process steps.....	23
Figure 2.9. STL example. ....	24
Figure 2.10. AM techniques. ....	25
Figure 2.11. Schematic diagram of Stereolithography. ....	25
Figure 2.12. Exemplary - SLA machine and products. ....	26
Figure 2.13. Schematic diagram of Poly-Jet Modelling.....	27
Figure 2.14. Exemplary - PJM machine and products. ....	28
Figure 2.15. Schematic diagram of Digital Light Processing. ....	28
Figure 2.16. Exemplary - DLP machine and products. ....	28
Figure 2.17. Schematic diagram of Layer Laminated Manufacturing using a knife as the cutting tool.....	29
Figure 2.18. Schematic diagram of Layer Laminated Manufacturing using a laser as the cutting tool. ....	29
Figure 2.19. Exemplary - LLM machine and products. ....	30
Figure 2.20. Schematic diagram of Fused Layer Modelling.....	31
Figure 2.21. Exemplary - FLM machine and products.....	31
Figure 2.22. Schematic diagram of 3D Printing. ....	32
Figure 2.23. Exemplary - 3DP machine and products. ....	32
Figure 2.24. Schematic diagram of SLS and SLM. ....	33
Figure 2.25. Exemplary - SLS machine and SLS and SLM products. ....	34
Figure 2.26. Selective Heat Sintering.....	35
Figure 2.27. Exemplary - SHS machine and products. ....	36
Figure 2.28. A typical setup for MAP. ....	36
Figure 2.29. Structure of a Machine Vision system. ....	38
Figure 2.30. IIP trends and goals. ....	38
Figure 2.31. IIP time for different equipment (including the process time). ....	39
Figure 2.32. Image enhancement methods: point transforms. ....	40
Figure 2.33. Image enhancement methods: neighbourhood operations.....	40
Figure 2.34. Printed peptide layer image before (left) and after a 2D IIP based system calibration (right).....	41
Figure 2.35. Inline 2D IIP: measure operations; presence and completeness inspection and self-adaptive defect detection of surfaces. ....	42
Figure 2.36. SLS layer: low scene contrast ratio. ....	43
Figure 3.1. Optical monitoring set-up into the in-house build LM-Q machine of KUL. ....	48
Figure 4.1. AM PBF production chain - influencing parameters. ....	51
Figure 4.2. 2-Finger angular gripper.....	52

Figure 4.3. Bionic Handling Assistant.....	52
Figure 4.4. Schematic diagram of the approach for the identification of inline defects and failures during AM PBF processes.....	53
Figure 4.5. Laser window viewed from building platform side (left), laser window zoom (right).....	54
Figure 4.6. View of the powder layer after application - serious defect in the powder layer.....	54
Figure 4.7. "Black sintered" object.....	57
Figure 5.1. Overall hardware concept of the modular inline QC system for AM PBF.....	68
Figure 5.2. AM-IQCP: general SW architecture.....	69
Figure 5.3. AM-IQCP: SW architecture flow diagram.....	70
Figure 5.4. Overview UML diagrams of the framework classes.....	71
Figure 5.5. Upper part of the UML diagrams of the framework classes.....	71
Figure 5.6. Lower part of the UML diagrams of the framework classes.....	72
Figure 5.7. A Parameter object created from a ParameterBuilder, by using the builder pattern.....	72
Figure 5.8. Detailed view of the DataCommand and FilterCommand class hierarchy.....	73
Figure 6.1. NIST test artefact.....	75
Figure 6.2. Hardware architecture and interfaces of the modular inline QC system for SLS with one IQCSM, the IQCSM2.....	77
Figure 6.3. Implemented Software Engine.....	79
Figure 6.4. Exemplarily material supplier movement (left-to-right) – first and last image of this sequence is automatically acquired for the inline quality control.....	80
Figure 6.5. Software routine <i>IIP for Inline Image Acquisition</i> .....	80
Figure 6.6. ROI's with blob analysis: triggers for image acquisition.....	81
Figure 6.7. General concept for the <i>IIP for Quality Control</i> routine.....	82
Figure 6.8. Low contrast images with subtle failures.....	83
Figure 6.9. Subroutine - from image calibration to layer type decision.....	84
Figure 6.10. LC image (left) and an I4QC (right).....	85
Figure 6.11. Square corners in chessboard, used for a part of the building platform, where the objects are manufactured.....	85
Figure 6.12. Undistorted area: I4QC in IM1 (left a sintered layer and right a powder layer).....	86
Figure 6.13. ROI corresponding to the inspected area (left sintered layer, right powder layer).....	86
Figure 6.14. ROI - Subtraction results in IM 3 (left sintered layer, right powder layer).....	87
Figure 6.15. IM 6 – ROI after the edge sharpening and local binarisation operation (left sintered layer, right powder layer).....	87
Figure 6.16. IM8 - ROI after median and dilatation subroutines steps (left sintered layer, right powder layer).....	88
Figure 6.17. Exemplary: powder layer with serious, wide failure.....	88
Figure 6.18. Subroutine - quality control for powder layers.....	89
Figure 6.19. From left to right results of subroutine steps II (IM9) and III (IM10).....	90
Figure 6.20. Displayed results in IM20 - no wide failures found.....	90
Figure 6.21. Displayed results in IM20 - subtle failures found.....	90
Figure 6.22. Exemplary: sintered layer with major failure.....	91
Figure 6.23. Subroutine - quality control for sintered layers.....	92
Figure 6.24. IM 15 (left) and IM4 (right) after the dilatation filters.....	93
Figure 6.25. Displayed results in IM20: major failures in sintered layer found.....	94

Figure 6.26. Slide on check: 13 (build job: NIST TestArtifact). STL slide nominal inner features (left), actual inner features (right) -> n.i.O. ....	94
Figure 6.27. Exemplarily histogram analysis of one sintered slide in IM3, the image after the lighting correction. ....	96
Figure 6.28. Subroutine - in-process measure procedure for sintered layers. ....	97
Figure 6.29. Annulus probe for one circular feature of the sintered layer (left). Zoom: points found and saved in <i>PL</i> (right). ....	99
Figure 6.30. IM20 - two measured circular features through subroutine steps II to IX. ....	100
Figure 6.31. Rectangular probe for one sintered layer: edge feature (left), probe zoom (right). ....	100
Figure 6.32. IM20 - two measured circular features and the two lines <i>L1</i> and <i>L2</i> determined using subroutine steps X to XIII. ....	101
Figure 6.33. IM20 - two measured circular features, the two lines <i>L1</i> and <i>L2</i> and the results from subroutine steps from XIV to XVII -> n.i.O. sintered layer. ....	102
Figure 6.34. Subroutines for results. ....	102
Figure 6.35. Defined <i>PartQualityProtocol</i> function symbol, used in the subroutines diagrams. ....	103
Figure 6.36. Extract from a part quality protocol. ....	103
Figure 6.37. GUI layout: <i>Inspection Program Setup</i> - setting the area criterion for the blob extraction and the grey value interval. ....	104
Figure 6.38. GUI layout: <i>Inspection Program Setup</i> - saving parametrised inspection programs. ....	104
Figure 6.39. GUI layout: <i>Inline Inspection</i> . ....	105
Figure 7.1. Overall advanced IIP SW routine of the inline QC for a reliable Additive Manufacturing. ....	108
Figure 7.2. Hardware and software integration in final system: conceptual level. ....	109
Figure 7.3. Hardware and software integration in final system: implemented level. ....	110
Figure 7.4. System HW with integrated fan. ....	110
Figure 7.5. STL model of the test artefact used as input data for the SLS machine EOS Formiga P100. ....	111
Figure 7.6. CT acquired 3D-Data for inline QC system classified parts: i.O.(top) and n.i.O.(down). ....	111
Figure 7.7. CT acquired data of a SLS build artefact part: CT slice 6. ....	112
Figure 7.8. Artefact build part, classified by the inline QC system as i.O., at layer 8. ....	112
Figure 7.9. SLS build artefact, in CT data projection YZ. ....	113
Figure 7.10. SLS build artefact, classified by the inline QC system as n.i.O., at layer 14. ....	113
Figure 7.11. Powder layer image: zoom of the one failure region (down). ....	114
Figure 7.12. Powder layer image: detected failures areas. ....	115
Figure 7.13. Powder layer image: i.O. region with detailed pixel values. ....	115
Figure 7.14. Powder layer image: one n.i.O. region with detailed pixel values. ...	116
Figure 7.15. Powder layer image: another n.i.O. region with detailed pixel values. ....	116
Figure 7.16. Sintered layer image: measure results. ....	117
Figure 7.17. Sintered layer image: fitted line. ....	117
Figure 7.18. Sintered layer image: fitted circle. ....	118

Figure 8.1. Inline QC system for the AM PBF processes: forming the main distinct contribution to the knowledge of the QC in the field of tomorrow's additive production. .... 121

## TERMS LIST

The terminology used throughout this thesis follows the ISO/ASTM 52900 international standard terminology [1].

Additive Manufacturing (AM)	"process of joining materials to make parts from 3D model data, usually layer upon layer, as opposed to subtractive manufacturing and formative manufacturing methodologies" [1]
Build cycle	"single process cycle in which one or more components are built up in layers in the process chamber of the AM system" [1]
Production run	"all parts produced in one build cycle or sequential series of build cycles using the same feedstock batch, and process conditions" [1]
Feedstock	Raw material supplied to the AM processes
Single-step process	"type of AM process in which parts are fabricated in a single operation where the basic geometric shape and basic material properties of the intended product are achieved simultaneously" [1]
AM machine	"section of the AM system including hardware, machine control software, required set-up software and peripheral accessories necessary to complete a build cycle for producing parts" [1]
Powder Bed Fusion (PBF) process	"AM process in which thermal energy selectively fuses regions of a powder bed" [1]
PA 12	Polyamide 12
STL format	"file format for model data describing the surface geometry of an object as a tessellation of triangles used to communicate 3D geometries to AM machines in order to build physical parts" [1]
Scan speed	Sintering, melting speed



# 1. INTRODUCTION

## 1.1. Problem statement

Additive Manufacturing (AM) processes are enabler of new concepts and approaches for product design, production and even business modelling. Thus, new demands appear on the build cycles and the production run itself.

The need for additive production is constantly increasing, as the digital fabrication of additive manufactured parts is the only single-step process allowing simultaneously:

- *functional integration* - means implementing as many technical functions as possible into as few parts as possible;
- *complex geometries* - from free-form surfaces to even weight-based topologies;
- and *individualisation* - a strong personalization and adaptation to the customers' needs being possible at any time.

Based on the characteristics of the AM tool-free layered build process, customer-specific or even complex variants of mass products are possible. Nowadays not only plastic and metal components are additive manufactured, but also function-integrating solutions for fibre-reinforced composites. For future applications, multiple materials as well as functionally gradient materials will be used in one build process. Even application-specific cavity structures will be possible. However, can AM technologies really find their way into industrial environments? Can highly customized unique parts be additively produced as efficiently as conventional mass-produced parts? Can we additively manufacture batch size one products?

Up to now the major international AM machine builders deliver nowadays only the equipment. They do not support the needs of the users, mostly SMEs, as far as Quality Assurance (QA) and manufacturing integration is concerned. Only the aerospace industry has driven, with extremely great efforts, individual solutions for the manufacturing of metallic components without finally debating this topic. All other industries, service providers, and end users of the AM products are literally forced to deliver, respectively to use parts with unpredictable reliability, having no direct control of the quality and reproducibility of the parts.

Beginning to view AM in an industrial environment, reliable statements about product quality are indispensable. Statements regarding compliance with geometric tolerances and exact quantifiable physical parameters, in terms of product certification are therefore imperative. The sooner the identification, acquisition and in-process influence of these parameters became reality, the higher the efficiency of the AM processes; a product certification being then also possible. Therefore, the quality of the AM parts must not only be sustainably secured but also reproducible at any time.

The AM Powder Bed Fusion (PBF) [1] processes, are currently on the cusp of becoming established production processes. The prerequisite for this is a corresponding standardisation and integration of these processes within the production. In this context, not only the horizontal production integration is

essential but also increasing the quality and reproducibility of the parts is mandatory.

Selective Laser Sintering (SLS) and Selective Laser Melting (SLM) are the two most important AM PBF techniques, being the genuine hopes for the industrial contenders, ranging from automotive and even aerospace applications to the medical field. The quality of the SLS and SLM parts and their designed functionality can be very easily impaired, taking into consideration the following items:

- Nowadays the Quality Management (QM) and QA is not ensured.
- Neither Quality Standards nor a generally accepted QM Standard or System are available.
- A larger variety of parameters are influencing the manufacturing processes based on these techniques and so the quality of the build parts.

Regarding the QA chain for the SLS and SLM processes, with respect to inline Quality Control (QC) and in-process optimisation, the scientific needs are given by the fact that part quality problems, appearing during the production process and being caused by different factors, are leading to vulnerabilities, fractures, or product failures.

Nowadays *there is no inline system, which at the same time:*

- can detect and classify comprehensively such errors,
- monitors and in real-time optimises the AM PBF processes,
- offers a complete quality report of the produced part, and
- can abort the production process in extreme cases.

A reliable inline part and process monitoring as well as a real-time optimisation of the AM PBF processes are crucially, not only for the service providers in this field but also for all end users of the AM parts, so that the products meet the required quality and safety standards in the different fields of application.

## 1.2. Thesis overview

The objectives of the present thesis are embedded in the presented context. Its concrete aim is the realisation of an AM PBF machine-independent, modular inline QC system based on Industrial Image Processing (IIP), using the example of the SLS technique. The focus on especially this AM PBF process is driven by the fact that SLS is one of the main technologies that can be used to produce functional parts, even for low series production.

The following features and advantages over the current efforts are expected and are the focus of the development work: the inline QC system should

- be able to detect quality failures during production process,
- offer a quality report of the produced part,
- cancel the production process in extreme cases,
- require a small number of input parameters,
- not need users with technical expert knowledge,
- have a clear-cut attractive price/performance ratio,
- provide the possibility of being extended at any time to further sensors and/or algorithms,
- offer the basis for a future in-situ process optimisation.

Briefly, a user-friendly machine-independent and modular system using machine vision technology will arise in order to ensure and log, through an inline layer-by-layer inspection, the quality of the SLS produced parts. Hereto, first, in chapter 2 of this thesis the "*Background*" of the *Additive Manufacturing and*



*Industrial Image Processing* will be outlined. In chapter 3, a literature review of the *quality assurance of the AM PBF processes* will be carried out.

Chapter 4 will present the research work conducted in order to determine the requirements specification for an inline QC system, namely the *"Tasks for inline quality control and in-situ optimisation of AM PBF processes"*. These results are forming the firm foundation of an *"Overall concept for an inline QC system in the AM PBF production"*, which is detailed in chapter 5. Following the overall concept, the development and implementation of the two innovative system components, the hardware platform and the software engine, is presented in chapter 6, *"Inline QC system components"*.

In the first part of chapter 7, *"Results"*, the advanced IIP software routine for the inline layer-by-layer inspection and the overall hardware and software integration in the final inline QC system are detailed. In the second part, an evaluation of the novel system is issued. The thesis outcomes will be analysed placing the new results in the context of the state of the art presented in chapter 3.

Chapter 8, *"Conclusions, distinct contributions and further research work"*, contains a synthesis of the conclusions and the distinct contributions gained through this doctoral thesis.



## 2. BACKGROUND

### 2.1. Additive Manufacturing

AM is the "process of joining materials to make parts from 3D model data, usually layer upon layer, as opposed to subtractive manufacturing and formative manufacturing methodologies", as defined in the ISO/ASTM 52900-15 standard [1].

The importance of the AM methods and their impressive further development is nowadays a fact that cannot be ignored. The AM machines enable rapid, tool-free production of sophisticated, highly complex, filigree components, and systems that go beyond the limits of conventional production technologies (Figure 2.1). Additive processes offer important advantages over traditional methods. Compared to subtractive processes such as cutting, drilling or milling the additive processes add material, instead of removing it. This is much more resource-efficient. Moreover, very complex shapes or even assemblies with functional properties can be produced in one piece. This reduces tremendous the expensive assembly work.

Using the layered production, a company's competitiveness can be strengthened: the number of manufacturing steps is reduced and new business models are developed. The digital process chain, the flexibility, and scalability of production gave birth to new services and production concepts. The integration of these innovative manufacturing processes in industrial process chains can offer far-reaching opportunities and possibilities for the redesign and optimization of products and product groups.

The top five industries using AM, presented in the Wohlers Report 2015, are the industrial/business machines (17.5%), consumer products/electronics (16.6%), motor vehicles (16.1%), aerospace (14.8%), and the medical/dental one (13.1 %). The quality of the AM parts is one of the five identified challenges for adoption of AM technologies [2].

The AM industry, the market for AM, consisting of all services and products, grew at a Compound Annual Growth Rate (CAGR) of 35.2% to \$4.1 billion in 2014 [2] and at a CAGR of 25.9% to \$5.165 billion in 2015 [3]. In 2011, 31 manufactures sold industrial AM systems, compared to 62 companies that sold industrial grade AM systems in 2015 [3]. This is a stunning industry, evolving continuously, growing by an impressive \$1 billion for two years successively.

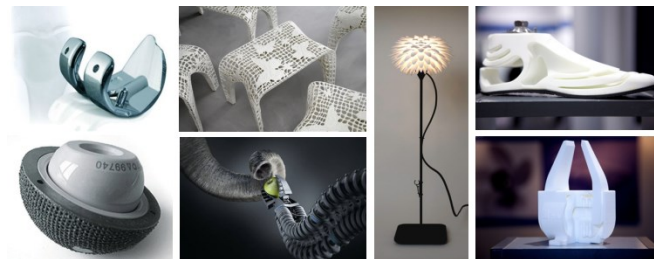


Figure 2.1. Examples to AM.  
[source: Fraunhofer IPA, Festo, Freedom of Creation, Arcam, EOS]

Taking into consideration the impressive development and growth of this industry over the years and the large field of applications based on this technology, an overview of the AM history, methods and techniques will be presented from the point of view of a manufacturing process for end-products.

### 2.1.1. History

In 1860 François Willème laid the foundation for AM technologies. He reproduced a sculpture of himself, as the sum of its profiles: the subject had to sit completely still within a circle of 24 cameras (see Figure 2.2). The resulting images have been used then to create sculptures. The photographs were projected onto a screen and then a pantograph used to match up the images to create in clay a 3D-like object. The process was called "photosculpture".

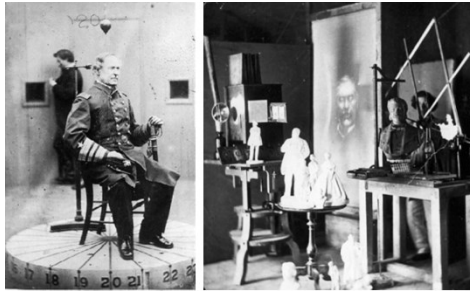


Figure 2.2. François Willème experiments.  
[4]

Three decades later, J. E. Blantner patented a layered method for producing 3-dimensional maps, the so-called "topographical relief maps". The method consisted of impressing topographical contour lines on a series of wax plates and cutting these wax plates along the lines.

By 1956, Otto John Munz had described in his US patent 2.775.758.1956. a system with impressive similarities to Stereolithography. The „Photo-Glyph Recording“-method was born. Much like Stereolithography, this method uses a photographic emulsion contained in a vat with an elevator platform.

The development of lasers by Theodore Maimann (1960), laid the foundation for the currently used Stereolithography process, and in a way also for the SLS. Therefore, Stereolithography is regarded as the first of the new additive techniques. The company 3D Systems, founded by Chuck Hull, introduced the first machine in 1987.

### 2.1.2. The need for additive production

Certain products' requirements are increasing presently the need for additive technologies: already an item, from the three ones in the Figure 2.3 (i.e. functional integration, complex geometries, and individualisation), may be enough to justify an additive production. In the design stage it should be already considered if a fulfilment of the other items is simultaneous possible, and if this could add additional value to the product. If all three criteria are fulfilled, other production processes and methods are virtually eliminated.

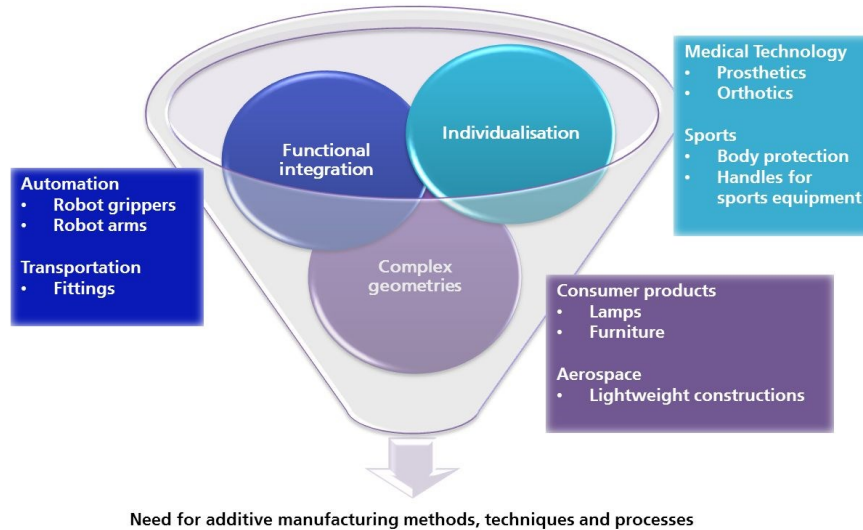


Figure 2.3. Individualised products with complex geometries and integrated functionalities require an additive production.

*Functional integration* means implementing as many technical functions as possible into as few parts as possible. AM offers a clear advantage for this requirement, often allowing the production of all required product parts in a single-step process, including functional components like e.g. hinged joints or even pneumatic actuators. In this way a large number of assembly steps, otherwise necessary, can be dispensed with. This saves money and minimises the likelihood of errors in production. With the newfound possibilities, simple components (e.g. springs or hinge pivots), as well as complex parts (e.g. pneumatic actuators) can be realised very quickly. The geometrical freedom and the high elasticity of the feedstock, for example PA 12, give the possibility to manufacture complicated geometries as leaf springs or coil springs.

One of the challenges in this field is to succeed providing the actual functionality by replacing multiple components with one single AM part. In Figure 2.4 the 2-finger angular gripper from Schunk GmbH & Co. KG is compared with the one-piece AM gripper from Fraunhofer IPA: here a bellows expands and the gripping movement is assured by the deflection realised via an integral hinge. The additively produced parts are extremely resistant: e.g., several thousand operations can be done without damages of the product and its functionality. Therefore, such PA 12 parts can be used nowadays in e.g. automation technology [5].

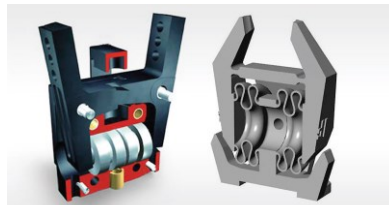


Figure 2.4. 2-Finger angular grippers from Schunk (left) and IPA (right).  
[source: Schunk GmbH & Co. KG, Fraunhofer IPA]

*Complex geometries*, understood in this context as three-dimensional structures, which by conventional manufacturing processes are very difficult to be produce; the undercuts and cavities of such structures are causing very often problems or at least very high production costs. Organic structures, such as the tree one shown in Figure 2.5, are problem-free build with AM. The main advantage of the layered process is that any shape generated in a 3D CAD program, is producible [5].

Free-form surfaces and weight-based topologies (see Figure 2.6), have become reality with the help of these manufacturing technologies and, are completing the widely spectrum of complex geometries [6] that can be additively build.

*Individualisation*: with the additive technologies a strong personalization and adaptation to the customer needs is possible, through [5]:

- An active personalisation: the customers are becoming indirectly producers, being responsible for the customized product design (e.g. jewellery), using directly the services in the layered manufacturing field.
- A passive personalisation: understood as a direct implementation of the AM technology for special customer requirements (e.g. prostheses and implants based on scan data as in Figure 2.7).



Figure 2.5. Additively manufactured bench.  
[7]

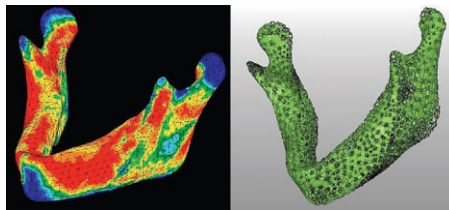


Figure 2.6. Simulation and CAD model of a jawbone.  
[source: Fraunhofer IFAM]



Figure 2.7. Additively manufactured prosthesis.  
[source: F.Gottinger Orthopädietechnik GmbH, Fraunhofer IPA]

### 2.1.3. Process chain

Despite last year's manifold developments and an increasing number of new technologies and applications, going from the laboratory phase into the industrialisation one, all-current additive technologies are based on the same principle of layering. The product development and production stages are summarised in Figure 2.8.

The starting point of the AM processes is the creation of the *3D-CAD model*, taking into account the characteristics of each technology. Once the design data is completed, in accordance with the design requirements, the data is saved in the specific AM machine data format.

The STereoLithography, also named Surface Tessellation Language format (STL), is the de facto industry standard, despite recent data formats such as "Additive Manufacturing File Format" (AMF) or the "3D Manufacturing Format" (3MF). STL is the interface between the CAD program and the AM machines. In this case, the surface geometry of an object is represented as a tessellation of triangles (see Figure 2.9). The resolution of an STL can be adjusted, by specifying the number of its triangular facets. A low resolution forms the model no longer correctly and therefore the accuracy of the object produced is reduced [5]. On the other hand, high resolution means a higher memory storage and usage, which makes the handling of the data sometimes difficult.

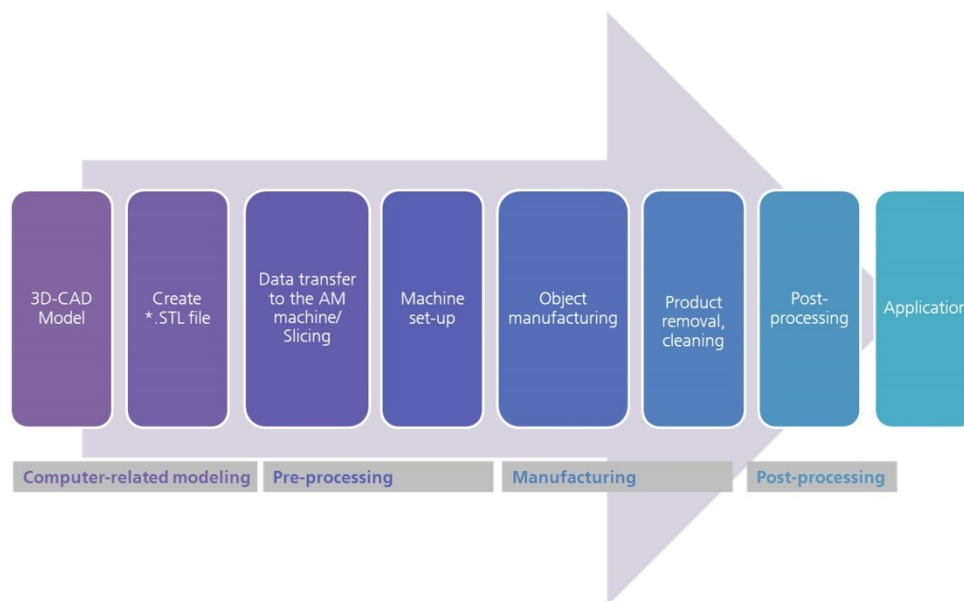


Figure 2.8. AM - process steps.

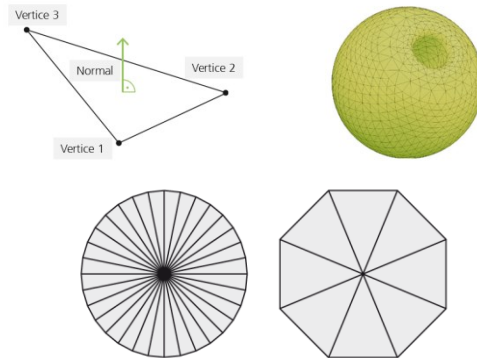


Figure 2.9. STL example.  
[5]

Once the STL data is loaded on a machine, the *pre-processing* stage can start: operations like fixing, repairing, enhancing and *slicing the data take place*. Each manufacturer has its own software already installed on the machine, which allows the user to make the mentioned operations. However, there are also companies that are providing directly these tools (e.g. Magics software platform from Materialise).

Depending on the machine type and the technology, various *machine set-up operations* are necessary: e.g. adjusting the corresponding parameter, selecting the right feedstock, preparing the build platform. The actual manufacturing of the part follows this step. Once executed, the excess material or support material must be removed. The nature and scope of the *post-processing* step are depending directly on the technical and quality requirements of the final part.

Looking at the production process as a whole, it is important to take into consideration the fact that each additive process has its own process steps (Figure 2.8) to be mandatory fulfilled.

#### 2.1.4. Process categories and techniques

In the standard ISO/ASTM 52900- 15 [1], the terms and definitions section, aside the AM PBF processes, other six process categories are mentioned: binder jetting, directed energy deposition, material extrusion, material jetting, sheet lamination, and vat photopolymerization.

Meanwhile there are wide ranges of different AM techniques. All of them are following the same layer-by-layer manufacturing principle, but they differ in feedstock type and in the technological principle of modelling. The VDI 3405 standard [8] and the work done by Breuninger et al. [5] have been considered as basis for a description of the techniques involved in commercially available systems, which is presented in the following. In Figure 2.10 an overview of the AM techniques is presented; the AM PBF ones being highlighted.



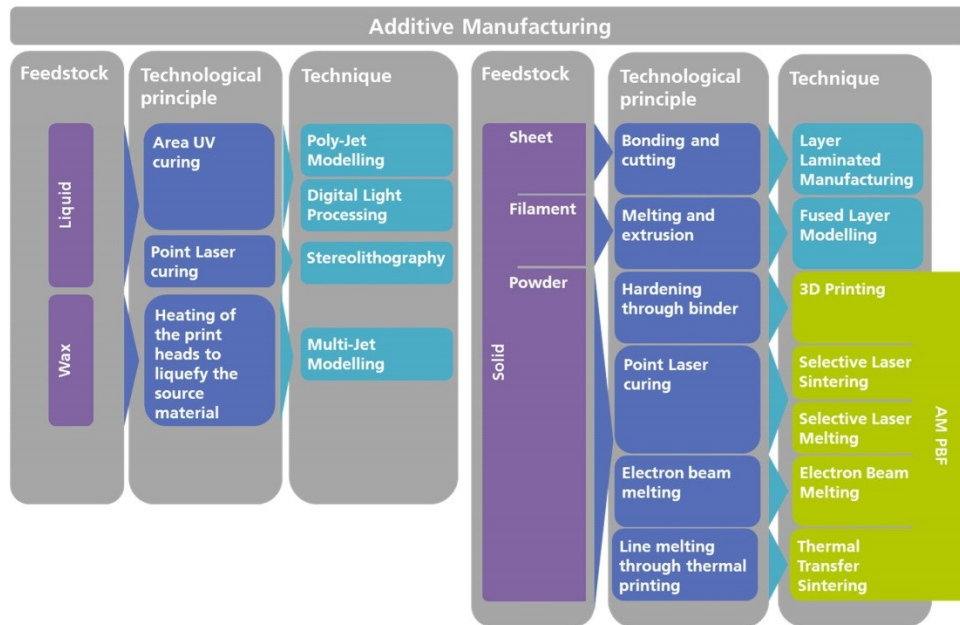


Figure 2.10. AM techniques.

**Stereolithography (SLA or SL)**

*Principle:* a photopolymer (photo-reactive resin) is polymerized, respectively cured in thin layers by a laser. This occurs in a liquid bath, where a gradually lowering construction platform is located. In order to hold the solidified model in the liquid, supporting structures are necessary. Post-processing steps as e.g. post-exposure in the UV chamber and removal of support structures are necessary.

A *schematic diagram* of this AM technique is shown in Figure 2.11 as well as a SLA machine and some exemplary products in Figure 2.12.

*Materials:* photosensitive polymers/resins.

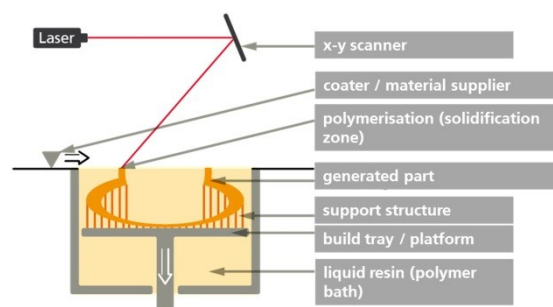


Figure 2.11. Schematic diagram of Stereolithography.



Figure 2.12. Exemplary - SLA machine and products.  
[9]

*Advantages:*

- According to the state of the art, SLA is the most accurate and precise AM technique. The accuracy, mainly determined by the diameter of the laser beam, can reach 0.025-0.05 mm per 25.4 mm of part dimension [10].
- A very good surface quality is available.
- Microstereolithography, a method used for the production of micro parts, is a derivation of the SLA method.

*Disadvantages:*

- Only the use of photosensitive materials is possible. This limits the number of available materials. In addition, these materials, at the present state of the art, have a limited life, mainly because of the harmful effect of the day light.
- Mechanical strength is currently worse than in relation to other AM methods.
- Thermal stability is not sufficient for product production.
- Cantilevered structures cannot be produced without supporting constructions, which implies a high workload.
- Also associated with an elaborate and extensive labour, is the UV post-exposure, necessary to cure completely the cross-linked material. This is only 95% cured after the manufacturing process [11].
- In addition, in order to provide employees protection a separate as well as a dust-free working space is needed.

*Outlook:*

- The big advantage of this method is the excellent accuracy; therefore, the development of this technique is heavily promoted in the field of Microstereolithography. The production of micro components offers here, beyond the prototypes area, promising areas of application of this technology.
- For the development of new materials for SLA, a lot of research work is being done. It is mainly about the thermal stability of the materials and the improvement of the mechanical properties, in order to close the gap with competing techniques such as Selective Laser Sintering and Fused Layer Modelling.

### Poly-Jet Modelling (PJM)

*Principle:* this procedure prints the photosensitive resin directly onto the build platform. UV lamps attached to the print head are curing the photosensitive plastic when passing the layer. A supplementary printing head is used in order to print the support material, which must be removed after the printing process in a post-processing working step.

A *schematic diagram* of this AM technique is shown in Figure 2.13 as well as a PJM machine and some exemplary products in Figure 2.14.

*Materials:* photosensitive polymers / resins.

*Advantages:*

- Similar to the SLA this method is also very accurate. With a resolution of 42 microns in X and Y axis and 16 microns in Z-axis, accuracies from 0.1 to 0.3 mm can be achieved.
- A post-exposure step is not necessary.
- The Connex3 multi-material 3D-printers from the Stratasys brought a family of machines on the market that can handle not only different coloured materials, but also materials with different mechanical properties in one printing process.

*Disadvantages:*

- Through the materials similarity used by SLA similar disadvantages appears. However, health and safety measures are significantly easier to implement, since the resins are in cartridges packed and directly cured during the construction process. A contact with liquid resin is therefore practically impossible.
- The material costs are approximately a factor four higher compared to laser sintering.

*Outlook:*

- Searching for materials with higher thermal stability and better mechanical properties is a topic for the researcher in this field.
- This method is unique in the AM world through the ability to handle several different materials at a time. However, significant improvements to the flexibility of materials are needed in order to replace the conventionally produced rubber.

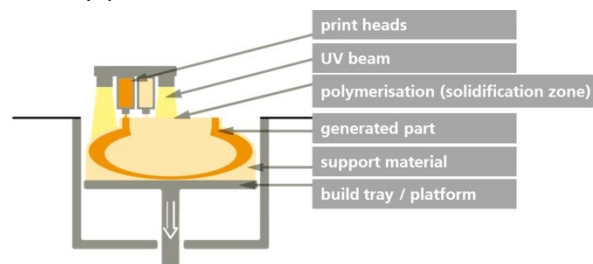


Figure 2.13. Schematic diagram of Poly-Jet Modelling.



Figure 2.14. Exemplary - PJM machine and products.  
[12]

### Digital Light Processing (DLP)

*Principle:* this process forms a layer-by-layer construction process by local solidification of a photopolymer liquid resin (polymers with photo activators) under the action of a light mask (by means of micro-mirror deflected light rays). A cleaning post-processing step is needed.

A *schematic diagram* of this AM technique is shown in Figure 2.15 as well as a DLP machine and some exemplary products in Figure 2.16.

*Materials:* photosensitive resins.

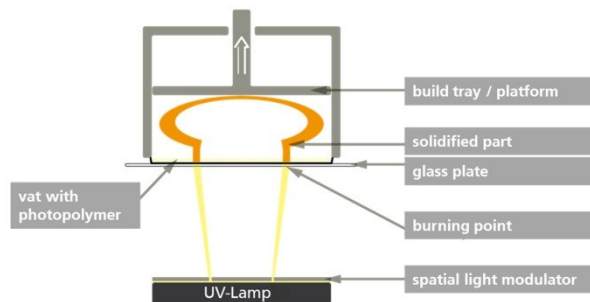


Figure 2.15. Schematic diagram of Digital Light Processing.

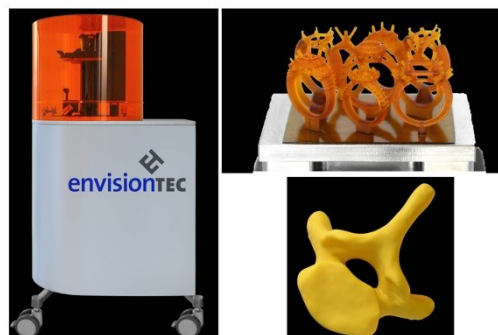


Figure 2.16. Exemplary - DLP machine and products.  
[13]

*Advantages:*

- Very compact and cheap AM machines.
- Used in the jewellery sector and for generating blanks, because of good resolution of details.

*Disadvantages:*

- The material used by this method has a low thermal stability and the mechanical properties are significantly lower than, for example, the ones by laser sintering.

**Layer Laminated Manufacturing (LLM)**

This method is also referred to as Laminated Object Manufacturing (LOM).

*Principle:* layers of paper or plastic sheets (experimentally also ceramic or aluminium films) are laminated to the respective previous layer. Then the contour is cut with e.g. a knife or a laser. A post-processing step of cleaning is necessary in order to obtain the manufactured part.

*Schematic diagrams* of this AM technique are shown in Figure 2.17 and Figure 2.18 as well as a LLM machine and some exemplary products in Figure 2.19.

*Materials:* plastics, paper and experimentally ceramics and aluminium.

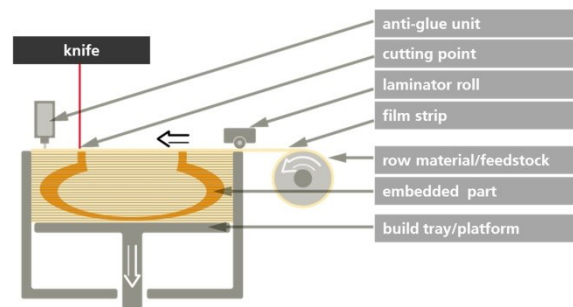


Figure 2.17. Schematic diagram of Layer Laminated Manufacturing using a knife as the cutting tool.

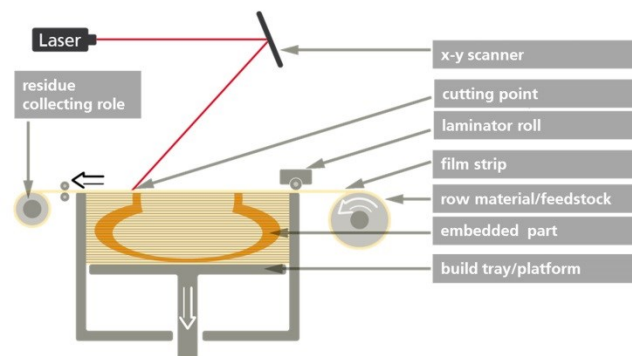


Figure 2.18. Schematic diagram of Layer Laminated Manufacturing using a laser as the cutting tool.

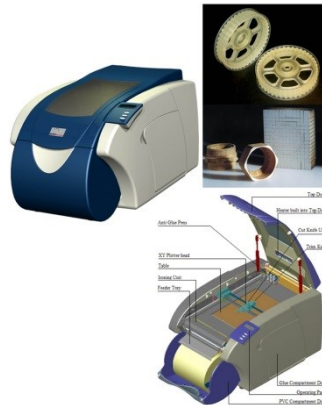


Figure 2.19. Exemplary - LLM machine and products.  
[14] [11]

*Advantages:*

- The manufacturing speed is not depending of the dimensions of the produced part. For large components, this ensures a relatively high production speed.
- Because of the layered bonding and lamination process, almost no internal stress is produced; thus, the parts are mostly free from distortion.
- Materials (e.g. paper) are very cheap.
- The machines are small and cheap.

*Disadvantages:*

- The mechanical strength of the parts is strongly varying depending on the building direction.
- Thin wall thicknesses in the Z direction can be realised only under certain conditions.
- The remaining material is not recyclable and depending on part geometry, often difficult to remove.

*Outlook:*

- These machines are rarely in practice. Consequently, there are only small investments for further developments, and so no clear trends exist for the future.

**Fused Layer Modelling (FLM)**

This method is also referred to as Fused Deposition Modelling (FDM).

*Principle:* a freely movable heating nozzle, in a build space, liquefies a filament-shaped plastic or wax material and builds the model in layers. At the end of the building process, the supporting structure must be removed, in a post-processing step.

A *schematic diagram* of this AM technique is shown in Figure 2.20 as well as a FLM machine and some exemplary products in Figure 2.21.

*Materials:* e.g. Acrylonitrile-Butadiene-Styrene (ABS) plastic.

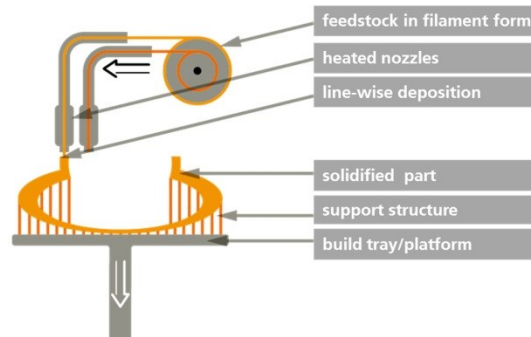


Figure 2.20. Schematic diagram of Fused Layer Modelling.

Figure 2.21. Exemplary - FLM machine and products.  
[12]*Advantages:*

- The purchasing price of these machines can be very low. Thus, companies with low to medium needs for prototypes can see an alternative in this technology.
- The small dimension of the machines and the uncomplicated material, mostly ABS, make the process very accessible, even for office use.
- A quite significant advantage of the machines sold by the company like Hewlett Packard (HP) and Stratasys is that the support material, used to clip around overhanging geometries, can be very quickly removed in an optional washing machine. Thereby the post-processing step, especially for parts with complex geometries, is hardly reduced.

*Disadvantages:*

- The process for small and complex structures is rather inappropriate, taking in consideration the diameter of the extruder nozzle. This leads (especially in the Z direction) to inaccuracies and, in comparison with other additive processes, to a poorer surface quality.
- The after-treatment of the surface is complex. A possibility, in case of ABS, is to vaporize the parts with acetone in order to melt the surface, and thus to smooth it.
- The material costs are relatively high: approximately a factor four higher in relation to laser sintering.
- For loads in build direction, it may happen that individual layers peel off from each other.

*Outlook:*

- Since the machines are cheap, compact, and suitable for the office, the sales by small businesses will probably increase in the future; therefore the trend is towards more cost-effective and compact devices for small companies, such as product design offices or modeller.
- Another noticeable trend is the presence of FLM machines also for home use.

**3D Printing (3DP)**

*Principle:* the feedstock spread in layers onto the build platform is selectively glued by a print head with an additional binder. There are two post-processing steps necessary: the cleaning and the infiltration one.

A *schematic diagram* of this AM technique is shown in Figure 2.22 as well as a 3DP machine and some exemplary products in Figure 2.23.

*Materials:* ceramic powder, plastic powder, metal powder, gypsum powder.

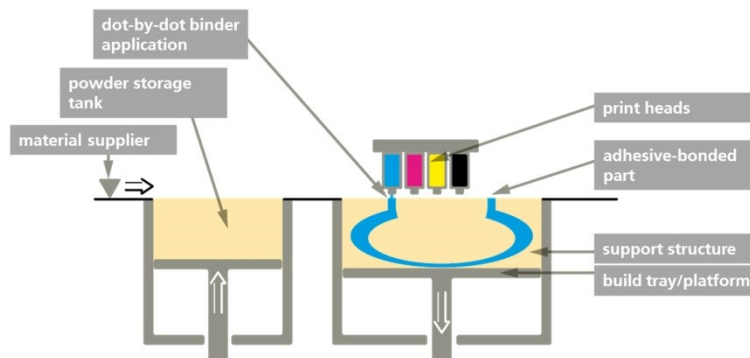


Figure 2.22. Schematic diagram of 3D Printing.



Figure 2.23. Exemplary – 3DP machine and products.  
[9]



*Advantages:*

- The greatest advantage of this method is that full-colour models can be manufactured using coloured binder.
- The residual powder can be fully recycled.
- Using technologically advanced inkjet printing heads the method allows a fast manufacturing process.
- No support constructions are necessary.
- A direct manufacturing of moulds is possible.

*Disadvantages:*

- The rough surface caused by the powder grain size.
- For filigree part geometries, there is a damage risk, already by removing the residual powder, as the resulting parts are unstable and friable. An elaborate infiltration with resin or other synthetic materials is necessary. However, many new infiltration materials, improved binder, and improved post-processing procedures are already on the way to solve this issue.

*Outlook:*

- Currently the technique is used in the area of visual models and of moulds manufacturing.
- Development and research projects are running in the field of ceramic printing, especially in the field of medical technology (e.g. dental implants). The need of such products is tremendous, so this method will gain importance in the future.

**Selective Laser Sintering (SLS) and Selective Laser Melting (SLM)**

*Principle:* these methods are similar in principle to the 3DP, but here the feedstock is selectively sintered with a laser beam. Not sintered powder acts as a support material during the build process. This powder, named used powder, can act as feedstock for further processes. A post-processing cleaning step is necessary for both techniques.

A *schematic diagram* of these AM techniques is shown in Figure 2.24 as well as a SLS machine and some exemplary SLS and SLM products in Figure 2.25.

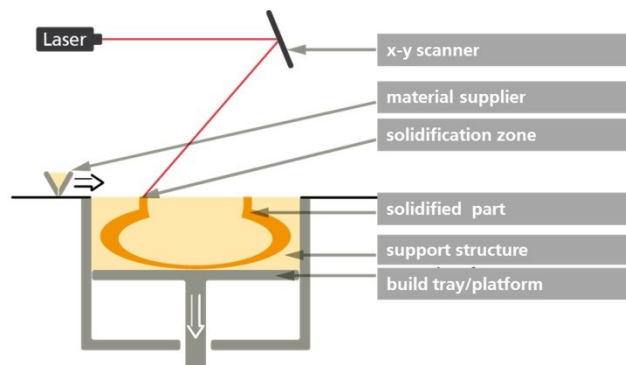


Figure 2.24. Schematic diagram of SLS and SLM.



Figure 2.25. Exemplary – SLS machine and SLS and SLM products.  
[15] [Fraunhofer IPA]

### **Selective Laser Sintering**

*Materials:* polyamides (PA), polystyrene (EPS), Polyetheretherketone (PEEK) and other thermoplastics; particle-reinforced polymers, polymer compounds, metal alloys, ceramics with fillers or binders.

*Advantages:*

- The method allows the use of a rich variety of plastics.
- The parts build with this method achieve the highest mechanical strength in comparison with all other AM techniques.
- The remaining material, the used powder [1] can be mainly recycled.
- Within the build chamber [1], the whole build space [1] is available for the manufacturing process; in X, Y and Z-axis only small spaces are necessary between the parts. When using a SLM this advantage does not apply, as only one layer of parts can be manufacture at a time.
- The material costs are low, compared to other AM methods.

*Disadvantages:*

- The powder grain size leads often to a rough surface.
- The solidified powder in vicinity of complicated part geometries leads partially to high cleaning efforts. In case of parts with cavities and long, thin channels, the removing of the used powder is sometimes time consuming.
- During the manufacturing process, depending on the material used, toxic gases can result.
- Another disadvantage is the high price of SLS machines. A good planning of the build space is necessary, since the machine can be operated economically only in case of a good capacity utilisation.

*Outlook:*

- Through good material properties, a trend towards batch production is recognisable (e.g. robot gripper).
- In addition, in the field of plastic SLS, flexible materials are just before the commercial launch.

### Selective Laser Melting

*Materials:* stainless steel powder and cobalt powder, usually metal alloys.

*Advantages:*

- Metal parts having a nearly 100% density with good mechanical properties can be achieved.
- The residual material can be recycled. Moreover, no additional material as support material is necessary.

*Disadvantages:*

- The powder grain size leads often to a rough surface.
- The powder handling and the required protective gas atmosphere needs additional safety precautions (e.g. ventilation system) and well-trained staff.
- After the manufacturing process, the build platform must be flat milled, because of the melting part process during the SLM process.

### Selective Heat Sintering (SHS)

*Principle:* compared to SLS here the powder is melted with a thermal print head. The print head is made of small heating resistors, which are applying heat on layers powder. A cleaning procedure is necessary in order to obtain the final produced part.

A schematic diagram of this AM technique is shown in Figure 2.26 as well as a SHS machine and some exemplary products in Figure 2.27.

*Materials:* thermoplastics.

*Advantages:*

- The produced parts have a high mechanical strength.
- The machines are having a low price in relation to other AM machines.

*Disadvantages:*

- SHS is less accurate than SLS.

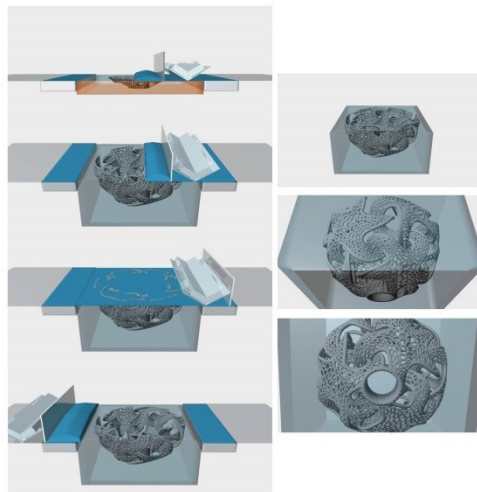


Figure 2.26. Selective Heat Sintering.

[16]



Figure 2.27. Exemplary – SHS machine and products.  
[16]

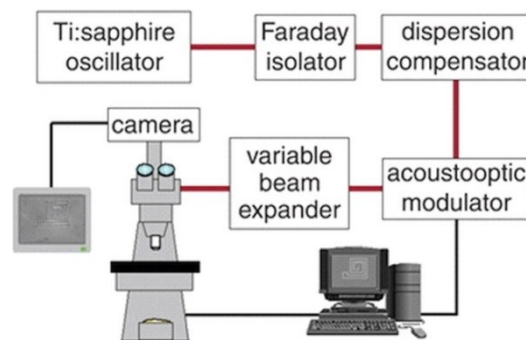


Figure 2.28. A typical setup for MAP.  
[17]

### Micro AM techniques

A detailed review of the micro AM processes and technologies is presented in [18], where these techniques are classified into three categories: scalable micro-AM systems, 3D direct writing, and hybrid 3D micro AM processes. An approach of using SLM in micro manufacturing, in order to produce parts with micro-sized elements, is issued in [19].

The multiphoton absorption polymerization (MAP) [17] is an AM technique based on the multiphoton absorption (MPA). Through MPA (see Figure 2.28), exposing a photoresist to one voxel (a volume element) at a time until the goal pattern has been scanned, emerged in the desired three-dimensional structure.

### 2.1.5. Selective Laser Sintering: from one-off production to batch production

Each of the techniques listed so far has its own advantages and disadvantages. Therefore, no method can be in principle excluded for a future batch production. Nevertheless, the plastics SLS enables a wide variety of innovative products, important from the economic point of view. The main reason lays in the parts good mechanical properties. The high toughness of the material allows production of flexible parts, such as hinges, grippers, and even springs. The material is resistant and can absorb high forces, being also suitable for stable structures [5].

In contrast to the resin based AM methods, the melting point by SLS is very high. The relatively high heat resistance is another aspect that makes this technique

to be one of the most important for a future AM batch and even mass production. In addition, no support material is required to stabilise the object during the building process. At a total capacity utilisation of the machines, the method is profitable through the low material cost; therefore is particularly suitable for small batch production [5].

Analysing Table 2.1, the SLS technology, despite limitations in terms of accuracy, cleaning efforts and surface finishing, is above the average in all areas of importance for a possible batch production. Therefore SLS, one of the AM PBF techniques, is the most suitable AM technique for the production of small quantity of identical products. Nowadays for products with high accuracy requirements, other methods are more appropriate [5].

Table 2.1. Evaluation of AM techniques [5].

	SLA	PJM	FLM/FDM	LLM/LOM	3D Printing	SLS
Mechanical properties	+++	+++	+++	+++	---	++++
Thermal Properties	---	---	+++	+++	+++	+++
Accuracy	++++	++++	---	+++	---	+++
Cleaning effort	+++	---	+++	---	+++	+++
Post-processing	---	+++	+++	+++	---	+++
Surface finishing	+++	+++	---	+++	---	+++
Number of positive ratings	12	12	12	10	7	16

## 2.2. Industrial Image Processing

Through Digital Image Processing (DIP) [20], using computer algorithms, useful information about a scene are extracted as well as features of interest are enhanced, while unimportant details are diminished.

A variety of image-producing sensors (e.g. x-ray devices, still and video cameras) can deliver digital images. A digital image embodies a finite number of pixels, being a two-dimensional discrete function, which describes the greyscale level of a pixel at a spatial coordinate.

Industrial Image Processing (IIP), known also as Machine Vision, based on DIP, is one of the most important computer vision field aside e.g. Medical Image Processing or face detection. IIP systems are extremely versatile: they can identify objects, verify the quality of a product, and even control different processes and machines. In Figure 2.29 the principal structure of an IIP system is presented, following the standard VDI/VDE Standard "Machine vision - Basics, terms, and definitions" [21].

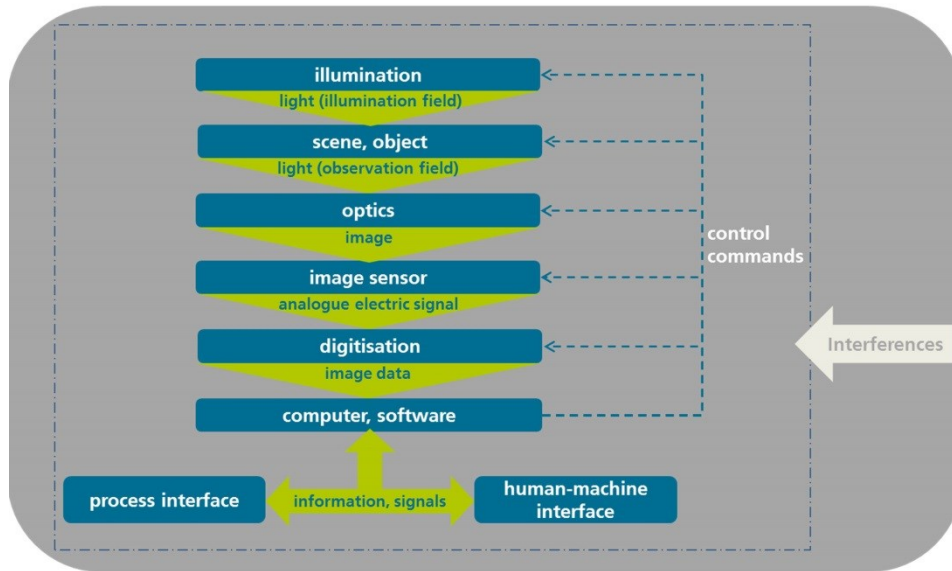


Figure 2.29. Structure of a Machine Vision system.

Theoretically, through continuous development of Machine Vision systems one can make measurable in a contactless, non-destructive way all that is visible. The IIP systems are not anymore only simple inspection systems, because meanwhile they can ensure even an early detection of production process trends: nowadays you can train in a dedicated software warning limits for e.g. dimensions and many other criteria in order to identify, at an early stage, creeping changes in the production process. The zero defect production, the short products life spans had a huge impact on the development of the IIP systems (see Figure 2.30). Noteworthy in this context, is though the IIP biggest challenge namely the fact that digital images are not directly measurements of the physical parameters of the scenes being inspected. Instead, it is about manifold interactions between different physical processes like e.g. the illuminating radiation and its interaction with the object being viewed [22].

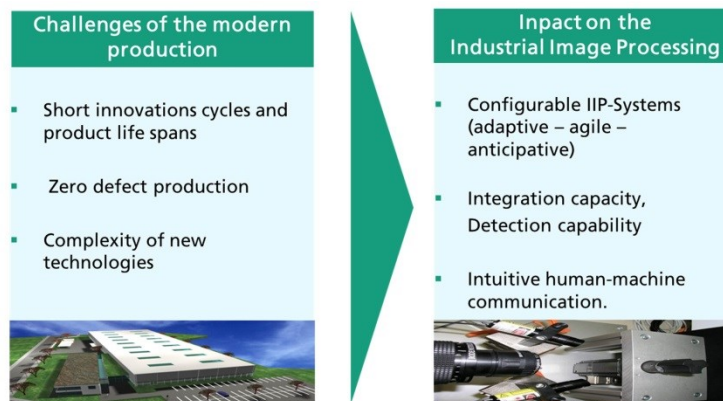


Figure 2.30. IIP trends and goals.

**2.2.1. Industrial Image Processing for inline Quality Control**

Different inspection and control tasks, during the manufacturing process, such as e.g. position and orientation detection, object identification, completeness checking, shape and dimensional inspection, surface inspection can be inline performed through real-time IIP. The Machine Vision system, meaning the IIP algorithms as well as the image sources, must run at a higher rate than the tact time of the manufacturing line.

3D IIP is still too slow (see Figure 2.31) for the most of the inline applications, in the context of higher data resolution and the computational time; 3D IIP being still very time consuming. However, in the last decade 3D digital image processing algorithms have known an unbelievable development, as for example in the area of automated segmentation and object recognition in 3D data sets [23]. Fast and flexible 3D object recognition solutions for Machine Vision applications are nowadays one of the hot topics in the applied research filed of 3D IIP [24].

In an era when the most companies are demanding dpm (defects per million) numbers of less than 100 [25], the 2D IIP is one of the most suitable and applied techniques for ensuring a 100% inline quality control. Two types of algorithms are for this purpose necessary: the image enhancement and the image analysis ones.

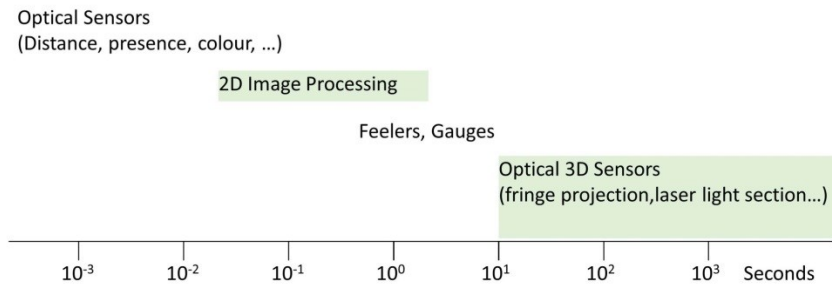


Figure 2.31. IIP time for different equipment (including the process time).

**2D IIP algorithms and methods for inline image enhancement**

Image enhancement algorithms, used as input for subsequent analysis, are known as low-level vision algorithms and can be divided in two classes: point transforms and neighbourhood operations. An overview of the most used enhancement operations in real-time IIP is presented in Table 2.2.

Table 2.2. Classification of digital image enhancement algorithms in common use.

Categorisation of digital image enhancement algorithms	Algorithms classes and enhancement operations - exemplary
Point transforms	Pixel mapping: <ul style="list-style-type: none"> <li>• thresholding, binarisation, Otsu method</li> <li>• histogram equalisation and specification</li> </ul>
	Colour space transforms
	Time averaging <ul style="list-style-type: none"> <li>• Kalman averaging</li> </ul>
Neighbourhood operations	Linear filtering: <ul style="list-style-type: none"> <li>• smoothing</li> <li>• sharpening</li> </ul>

Neighbourhood operations	Boundary detection
	Non-linear filtering <ul style="list-style-type: none"> <li>• median filter</li> <li>• morphology: erosion and dilatation; opening and closing</li> </ul>
	Re-sampling <ul style="list-style-type: none"> <li>• coordinate transforms: nearest neighbour, bilinear and bicubic interpolation</li> </ul>
	Adaptive filtering: <ul style="list-style-type: none"> <li>• mean</li> <li>• standard deviation</li> </ul>

*Point transforms operations:* are the simplest operations performed on a digital image, consisting in pixel values modification based on their intensity. Each output pixel is a function  $T$  (see Figure 2.32) of the corresponding input pixel;  $T$  is the same for every pixel. In most of the cases  $T$  is set up based on an overall statistics of the image input pixels. These operations are very fast and are used for e.g. global transformations (e.g. image contrast adjustment).

*Neighbourhood operations:* here each output pixel is a function  $T$  of a batch of corresponding input pixels (see Figure 2.33). This batch is called a neighbourhood, being usually a region surrounding a corresponding centre pixel (e.g. a 5x5 neighbourhood). Because the neighbourhood must be recalculated for each output pixel, these operations are more slowly, but have the advantage of implementing sophisticated enhancements (e.g. frequency and even shape filtering) [22].

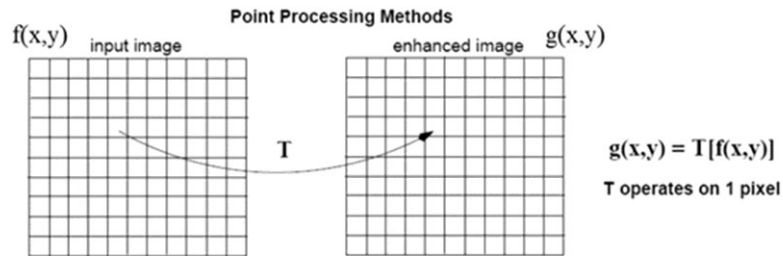


Figure 2.32. Image enhancement methods: point transforms. [26]

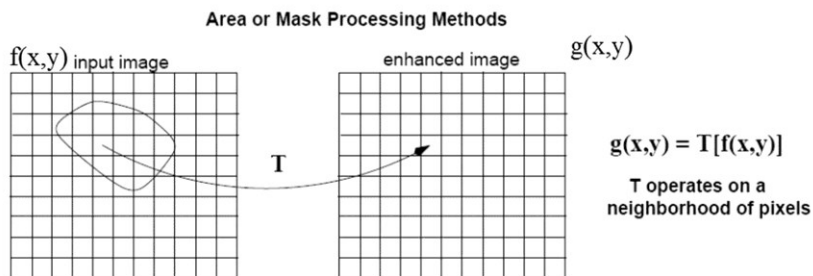


Figure 2.33. Image enhancement methods: neighbourhood operations. [26]



### 2D IIP algorithms and methods for inline image analysis

Image analysis, known as medium-level and high-level vision, by contrast to image enhancement operations, produces a smaller quantity of information. However, the information obtained is much more significant and informative (e.g. the orientation of a part and its position or an OK/ not OK decision). The speed of these kinds of operations is difficult to outline, as image analysis algorithms are often main intellectual property [22] of individuals or companies.

Pattern recognition is one powerful image analysis technique in order to solve tasks like e.g. inline detection and/or tracking, motion or shape estimation. With proper algorithms, image patterns corresponding to physical parts can be identified, for the purpose of calculate their e.g. position and size. Some of the most important methods for pattern recognition are the blob analysis, the template matching (e.g. based on correlation or edge based) or the Hough transform.

#### 2.2.2. Inline IIP enabler of SLS batch production

2D IIP is already a proven enabler of extremely efficient digital production processes, as in the case of the novel manufacturing process for peptide arrays based on electrophotography [27]. Here the biochips fabrication is in principle based on a layer-by-layer peptide laser printer system (see Figure 2.34).

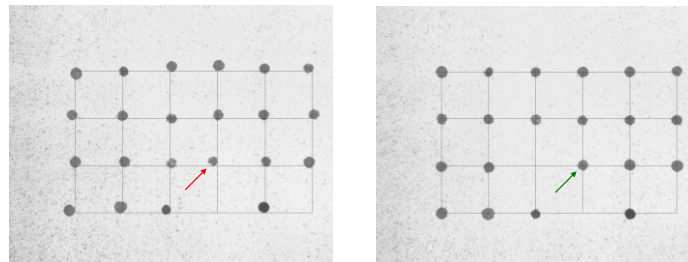


Figure 2.34. Printed peptide layer image before (left) and after a 2D IIP based system calibration (right).

The inline 2D IIP was also for many industry sectors the enabler of ensuring the part quality inline as well as of increasing the demand on a higher automation degree of the manufacturing processes. This fact is still valid for new manufacturing processes being on their way to become established production processes, as the AM PBF processes.

As exemplarily shown in Figure 2.35, nowadays, 2D IIP enables during the manufacturing processes:

- Measuring operations, with execution time of 0.1-10 milliseconds per operation as e.g. measure of length, diameters, angle, and concentricity;
- Inspection operations, with inspection time of 1-20 milliseconds per operation as e.g. inspection of presence, completeness, colour, and different codes types (e.g. Data Matrix Code). Even inline surface inspection takes nowadays place without prior training of surfaces, as in [28]. Here the surface model is derived during runtime from the dominant texture in image. In this way, not only an independence of fluctuations in target texture is achieved but also there is no need for prior training of target textures anymore. This method offers a big advantage for the inline quality control of object surfaces, being independent of the product fluctuations in the process line.

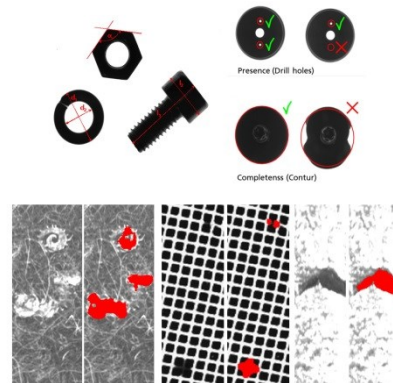


Figure 2.35. Inline 2D IIP: measure operations; presence and completeness inspection and self-adaptive defect detection of surfaces.  
[source: Fraunhofer IPA]

In this context, the question still without a comprehensive scientific answer is: can 2D IIP ensure an inline quality control for an AM PBF process? Can 2D IIP be the enabler of the SLS batch production, by ensuring the quality and reliability of the manufacturing parts? This topic will be debated in the next chapter, nevertheless the big challenges for the AM PBF processes, using the example of SLS are:

- If a Machine Vision system will be installed into a SLS machine, cooling houses for each of the IIP system components are inevitable, taking into account the fact that the temperature into the build chamber is, during manufacturing process, over 160 °C. The presence of these continuously cooled elements into the build chamber will directly influence the temperature distribution on the powder bed surface. However, the temperatures on the build platform, respectively on the powder bed surface must be as homogeneous as possible [29], in order to manufacture equal part properties throughout the whole build envelope.  
=>A 2D IIP system cannot be actually installed into the AM machine.
- If a Machine Vision system will be installed on the SLS machine other questions arise:  
=>Will it be possible to achieve the necessary accuracy, taking into account the given conditions as e.g. the big working distance or the angle between the IIP system and the inspection field?  
=>Will it be possible to identify the inline defects having a low scene contrast ratio (see Figure 2.36)?  
=> Will it be possible to generate a universal interface to use a specific developed IIP system with all type of SLS machines?

**Conclusion:** IIP, through its variety of solutions in the field of inspection and measuring, could be one enabler of the SLS batch production, but big challenges have to be overcome in order to extract the important data from the low contrast images.

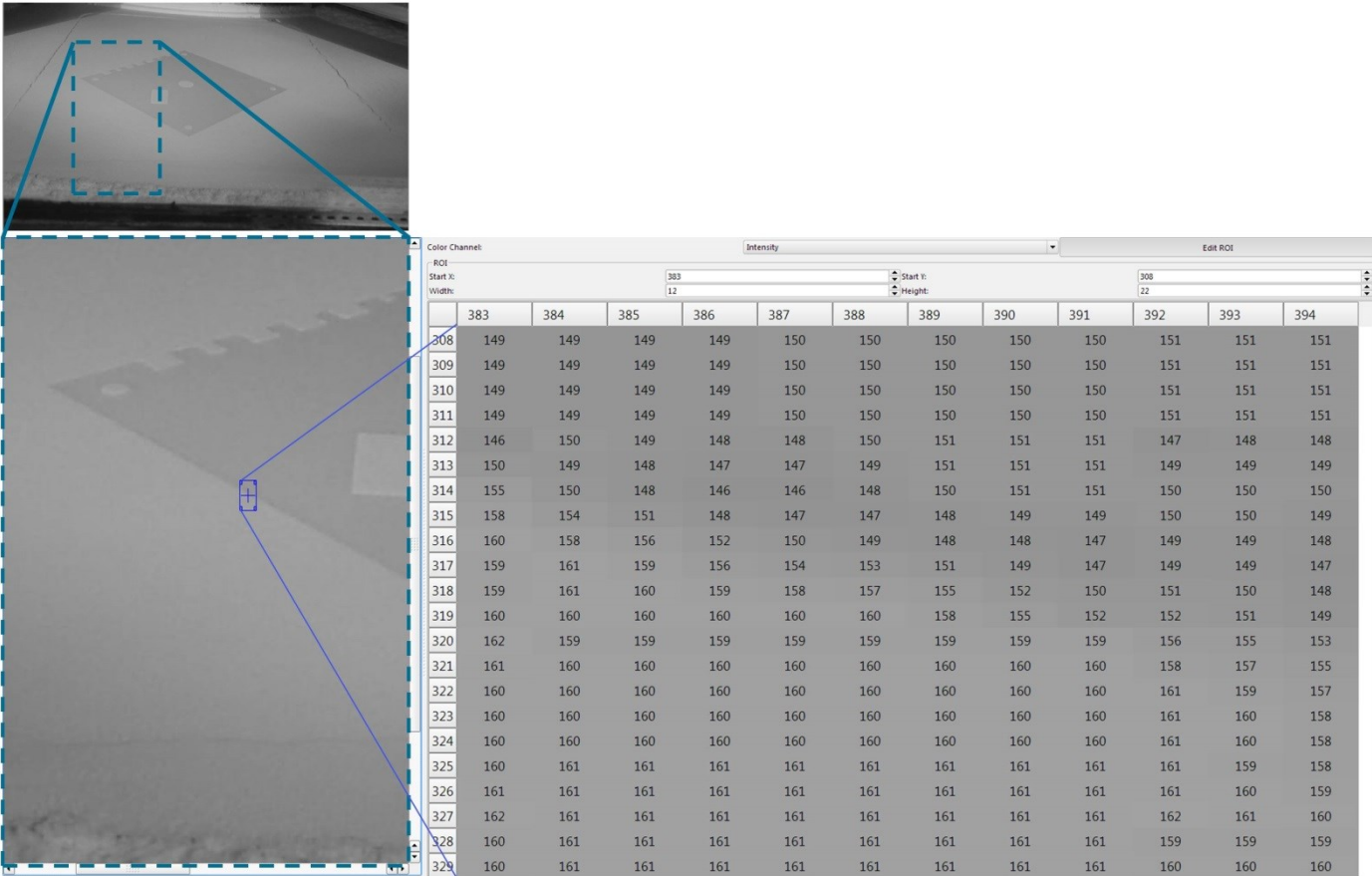


Figure 2.36. SLS layer: low scene contrast ratio.



## 3. STATE OF THE ART REVIEW ON QUALITY ASSURANCE OF AM PBF PROCESSES

### 3.1. Quality assurance chain

The prerequisite for highly customised unique parts, or even batch size one AM PBF products, which can be produced by additive manufacturing, as efficiently as conventional mass-produced parts [30], is the QA by the way of immediate product and process monitoring.

The scientific needs for the QA chain of the AM PBF product manufacturing process, by the example of SLS, can be summarised with respect to [31]:

- *Data-management*: presently there is no uniform data management. There are no standards that define the digital process chain for the different AM facilities. Thus, for example, geometrical and process data are widely mixed in machine-specific formats. There is no possibility to perform a recording of the entire product manufacturing process for each part produced. Therefore, the reproducibility is not provided in the AM production.
- *Real-production integration*: today additive production facilities are usually independently working laboratory machines, which are not integrated in a process chain. The process variables are partly recorded during the manufacturing process but not long-term logged.
- *3D-input-data* and their effect on the quality of the AM parts: the STL format is the de facto industry standard, despite recent data formats such as "Additive Manufacturing File Format" (AMF) or the "3D Manufacturing Format" (3MF). Up to date, no standards or guidelines that describe and/or specify the required quality characteristics for the STL models are available. 3D input data affect massively the quality of the final product and must therefore be suitable prepared with the requirements of a production process [32]. This is often carried out by manual, time and costs consuming, post processing steps.
- *Inline QC and in-process optimisation*: by the parts, components, or objects, which are additively manufactured, quality problems appear during the production process caused by different factors. These are leading to vulnerabilities, fractures, or product failure [33]. There is no AM PBF inline approach, which at the same time: can detect and classify comprehensively such errors; monitors and optimises in real-time AM PBF processes; offers a complete quality report of the produced part, and can abort the production process in extreme cases [31].

Typical errors and quality problems that may occur in the additive production and are especially affecting the additive manufactured products [31] are:

- lack of geometric accuracy of the parts, which is dependent on e.g. the raw material conditioning, the temperature control, the laser offset, the cooling process, the layer thickness or the slicing procedure;
- component distortions caused e.g. by the cooling process and the temperature control in the machine;

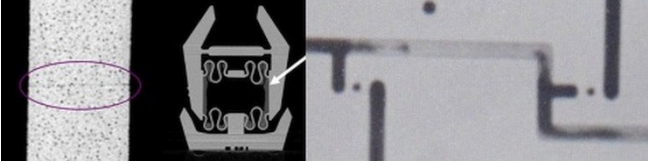
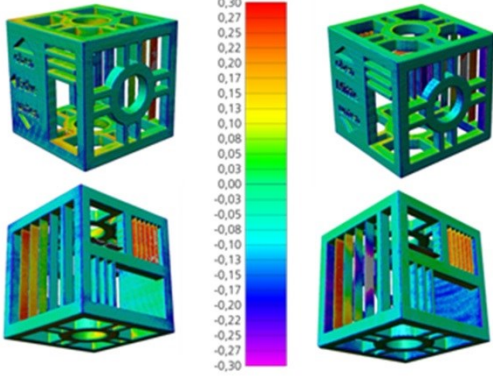
- fluctuations in the quality, depending on the placement of the parts in the space of the machine;
- reduced mechanical strength: deviating density through e.g. too low laser power or too high laser speed;
- surface defects: aged material, unsuitable material mixture, contamination by extraneous substances;
- closure of narrow or deep channels and holes in the component by partial melting of the marginal zone, dependent on the geometry.

### 3.2. Quality assurance by means of industrial computer tomography

Using the advantages of Industrial Computer Tomography (CT), combined with the versatility of 3D data processing and evaluation methods, a quality criteria analysis of the AM PBF processes have been carried out by Kroll et al. so that part defects occurring after the production as e.g. component deformation, mechanical properties, could be detected [32]. Based on these results (see Table 3.1), appropriate test objects were designed and additively build under pre-defined conditions, in order to:

- achieve for each AM PBF process step an optimal quality prediction by 3D-sensor and software solutions;
- monitor the quality and efficiency during the product development process.

Table 3.1. Exemplary - 3D-evaluation results for the quality criteria of AM PBF [32].

Off-line inspection	Results
<p>Detection of inner material</p>	<p>Voxel slices with visible defect (missing layer in the left) or inner rest material (right)</p> 
<p>Geometrical accuracy of the build process</p>	<p>Colour-coded results of nominal-actual comparison of CT-surface data of laser sintered test parts with CAD data.</p> 

The experimental designs reveals that off-line QA, by means of CT, modern 3D sensors, data processing and evaluation methods, offers a good opportunity in order to evaluate the quality characteristics of laser-sintered components, and to predict off-line their behaviour, in different industrial applications [32].

QA by means of CT shows that, based on detailed off-line analysis of special designed and build objects, corrections of 3D-input-data and even modification of process parameter are possible, in order to achieve for the next manufacturing process a higher product quality. In addition, a plurality of operating instructions and statements regarding optimal design and component production can be derived, as e.g. thin wall thicknesses (<0.5 mm) are to be built in an upright position [32].

As the AM PBF manufacturing process is continuously affected by different parameters, as e.g. the feedstock or the AM machine fitness influenced by errors of system components, QA through off-line methods is very useful, but not sufficient in order to ensure the quality and reproducibility of the AM PBF parts.

### 3.3. Science and technology needs in metrology

The reported studies, in the field of real-time measurements and control of AM PBF processes and parts, focuses almost only on the SLM process. The developments of this technology being also pushed by the aerospace industry; here for critical applications (e.g. aerospace engine components) a less than 100% production quality qualified process or machine is out of question.

The U.S. National Institute of Standards and Technology (NIST), along with experts and stakeholders, worked out a summary of the challenges and barriers with respect to the AM processes monitoring, control, and measurements [34]. The most rated categories with the highest priority have been mentioned to be:

- The deficit of sensors for process monitoring and in-situ control, e.g. a complete real-time system sensor scheme, knowing what and how should be inspected and/or measured.
- The absence of measurement methods for inline process control e.g. in-process powder bed distortion monitoring.
- The optimisation of non-destructive evaluation (NDE) techniques for AM PBF using metal powders.

For a higher quality and part-to-part reproducibility in production environments, analysing the priority road map of future topics for the AM processes and equipment, NIST identified the core of the scientific needs [34]:

- *Inline measurement methods for inline non-destructive inspection (NDI) based on specific AM PBF imaging and processing NDE methods.*
- *Systems for in-situ measuring and monitoring AM PBF products and processes, hinting "standardised, repeatable, and reliable sensor technology and validated models that can be applied across multiple AM platforms" [34].*

Mani et al., presenting a literature review of AM PBF control schemes, process measurements respective modelling, and simulation methods [35], mention that two critical factors are most often assumed to influence the output of the AM PBF processes: the melt pool temperature and its size. The melt pool, being directly affected by high local temperatures, is a small molten powder area caused by the absorption of the laser beam spot by the exposed powder [35].

### 3.4. Control schemes

A team from the University of Leuven (KUL), developed different experimental setups for control schemes in SLM-related processes [36] [37] [38]. All these setups have been conducted on in-house developed SLM machines, e.g. the LM-Q. For a feedback control system [36] for the laser power parameter, the melt pool was monitored using a laser coaxial photodiode and a CMOS (complementary metal-oxide-semiconductor) camera. The 8-bit greyscale values delivered by the CMOS camera have been related to temperature values and a correlation between the powder melt temperatures has been acknowledged. The melt pool geometry, e.g. size and form, has been also determined by camera images. The photodiode signal, which correlated well with the melt pool, was used as feedback to control the laser power. In this way, on the specific hardware setup, improvements of the surface roughness have been achieved. Craeghs et al. report that more than 50 parameters are influencing the melting process [37]. After the integration into the LM-Q machine of a powder deposition visual inspection set-up and of a real-time monitoring system, they concede in [37] that controlling or even monitoring all SLM process parameters is a significant challenge [35]. The visual inspection set-up that was developed, used a "line profile", defined as an average (mean) of five vertical lines in an image of a powder layer [37], as reference profile for comparison the profiles of other layers. Into the LM-Q machine of KUL (see Figure 3.1) the visual system was further developed by mapping the temperature signatures of the molten pool, as a function of the laser beam position (X-Y) on the powder bed [38]. Using the "mapping of melt pool data" [38] the KUL group was able to detect deformation due to thermal stresses and overheating zones due to overhangs [35].

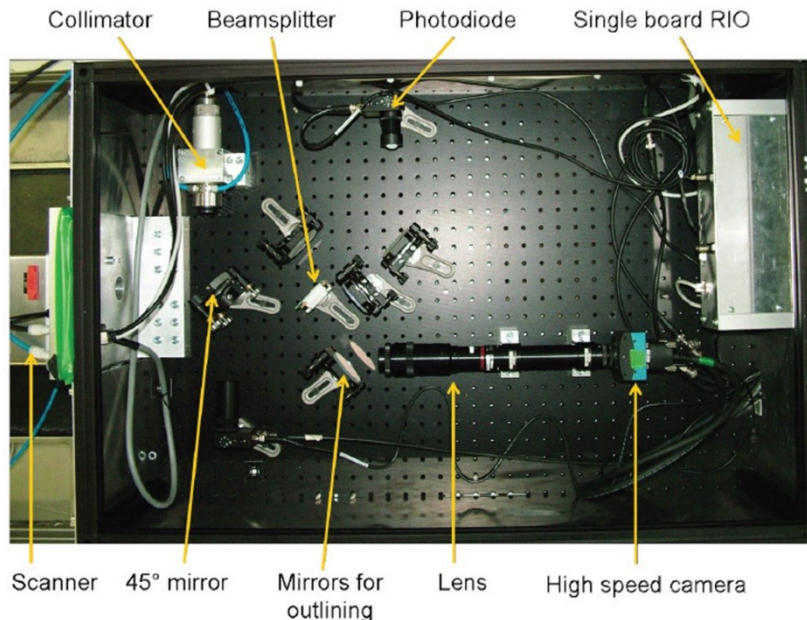


Figure 3.1. Optical monitoring set-up into the in-house build LM-Q machine of KUL. [38]



In order to determine the roughness of the surface generated by the re-solidification of the melt pool, Mumtaz and Hopkinson studied in [39] the effect of heat being delivered to these regions by a laser beam. They experimented with a pulsed laser system, respectively with various pulse shapes, and obtained through pulse shaping an added degree of control of the SLM process, emerged into a lower surface roughness on the side and top of the manufactured parts [35].

For higher part accuracy, the research concentrate on methods involving the laser beam. Exemplarily mentioned here are the controlling of the laser power and of the scan speed [40] or different laser beam compensation techniques [35] [41]. Other related works based on laser beam offset are issued in [42], [43].

### 3.5. In-process measurements

Related to AM PBF in-process measurements, the primary focus of research has been associated with determining the geometry and the temperature profile of the heat affected zone (HAZ); an area which includes the melt pool and its vicinities [30] [35]. Works related to surface temperature measurements as well as residual stress and geometrical measurements have been reported.

*Surface temperature measurements using infrared (IR) thermography and pyrometry.*

Here are two kind of thermographic imaging systems that have been experimented:

- Co-axial systems: the working field of the systems aligns with the laser beam, directly through the scanning optics [36], [38], [44], [45], [46] [47].
- External systems: the imaging system is set outside the build chamber, viewing through the AM machine operational window as in [29] [48] [49] [50] [51].

Santosprito et al. used a pulsed laser transient thermography system to evaluate the heat movement through the inspected area [52]. Here a defect size of around 400  $\mu\text{m}$  was reported as minimum detectable. Krauss et al. using an uncooled microbolometer have investigated the HAZ radiance images with respect to the area, aspect ratio, and circularity [53].

In the field of pyrometric measurements, co-axial with the laser beam, Pavlov et al. found that the pyrometer signal from the melt pool is sensitive to variation of e.g. scanning velocity or powder layer thickness. These acknowledgments could be the basis for an inline quality control system [54].

*Residual stress*

In [55] Shiomi et al. discussed an inline approach for measuring the residual stress by means of strain gages installed on a SLM build platform. Van Belle et al. also researched the residual stresses induced during an AM PBF process, through a strain gauge rosette being fixed under a support platform [56]. Using force balance principles and monitoring the strain gauge data they determined residual stress in the part and the support, corresponding to elastic bending [35].

*Geometric Measurements*

In this field, of in-process geometric measurements, there is not much research work reported [35]. This concludes in the background study done in 2015

by NIST [35] as well as in the work from Vlasea et al., which focuses on understanding the need for developing a new generation of control strategies [57].

Nevertheless, Cooke and Moylan reported that two-dimensional geometric measurements, during the manufacturing process on a 3D Printing machine, can be feasible for the characterisation of the inner part geometries as well as for a process enhancement [58].

Pedersen et al. [59] discussed a vision system that, during a build job on a 3D Printing machine, acquires and stores images corresponding to each layer for numerical post treatments. Taking into consideration the nature of the layered manufacturing, a generic geometry reconstruction method was suggested.

Kleszczyński et al. use a CCD camera, with shift and tilt lens for the image correction, installed on the viewport of a SLM machine [60]. With the system, they collected images representing potential error sources during the build process.

### 3.6. Conclusions

Despite existing research activities, the AM PBF part quality control continue to be an expert issue by means of using off-line destructive methods or NDT technologies as e.g. the industrial computer tomography. There is no inline quality control system, to ensure the part quality and the part-to-part reproducibility during the AM PBF process. There is no inline quality control system that can be applied across different AM machines to ensure a reliable PBF production.

In detail, the deficiencies of the actual approaches in the field of AM PBF control schemes and in-process measurements consist of:

- All control schemes approaches, based on different defect detection strategies, are using complex process monitoring systems that require a modification of the entire scanning system (laser beam sintering/melting system) or at least of its optical components.
- In-process surface temperature measurements, using infrared thermography or pyrometry, concentrate mostly only on the melt pool and its vicinities. By both reported thermography system types, big challenges have been identified:
  - the thermal image can give unreliable temperature data because the emissivity of feedstock, melt pool, and solidified surface differ, as e.g. in [51];
  - the laser light focusing lens, so called f- $\theta$  lens, can induce spectral and chromatic aberrations [35];
  - powder particles from the HAZ can cover partially or totally, the AM machine window dedicated for the thermographic system and so can change the radiation transmission through the window [48] [50] [61].
- No research work ensures inline geometric measurements, even if in-process images have been acquired. None of the approaches reports inline non-destructive inspection measurements or AM PBF specific imaging and processing evaluation methods.
- There is no approach for the identification of the tasks for an inline quality control system, in order to ensure the quality and the reliability of the AM PBF manufactured parts.

## 4. TASKS FOR INLINE QC AND IN-SITU OPTIMISATION OF AM PBF PROCESSES

### 4.1. Overview

The primary focus of the research in measurements for real-time control of AM PBF has been associated with determining the geometry and the temperature profile of the heat affected zone [35]. For the part's quality monitoring, during the build cycles, the question, still without a comprehensive scientific answer, is: which are the overall tasks for an inline QC system? Which are the overall effects of the influencing parameters (see Figure 4.1) on AM processes having an appearance during the manufacturing process itself? Only when these effects are clearly determined, the defects and failures during AM PBF processes, as well as the tasks for an inline QC system, can be accurately defined and used for the development of an overall inline QC system. In this way, an inline part quality control and even a real time optimisation of the PBF processes can be achieved.

It is widely known that the relationships between the influencing parameters in the PBF process are complex [35], a fact that hinders the development of inline QC and in-process optimisation systems. In this context, the scope of the work described in this chapter, is to identify and determine the inline defects and failures during AM PBF processes and the tasks for an inline QC system, using the example of the SLS process.

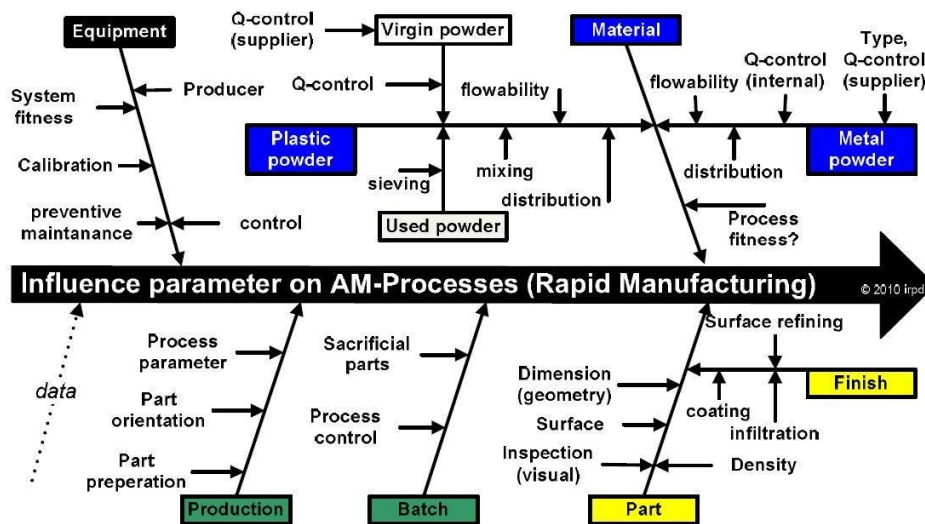


Figure 4.1. AM PBF production chain - influencing parameters. [62]

The obtained results, being a *novelty*<sup>1</sup>, should be the basis for a conceptual approach for an inline QC system. The focus on the SLS process is driven by the fact that SLS is one of the main technologies for a future batch production. Nevertheless, the proposed approach can be applied to other PBF processes through a deductive procedure.

## 4.2. Quality control in AM PBF processes

The quality of the AM PBF parts and their designed functionality (Figure 4.2 and Figure 4.3) can be very easily impaired, taking into consideration the following items:

- nowadays the quality management and the quality assurance is not ensured,
- neither quality standards, nor a generally accepted quality management standard or system are available,
- a larger variety of parameters are influencing the manufacturing processes based on these techniques [62].

In addition, the reproducibility and good dimensional stability of the parts is nowadays a big challenge. Typical errors and quality problems that may occur in the additive production and which especially affect the additive manufactured end-products have been presented in chapter 3.1.

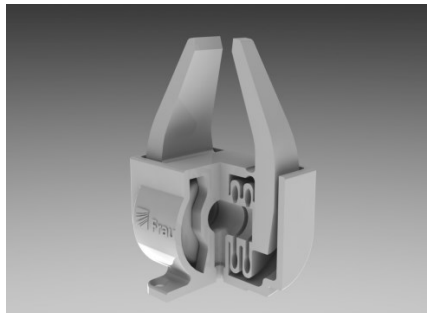


Figure 4.2. 2-Finger angular gripper.  
[source: Fraunhofer IPA]

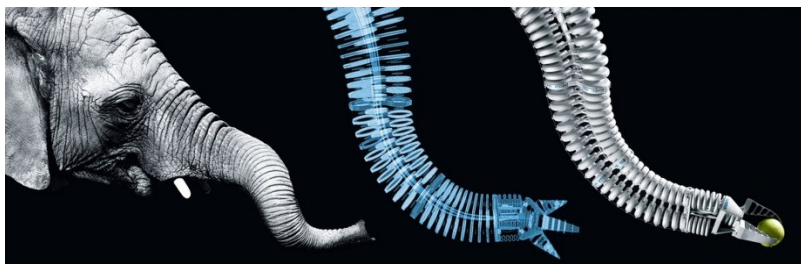


Figure 4.3. Bionic Handling Assistant.  
[source: Festo/Fraunhofer IPA]

---

<sup>1</sup> "forming a distinct contribution to the knowledge of the subject and/or evidence of originality by the discovery of new facts and/or by the exercise of independent critical power" [79]

### 4.3. Identification of inline defects and failures

Up to date preventing and/or detecting all defects and faults in PBF AM manufactured products is nearly impossible, because the processes are influenced by numerous parameters (see Figure 4.1). These can be summarised in four QM categories: *Equipment*, *Feedstock* (material) [1], *Production* including Batch, and *Part* including Finish (see Figure 4.1) [62].

A comprehensive approach was developed in order to identify all defects and failures having an appearance during the build cycles of the AM PBF processes. For all AM PBF techniques, the approach's starting point are the four categories of the QM aspects: *Equipment*, *Feedstock*, *Production* including Batch, and *Part* including Finish. In the first, through a deductive approach, the *Generators of Potential Quality Failure Modes* corresponding to each of the four QM categories, which are actually describing the *Potential Quality Failure Mechanism* of the respective AM PBF process, will be determined. In the next step using a deductive procedure, based on a qualitative top-down Fault Tree Analysis, going through each *Generator of Potential Failure Modes*, the corresponding *Potential Quality Failure Causes*, respectively all *Potential Quality Failure Effects* will be identified. The overall inline defects and failures for the respective AM PBF process are given by the in-process appearance of the *Potential Quality Failure Effects* (Figure 4.4).

In the following subchapters, applying the developed approach for one of the AM PBF processes, namely the SLS process, for each of the QM categories the identified inline defects and failures are determined.

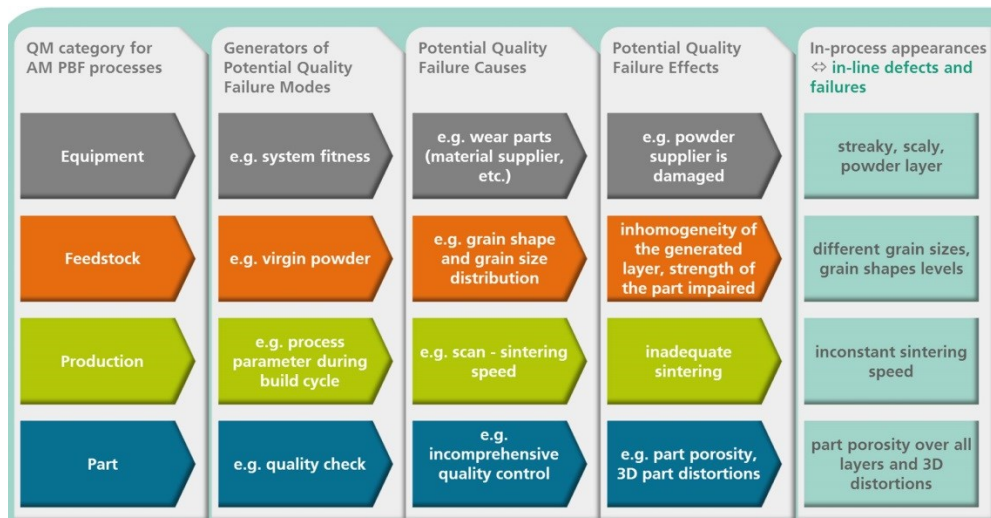


Figure 4.4. Schematic diagram of the approach for the identification of inline defects and failures during AM PBF processes.

### 4.3.1. Equipment

Considering the QM category *Equipment*, three *Generators of Potential Quality Failure Modes* were identified: uncleanness of the system, system fitness influenced by errors of system components and the system performance, influenced by inadequate functionality of the system (see Table 4.1). For example, taking into consideration the uncleanness of the system: one of the *Potential Quality Failure Causes* in this case are the depositions on the laser window (see Figure 4.5), which directly can influence the laser power respectively the elongation at break and the strength of the build part.

The in-process appearances of these *Potential Quality Failure Effects*, being actually inline defects and failures, are not only the deposits on the laser window themselves, but also geometrical distortions and even inappropriate layers adhesion.

In the same way considering the system fitness respectively, the wear parts (e.g. material supplier) as one of the *Potential Quality Failure Causes*, the *Potential Quality Failure Effects* and the corresponding in-process appearances of these effects are:

- damaged powder wiper -> streaky, scaly powder layer (see Figure 4.6),
- tilted build platform -> tilted layer geometry, and
- improper functionality of build platform generated by e.g. defects of the stepper motor -> layers overlapping, layers sintering totally failed.

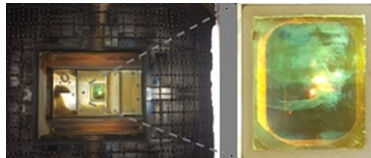


Figure 4.5. Laser window viewed from building platform side (left), laser window zoom (right).  
[source: Fraunhofer IPA: EOS FORMIGA P 100]



Figure 4.6. View of the powder layer after application - serious defect in the powder layer.  
[source: Fraunhofer IPA: EOS FORMIGA P 100]

Table 4.1. Equipment – identified inline defects and failures, using the example of SLS.

Generators of Potential Quality Failure Modes	Potential Quality Failure Causes	Potential Quality Failure Effects	In-process appearances ⇔ inline defects and failures
Uncleanness of the system	deposits on laser window	low laser power, small elongation at break, low strength of the part	<i>inappropriate layers' adhesion, geometrical layer and part distortions, deposits on laser window</i>
	impurities	strength and even geometry of part impaired	<i>impurities in layer</i>

System fitness influenced by errors of system components	laser and optical system	to low or to high laser power, small elongation at break, low strength of the part, thermal damage in case of too high laser power	<i>inappropriate layers' adhesion, geometrical layer and part distortions, melting of the edge zone depending on the geometry, porosity</i>	
	temperature control	geometry and strength of part vitiated	<i>inconstant temperature</i>	
	inert gas supply	burning of the layer	<i>melting of the part, black sintered</i>	
	wear parts (e.g. seals, material supplier)	powder supplier damaged		<i>streaky, scaly, powder layer</i>
		tilted build platform		<i>tilted layer geometry</i>
		build platform got stuck or lost the height information because of defects of the stepper motor	<i>Layers' overlapping, layers' sintering failed</i>	
System performance influenced by inadequate functionality of system	powder application performance	homogeneity of the generated powder layer affected	<i>inhomogeneity of the layer thickness</i>	
		density variation of the powder layer	<i>density variation of the powder layer</i>	
	scaling	scaling factor variations	<i>geometrical deviations and distortions of the sintered layer</i>	
	tolerances	inadequate tolerances	<i>geometrical deviations of the sintered layer</i>	
	beam offset	inadequate layer sintering	<i>geometrical deviations and distortions of the sintered layer</i>	
	low scan speed	high porosity and low density	<i>porosity, inhomogeneity</i>	
	surface roughness	layer and part surface roughness variation	<i>layer and part surface roughness</i>	

### 4.3.2. Feedstock

Two *Generators of Potential Quality Failure Modes* were identified: the virgin and the refreshed powder. Exemplarily, taking into consideration the virgin powder: the grain shape and its size distribution, as one of the *Potential Quality Failure Causes*, are influencing the homogeneity of the applied powder layer and of the final part as well as the strength of the SLS manufactured object. The in-process appearance of the *Potential Quality Failure Effects* is the grain size.

The *Potential Quality Failure Causes*, their effects and the identified inline defects and failures are presented in Table 4.2.

Table 4.2. Feedstock – identified inline defects and failures, using the example of SLS.

Generators of Potential Quality Failure Modes	Potential Quality Failure Causes	Potential Quality Failure Effects	In-process appearances ⇔ inline defects and failures
Virgin powder	grain shape and grain size distribution	inhomogeneity of the generated powder layer and final part, strength of the part impaired	<i>different grain sizes, grain shapes levels</i>
	thermal properties like melting point and recrystallization	inadequate layer sintering	<i>inconstant temperature on sintering point, inappropriate layers' adhesion</i>
	bulk density	affected processability	-
Used powder	type and mesh size of sieve (preferred mesh 140 µm)	grain size variation, strength of the part vitiated	<i>different grain size, inhomogeneity of the powder layer</i>
	Melt Flow Index (MFI)	mechanical strength	-

### 4.3.3. Production

In this case, also two *Generators of Potential Quality Failure Modes* were identified, namely: the part preparation and orientation as well as the process parameters during manufacturing (see Table 4.3). Also in this case, considering the atmosphere in the process chamber, as one of the *Potential Quality Failure Causes*, the burning of the layer was deducted as a *Potential Quality Failure Effect*. The in-process appearance of this effect is the melting of the part (Figure 4.7) respectively a "black sintered" layer.





Figure 4.7. "Black sintered" object.

Table 4.3. Production – identified inline defects and failures, using the example of SLS.

Generators of Potential Quality Failure Modes	Potential Quality Failure Causes	Potential Quality Failure Effects	In-process appearances ⇔ inline defects and failures
Part preparation and orientation	resolution of the STL file	part accuracy affected	<i>high surface roughness</i>
	part orientation	inadequate sintering	<i>part and layer orientation failure</i>
Process parameter during manufacturing	laser power	inadequate sintering	<i>inappropriate layers' adhesion, geometrical layer and part distortions, melting of the edge zone depending on the geometry, layer porosity</i>
	scan, sintering speed	inadequate sintering	<i>inconstant sintering speed</i>
	scan, sintering line	inadequate sintering	<i>scan line deviation</i>
	temperature profile	inadequate layer sintering	<i>inconstant temperature on sintering point, inappropriate layers' adhesion</i>
	layer thickness	part homogeneity affected	<i>variation of layer thickness, inhomogeneity</i>
	laser exposure style	-	-
	hatch distance	overlap area between two hatch lines	<i>inappropriate hatch distance, inappropriate layers' adhesion</i>
	atmosphere	burning of the layer	<i>melting of the part, black sintered</i>
	skywriting	melting of the edge zone is not correct	<i>skywriting length variations, melting of the edge zone, geometrical layer and part distortions</i>

#### 4.3.4. Part

The *Generators of Potential Quality Failure Modes*, the QM subcategories with respect to the Total Quality Management (TQM) of the part, are the part protocol and the quality check. For example, considering the recording of the entire product manufacturing process for each part produced, as one of the *Potential Quality Failure Causes*, we deduct as *Potential Quality Failure Effect* the fact that a part quality report is unaccounted for the end-user of the respective SLS part. If the SLS produced part has complex functionalities integrated, and not only then, a quality protocol of each layer, respectively of the part as a whole is imperative. In this case, the task for the inline QC system is the recording of each and any quality part information in a protocol.

The Potential Quality Failure Causes, their effects respectively the tasks for the inline QC system are presented in Table 4.4.

Table 4.4. Part – identified future tasks for an inline QC system.

Generators of Potential Quality Failure Modes	Potential Quality Failure Causes	Potential Quality Failure Effects	Inline QC system tasks
Part protocol	recording the entire product manufacturing process for each part	no reproducibility and reliability is given	<i>collect any quality part information in a protocol</i>
Perform quality check	incomprehensive quality control	part distortions, vitiated mechanical strength, 3D distortions, surface roughness, 3D dimensional deviations, etc.	<i>identification of all inline defects and failures (see Table 4.1, Table 4.2, Table 4.3), part distortions, surface roughness, 3D dimensional deviations, porosity analysis over all layers, etc.</i>

#### 4.3.5. Results

In Table 4.5 the overall identified inline defects and failures for the SLS processes [30] are summed up in connection with the *Potential Quality Failure Causes*, as an example for one of AM PBF processes.

Table 4.5. Overall identified SLS inline defects and failures.

Potential Quality Failure Causes	Inline defects and failures during the SLS process
deposits on laser window	<i>inappropriate layers' adhesion, geometrical layer and part distortions, deposits on laser window</i>
impurities	<i>impurities in layer</i>
laser and optical system	<i>inappropriate layers' adhesion, geometrical layer and part distortions, melting of the edge zone depending on the geometry, porosity</i>
temperature control	<i>inconstant temperature</i>
inert gas supply	<i>melting of the part, black sintered</i>
wear parts (e.g. seals, material supplier)	<i>streaky, scaly powder layer</i>
	<i>tilted layer geometry</i>
	<i>layers' overlapping, layers' sintering failed</i>
powder application performance	<i>inhomogeneity of the layer thickness</i>
	<i>density variation of the powder layer</i>
scaling	<i>geometrical deviations and distortions of the sintered layer</i>
tolerances	<i>geometrical deviations of the sintered layer</i>
beam offset	<i>geometrical deviations and distortions of the sintered layer</i>
low scan speed	<i>porosity, inhomogeneity</i>
surface roughness	<i>layer and part surface roughness</i>
grain shape and grain size distribution	<i>different grain sizes, grain shapes levels</i>
thermal properties like melting point and recrystallization	<i>inconstant temperature on sintering point, inappropriate layers' adhesion</i>
type and mesh size of sieve	<i>different grain size, inhomogeneity of the powder layer</i>
resolution of STL file	<i>high surface roughness</i>
part orientation	<i>part and layer orientation failure</i>
laser power	<i>inappropriate layers' adhesion, geometrical layer and part distortions, melting of the edge zone depending on the geometry, layer porosity</i>
scan, sintering speed	<i>inconstant sintering speed</i>
scan, sintering line	<i>scan line deviation</i>
temperature profile	<i>inconstant temperature on sintering point, inappropriate layers' adhesion</i>
layer thickness	<i>variation of layer thickness , inhomogeneity</i>
hatch distance	<i>inappropriate hatch distance, inappropriate layers' adhesion</i>
atmosphere	<i>melting of the part, black sintered</i>
skywriting	<i>skywriting length variations, melting of the edge zone, geometrical layer and part distortions</i>

#### 4.4. Tasks prioritisation

Using the approach presented in chapter 4.3, the determined inline defects and failures, as e.g. in Table 4.5 for the SLS process, are in principle the overall tasks for an inline QC and in-situ optimisation of AM PBF processes. Nevertheless, a prioritisation of the inline defects and failures is necessary in order to assure a firm foundation for the development of an inline QC system, respectively for future in-process control schemes. In this circumstance, the next step is to determine the ranking position of each of the *Potential Quality Failure Causes*. Their associated inline defects and failures are the basis for the correction of the failures in SLS processes with respect to the quality aspects.

For the ranking of the probability and severity of the *Potential Quality Failure Causes* and of their corresponding effects, respectively of the inline defects and failures, at least two factors are necessary and should be taken into account: their occurrence frequency and the sunk costs directly implicated. These two factors, especially the occurrence frequency, are strongly depending on the AM machine and on the feedstock used. Therefore these two factors cannot be exactly determined for all AM PBF processes, but can be approximated for particular AM machines based on an e.g. Design of Experiments (DoE) approach.

A DoE has been implemented in order to generate clear-cut conclusions for the ranking of the frequency of the inline defects and failures. The parts' production process on a SLS machine has been monitored over a determined period of time. The overall defects and failures occurred over the screening period have been logged as well as their associated *Potential Quality Failure Causes* and effects. Thus, the occurrence frequency of the *Potential Quality Failure Causes* and their associated inline defects and failures has been established. For each defect and failure logged over the screening period, the sunk costs in percent of one build cycle have been determined. For the sunk costs' factor the following data have been used: the year turnover of a fully loaded SLS production machine (newest generation) is about 1 Million €, calculated over an average of: 100 builds per year, with 40 parts per build and a part price of 250€; one build having a value of about 10.000€.

The ranking of the top inline defects, to be identified by the inline QC system, is presented in Table 4.6. The criteria for the ranking position were established taking into consideration two factors: the normalised value of the frequency of defects in percent of build cycles ( $F_1$ ), and the normalised value of the implicated sunk costs, in percent of one build cycle turnover ( $F_2$ ). The ranking scores  $R_S$  have been determined using the following function:

$$R_S = (w_1 * F_1 / F_{1max}) + (w_2 * F_2 / F_{2max}) \quad (4.1)$$

where  $w_1 = 2$  and  $w_2 = 1$  are the assigned weights, and  $F_{xmax}$  are the maximal values of the factors.

The interval of ranking scores has been split up in three. The ranking position values from Table 4.6 represent the interval, to which the associated ranking score belongs.

*The determined future tasks for an inline QC system for the SLS processes are:*

- to identify the ranked inline defects and failures presented in Table 4.6, and
- to collect, during the additive process, all quality information in a part protocol.

Table 4.6. Ranking of the top inline defects and failures to be identified by the QC system.

Ranking position	Frequency of defects in percent of build cycles	Sunk costs in percent of one build cycle turnover	Quality Failure Causes, logged over the screening time	Inline defects to be identified by the inline QC system ⇔ tasks of the inline QC system
1	<10%	≤100%	deposits on laser window	<i>inappropriate layers' adhesion, geometrical layer and part distortions, deposits on laser window</i>
2	<15%	<25%	impurities	<i>impurities in layer</i>
1	<10%	≤100%	laser and optical system	<i>inappropriate layers' adhesion, geometrical layer and part distortions, melting of the edge zone depending on the geometry, porosity</i>
2	<2%	<90%	temperature control	<i>inconstant temperature</i>
2	<3%	<50%	inert gas supply	<i>melting of the part, black sintered</i>
1	<5%	100%	wear parts: material supplier	<i>streaky, scaly powder layer</i>
1	<5%	100%	wear parts: tilted build platform	<i>tilted layer geometry</i>
1	<5%	100%	wear parts: build platform get stuck or lose the high information because defects of the stepper motor	<i>Layers' overlapping, layers' sintering failed</i>
1	<7%	≤90%	powder application performance: inhomogeneity of the generated powder layer	<i>inhomogeneity of the layer thickness</i>
2	<5%	<60%	powder application performance: density variation of the powder layer	<i>density variation of the powder layer</i>
3	<1%	<50%	scaling: scaling factor variations	<i>geometrical deviations and distortions of the sintered layer</i>

**62** Tasks for inline QC and in-situ optimisation of AM PBF processes – 4

3	<1%	<50%	tolerances: inadequate tolerances	<i>geometrical deviations of the sintered layer</i>
1	<20%	≤100%	improper beam offset	<i>geometrical deviations and distortions of the sintered layer</i>
2	<8%	≤80%	low scan speed	<i>porosity, inhomogeneity</i>
1	<20%	≤75%	surface roughness	<i>layer and part surface roughness</i>
1	<10%	≤90%	grain shape and grain size distribution	<i>different grain sizes, grain shapes levels</i>
1	<20%	≤100%	thermal properties like melting point and recrystallization	<i>inconstant temperature on sintering point, inappropriate layers' adhesion</i>
2	<3%	≤75%	type and mesh size of sieve	<i>different grain size, inhomogeneity of the powder layer</i>
3	<3%	≤25%	part orientation	<i>part and layer orientation failure</i>
1	<20%	≤100%	laser power	<i>inappropriate layers' adhesion, geometrical layer and part distortions, melting of the edge zone depending on the geometry, layer porosity</i>
1	<2%	≤100%	scan, sintering speed	<i>inconstant sintering speed</i>
1	<2%	≤100%	scan, sintering line	<i>scan line deviation</i>
1	<20%	≤100%	temperature profile	<i>inconstant temperature on sintering point, inappropriate layers' adhesion</i>
2	<2%	≤90%	layer thickness	<i>variation of layer thickness , inhomogeneity</i>
3	<1%	≤25%	hatch distance	<i>inappropriate hatch distance, inappropriate layers' adhesion</i>
2	<3%	<70%	atmosphere: burning of the layer	<i>melting of the part, black sintered</i>
3	<1%	<50%	skywriting	<i>skywriting length variations, melting of the edge zone, geometrical layer and part distortions</i>

## 5. OVERALL CONCEPT FOR AN INLINE QC SYSTEM IN THE AM PBF PRODUCTION

In order to be able to develop a concept for a hardware and software platform for an inline QC system [31], respectively for an in-situ optimisation system, it is necessary to classify the ranked tasks (see Table 4.6), taking into account the AM PBF production steps. The results for the SLS processes are presented in Table 5.1.

Table 5.1. Ranked tasks during the AM PBF production steps: basis for a firm foundation of a hardware and software platform for an inline QC system and a future in-situ optimisation.

AM PBF production steps	Tasks (ranking position)
During all steps	deposits on laser window (1)
	inconstant temperature (2)
After each powder layer application	streaky, scaly powder layer (1)
	layer porosity (2)
	variation of layer thickness (2)
	density variations of the powder layer (2)
	impurities in powder layer(2)
	different grain size (2)
During sintering process of the powder layers	inconstant temperature on sintering point (1)
	melting of the edge zone depending on geometry (1)
	inconstant sintering speed (1)
	scan line deviation (1)
	melting of the part, black sintered (2)
	check the hatch distance (3)
	skywriting length check (3)
After sintering of each powder layer	layers' overlapping, layers' sintering failed (1)
	inappropriate layers' adhesion (1)
	layer and part surface roughness (1)
	geometrical deviations and distortions of the sintered layer (1)
	layer porosity (2)
	layer/part orientation within the build volume, only after sintering the first layer (3)
After part is finished sintered	geometrical deviations and distortions of the part (1)
	porosity analysis of the part over all layers (1)
	part surface roughness (1)

### 5.1. Hardware platform

The hardware (HW) platform must be a modular one, in order to solve all of the inline QC system tasks identified. The modularity of the HW platform will allow in the future:

- to easily add other sensors in order to solve new quality issues that will appear with the development of the AM PBF machines,
- to easily adapt the system to other AM PBF processes,
- to allow a data fusion of the sensors on the Software (SW) side of the future system.

#### 5.1.1. Appropriate technologies and sensors for data and signal acquisition

The appropriate technologies, respectively the sensors' requirements for the multi-sensor analysis and field monitoring, have been identified regarding the identified tasks during the AM PBF production steps (see Table 5.1) and the best price/performance ratio. For each task, the most appropriate technologies and sensors have been exemplarily identified. The results are presented in Table 5.2.

Table 5.2. Technologies and sensors necessary for solving the overall tasks – exemplary.

AM PBF production steps	Inline QC system tasks (ranking position)	Technology for the HW platform	Machine Vision sensors for the data and signal acquisition
During all steps	deposits on laser window (1)	2D Image Processing	camera->visual spectral sensitivity
	inconstant temperature (2)	Temperature measurement	pyrometer and or camera -> infrared spectral sensitivity
After each powder layer application	streaky, scaly powder layer (1)	2D Image Processing	camera->visual spectral sensitivity
	layer porosity (2)		
	variation of layer thickness (2)	Active Thermography	camera -> infrared spectral sensitivity
	density variations of the powder layer (2)		
	impurities in powder layer(2)	2D Image Processing and/or Passive Thermography	camera in visual and/or infrared spectral sensitivity
different grain size (2)	2D Image Processing	camera->visual spectral sensitivity	
During sintering process of the powder layers	inconstant temperature on sintering point (1)	Passive Thermography	camera -> infrared spectral sensitivity
	melting of the edge zone depending on geometry (1)	2D Image Processing	camera->visual spectral sensitivity
	inconstant sintering speed (1)	Passive Thermography	camera -> infrared spectral sensitivity
	scan line deviation (1)		



During sintering process of the powder layers	melting of the part, black sintered (2)	2D Image Processing	camera->visual spectral sensitivity
	check the hatch distance (3)	Passive Thermography	camera -> infrared spectral sensitivity
	skywriting length check (3)		
After sintering of each powder layer	layers' overlapping, layers' sintering failed (1)	2D Image Processing	camera ->visual spectral sensitivity
	inappropriate layers' adhesion (1)	Lock-in Thermography	camera -> infrared spectral sensitivity
	layer and part surface roughness (1)	2D Image Processing	camera ->visual spectral sensitivity
	geometrical deviations and distortions of the sintered layer (1)		
	layer porosity (2)		
	layer/part orientation within the build volume, only after sintering the first layer (3)		
After part is finished sintered	geometrical deviations and distortions of the part (1)	3D Image Processing based on e.g.3D reconstruction from multiple 2D images	camera->visual spectral sensitivity
	porosity analysis of the part over all layers (1)		
	part surface roughness (1)		

Specific for a SLS machine (Fa. EOS), in Table 5.3, all technologies are grouped in necessary system modules, in order to solve the classified tasks. Three Inline Quality Control System Modules (IQCSM) are necessary, in order to cover the overall tasks [31]. For exact requirements of the IQCSM components the inspected field, the working distance, and the identified tasks must be grasped.

Table 5.3. Inline Quality Control System Modules - exemplary for a SLS machine.

IQCSM no.	Sensors for the data and signal acquisition	Tasks	IQCSM $n$ components
1	Machine Vision System	<ul style="list-style-type: none"> <li>• must acquire <i>during all AM production steps</i> an image of the laser window as basis for <b>the laser window clean check</b></li> </ul>	industrial image processing camera (visual spectral sensitivity), lens, appropriate illumination <i>Obs. must be integrated in the machine, therefore needs a cooled housing</i>

2	Machine Vision System	<ul style="list-style-type: none"> <li>• must acquire <i>after each powder layer application</i> at least one image of the powder layer, for the inspection of: <b><i>streaky, scaly powder layer; layer porosity; impurities in powder layer and the powder grain size</i></b></li> <li>• must assure <i>during the sintering process</i> a "live stream" of the sintering process for the inspection of: <b><i>melting of the edge zones; melting of the whole layer "black sintered"</i></b></li> <li>• must assure <i>after sintering of each powder layer</i> at least one image of the sintered layer as basis for the inspection of: <b><i>layers' overlapping; geometrical deviations and distortions of the sintered layer; porosity; layer/part orientation within the build volume, only after sintering the first layer</i></b></li> <li>• must assure <i>after the whole part is finished</i> all the images corresponding to all sintered layers of the part in order to reconstruct from all the 2D images the 3D Real Model as basis for the 3D inspection of: <b><i>geometrical deviations and distortions of the part on 3D level; porosity analysis of the part over all layers; part surface roughness</i></b></li> </ul>	<p>industrial image processing camera (visual spectral sensitivity), lens, appropriate illumination</p> <p><i>Obs. If the camera is integrated in the machine needs a cooled housing; else must work through a dedicated window.</i></p>
3	Thermography system	<ul style="list-style-type: none"> <li>• must assure <i>after each powder layer application</i> the necessary images for a Lock-in Thermography for the inspection of: <b><i>homogeneity of the layer thickness; density variations of the powder layer; impurities in powder layer</i></b></li> </ul>	<p>infrared camera, appropriate excitation source</p> <p><i>Obs. If the camera is integrated in the machine needs a cooled housing; else must work through a dedicated window.</i></p>

	Thermography system	<ul style="list-style-type: none"> <li>• must assure <i>during the sintering process</i> of the powder layers a "live stream" of images of the sintering process as basis for the inspection of:  <b>temperature variation on the sintering point, temperature distribution; scan speed variation; scan line deviation; check the hatch distance and skywriting length check</b></li> <li>• must assure <i>after sintering of each powder layer</i> the necessary images for a Lock-in Thermography of the sintered layer for the inspection of:  <b>inappropriate layers' adhesion</b></li> </ul>	infrared camera, appropriate excitation source <i>Obs. If the camera is integrated in the machine needs a cooled housing; else must work through a dedicated window.</i>
--	---------------------	--	---

### 5.1.2. Hardware architecture design: concept and interfaces

The hardware architecture is a modular one. New hardware modules can be added to the inline QC system at any time in order:

- to adapt the system to other AM PBF machines, and
- to solve new inline quality tasks.

The hardware concept and the corresponding interfaces [31] are presented in Figure 5.1. Between the sensors identified and defined in chapter 5.1.1, respectively all the other sensors necessary for the system implementation (e.g. emergency stop button, machine door open), and between the PC, the communication is based on a Gigabit Ethernet (GigE) interface. The connection for the communication to the machine is established over a Programmable Logic Controller (PLC) based on PROFINET protocol and the connection for the Sensors (e.g. Cameras) with a GenICam (Generic Interface for Cameras) protocol. Both are GigE based protocols.

GigE compatible sensors will be connected directly to the ports of the GigE-Network Boards. The boards will have Power over Ethernet (POE) support. In this way the sensors with POE support (e.g. the machine vision systems sensors) will be supplied with power directly over the standard Ethernet cables. A big advantage for the inline system: the data communication and the power supply will be assured only over one cable.

The PLC, a digital computer used for the automation of the inline QC system processes, can be connected over one port of a GigE Board or directly to a PROFINET Board. A PROFINET Board allows at least 90% of the PROFINET stack to work directly on the board.

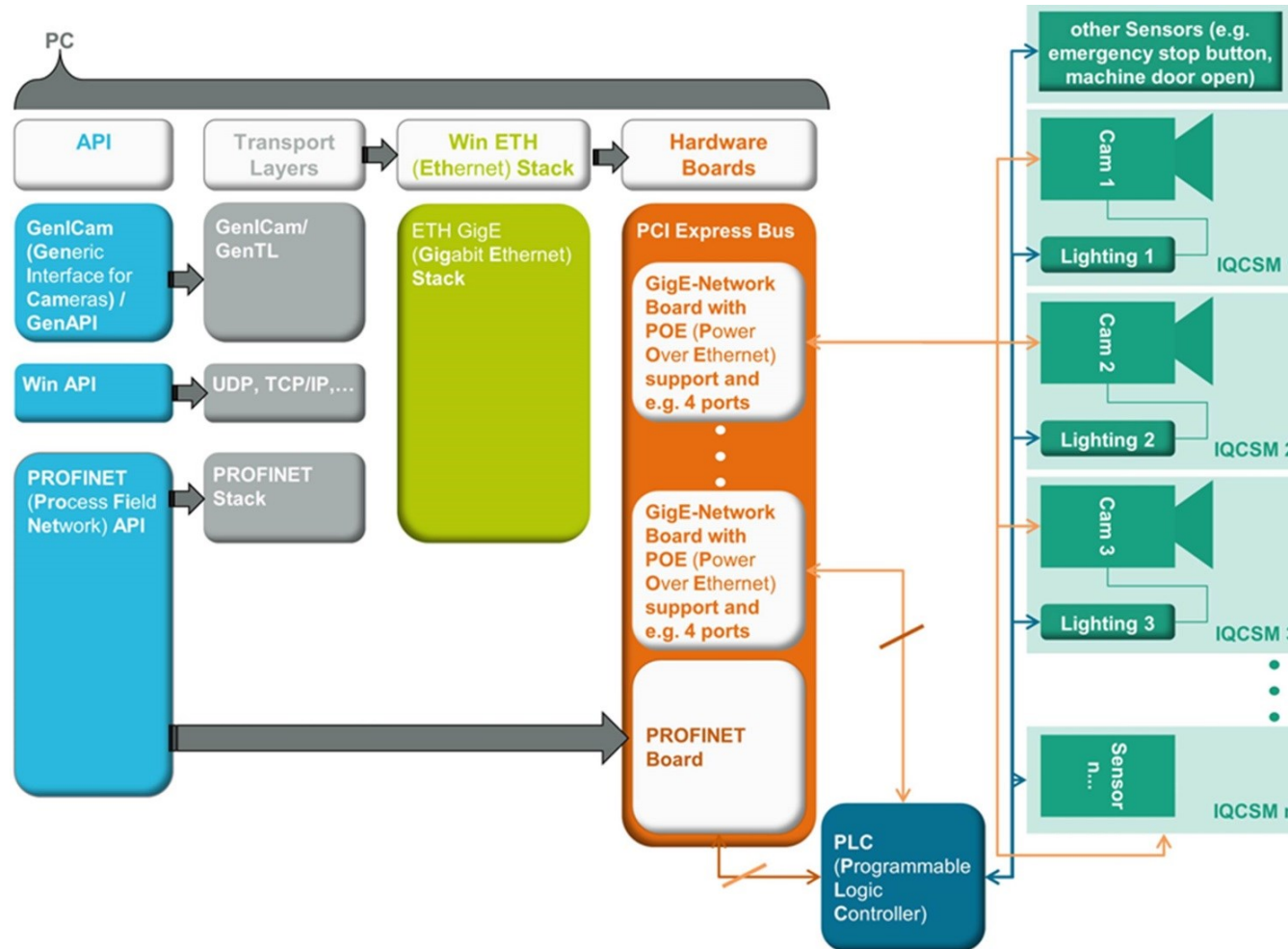


Figure 5.1. Overall hardware concept of the modular inline QC system for AM PBF.

## 5.2. Software platform

A dedicated common software (SW) framework, the AM PBF Inline Quality Control Platform (AM-IQCP), was conceptual developed based on [63]. AM-IQCP, emphasising an automatic data evaluation platform, includes different modules for the different evaluation strategies. All suitable interfaces, algorithms (the ones specifically for each technology as well as the ones corresponding to the data fusions), together with their complex functionality, can be then modular implemented under the AM-IQCP framework (Figure 5.2).

### 5.2.1. Concept

The AM-IQCP is a modular software platform, which will integrate the complex functionality of all the developed algorithms and routines as follows:

- the ones specifically for each technology used as described in chapter 4.1, and
- the ones corresponding to the data fusions.

The functionality of the platform is controlled from a user friendly Graphical User Interface (GUI). The user can start and parametrise the inline QC system; can choose what kind of the system parts will be activated; can visualise the results of the data analysis from each technology or as data fusion.

The AM-IQCP application is developed under Microsoft Visual Studio (Visual C++) with Qt (32-bit and 64-bit), following a V-model software development process. Designed as a modular software framework, AM-IQCP allows the integration of different plug-ins and libraries at any time and the separation of the functionality of the platform into independent, interchangeable modules (Figure 5.3).

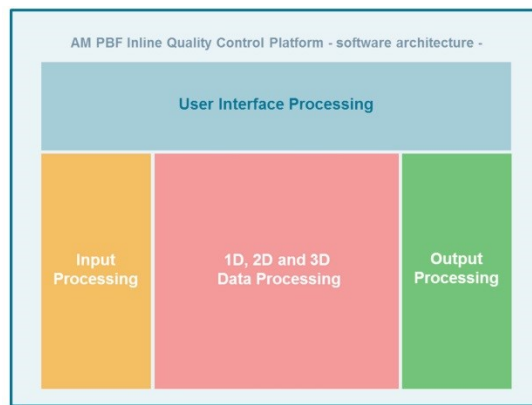


Figure 5.2. AM-IQCP: general SW architecture.

Data from different sensor technologies, through the modules corresponding to the Input Data Interface, can be transferred to the Data Processing block. After the data pre-processing or data handling operations (e.g. scaling, coordinate systems operations), the data is analysed in the modules corresponding to each data processing: 1D, 2D or 3D. The results of these operations can be then:

- directly transferred into the Automatic Evaluation Module, or
- first in the Data Fusion Module and just after the fusion, operations are executed in the Automatic Evaluation Module.

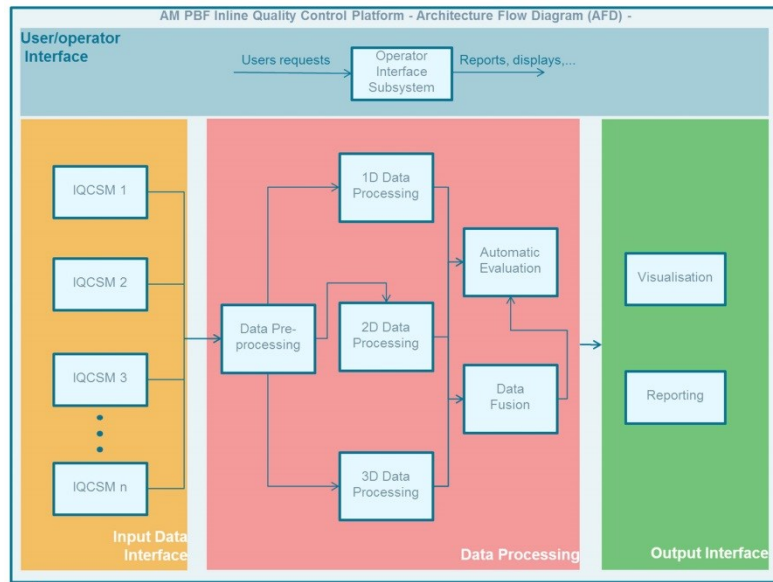


Figure 5.3. AM-IQCP: SW architecture flow diagram.  
(IQCSM - Inline Quality Control System Modules, see 5.1.1)

Concerning the Visualisation Module, the input can be the output from each module of the Data Processing block. In this way the user has the possibility to visualise the data after different data processing modules, corresponding to the data pre-processing, data processing, data fusion and of course after the automatic evaluation operations. Through the Reporting Module, on user demand, detailed information can be retrieved about the results of the inline automatic inspection.

### 5.2.2. Modular Software Framework

The goal of the AM-IQCP software framework is to combine various data sources, from different sensors technologies, in order to allow the implementation of improved error detection algorithms. The challenge faced is to combine all the different data formats within one interface in order to allow e.g. one algorithm to work on several sources. Furthermore, the algorithm side should be highly extensible to allow later inclusion of developments. Since for an inline QC system the speed is crucial, C++ was chosen as programming language, to allow low-level optimizations of the software [63]. Object-oriented principles and design patterns should be used, taken into consideration their proven flexibility in real-world software [64].

The initial data sources, with which the framework has to deal with, are in principle (see Table 5.3) 2D, 3D machine vision data and thermography data. As one can see, the big difference between these data is the dimensionality. The framework must be able to work with both, two-dimensional and three-dimensional data. Therefore, the designed framework is split into three main components: data (data handling and processing), core (utility and configuration classes) and report (generation of reports). A complete overview of the whole classes and the interaction between them is shown in Figure 5.4; a detailed view in Figure 5.5 and Figure 5.6.

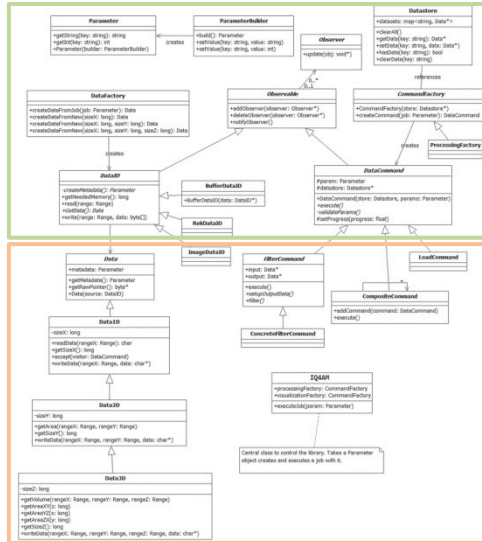


Figure 5.4. Overview UML diagrams of the framework classes.

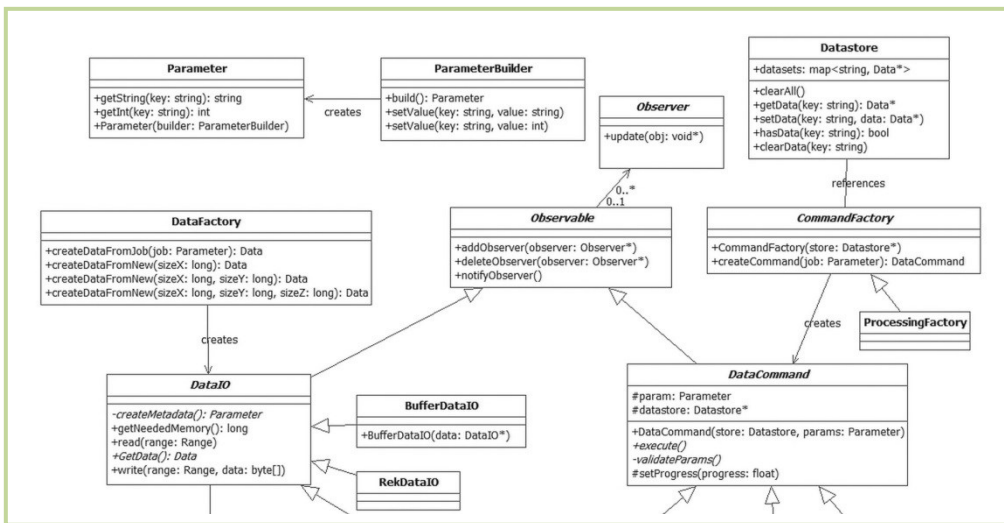


Figure 5.5. Upper part of the UML diagrams of the framework classes.

The IQ4AM class, inherited from the IQ4AMInterfacingClass, is the central interface in order to interact with the framework. It needs a Parameter object that describes what operation should be done. After receiving this object, a suitable command is created and executed. The core contains some utility methods and the configuration classes. Most important here is the ParameterBuilder used to configure and create a Parameter object. With these objects, interactions with the data processing engine itself are possible. This helps to decouple the application from third party programs, which wish to interact with the data processing core and keeps the public exposed application programming interface (API) small and stable.

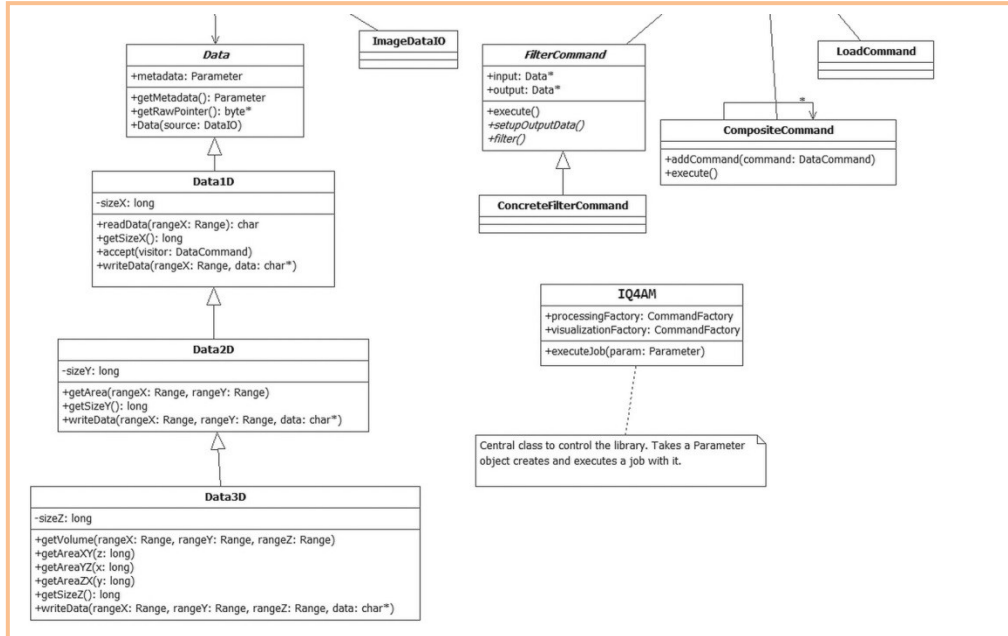


Figure 5.6. Lower part of the UML diagrams of the framework classes.

*Configuration and Control:* in order to control the engine, commands can be sent via the Parameter objects. These objects hold a key-value representation of the configuration information. Since the IQ4AM object accepts these parameters and builds commands out of it, a highly decoupled solution is found. The programmer, who wants to interact with the engine, can set whatever configuration he likes; the commands would pick this information and perform their work. To simplify the creation of these Parameter objects (see Figure 5.7), a builder pattern was used, which returns references to the builder itself. Thus, a procedure called *method chaining* is possible.

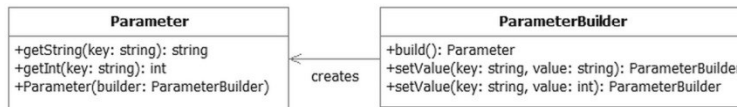


Figure 5.7. A Parameter object created from a ParameterBuilder, by using the builder pattern.

*Data Handling:* two design goals were identified during the design phase of the software framework: first, the engine has to support different formats of data: ranging from 1D, over 2D, to 3D volumetric data. In the same time, the data sources of these formats could be different and incompatible with each other. Therefore, the best way is to separate the actual reading of the data format from the internal API. With this extra layer of abstraction, it is possible to change the underlying data format without altering the rest of the program. All of the raw data handling work is encapsulated and hidden away in concrete implementations of the DataIO class. Since the classes that interact with the raw data know which API data they want to express (1D, 2D or 3D), they are responsible for creating these Data objects on their own. The DataIO objects hide themselves behind this facade and returns only the facade. In this way, the rest of the library is completely



independent of the underlying data format. All of these operations are designed to be done inside the DataFactory class.

*Commands and processing:* the designed implementation of these operations implies two classes: the DataCommand and the FilterCommand one (see Figure 5.8). Commands are used to interact with the data in any way. The DataCommand class header describes the interface of this type of classes and implements fundamental functions in order to work with the SW engine. Extending the SW engine is as easy as implementing further sub-classes of DataCommand and extending for e.g. the CommandFactory header file so that the new command will be recognised. Each command has access to the DataStore from which the loaded data can be accessed to do calculations with it or start external algorithms. The FilterCommand class header extends the command and provides some additional functionality in order to interact with the data. Another abstract method was implemented, which is responsible for setting up the output data, since all the data in the DataStore is read-only. Even though, the data is still available, so that the algorithm itself can decide which format will be used for the output data. The most common change in data type is if an algorithm extracts a 2D image from volumetric 3D data and so a new Data2D object must be created in order to hold the new image information.

*Reporting:* the report component is quite compact but easy to extend to own needs. It contains model classes to represent the report in a tree-like data structure and a decoupled formatting component to transform the generated tree structure into the desired final documentation format.

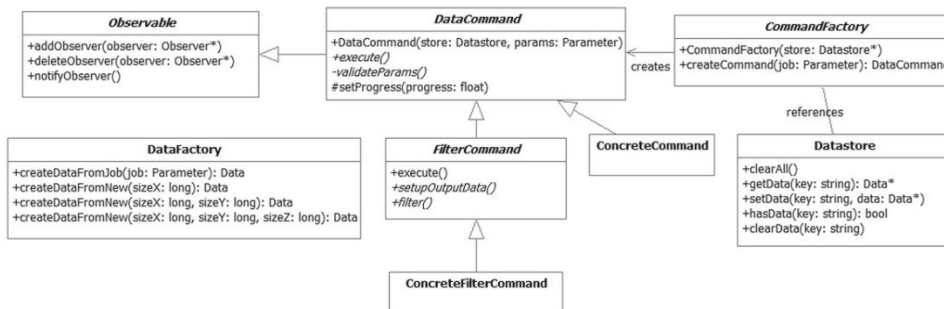


Figure 5.8. Detailed view of the DataCommand and FilterCommand class hierarchy.

**Conclusion:** based on the identified tasks (see chapter 4), an overall concept for an inline QC system in the AM PBF production was worked out as a novelty<sup>1</sup>.



## 6. INLINE QC SYSTEM COMPONENTS

*Novelty*<sup>1</sup>. Following the overall concept compiled in chapter 5, a HW setup, which ensure a firm foundation of an inline QC system, and a SW engine with two main IIP routines, the *Inline Image Acquisition*, and the *Quality Control* have been designed and developed, using the example of one AM PBF process, namely the SLS one.

For the development, implementation and validation of the modular inline QC system components the US National Institute of Standards and Technology (NIST) test artefact has been used, as it was developed by NIST in order to be standardised [65]. A schematic of the current version of this artefact is shown in Figure 6.1.

### 6.1. Experimental hardware platform

*Novelty*<sup>1</sup>. The inline QC system hardware platform is machine-independent and has a clear-cut attractive price/performance ratio for all users of PBF technologies. The developed hardware (HW) platform is modular. The modularity of the machine-independent implemented HW platform allows:

- to easily add other sensors in order to solve new quality issues that will appear with the development of the SLS or any other PBF machines,
- a future multi-sensors data fusion on Software (SW) side.

Starting from the HW concept, respectively from the appropriate technologies and sensors for the data and signal acquisition necessary to cover all the ranked tasks, the HW implementation concentrated on one inline QC system module, the IQCSM 2 (see Table 5.3), as conceptual presented in chapter 5.1. The research work concentrated on the IQCSM 2. Its' successful implementation can cover the most ranked tasks determined in chapter 4, as in Table 6.1 presented.

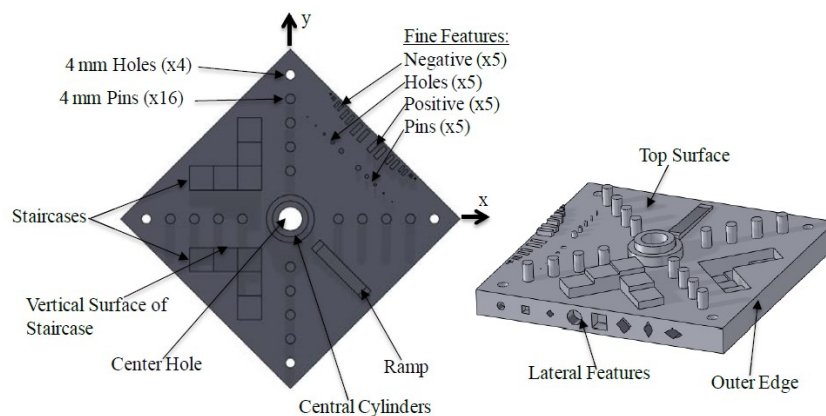


Figure 6.1. NIST test artefact.

Table 6.1. IQCSM 2 overall tasks.

AM PBF production steps	Tasks (ranking position)
After each powder layer application	streaky, scaly powder layer (1)
	layer porosity (2)
	impurities in powder layer(2)
	different grain size (2)
During sintering process of the powder layers	melting of the edge zone depending on geometry (1)
	melting of the part, black sintered (2)
After sintering of each powder layer	layers' overlapping, layers' sintering failed (1)
	layer and part surface roughness (1)
	geometrical deviations and distortions of the sintered layer (1)
	layer porosity (2)
	layer/part orientation within the build volume, only after sintering the first layer (3)
After part is finished sintered	geometrical deviations and distortions of the part (1)
	porosity analysis of the part over all layers (1)
	part surface roughness (1)

The development of the IQCSM 2, which is an AM PBF specific Machine Vision System, was conducted on a SLS machine, namely the EOS Formiga P100.

### 6.1.1. Technologies and sensors for data acquisition

The evaluated technologies and sensors necessary for the development and implementation of the hardware are:

- A **machine vision camera** with visual spectral sensitivity is fixed on the machine through a mechanical modular construction. The inspected field, in case of Formiga P100, is of 200 x 250 mm and the working distance is about 350 mm. This camera:
  - acquires *after each powder layer application* one image of the powder layer, as basis for the inspection of: streaky, scaly powder layer; layer porosity; impurities in powder layer and the powder grain size;
  - ensures *after sintering of each powder layer* one image of the sintered layer as basis for the inspection of: melting of the edge zones; melting of the whole layer "black sintered"; layers' overlapping; geometrical deviations and distortions of the sintered layer; porosity; layer/part orientation within the build volume, only after sintering the first layer;
  - ensures *after the whole part is finished* all the images corresponding to all sintered layers of the part in order to reconstruct from all the 2D images the 3D Real Model as basis for the 3D inspection of: geometrical deviations and distortions of the part on 3D level; porosity analysis of the part over all layers; part surface roughness.

*Hardware component:* Basler Low-Cost Camera acA3800-10gm (GigE camera with the Aptina MT9J003 CMOS sensor delivers 10 frames per second at 10 MP resolution, Power-over-Ethernet PoE); Basler PoE single port injector POE21U for GigE Vision.

- **Specific lens** in order to achieve the accuracy necessary for the quality tasks (e.g. real powder grain size used > 100 micrometres), in combination with the machine vision camera.

*Hardware component:* Basler Lens C125-0618-5M.

- Machine vision **lightings** in order to achieve the necessary image contrast for the failure analysis during the AM PBF production.

*Hardware components:* two SMD high density, rectangular illuminations: EVT EyeLight, model LBRX-00-080-3-W; two EVT corresponding diffusers; Analog Power Supply (two channels): ANG-2000-CH2-24V.

### 6.1.2. Hardware architecture

Based on the hardware concept and the corresponding interfaces, developed and presented in 5.1, the implemented hardware architecture is presented in Figure 6.2.

Between the sensors and the PC the communication is based on a Gigabit Ethernet (GigE) interface. The machine vision camera connection is established over a GenICam (Generic Interface for Cameras) protocol, a GigE based protocol.

The GigE compatible machine vision camera is connected directly over the ports of the GigE-Network Board. The board has Power over Ethernet (POE) support. In this way, the camera power supply works directly over the standard Ethernet cables. This configuration allows a flexible integration of the machine vision system into an AM machine, as the data communication and power supply are ensured only over one cable.

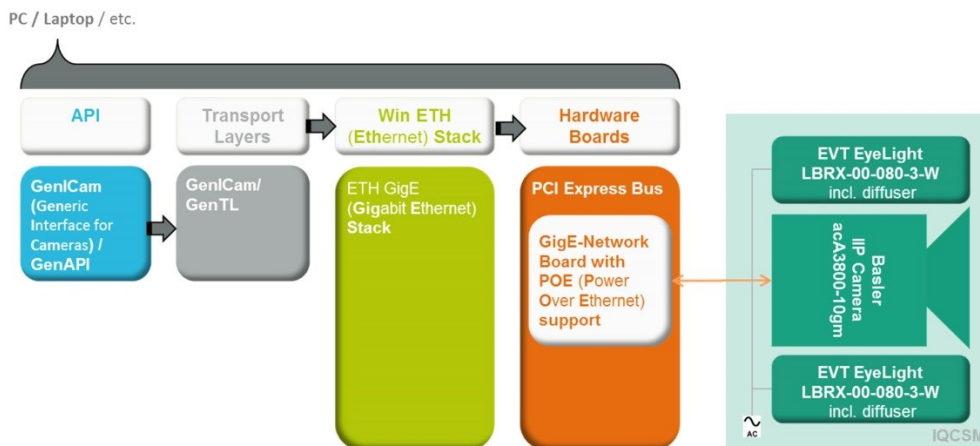


Figure 6.2. Hardware architecture and interfaces of the modular inline QC system for SLS with one IQCSM, the IQCSM2.

### 6.1.3. Hardware setup

The HW setup, as implemented directly on the EOS Formiga P100, is presented in Table 6.2. The basis of the construction is an aluminium modular profile system. The modularity is available at any time in any place. For each component (e.g. machine vision camera, lighting sources) special connectors have been developed, so that comprehensive adjustment possibilities for every sensor added are given (e.g. position in (x,y,z), angle).

*Novelty*<sup>1</sup>. The experimental HW setup is mounted on the machine and not into the machine. The advantages of this setup are considerable:

- The system is machine-independent: the system can be mounted on other AM PBF machines and through its modularity concept it can be easily adapted, respectively other sensors can be added at any time.
- The end price of the system is considerably reduced (over 70 %), as no cooled housings for the sensors and cable are necessary. Thus, not only a HW setup for an inline QC system for the AM PBF was achieved, but also a low-cost one, with a clear-cut reasonable price/performance ratio.
- The inline QC system can be used at any time by every AM PBF machine owners, without the approval of the AM PBF systems manufacturers. No HW interfaces of the AM machines must be open and no ISO standards certificates will be necessary for the integration into machines; the certification procedure being a very long, strongly machine-dependent, process.

Table 6.2. Experimental HW setup and components.

#### Lighting HW setup



#### Machine Vision Camera setup



**Final HW setup****6.2. Software engine**

Another component of the machine vision system, as critical as the image acquisition hardware system (e.g. camera, lighting), is the inline image acquisition and processing engine itself, that renders and communicates the final result.

*Novelty*<sup>1</sup>. The implemented SW engine of the inline QC system for reliable AM PBF processes, using the example of SLS, consist of two main IIP routines: the *Inline Image Acquisition* and the *Quality Control* one, as shown in Figure 6.3. The modularity of the platform easily allows the integration of new algorithms at any time. In this way, the HW modularity is strongly supported by the SW.

The development of the SW modules follows the architecture worked out in chapter 5.2. All algorithms have been implemented in C++, in form of plugins. A modular framework, the EyeVision3 framework, was integrated in order to assure a basic development environment (e.g. image memories are directly managed by the framework). The Open CV "imgproc" and "calib3D" modules have been integrated.

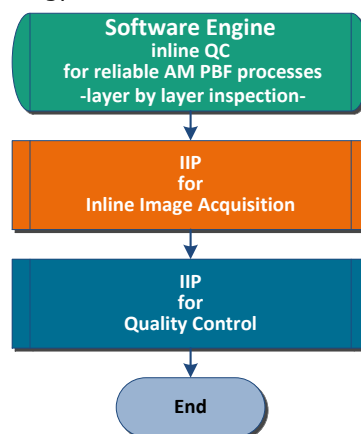


Figure 6.3. Implemented Software Engine.

### 6.2.1. Advanced IIP routine for data acquisition during the AM PBF manufacturing processes

The developed machine vision system is a machine-independent one. That means that the data acquisition process cannot be triggered by a machine signal, as there is no connection with the machine itself. The *scientific challenge* is, based on the live image information, to develop an IIP routine, which acquires the significant images at the right time.

*Novelty*<sup>1</sup>. For the AM PBF processes the image acquisition trigger is given by the developed software routine itself, namely by the *IIP for Inline Image Acquisition*.

The image acquisition will be triggered by the material supplier movement (see Figure 6.4):

- when the material supplier starts (runs into the image, right-to-left or left-to-right): a layer was finished, a new powder layer must be spread;
- when the material supplier stops (runs out of the image, right-to-left or left-to-right): a new powder layer was spread.

After a detailed evaluation of live images, the advanced IIP routine for data acquisition during the SLS process was implemented. (see Figure 6.5).

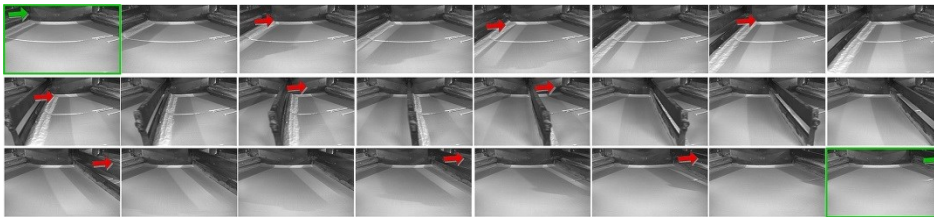


Figure 6.4. Exemplarily material supplier movement (left-to-right) – first and last image of this sequence is automatically acquired for the inline quality control.

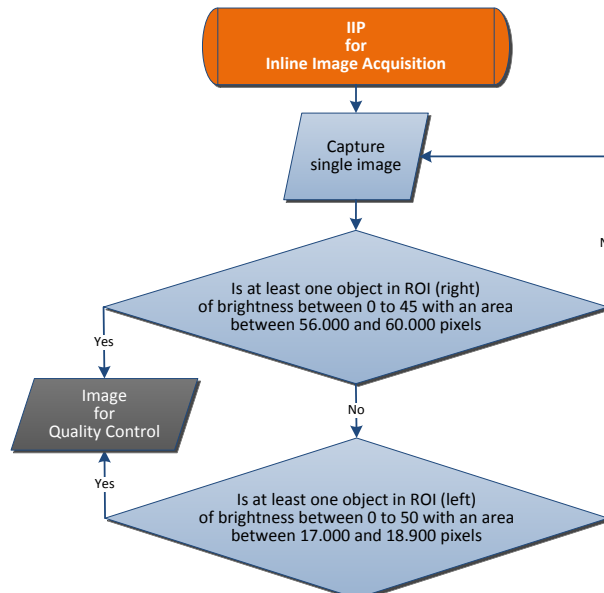


Figure 6.5. Software routine *IIP for Inline Image Acquisition*.



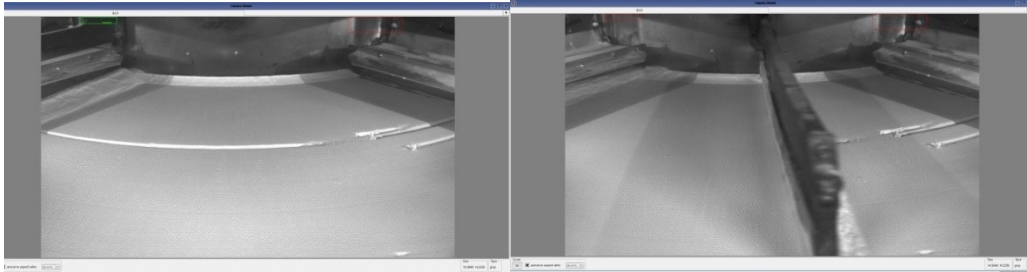


Figure 6.6. ROI's with blob analysis: triggers for image acquisition.

Two regions of interest, abbreviated ROI's were selected: one on the upper left side and the other one on the upper right side of the image field (see Figure 6.6). In these ROI's, a blob analysis was implemented in order to detect the material supplier status. When the material supplier "goes" through these regions, it will be identified through the blob analysis; the correspondent layer image at that moment of time is the input image for the quality control routine.

*Blob Analysis.* Connected pixels in a binary image represent a Binary Large Object. The more accurate existing connectivity criterion, the 8-connectivity, was considered [66]. The connected component analysis [67] have been implemented: suppose that  $I$  is a binary image and that

$$I(x; y) = I(x'; y') = t \quad (6.1)$$

where, either  $t = 0$  or  $t = 1$ . The  $(x; y)$  and  $(x'; y')$  pixels are connected in relation with  $t$  if there is a pixels sequence

$$(x; y) = (x_0; y_0), (x_1; y_1), \dots, (x_n; y_n) = (x'; y') \quad (6.2)$$

for which  $I(x_i; y_i) = t; i = 0, \dots, n;$  and  $(x_i; y_i)$  neighbours  $(x_{i-1}; y_{i-1})$  for each  $i = 1, \dots, n$ . A set of pixels  $J$  where each pixel has the value  $t$  and every set's pair of pixels has a connection with respect to  $t$  is a connected component [67].

After extracting the blobs, based on the connected component analysis, the blobs of interest corresponding to the material supplier have been extracted using the blobs area criterion.

### 6.2.2. Advanced IIP routine and dedicated algorithms for data evaluation during the AM PBF manufacturing processes

For the layer-by-layer inspection two kinds of images are analysed, corresponding to the powder layers and to the sintered layers. In both cases, quality control checks, as well for serious as for subtle failures, have been implemented. In case of sintered layers, measurements in world coordinates have been implemented in order to execute a dimensional control of the layers.

*Novelty*<sup>1</sup>. Quality issues during the AM PBF production process are inline detected. The user has the possibility to cancel the production process in extreme cases. The basis for a future in-process optimisation of the process is in the same time ensured.

The general concept for the *IIP for Quality Control* routine is presented in Figure 6.7. After an image distortion operation takes place, as the camera system is not parallel to the inspected field, the layer type is determined. If a powder layer has been identified, the system is checking for serious, wide failures. If at least one

failure was found, the inspection routine ends and the detected wide failures are logged. If none serious failure was found, the next step is to check for subtle failures. In this case, the failure size is at least three physical pixels. With the end of this subroutine, the inspection of the powder layers is finished. Now, if there is a sintered layer, the system will check first if serious or subtle failures are present. If none, than a comparison between the nominal and actual features of the sintered layer take place, based on the nominal layer information obtained from the STL machine input data. If the nominal and actual features are not different from each other than the system continues with the *Measure* procedure, where specific features will be inline measured. After the execution of each subroutine, the detected failures will be logged in the part quality protocol.

Taking into consideration the very low contrast of the images (Figure 6.8), the different range of failures that should be identified, with an accuracy of approx. 100 micrometres on the complete building platform (200 x 250 mm), the distinct contribution is hereby the resulted *IIP for Quality Control* routine, which solves at the same time all the tasks.

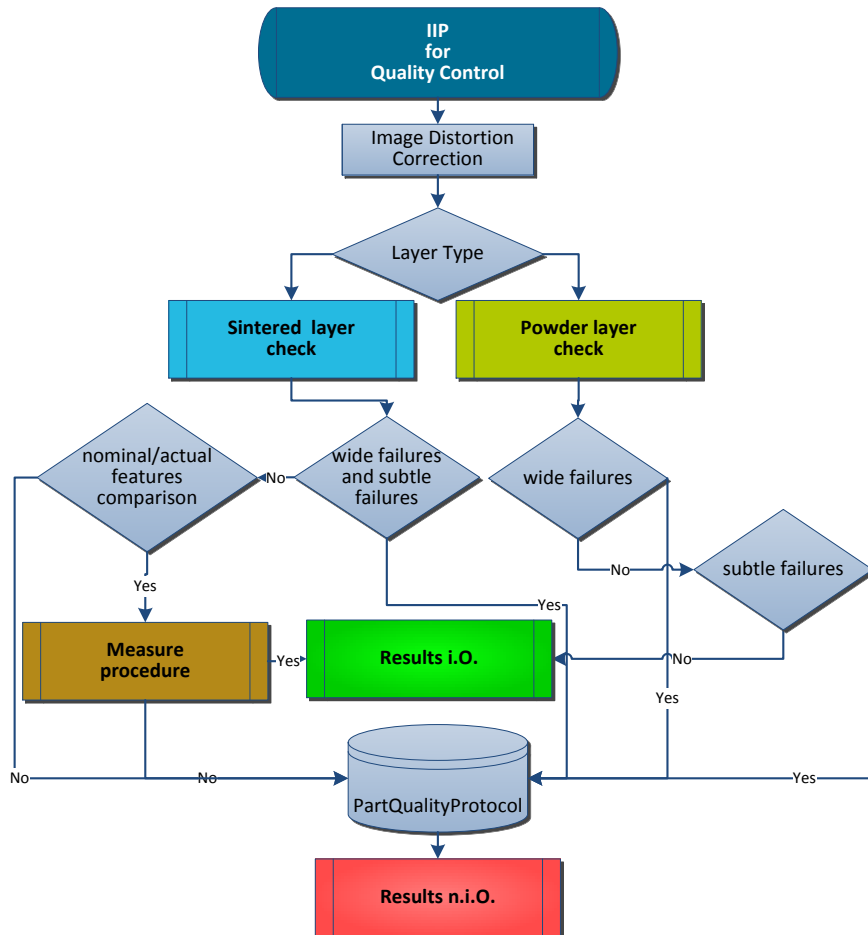


Figure 6.7. General concept for the *IIP for Quality Control* routine.

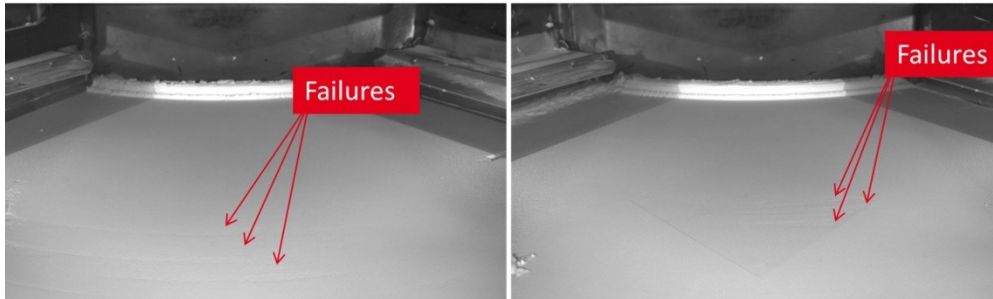


Figure 6.8. Low contrast images with subtle failures.

In order to ensure the inline QC of the manufactured parts it is necessary to separate the different image features (e.g. sintered layer from powder layer, failures in powder or sintered layer). As the contrast of these features is very low (see Figure 6.8), a research work over 10000 images was conducted, and the statistical result of the analysis, between the feature and non-feature area's values, resulted in a Gaussian distribution. Based on it, the brightness interval limits, for all developed algorithms over the subroutines presented in the following, have been set based on the Full Width at Half Maximum (FWHM) values. The values are not machine dependent, means that these obtained values can be used with the inline QC system also on other SLS machines type EOS Formiga P100.

The feature extraction dimensional criteria are depending on the build job. In order to set these criterions, the user, depending on the parts he want to build, has the possibility in the Graphical User Interface (GUI) to parametrise the main decisional algorithms (e.g. decisional blob extraction with area criterion for determination of layer type) (see subchapter 6.3 for details). For this, a catalogue with recommended values for the system users could be issued: e.g. values for parts with integrated functionality.

In the GUI, for the measurement tasks, the user has also the possibility to define the specific tolerances of the parts, in real world units. The subroutines presented in detailed have the parameters set for the NIST artefact test part (see Figure 6.1).

The first **subroutine** of the *IIP for Quality Control* routine, **from image distortion correction to layer type identification**, is presented in Figure 6.9. The initialisation of global variables takes place at the beginning of the main program (see subchapter 7.1.1):

- a. *nNominalSliceNr* - number of the nominal sintered layer, layer of the STL model to be sintered.
- b. *nBlobsNominal* - number of blobs extracted in the nominal image.
- c. *nBlobsActual* - number of blobs extracted in the actual image.
- d. *nFinalSliceCheckBlobResult* - difference between *nBlobsNominal* and *nBlobsActual*.
- e. *nLayerNumber* - actual layer number.

In this subroutine the image is loaded from the camera, the distortion corrections are done and after running the specific developed pre-processing sequence of algorithms, the layer type decisional evaluation takes place. At the end, if a powder layer has been identified then the subroutine goes further to the *quality control for powder layer* subroutine, else to the *quality control for sintered layers*. The detailed steps are presented in the following.

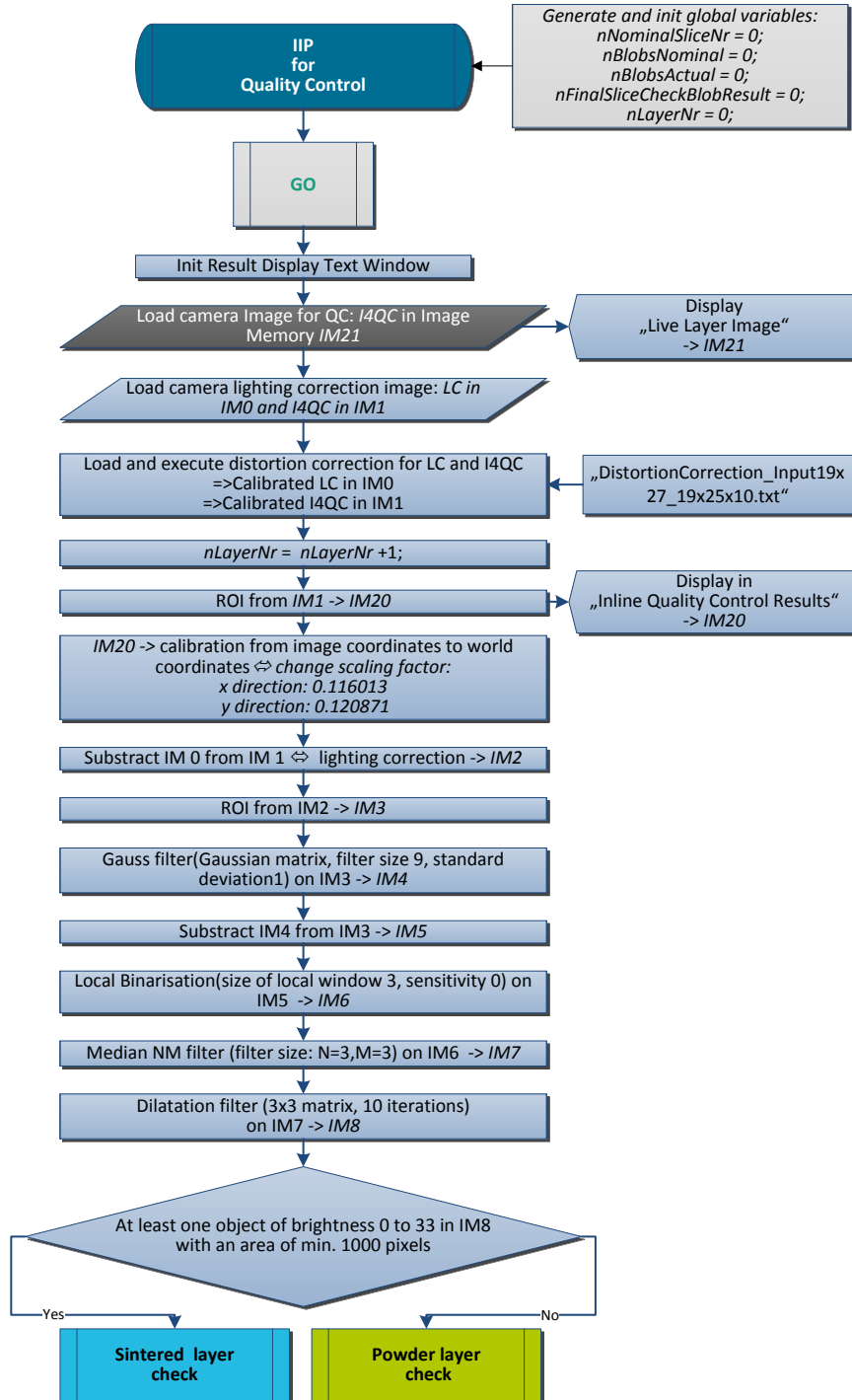


Figure 6.9. Subroutine - from image calibration to layer type decision.

**Subroutine steps:**

- I. *GO* is the mark where, after one layer was classified as i.O.(layer OK) or n.i.O. (layer not OK), the program returns for the inspection of the next layer.
- II. *Init Result Display Text Window* - initialisation of text window where the in between results are displayed for user.
- III. Load camera image for quality control (I4QC) in Image Memory (IM) 21, IM21 and display it.
- IV. Load camera lighting correction image (LC) in IM0 and I4QC in IM1 (see Figure 6.10).

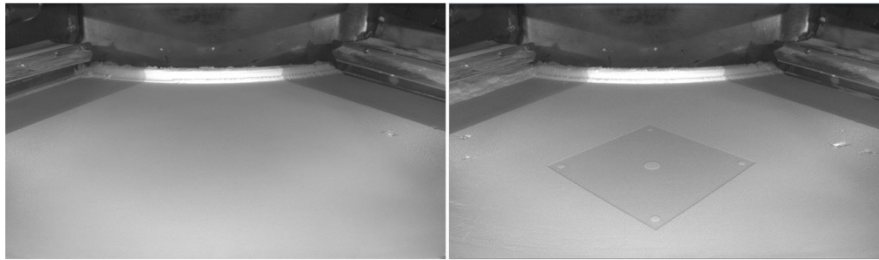


Figure 6.10. LC image (left) and an I4QC (right).

- V. Load and execute distortions correction for LC and I4QC  
*Distortion correction.* Because the camera lenses are not parallel to the imaging plane, a distortion correction [68], was implemented using the OpenCV library *calib3d*. The lens distortion modelling approach is based on the Brown-Conrady distortion model [69]. The relation between the camera pixels and the metric systems' units (e.g. millimetres) is also determined. A chessboard, as a well-defined pattern with 10mm distance between the cells, has been acquired by the camera. Finding the square corners of the chessboard (see Figure 6.11) and knowing the chessboard coordinates, as well as its coordinates in image, the distortion coefficients are calculated.



Figure 6.11. Square corners in chessboard, used for a part of the building platform, where the objects are manufactured.

At this point of the subroutine, the corresponding distortion correction file is loaded, and the LC and I4QC are undistorted. The operation takes place on the same area covered by the chessboard (see Figure 6.12), which is the area corresponding to the part to be build.



Figure 6.12. Undistorted area: I4QC in IM1 (left a sintered layer and right a powder layer).

- VI.  $nLayerNr = nLayerNr + 1$ ; - incrementing the layer number by one.
- VII. The ROI, corresponding to the sintered area (Figure 6.13), will be copied and displayed in IM20, which is the *Inline Quality Control Results* image, where the evaluations results will be displayed.

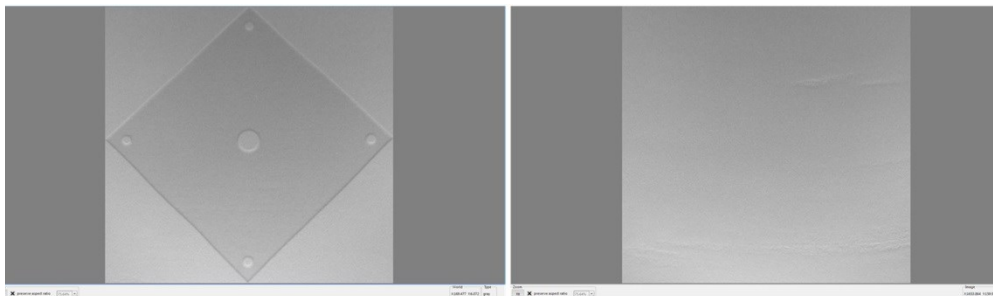


Figure 6.13. ROI corresponding to the inspected area (left sintered layer, right powder layer).

- VIII. IM20 -> calibration from image coordinates to world coordinates through the calculated scaling factors: in x direction  $0.116013$  and in y direction  $0.120871$ .  
The dimensions of the additive manufactured object are known: max. *width*= $141.42$  mm, max. *high*= $141.42$  mm. The ROI pixel dimensions are also known:  $w=1219$  pixels,  $h=1170$  pixels. In this way, the scaling factors can be achieved=> in x direction we have a scaling factor given by  $141.42/1219$  and in y direction given by  $141.42/1170$ .
- IX. The lighting correction is executed through an arithmetic filter: subtraction (see Figure 6.14). The result goes to IM3.

*Subtraction*: is a point operation method, based on simply image arithmetic. Here the LC image is subtracted [70] from the camera image at the inspection time, which can be corresponding to a sintered layer or a powder layer.



Figure 6.14. ROI - Subtraction results in IM 3 (left sintered layer, right powder layer).

- X. In order to accentuate the edges of the image, first a Gauss filter (Gaussian matrix, filter size 9, standard deviation 1) is applied on IM3. The result is saved in IM4. Then the IM4, the Gauss filtered image is subtracted from its original IM3. The result is saved in IM5. A local binarisation (size of local window 3, sensitivity 0) is applied on IM5. The result goes in IM6 (see Figure 6.15).

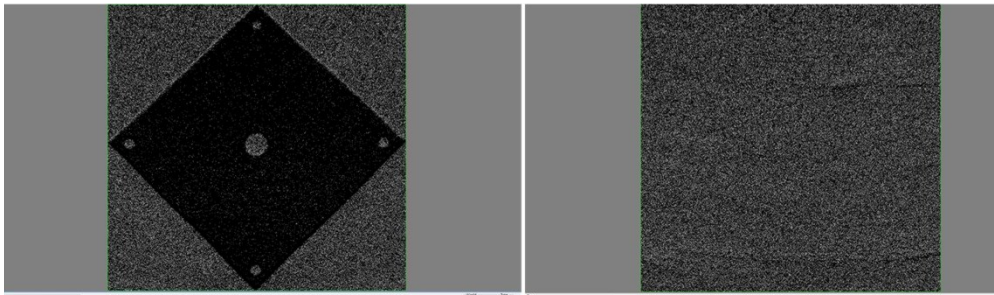


Figure 6.15. IM 6 - ROI after the edge sharpening and local binarisation operation (left sintered layer, right powder layer).

*Gauss filter*: is a linear filter algorithm used for image noise reduction. Parallel with good noise suppression an edges sharpening can be achieved. The Gaussian kernel in 2D is given by:

$$G_{2D}(x,y;\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (6.3)$$

where  $\sigma$  is the standard deviation [71].

*Local binarisation*: this filter creates a binary image. The implemented algorithm is a variation of the Bersen approach [72]. A local threshold is detected for each pixel within a given local  $n \times n$  window around the pixel. Dark in front of bright background becomes black, if the pixel value is at least "sensitivity" smaller than the median inside the local window. All other pixels become white.

- XI. A median NM filter (filter size:  $N=3$ ,  $M=3$ ) is applied on IM6 in order to eliminate the Salt-and-Pepper Noise and outliers; results in IM7.

*Median NM filter*: is a nonlinear averaging filter [73] that calculates the median of each pixel with a given  $N \times M$  matrix.

- XII. A dilatation filter ( $3 \times 3$  matrix, 10 iterations) is applied on IM7 in order to accentuate the brighter regions in image; results in IM8 (see Figure 6.16). In this way, we assure the correct detection of the sintered layer in the next step.

*Dilatation and erosion*: are neighbourhood operations (see subchapter 2.2.1) for binary images [73]. Dilatation operation dilates an image: bright regions are increased and darker regions reduced. In contrast, the erosion operation erodes an image: darker regions will be increased and brighter regions reduced.

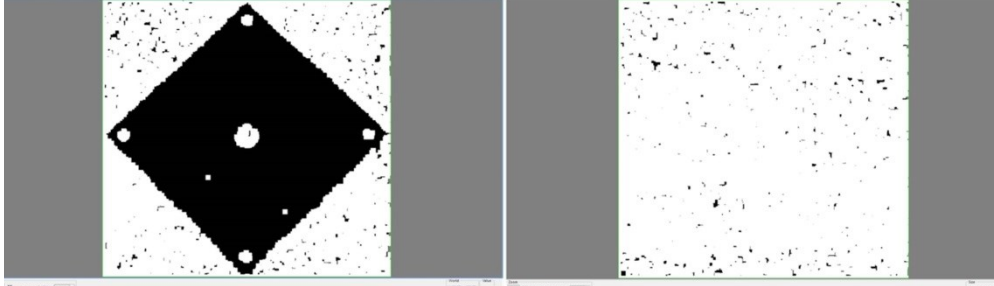


Figure 6.16. IM8 - ROI after median and dilatation subroutines steps (left sintered layer, right powder layer).

XIII. *At least one object of brightness 0 to 33 in IM8 with an area of min. 1000 pixels* ⇔ a blob extraction (see Blob Analysis, subchapter 6.2.1), with area as criterion for blobs classification.

If one dark object (sintered layer) with an area of min. 1000 pixel is found, the program goes to the *subroutine: quality control for sintered layers*, otherwise, a powder layer has been automatically identified and the program continues with the *subroutine: quality control for powder layers*.

**Subroutine: quality control for powder layers**

The powder layers check subroutine is shown in Figure 6.18.

In Figure 6.17 exemplarily, a wide failure in the powder layer is depicted.



Figure 6.17. Exemplary: powder layer with serious, wide failure.

First, a specific developed pre-processing algorithms sequence for serious, wide failures is running on the powder layer image in order to apply the decisional evaluation algorithm. If no wide failures have been found, than a specific developed pre-processing algorithms sequence for subtle failures runs on the original powder layer image in order to apply the decisional evaluation algorithm. If no failures, serious or subtle, have been found the quality control of the actual image is finished. The user is informed if the powder layer was i.O. or n.i.O., respectively all failures found will be logged in the part quality protocol. The detailed steps of this subroutine are presented in the following.



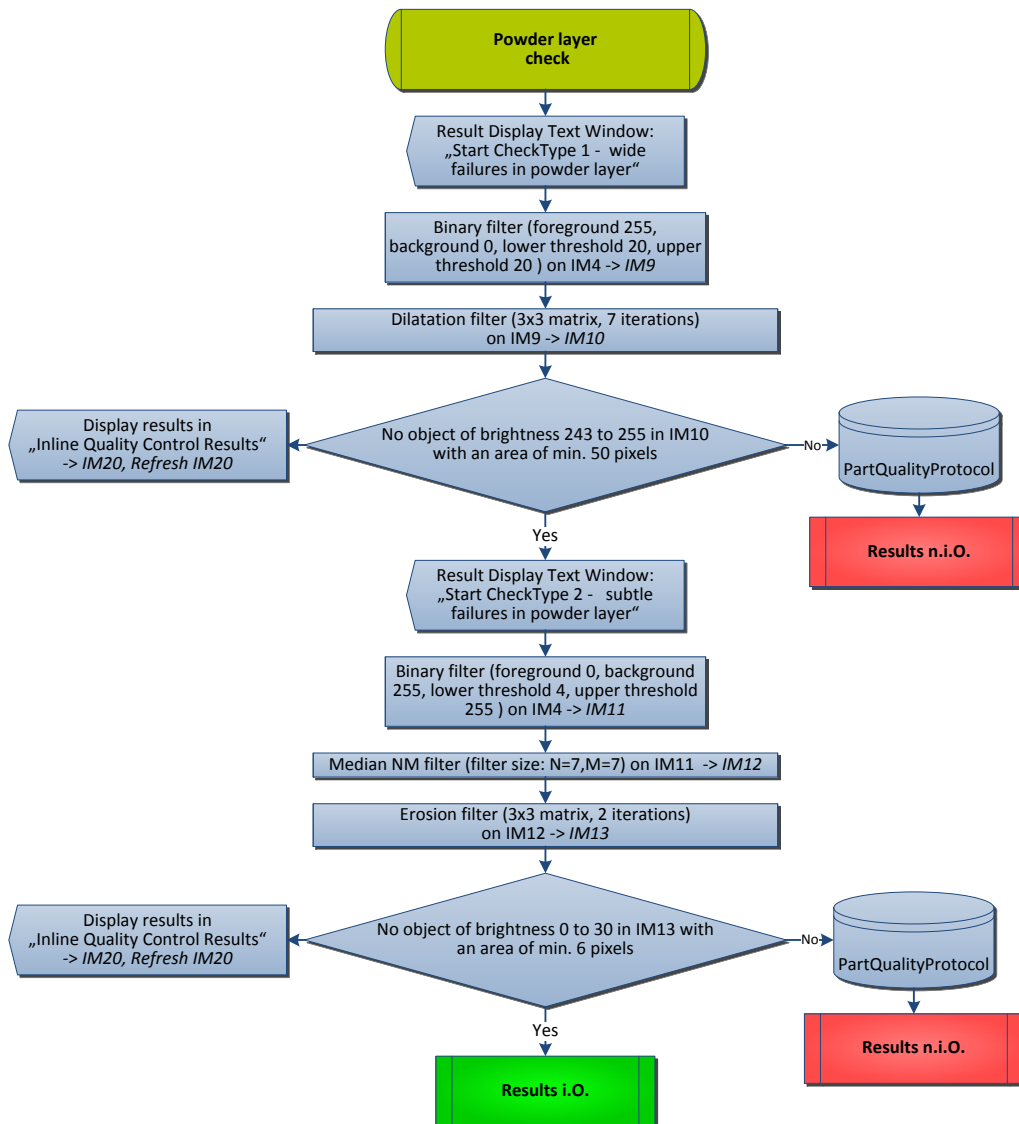


Figure 6.18. Subroutine - quality control for powder layers.

**Subroutine steps:**

- I. Message for user: "Start CheckType 1 - wide failures in powder layer".
- II. Binary filter (foreground 255, background 0, lower threshold 20, and upper threshold 20) on IM4, results in IM9. In this way, the possible wide failure structures will be separated from the rest of the image.

*Binary filter.* This filter does a binarisation of the image. Pixels with values above the upper threshold will be filled with foreground colour and pixels with values below the lower threshold will be filled with background colour.

- III. With the following dilatation filter (3x3 matrix, 7 iterations) applied to IM9, results in IM10 (Figure 6.19), the possible small gaps between the wide failures will be eliminated => if after step II wide failures structures were found, now these are well defined, closed, bright regions.

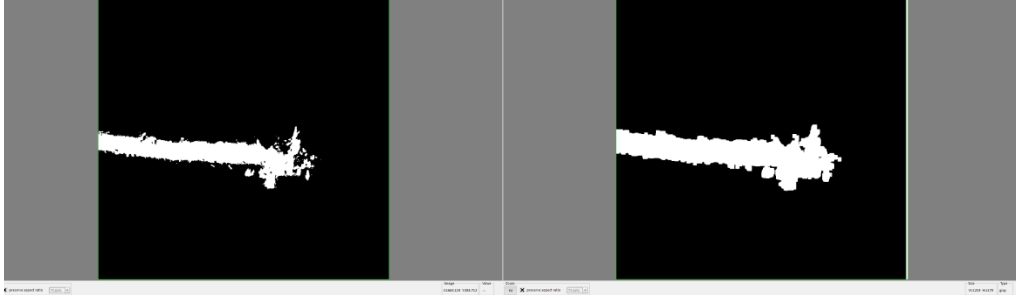


Figure 6.19. From left to right results of subroutine steps II (IM9) and III (IM10).

- IV. No object of brightness 243 to 255 in IM10 with an area of min. 50 pixels ⇔ blob extraction (see Blob Analysis, subchapter 6.2.1), with area as criterion for blobs classification. Display results in IM20.
- a. If no bright object with an area of min. 50 pixels was found (see Figure 6.20) then go to the check for the subtle failure and display the message for user: "Start CheckType 2 - subtle failures in powder layer". The next step is checking for subtle failures.



Figure 6.20. Displayed results in IM20 - no wide failures found.

- b. Else, at least one wide failure was found (see Figure 6.21). In this case, write the results in the *PartQualityProtocol* text file (see subchapter 6.2.3) and go to *Subroutine: Results n.i.O.*

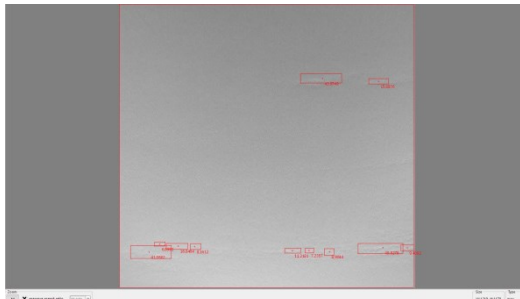


Figure 6.21. Displayed results in IM20 - subtle failures found.

**Subroutine: quality control for sintered layers**

The sintered layers check subroutine is presented in Figure 6.23.

In Figure 6.22 exemplarily, a sintered layer with a major failure is depicted.

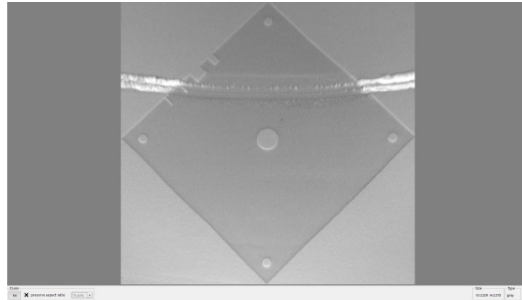


Figure 6.22. Exemplary: sintered layer with major failure.

For the QC of the sintered layers, in the first line the global string containing the path of the STL nominal layers (used as input for the SLS machine) will be reset and then filled out with the path corresponding to the sintered layer to be inspected. A specific developed pre-processing sequence of algorithms, for serious and subtle failures in the sintered layer, will run on the input image. Then, after the decisional evaluation algorithm runs, if failures have been found then other further implemented image algorithms are executed in order to display to the user the failures found. If no failures have been found, the reference slide image will be loaded (the nominal STL layer) from the path saved in the specific global string. Then in the nominal loaded image as well as in the actual sintered layer image, the blob number, representing the number of features existing in the layers will be determined. If these numbers are different, means that the actual features are different from the nominal ones and that the sintered layer has a quality problem, which will be displayed to the user, and logged at the same time. If the features determined in the nominal layer are identical with the ones in the actual layer, the *in-process measure procedure for sintered layers* subroutine is called. The subroutine steps are presented in detailed in the following.

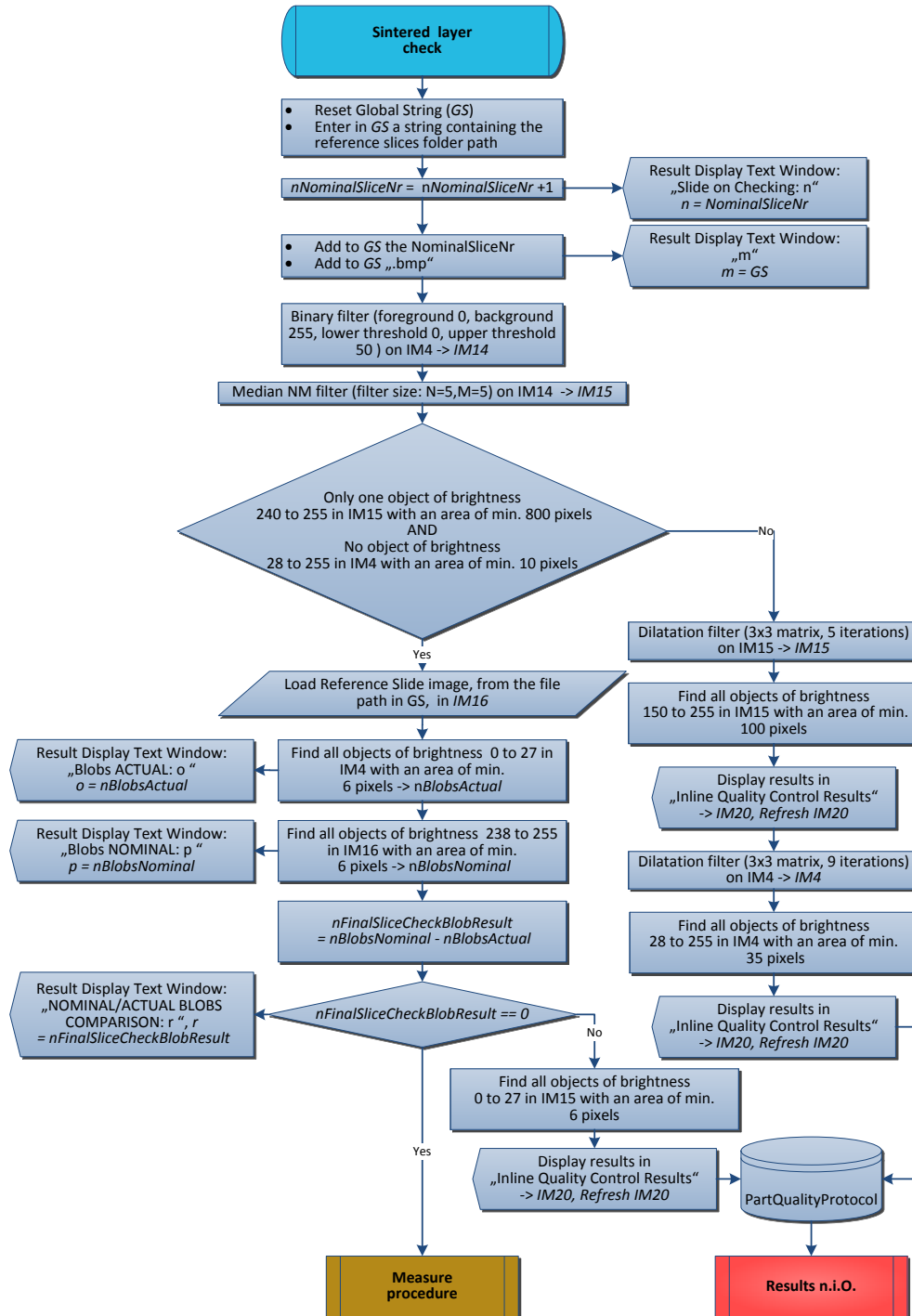


Figure 6.23. Subroutine - quality control for sintered layers.

**Subroutine steps:**

- I. Reset Global String (*GS*) and enter in *GS* a string containing the folder path for the reference slices.
- II. Incrementing the number of the nominal slices (the STL slices):  
 $nNominalSliceNr = nNominalSliceNr + 1$ ;
- III. Message for user: "Slide on checking: *n*",  $n = NominalSliceNr$ ;
- IV. Add to *GS* the nominal slide number and the file extension "bpm".
- V. Message for user: the content of *GS* (path to the nominal slide, which was sintered and will be in the following inspected).

The next three steps of the subroutine are necessary in order to check for major failures in the sintered layer image.

- VI. Binary filter (foreground 0, background 255, lower threshold 0, and upper threshold 50) on *IM4*, results in *IM14*, is implemented in order to separate from the rest of the image all sintered structures.
- VII. In order to eliminate the noise a median NM filter (filter size:  $N=5, M=5$ ) on *IM14* is applied, results in *IM15*.
- VIII. If NOT ((*only one object of brightness 240 to 255 in IM15 with an area of min. 800 pixels*) && (*no object of brightness 28 to 255 in IM4 with an area of min. 10 pixels*)) major failures have been detected.

In order to better display the results to the user, following steps have been implemented:

- a. Dilatation filter (3x3 matrix, 5 iterations) on *IM15*->*IM15* (see Figure 6.24).
- b. Find all objects of brightness 150 to 255 in *IM15* with an area of min. 100 pixels -> display it in *IM20*.
- c. Dilatation filter (3x3 matrix, 9 iterations) on *IM4*->*IM4* (see Figure 6.24).

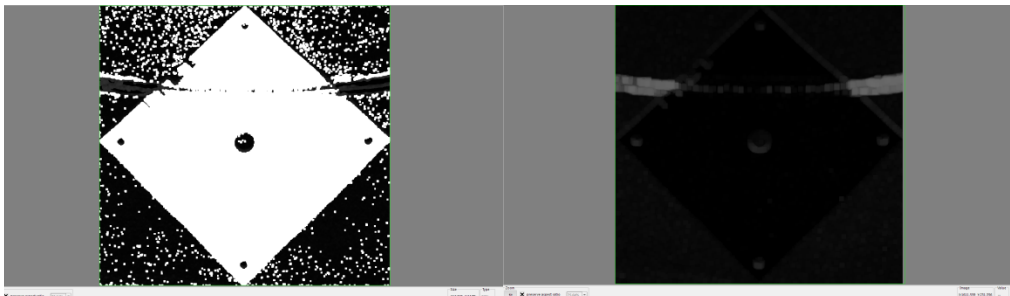


Figure 6.24. *IM 15* (left) and *IM4* (right) after the dilatation filters.

- d. Find all objects of brightness 28 to 255 in *IM4* with an area of min. 35 pixels -> display it in *IM20* (see Figure 6.25).  
Write results in *PartQualityProtocol* text file (see subchapter 6.2.3) and go to *Subroutine: Results n.i.O.*
- IX. If ((*only one object of brightness 240 to 255 in IM15 with an area of min. 800 pixels*) && (*no object of brightness 28 to 255 in IM4 with an area of min. 10 pixels*)) then no major failures have been detected and the next check will be done: namely, the nominal/actual features check.

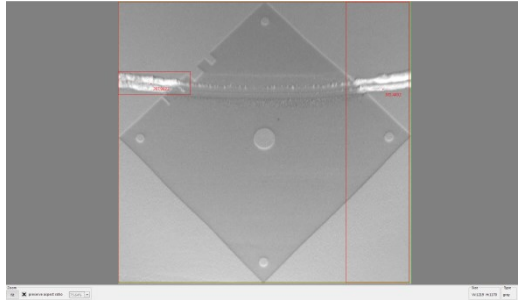


Figure 6.25. Displayed results in IM20: major failures in sintered layer found.

- X. Load the nominal slide, the reference slide, in IM16.
- XI. Over a blob extraction (see Blob Analysis, subchapter 6.2.1) all objects of brightness 0 to 27 in IM15 with an area of min. 6 and max 1500 pixels will be found and counted in *nBlobsActual*. These objects represent the actual inner features of the sintered layer.
- XII. Over a blob extraction, all objects of brightness 238 to 255 in IM16 with an area of min. 6 pixels will be found and counted in *nBlobsNominal*. These objects represent the nominal inner features of the STL slide.
- XIII. User is informed in the *Result Display Text Window* over the number of actual and nominal blobs.
- XIV. Over the  $nFinalSliceCheckBlobResult = nBlobsNominal - nBlobsActual$ , the nominal/actual inner features comparison take place. User is informed in the *Result Display Text Window* over the comparison result.
- XV. If *nFinalSliceCheckBlobResult* is not zero, the inner features of the sintered layer are not identical with the STL slide corresponding ones => we have at least one failure in the sintered layer. Failures are displayed in IM20, over a blob extraction (see Figure 6.26): all objects of brightness 0 to 27 in IM15 with an area of min. 6 pixels will be found and displayed in IM20. Write results in *PartQualityProtocol* text file (see subchapter 6.2.3) and go to *Subroutine: Results n.i.O.*

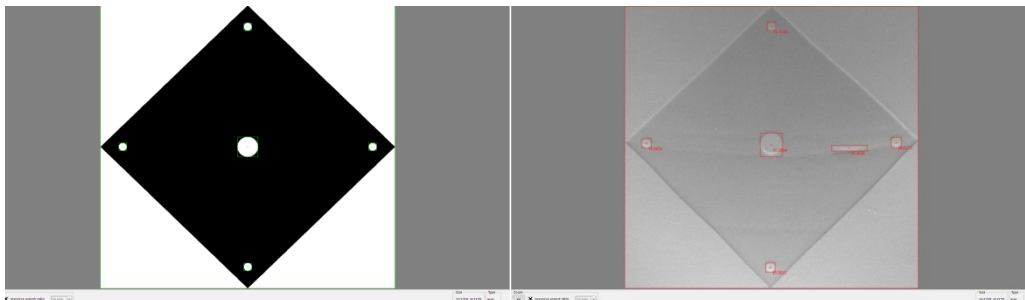


Figure 6.26. Slide on check: 13 (build job: NIST TestArtifact). STL slide nominal inner features (left), actual inner features (right) -> n.i.O.

- XVI. If *nFinalSliceCheckBlobResult* is zero, the actual inner features of the sintered layer are identical with the STL layer corresponding ones; no failure in the actual sintered layer; go to *Subroutine: in-process measure procedure for sintered layers.*

**Subroutine: in-process measure procedure for sintered layers**

In this subroutine, no morphological operations, respectively no non-linear filters are used in order to sustain the measurements accuracy in world coordinates, the actual dimensions of the sintered layer. The in-process measure procedure is presented in Figure 6.28. Here, after a contrast stretching operation, a points list will be reset and filled with the first determined points corresponding to a circular feature, through an annulus probe (two concentric circles). Over a best fit algorithm a circle will be determined, corresponding to the points saved before in the points list. Then a dimensional check, in real world coordinates, of the diameter of the determined circle take place. The user must give all the nominal values of the features that he wants to measure at the beginning. In the same exemplary way, another circular feature is determined and measured. Then the points list will be again reset, in order to save the determined points corresponding to a line (object margin), through a rectangular probe. Over a best fit algorithm a line will be determined, corresponding to the points saved in the points list. Following the same procedure, a second line will be determined and then the distance between these two lines will be calculated. In this way, the actual part width is calculated and will be then compared with its nominal value. If all measuring operations are successful then the layer is i.O., else if for one feature the actual measured value is out of tolerance then the layer is n.i.O. The user is informed continuously about the results and the running operations. All failures are logged in the part quality protocol. All detailed steps of the subroutine including the operations and algorithms developed are presented in the following.

As the image contrast was very low, in order to implement a contrast stretching operation, first a detailed histogram analysis (see e.g. Figure 6.27) of over 10.000 images from different sintered layers, after the lighting correction operation took place. The results of the histogram analysis shown that in all these images the grey values are between [0 40].

*Histogram.* The histogram of an  $N \times M$  digital image having intensity levels  $\{0, 1, \dots, L-1\}$  is a discrete function:

$$h(r_k) = n_k; \quad k=0, 1, \dots, L-1; \quad (6.4)$$

where  $r_k$  is the  $k$ th intensity value and  $n_k$  is the number of pixels in the image with intensity  $r_k$ . [74]

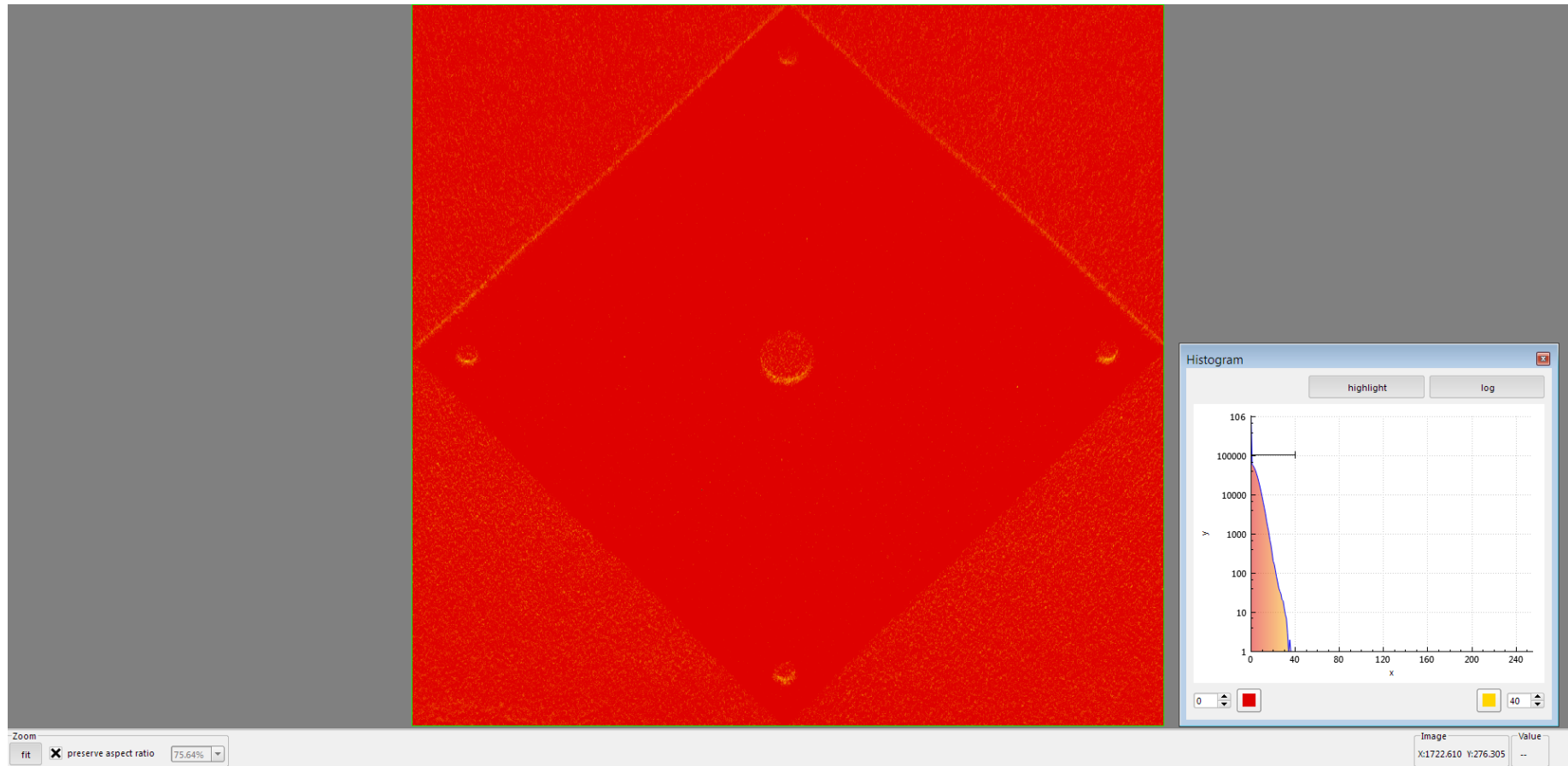


Figure 6.27. Exemplarily histogram analysis of one sintered slide in IM3, the image after the lighting correction.



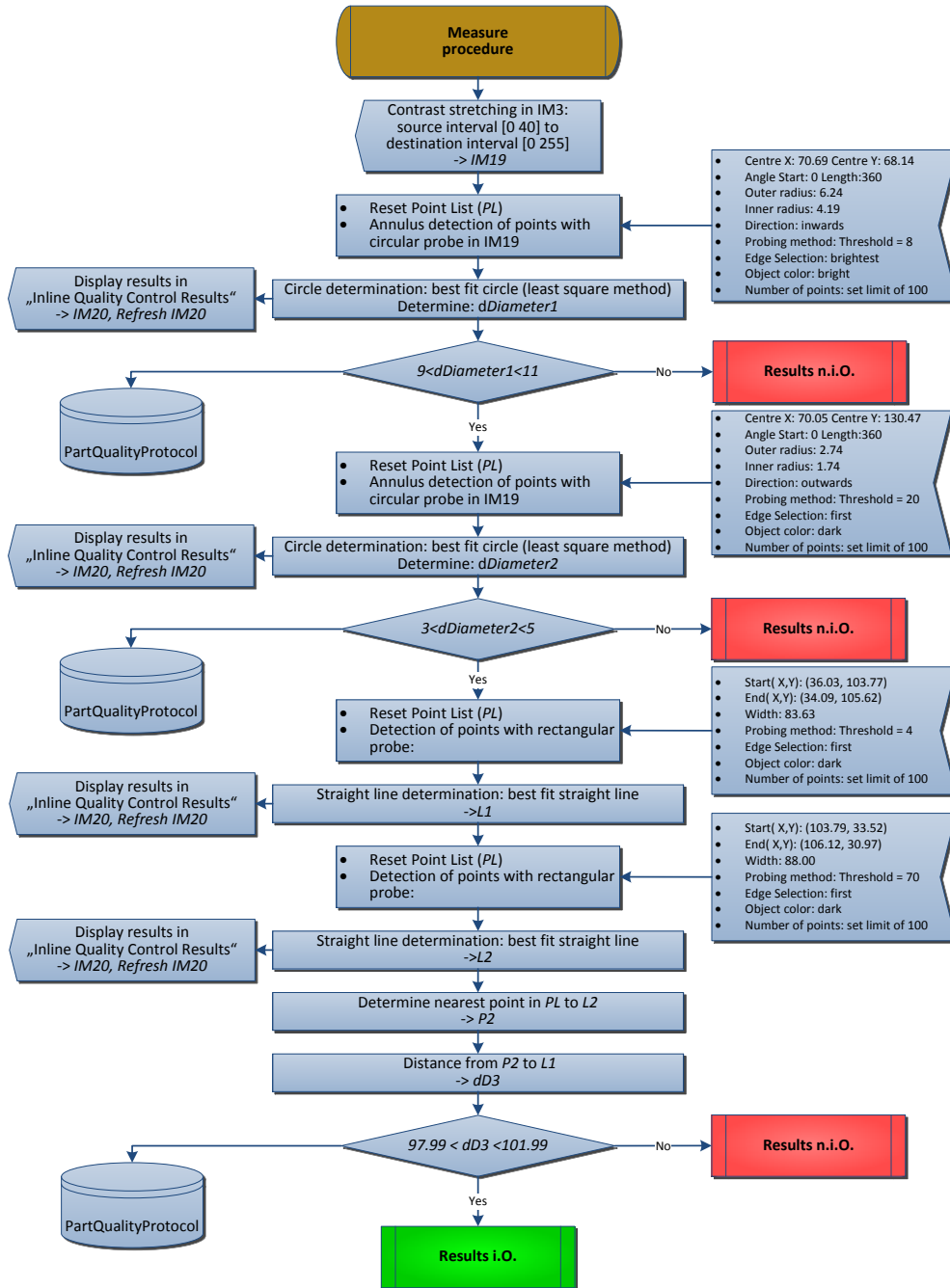


Figure 6.28. Subroutine - in-process measure procedure for sintered layers.

**Subroutine steps:**

- I. Based on the results from the histograms analysis, the first step of this subroutine was to implement a contrast stretching in IM3 (the image after the lightning correction): source interval [0 40] to destination interval [0 255], results in IM19.

*Contrast stretching.* Also named linear contrast enhancement [70], the operation spread the contrast from a given source to a given destination interval. If the lower and upper limits of the destination interval are  $p$  and  $q$ , and if the lower and upper value limits of the original image, determined through the histogram analysis, are  $r$  respectively  $t$ , then the original value of each pixel  $a$  is mapped to output value  $b$  as followed:

$$b = (a-r) \left( \frac{q-p}{t-r} \right) + p \quad (6.5)$$

- II. Reset Point List (*PL*) – a function that clears the content of the point list.

In case of circular features of the sintered layers, the next two algorithms (step III and step IV) have been implemented in order to locate a circular edge in a search area and to measure the respective circle.

- III. Using an annulus sampling form, namely a region between two concentric circles of different radii, the contour points of the respective circular feature are found (see Figure 6.29) and saved in the point list *PL*.

The input parameters for the algorithm are:

- a. Centre X: 70.69 Centre Y: 68.14 -> centre of the annulus.
- b. Angle Start: 0->is the first angle of the circular strip. The angle is measured counter clockwise from the x-axis of a coordinate system centred on the centred of annulus.
- c. Length: 360->arc length.
- d. Outer radius: 6.24-> radius of the outer circle.
- e. Inner radius: 4.19-> radius of the inner circle.
- f. Direction: inwards-> specifies the direction in which the search lines are scanned for edges.
- g. Probing method: Threshold = 8 -> specifies the threshold for the contrast of the edge. Only edges with a contrast greater than this value are used in the detection process. Contrast is defined as the difference between the average pixel intensity before the edge and the average pixel intensity after the edge.
- h. Edge Selection: brightest -> the edge points selection criterion.
- i. Object colour: bright -> object colour.
- j. Number of points: set limit of 100 -> parameter to set the maximum number of points to be added to the *PL*.

*Note.* The user must set once these parameters for each feature that he wants to measure.

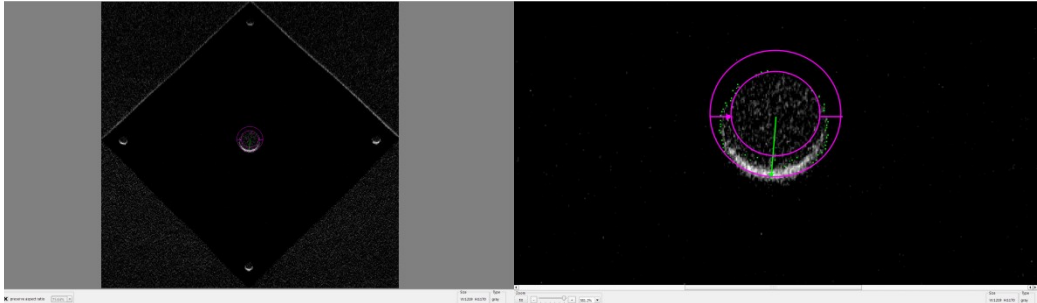


Figure 6.29. Annulus probe for one circular feature of the sintered layer (left). Zoom: points found and saved in *PL* (right).

- IV. For the circle determination, a circle best fit using the Least Squares method was implemented over the 2D points in *PL*. The diameter of the circle will be calculated, measured, and saved in the variable *dDiameter1*.  
*Least Squares Circle Fit*. It is an algebraic fit, based on Taubin's method [75]. The method is based on the minimisation of the function:

$$T = \frac{\sum [(x-a)^2 + (y-b)^2 - R^2]^2}{\sum [(x-a)^2 + (y-b)^2]} \quad (6.6)$$

with,  $x$  coordinates of the points in *PL*,  $y$  coordinates of the points in *PL*,  $a$ :  $x$ -coordinate of the centre of the fitting circle,  $b$ :  $y$ -coordinate of the centre of the fitting circle and  $R$  radius of the fitting circle.

- V. A nominal/actual comparison of the circle diameter *dDiameter1* is done in order to verify it. The nominal value of this diameter is of 10 mm (known from the STL). A tolerance of 1 mm is set in both directions. The results will be displayed in IM20.
- VI. If  $9 < dDiameter1 < 11$  go to the next feature measurement operation, else write results in *PartQualityProtocol* text file (see subchapter 6.2.3) and go to *Subroutine: Results n.i.O.*
- VII. Steps II to IV for another circular feature measurement of the sintered layer.
- VIII. A nominal/actual comparison of the circle diameter *dDiameter2* is done in order to verify it. The nominal value of this diameter is of 4 mm (known from the STL). A tolerance of 1 mm is set in both directions. The results will be displayed in IM20 (see Figure 6.30).
- IX. If  $3 < dDiameter2 < 5$  go to the next feature measurement operation, else write results in *PartQualityProtocol* text file (see subchapter 6.2.3) and go to *Subroutine: Results n.i.O.*

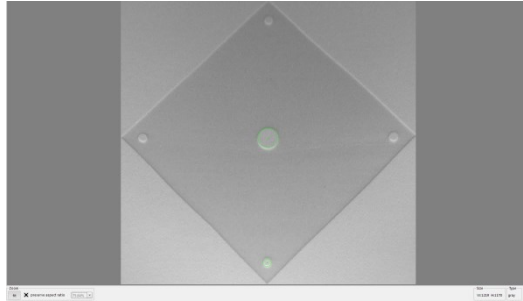


Figure 6.30. IM20 - two measured circular features through subroutine steps II to IX.

X. Reset Point List (*PL*) – a function that clears the content of the point list.

In case of rectangular features of the sintered layers, the next two algorithms (step XI and step XII) have been implemented in order to determine the edges, the lines describing the rectangular features.

XI. Using a rectangular sampling form, the contour points of the respective rectangular feature are found (see Figure 6.31) and saved in the point list *PL*.

The input parameters for the algorithm are:

- a. Start(X,Y): (36.03, 103.77)->point start coordinates.
- b. End(X,Y): (34.09, 105.62)->point end coordinates.
- c. Width: 83.63->width of the rectangular probe, if 1 then we have only one line from Strat to End.
- d. Probing method: Threshold = 4 -> specifies the threshold for the contrast of the edge. Only edges with a contrast greater than this value are used in the detection process. Contrast is defined as the difference between the average pixel intensity before the edge and the average pixel intensity after the edge.
- e. Edge Selection: first->the edge points selection criterion.
- f. Object colour: dark->object colour.
- g. Number of points: set limit of 100->parameter to set the maximum number of points to be added to the *PL*.

*Note.* The user must set once these parameters for each rectangular feature that should be measured.

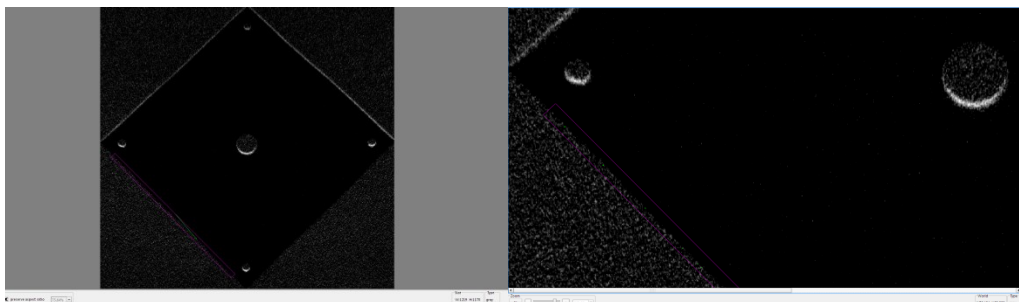


Figure 6.31. Rectangular probe for one sintered layer: edge feature (left), probe zoom (right).

- XII. A straight line determination using a best fit straight line method, the Least Squares Regression Line, was implemented over the 2D points of the founded edge and saved in *PL* (step XI). Results in *L1*. *L1* is display in IM20 (see Figure 6.32).

*Least Squares Regression (LSR) Line.* Linear regression is a method to best fit a straight line of the form  $y = a + bx$  to a collection of points  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ , where  $b$  is the slope and  $a$  is the intercept on the y axis. The algorithm requires the minimization of the sum of the squared distances from the data points to the proposed line.

$$b = r \frac{S_y}{S_x} \quad (6.7)$$

with  $S_y$  and  $S_x$ , standard deviations of y and x and

$$a = \bar{y} - b \bar{x} \quad (6.8)$$

with  $\bar{y}$  and  $\bar{x}$ , the mean of y and x.

- XIII. Following the steps from X to XII, a second line *L2* of the sintered layer is determined. *L2* displayed in IM 20 (see Figure 6.32).

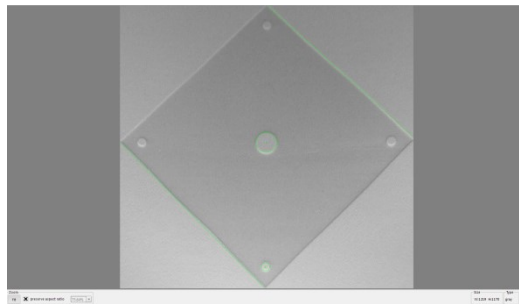


Figure 6.32. IM20 - two measured circular features and the two lines *L1* and *L2* determined using subroutine steps X to XIII.

In order to find the width of the sintered layer, the distance between lines *L1* and *L2* is calculated using subroutine steps XIV and XV.

- XIV. Determine nearest point from *PL* (points list where all the point's coordinates used for the LSR calculation of *L2* are still saved) to *L2*, results in variable *P2*.
- XV. Calculate the distance between *P2* and *L1*, results in variable *dd3*.
- XVI. A nominal/actual comparison of the sintered layer width and *dd3* is done. The nominal value of it is of 100 mm (known from the STL). A tolerance of 2 mm is set in both directions. The results will be displayed in IM20.
- XVII. If  $97.99 < dd3 < 101.99$  go to the next feature measurement operation, if wanted, or to the *Subroutine: Results i.O.*; else (see Figure 6.33) write results in *PartQualityProtocol* text file (see subchapter 6.2.3) and go to *Subroutine: Results n.i.O.*

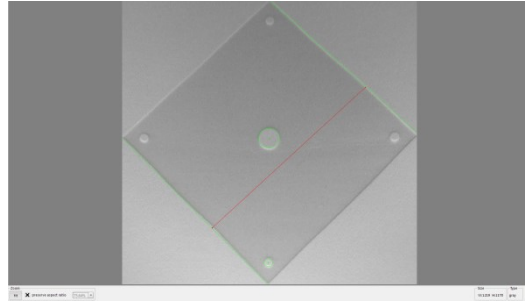


Figure 6.33. IM20 - two measured circular features, the two lines  $L1$  and  $L2$  and the results from subroutine steps from XIV to XVII -> n.i.O. sintered layer.

**Subroutine: Results i.O.** and **Subroutine: Results n.i.O.**

The results subroutines are presented in Figure 6.34.

**Subroutines steps:**

- I. Message to user in the *Result Display Text Window* about the inspected layer: **i.O.** or **n.i.O.**
- II. Refresh IM20, the *Inline Quality Control Results* image.
- III. Wait 1000 ms, so that the user gets the results.
- IV. Delete all result graphics from all IMs (image memories).
- V. Fill with grey the IM16.
- VI. Jump to mark *GO*, in order to inspect the next layer (see Figure 6.9).

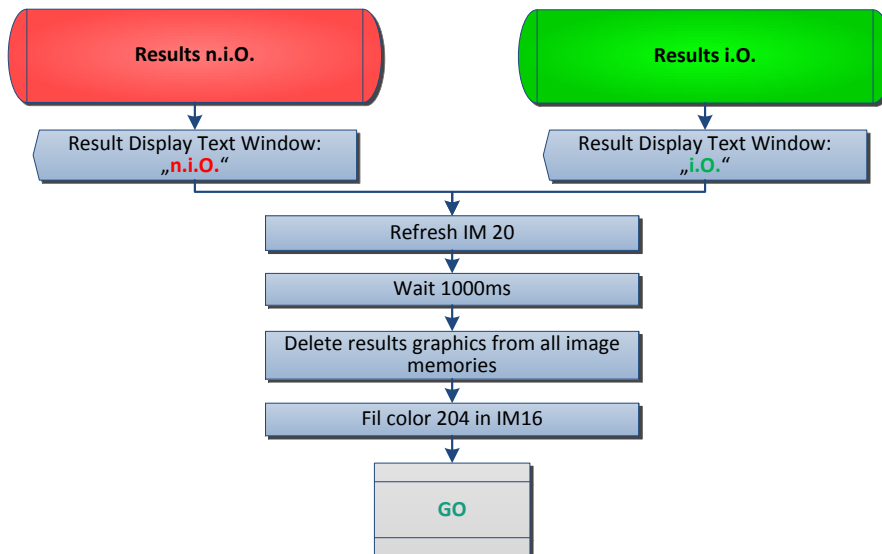


Figure 6.34. Subroutines for results.

### 6.2.3. Quality protocol

*Novelty*<sup>1</sup>. At the end of any AM PBF process, every additive manufactured part can be now delivered with a detailed quality protocol.

In the advanced IIP routine of the inline QC system, a part quality protocol was implemented, so that the results of the layer-by-layer inspection are logged, in form of a text file, during the execution of all subroutines presented in subchapter 6.2.2.



Figure 6.35. Defined *PartQualityProtocol* function symbol, used in the subroutines diagrams.

The file contains following data: time stamp, layer number, sintered layer number and quality control results as failure type, number of failures found, measurement results. The file can be extended at any time further, with other values that are important for the system users.

Exemplary in Figure 6.36 an extract from a part quality protocol is presented.

```
2016_05_31_18_12_15_451 Layer:3 QC:
wide failures in powder layer:2,00
2016_05_31_18_12_17_088 Layer:4 QC:
subtle failures in powder layer:13,00
2016_05_31_18_13_25_453 Layer:44 Sintered Layer:1 QC:
Diameter Circle1:10,03;
Diameter Circle2:4,07;
DistancePointToLine1:102,27
2016_05_31_18_14_44_379 Layer:68 Sintered Layer:13 QC:
Failure -> NOMINAL/ACTUAL BLOBS COMPARISON:-1,00
```

Figure 6.36. Extract from a part quality protocol.

## 6.3. Graphical user interface

The developed user interface is a very friendly one. Two layouts have been implemented: the *Inspection Program Setup* and the *Inline Inspection* one. The first one is for an advanced user that can parametrise the SW for a specific build job; the other, independent from the user qualification degree, where the user is just loading a build specific inspection program and just runs it.

In the *Inspection Program Setup* layout, following procedures should be followed:

- Setting the folder path containing the slices (nominal layers) in \*.bmp format. These slices must be exported in this format by the dedicated AM PBF machine software after they were set as input for the process, as input data for the AM PBF machine.
- The layer thickness and all dimensions of the part must be introduced.
- Defining the measurements that must be performed during the manufacturing process, e.g. for the features dimensional check.
- Set the limits for the decisional algorithms (see Figure 6.37): as e.g. decisional blob extraction with area criterion for determination of layer type.

## 104 Inline QC system components – 6

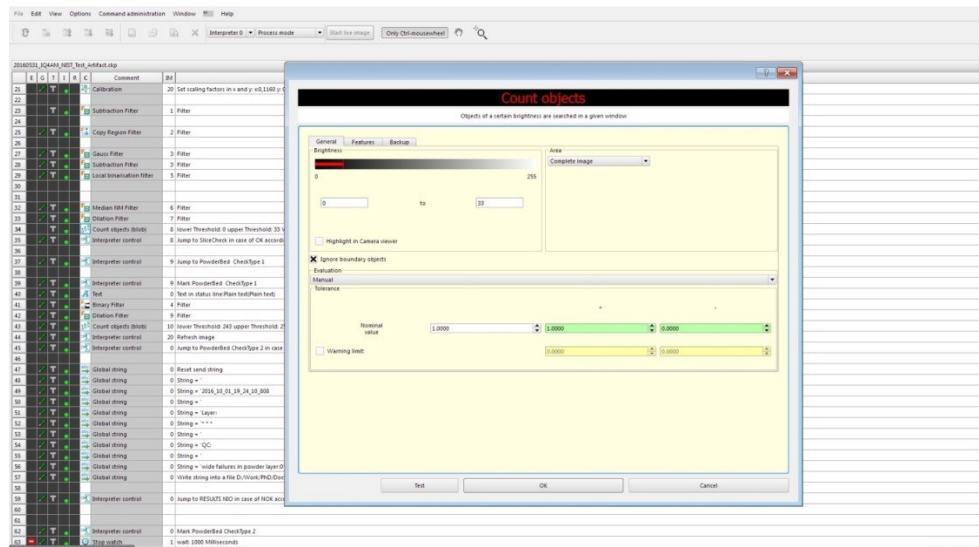


Figure 6.37. GUI layout: *Inspection Program Setup* - setting the area criterion for the blob extraction and the grey value interval.

After the user executed these steps, there is the possibility (see Figure 6.38) to save each inspection program. In this way, if he wants to produce the same build job later again, for which he already parametrised an inspection program, he can simply load and run it.

After the program was loaded, the inline inspection layout of the GUI can be chosen and the software can be started. In the *Inline Inspection* modus, no advanced user is needed; any user can start or stop the program (see Figure 6.39). Based on the quality information delivered by the SW, the user can stop the AM PBF machine when e.g. serious quality failures are detected during production time.

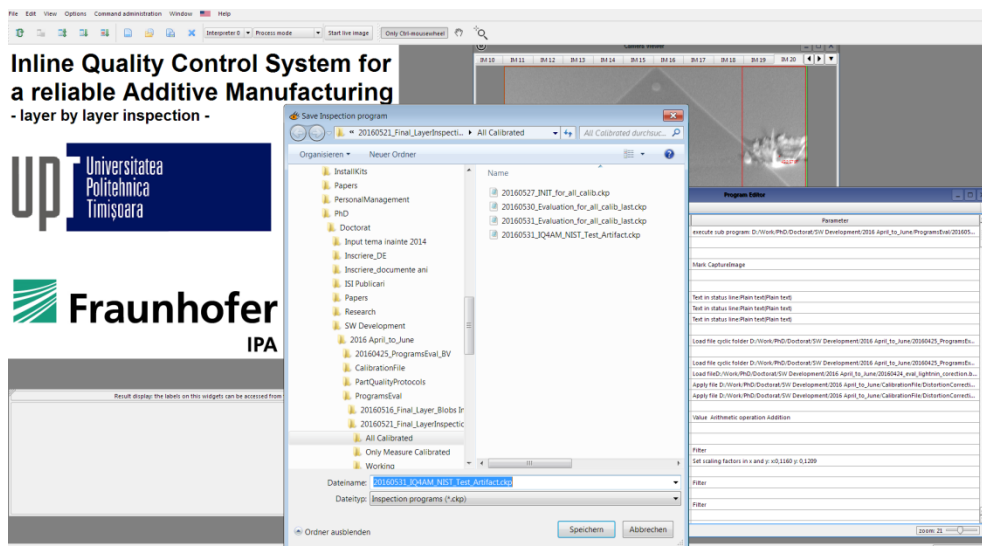


Figure 6.38. GUI layout: *Inspection Program Setup* - saving parametrised inspection programs.



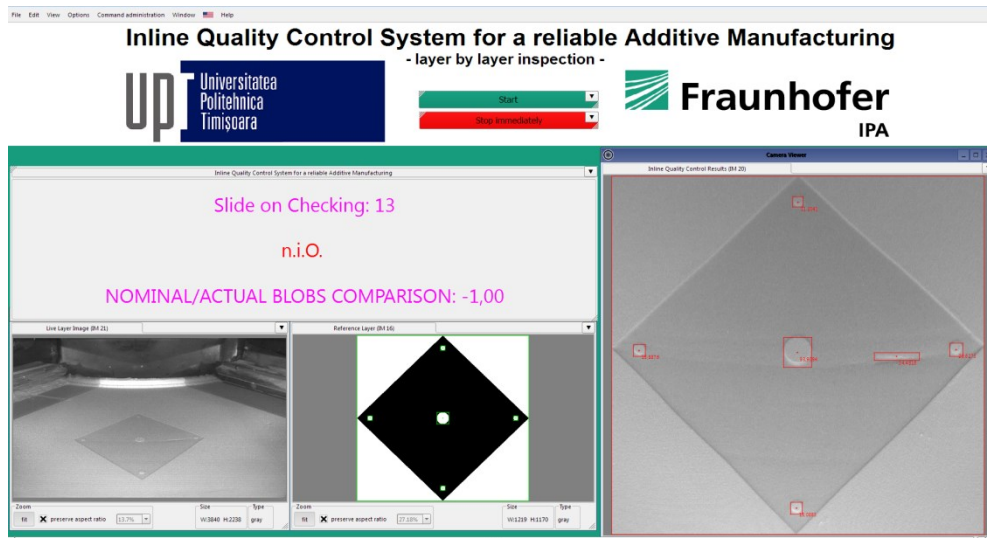


Figure 6.39. GUI layout: *Inline Inspection*.

*Novelty*<sup>1</sup>. At the same time, because the manufacturing process can take hours, another necessity of the machine users is herewith covered, namely the screen of the inline quality control software can be shared, so that at any time, from another place (e.g. office) the users of the AM PBF machines can have an overview of the manufacturing process.

The information got by the machine user, in the inline inspection layout, is concentrated in four windows:

- A step-by-step result window: here the user can see which layer is inspected at the moment, what kind of inspection operations is in progress (e.g. subtle failures in powder layer) and the inspection results; in case of failures including the failure type identified is displayed.
- The live image acquired.
- A window showing the nominal slide.
- A window displaying the distorted inspected ROI including the results in form of intuitive graphic overlays.



## 7. RESULTS

In the actual context of the state of the art in the field of inline QC and in-situ optimisation of AM PBF processes, taking into consideration the deficiencies of the actual approaches (see subchapter 3.6), first an inline QC system's tasks identification approach (see chapter 4) was developed. Based on it, after the tasks determination, an overall concept for an inline QC system has been worked out (see chapter 5). Further, the design, development, and implementation of the system components have been conducted (see chapter 6). The results of this research work, being a *novelty*<sup>1</sup>, culminate into a novel and innovative inline QC control system for a reliable AM, having a high degree of applicability in the AM PBF processes. The system as a whole and the achieved results are described in detail in the following subchapters.

### 7.1. Inline QC system for a reliable AM

The developed HW platform and the SW engine (see chapter 6), have been integrated in the inline QC system for the AM PBF processes, using a SLS system.

*Novelty*<sup>1</sup>. The developed system is an AM PBF machine-independent one: the data acquisition is triggered directly by a dedicated SW subroutine over a data analysis, the data being delivered by the machine vision HW components. The overall advanced IIP SW routine (see subchapter 7.1.1), not only ensures an inline layer-by-layer inspection of the additive manufacture parts, but generates at the same time a quality protocol for each produced part. The standalone SW application, with a friendly GUI (see subchapter 6.3), is thanks to the Qt library implementation, operating system (OS) independent. In this way, the software application can run not only on Intel or on AMD, but also on ARM and MIPS architecture based systems. Through the SW and HW integration (see subchapter 7.1.2) a modular system has been achieved that can continuously inspect the quality delivered by the AM PBF production and give an inline feedback to the machine operator.

#### 7.1.1. Advanced IIP SW routine for the inline layer-by-layer inspection

The two main software routines, the *IIP for Inline Image Acquisition* (see subchapter 6.2.1) and the *IIP for Quality Control* (see subchapter 6.2.2), and their subroutines presented in subchapter 6.2 have been integrated, connected with each other, in a final advanced IIP SW routine, which ensures an inline QC of the AM PBF SLS process (see Figure 7.1.). The image acquisition corresponding to a sintered or a powder layer will be triggered by the position of the material supplier. After the image is captured, the image distortion correction takes place as well as the layer type identification.

Then, in case of powder layers: if no serious failures are found then a subtle failure inspection takes place. If no further failures are found, the layer will be classified as i.O. If one wide or subtle failure is found during the inspection of the powder layer then the failure will be logged in the part quality protocol.

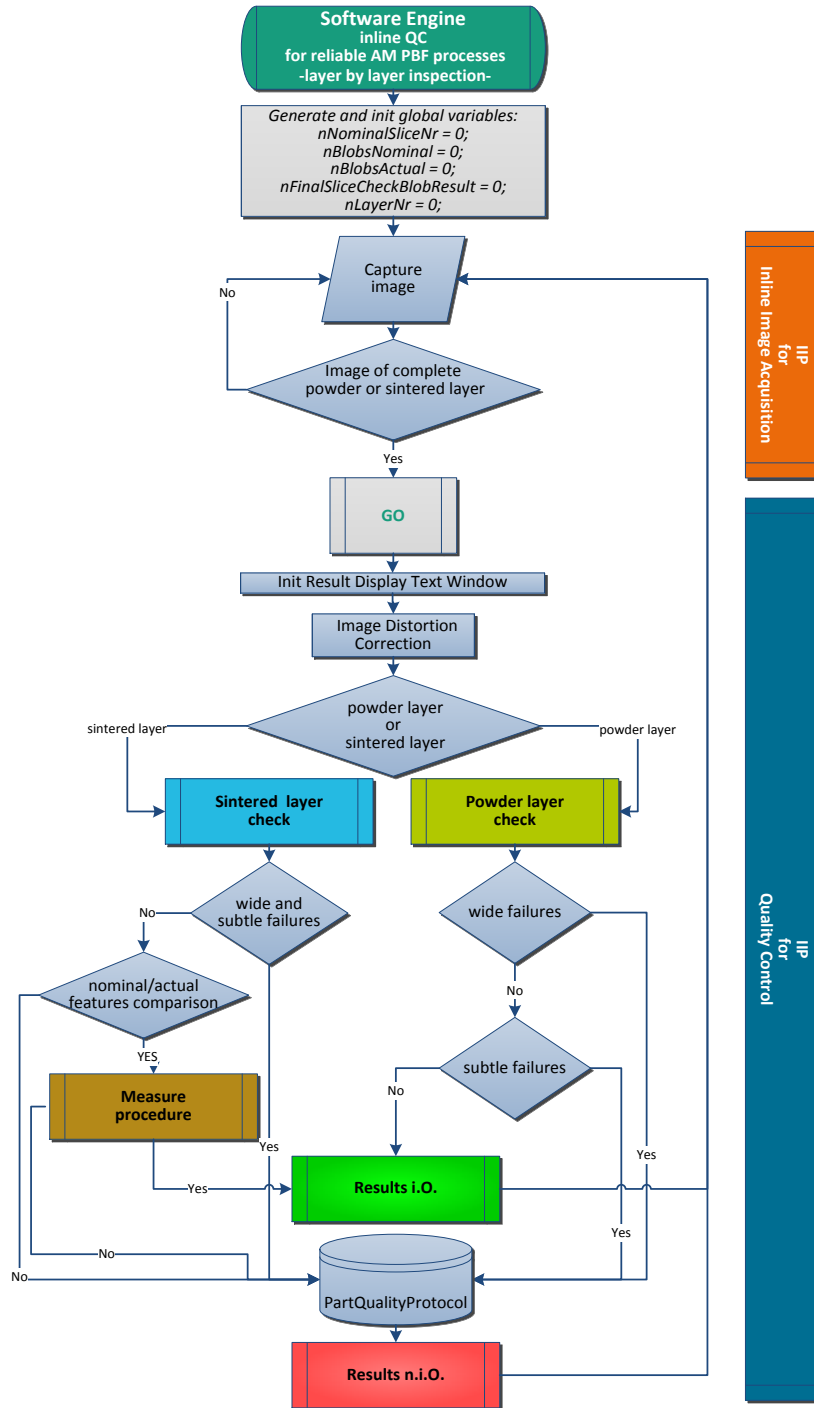


Figure 7.1. Overall advanced IIP SW routine of the inline QC for a reliable Additive Manufacturing.

In case of sintered layers: if no serious, wide, or subtle failures are found, a nominal/actual comparison of the sintered features numbers takes place. If the numbers are different, the layer is classified as n.i.O.; a quality problem appeared, as in the sintered layer the number of features differ to the one in the nominal layer. If the number of the features is the same, the measuring operations for different features are executed and the results are compared with nominal values, taking into consideration the tolerances.

In both cases, after the evaluation procedures, the user is informed about the results and the system will wait for the trigger signal, given by the SW routine *IIP for Inline Image Acquisition*, in order to acquire the next image.

### 7.1.2. Hardware and software integration

In order to bring together the two components of the system, only an Ethernet cable is necessary, which ensures the data transfer from the Machine Vision Camera over the GigE Network Board to the system running the application.

The overall implemented system is presented on conceptual level in Figure 7.2, and on implemented level in Figure 7.3.

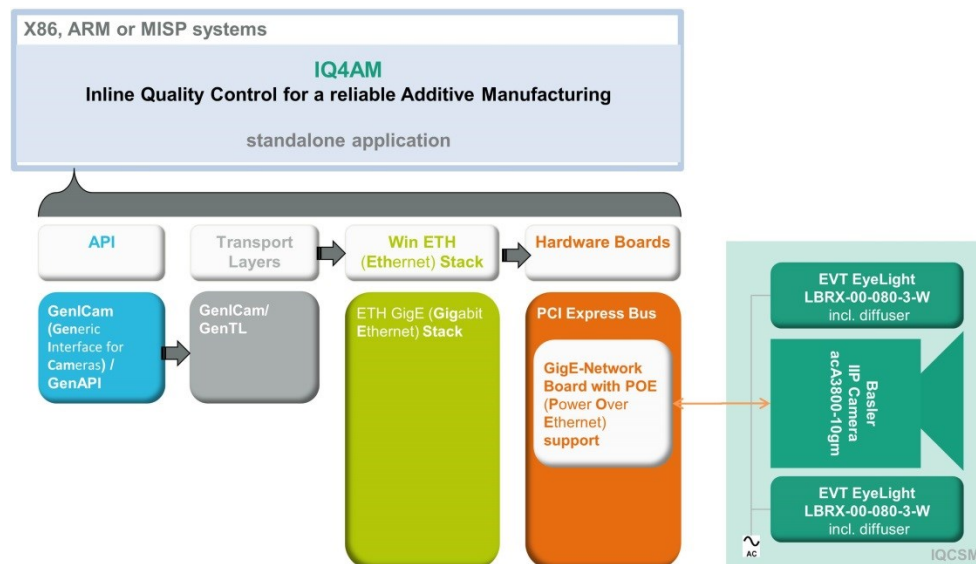


Figure 7.2. Hardware and software integration in final system: conceptual level.

The system has been installed on the EOS Formiga P100 at Fraunhofer Institute for Manufacturing Engineering and Automation IPA (Fraunhofer IPA), where, in-process, the quality of the manufactured parts was inspected. The user has the possibility to stop the process in case of serious failures and has for each manufactured part a quality protocol.

By manufacturing processes with a building time of over 10 hours, behind the observation window, were the system was installed, the temperature was increasing over 38°C. In order to ensure an uninterrupted system utilisation, a system improvement on the HW side took place. A fan type FONSONIC FSY80X24L, which ensures a system working environment temperature under 25°C, was installed (see Figure 7.4).



Figure 7.3. Hardware and software integration in final system: implemented level.



Figure 7.4. System HW with integrated fan.

## 7.2. Verification of the results

For the development and implementation as well for the verification of the modular inline QC system the NIST test artefact (see chapter 6) was used (see Figure 7.5).

The verification stage is very important in order to further improve the system and to verify the results achieved. Is a part, for which one or more layers have been classified as n.i.O. or i.O. by the inline QC system, really n.i.O. or i.O.? As up to date, no inline system ensures the quality control of AM PBF produced parts, it was necessary to use, as reference for the verification of the results, an offline technology; therefore, the advantages of industrial computed tomography (CT) have been used [32].

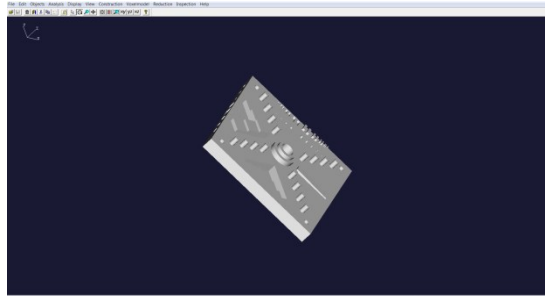


Figure 7.5. STL model of the test artefact used as input data for the SLS machine EOS Formiga P100.

### **7.2.1. CT as reference technology for overall results: i.O. vs. n.i.O.**

SLS manufactured NIST test artefacts have been integrally scanned (Figure 7.6) with the Werth TomoScope® HV 500 [76], an industrial CT at Fraunhofer IPA. The resolution of the CT is of 0.134566 mm; the artefacts being scanned at once and having the following dimensions: X=141.2 mm, Y=141.2 mm, Z=17mm.

With adequate software programs, the CT data obtained was analysed in detail on 2D level. The 2D CT slices are available, since the 2D X-ray images of the parts, used in order to obtain the spatial representation of the objects, are part of the 3D-Data.

Exemplary, in the following, verification results corresponding to i.O. and n.i.O. parts, classified during the manufacturing process by the inline QC system, are presented.

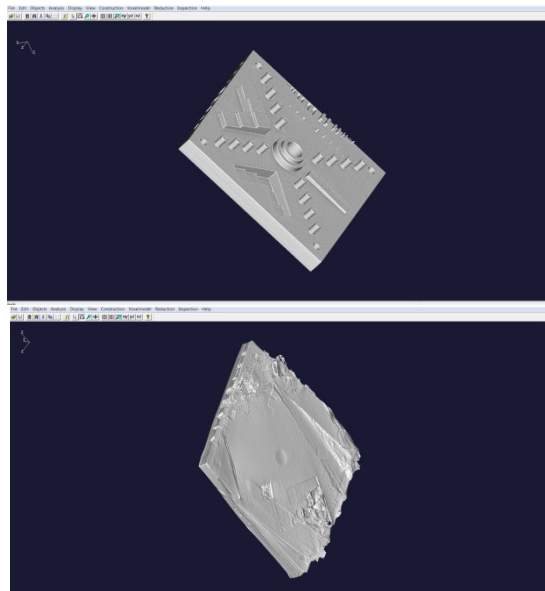


Figure 7.6. CT acquired 3D-Data for inline QC system classified parts: i.O.(top) and n.i.O.(down).

## 112 Results – 7

For example, taking a depth of 0.8 mm, are the results from the inline QC system and the ones from the CT the same? As the CT resolution is of 0.134566 the 2D CT slice corresponding to a depth of 0.8 mm is the slice 6 (see Figure 7.7).

In order to compare the results on 2D level, the inspection results of the inline QC system for the same part, same layer, at the same depth of 0.8 mm have been evaluated. As the EOS Formiga P100 "layer thickness" production parameter was of 0.1mm, the results of the 8th sintered layer have been considered.

The CT results (see Figure 7.7) are confirming the results obtained by the inline QC system (see Figure 7.8), namely that the sintered layer 8 is i.O.

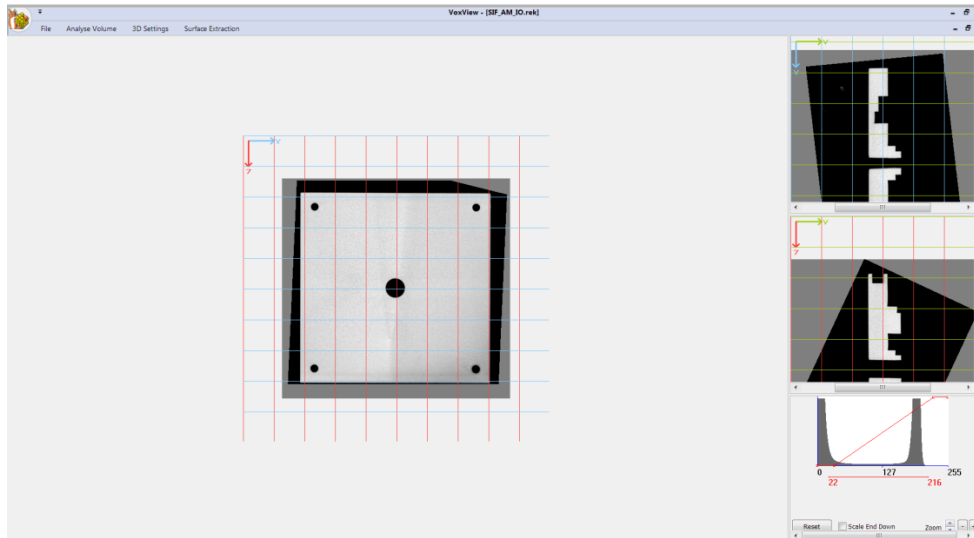


Figure 7.7. CT acquired data of a SLS build artefact part: CT slice 6.

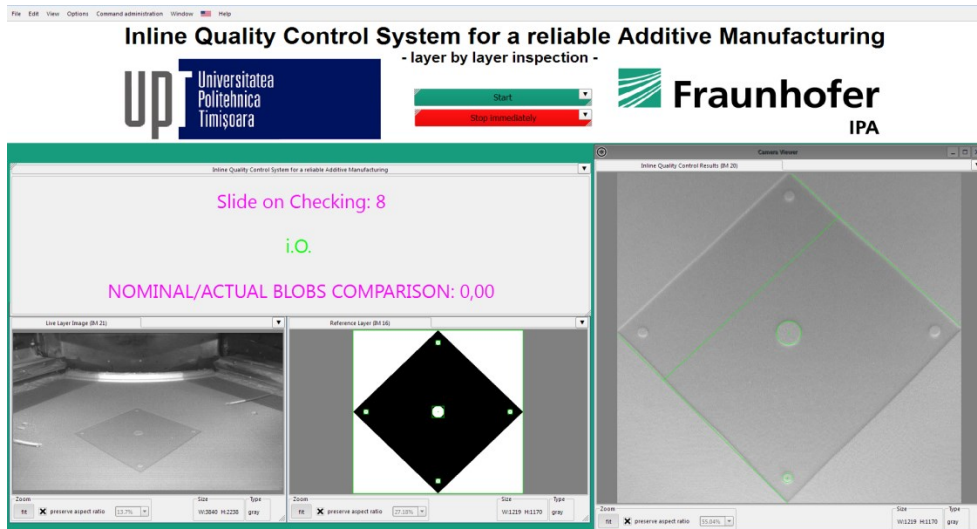


Figure 7.8. Artefact build part, classified by the inline QC system as i.O., at layer 8.



In the same way, exemplary, the results of one n.i.O. part, classified by the inline QC system have been confirmed by the CT scan and are depicted in Figure 7.9 and Figure 7.10. In this case, based on the results of the inline QC system, namely "Slide on Ckecking:14 n.i.O.", the user of the AM PBF machine, from his office could go and stop the build process, as the identified failure was very serious. The AM PBF machines are normally placed in special machine shops or laboratories, with special environmental conditions, where no user can wait hours until a build job is finished. Until the developed inline QC system, the build process has been continued hours after hours, up until the build job had completely finished: an extremely cost-inefficient build process, taking into consideration the machine time and the feedstock sunk costs.

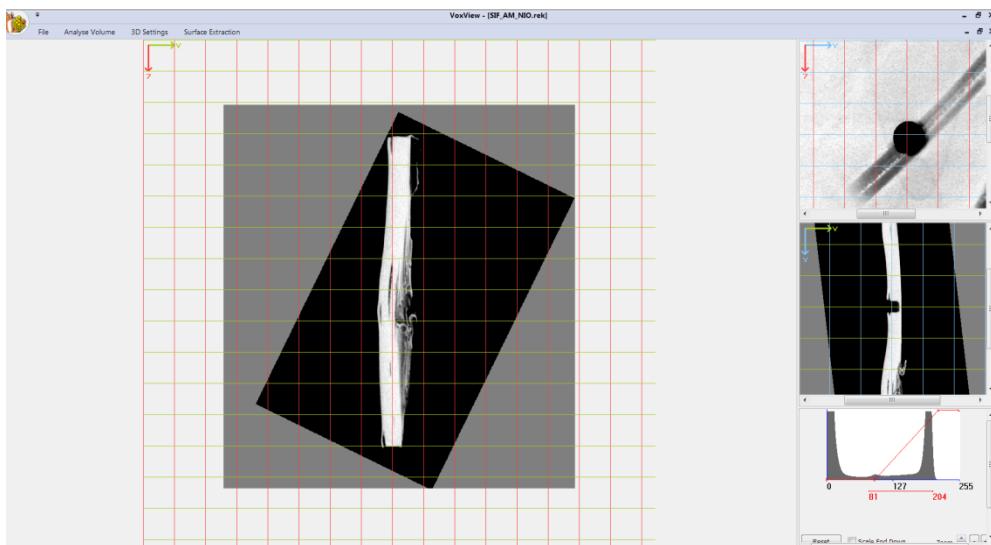


Figure 7.9. SLS build artefact, in CT data projection YZ.

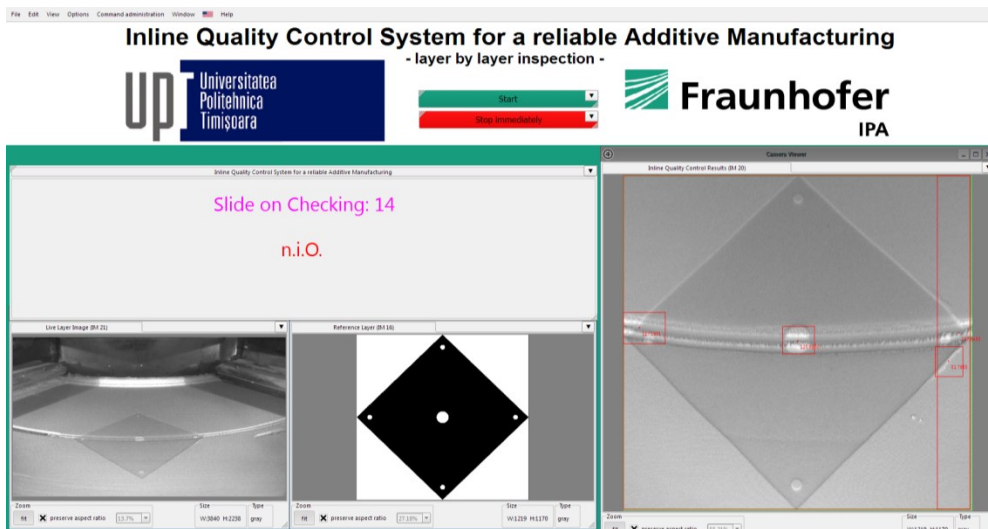


Figure 7.10. SLS build artefact, classified by the inline QC system as n.i.O., at layer 14.

### 7.2.2. Accuracy of the inline QC system

The system accuracy is first given by the hardware components (see subchapter 6.1.1). The machine vision camera has a resolution of 10 MP (3856 x 2764 pixels) with an optical sensor size of 1/2.3" and a pixel size of 1.67 x 1.67 [ $\mu\text{m}^2$ ]. For the developed system, the lenses gave the only accuracy limitation, as at the system hardware development time, 2015, no lenses having a Modulation Transfer Function (MTF) of 10MP, with a focal length of 6mm, and a corresponding field of view on pixel level were economical efficient (lens twice more expensive as the whole system).

Taking into consideration the MTF of the used lenses, which allows only 5MP, the focal length of 6mm, and the working distance of 350mm the resulted field of view is 320x240 mm. Based on this, the system optical resolution is 82 microns. This accuracy is the same for the 200x250mm inspected area, corresponding to the build platform. However, because the necessity of an image distortion correction, given by the fact that the camera system is not parallel with the build platform to be inspected, the pixel size in real world units lowered to: 116 microns in x direction and 120 microns in y direction. As the prices for the, in 2015, very expensive lenses decrease today (end of 2016) the pixel size in real world units for the system can be increased with over 50 %, by simply changing the lenses.

Therefore, in order to evaluate the inspection and measure accuracy of the developed advanced IIP routine dedicated algorithms for data evaluation, during the AM PBF manufacturing processes, we will refer to pixel units.

Taking into consideration an image of one powder layer, as shown in Figure 7.11, the failures detected by the system can be seen in Figure 7.12.

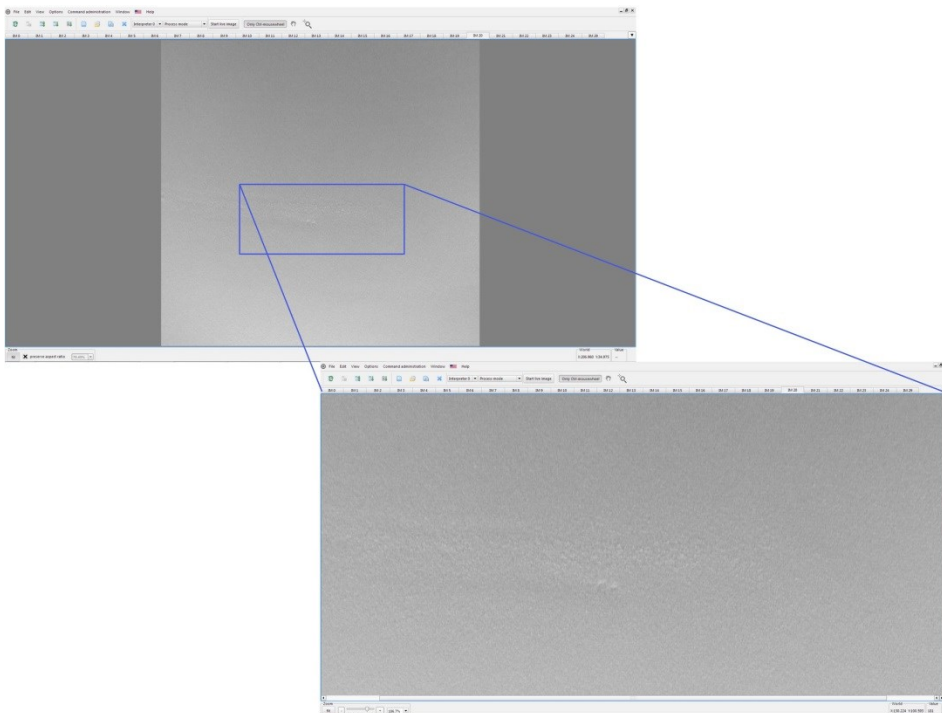


Figure 7.11. Powder layer image: zoom of the one failure region (down).

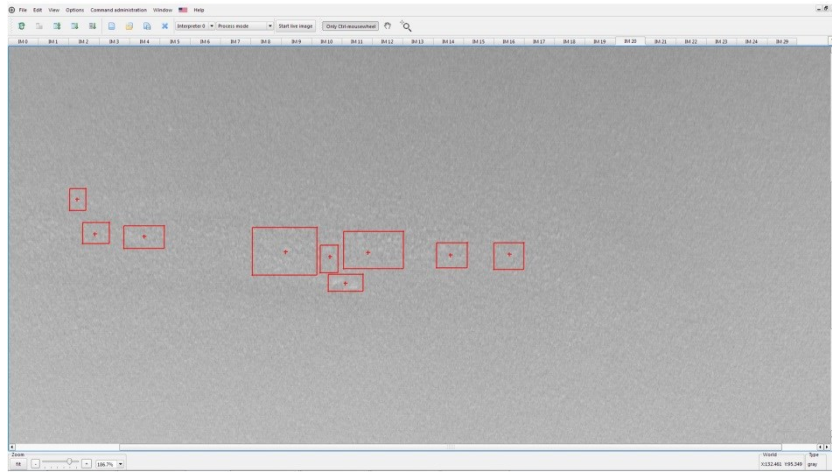


Figure 7.12. Powder layer image: detected failures areas.

In Figure 7.13 a i.O. region (blue coloured) is zoomed in order to see the pixel values. In Figure 7.14 and Figure 7.15 n.i.O. regions (blue coloured) are zoomed in order to see the corresponding pixel values. In these images, is shown that the developed system can identify failures having a size of at least three pixels even if the image contrast is extremely low.

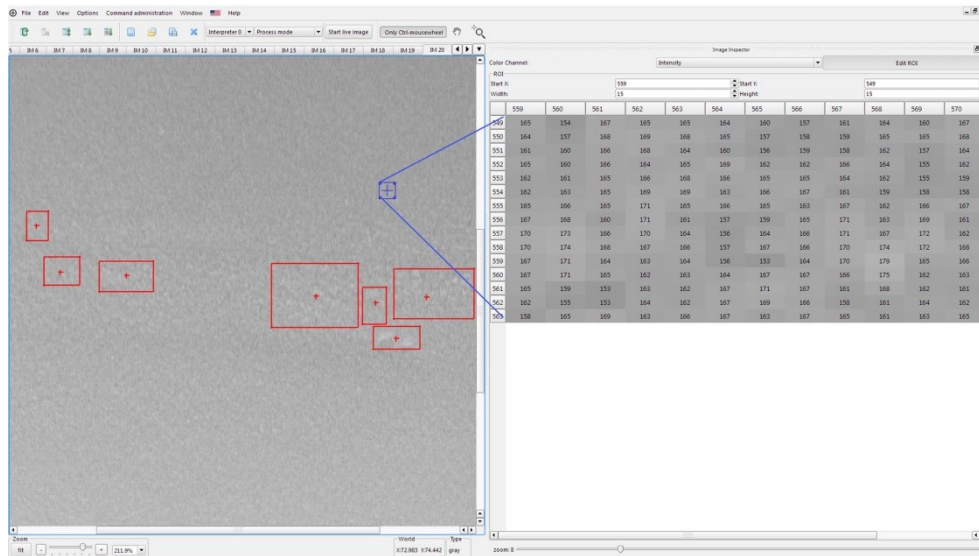


Figure 7.13. Powder layer image: i.O. region with detailed pixel values.

## 116 Results – 7

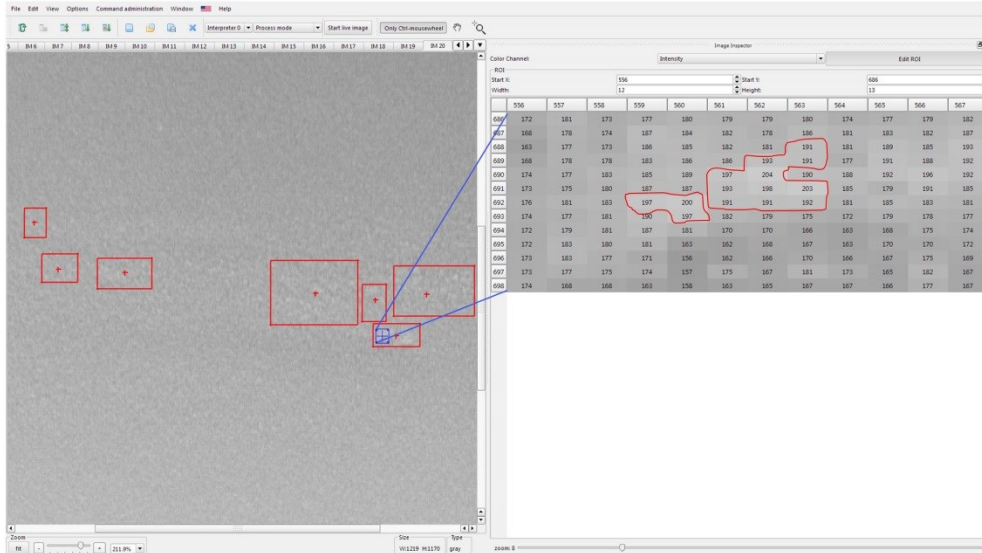


Figure 7.14. Powder layer image: one n.i.O. region with detailed pixel values.

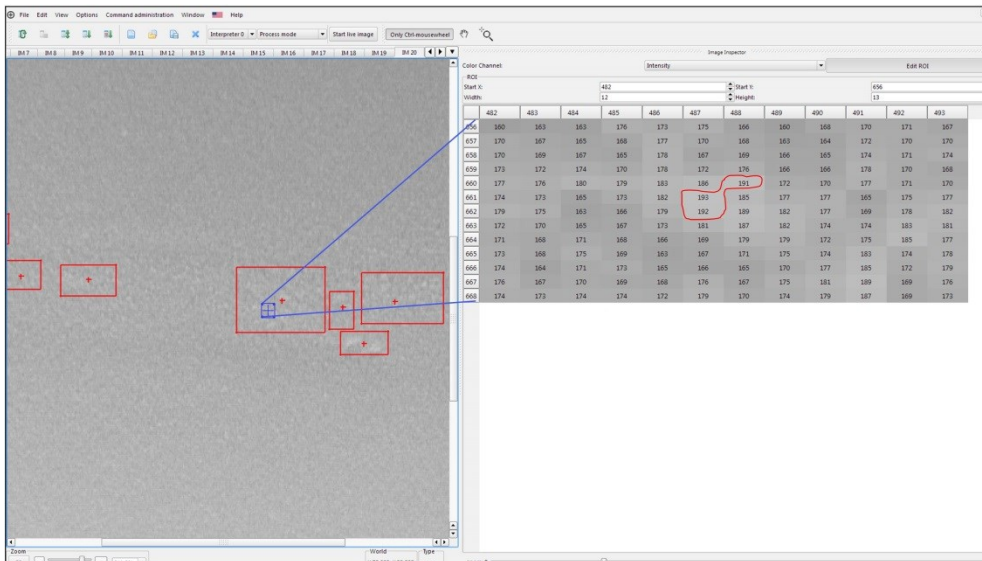


Figure 7.15. Powder layer image: another n.i.O. region with detailed pixel values.

In Figure 7.16 a sintered layer and the system measurements results (part width and diameter of two circular features – see subchapter 6.2.2), are depicted.

For the measured features, as circle diameters and distances, the accuracy is given by the accuracy with which the features are automatically found in image. For a sintered layer the Figure 7.17 and Figure 7.18 demonstrate the system high capability to fit such geometries in such low contrast images.



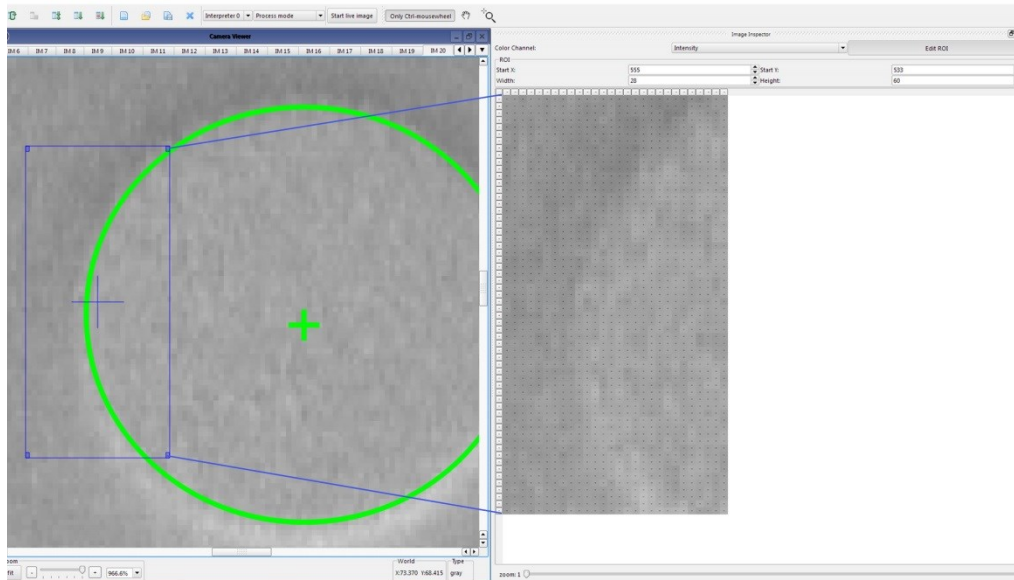


Figure 7.18. Sintered layer image: fitted circle.

**Conclusion:** *the overall obtained results are, taking into consideration the given low image contrast, an achievement which reflects the distinct contribution to the field of inline QC for the AM PBF processes.*

## **8. CONCLUSIONS, DISTINCT CONTRIBUTIONS AND FUTURE RESEARCH WORK**

### **8.1. Conclusions**

Starting from the critical analysis of the state of the art of the QC and QA of the AM PBF processes, the main goal of the present thesis was to bring distinct contributions in the field of inline QC, aiming to create a firm foundation for a reliable future additive batch production.

The thesis treats a series of theoretical issues concerning the parts' quality monitoring during the build cycles, which is influenced by a huge number of parameters. An approach for the determination of the overall effects of this influencing parameter, having an appearance during the AM PBF processes, have been carried out. The overall tasks for an inline QC system have been in this way determined and a comprehensive concept for an inline QC system in the AM PBF production was accomplished.

The applicative part of the thesis presents the development and implementation of an inline QC system for the AM PBF, based on IIP, using the example of one of the most promising technology, for a future batch and even serial AM production, namely the SLS manufacturing process. The development and implementation of the experimental research work took place at the Fraunhofer Institute for Manufacturing Engineering and Automation IPA.

Over the determination of the technologies and sensors necessary for data acquisition and the design of the hardware architecture, an experimental hardware setup was worked out. During the development stage, important aspects have been considered as e.g. the price/performance ratio and the modularity of the HW platform. A very important aspect is here the achievement of a machine independent HW platform.

Further, the software components for the layer-by layer inspection have been developed, in form of two main SW routines based on IIP: one for the machine independent data acquisition and the other one for dedicated algorithms for data evaluation during the AM PBF manufacturing processes. During the inline QC of the part being produced, all detected failures are logged in a dedicated part quality protocol, over the AM PBF production time. In this way, not only the quality of the parts, but also the prerequisite for using the AM PBF parts in industrial environments are ensured. The applicative part of the thesis concludes with the implementation and testing of an innovative and novel inline QC system mounted on a SLS machine from the company EOS [15], namely the Formiga P100.

The fact that the conducted research work is closing a gap in the state of the art concerning the QC and QA of AM PBF processes, is also given by its high degree of applicability. In order to install the system no approval of the manufacturers of the AM PBF systems is needed, as there is no necessity to access the HW interfaces of the AM machines. Thus, no ISO standards certificates are necessary for the integration of the system on the machines. The system provides at the same time the possibility of being extended at any time with further sensors

and/or algorithms. Herewith a firm basis for a future in-situ AM PBF process optimisation was attained.

The doctoral thesis statements are proved to be true, as the achieved QC system ensures inline, tirelessly the quality of the parts build on a SLS machine at the company cirp GmbH [77]. Here continuously, during the AM PBF production, the inline QC system:

- detects quality failures during production process,
- offers a quality report of the produced parts,
- offers to the user the possibility to cancel the production process in extreme cases,
- needs no users with technical expert knowledge, since it requires a small number of input parameters.

## 8.2. Distinct contributions and future research work

In the thesis content, the following distinct contributions of the author, have been identified as *novelties*<sup>1</sup>:

- Theoretical contributions:
  - A critical analysis of the AM technologies, highlighting their suitability for a future batch production (see subchapter 2.1.5).
  - A critical analysis of the QA chain of the AM PBF processes (see subchapter 3.1).
  - A critical analysis of the existing control schemes and in-process measurements approaches for the AM PBF processes (see subchapters 3.4, 3.5 and 3.6).
  - A comprehensive approach for the determination of the inline defects and failures appearing during the build cycles of the AM PBF processes (see subchapter 4.3).
  - Determination of the tasks for inline QC and in-situ optimisation of AM PBF processes (see subchapter 4.4).
  - Development of an overall concept for an inline QC system in the AM PBF production: appropriate technologies and sensors for data and signal acquisition, design of the HW architecture and the SW modular framework (see chapter 5).
- Applied research contributions:
  - Development and implementation of a modular HW platform for an inline QC system for the AM PBF production, using the example of SLS (see subchapter 6.1)
  - Development and implementation of a SW engine for the inline quality control of the parts during the SLS build process, with two main IIP routines: the *Inline Image Acquisition* and the *Quality Control* one (see subchapter 6.2)
  - Definition and implementation of a part quality protocol, for overall description of part's quality data, obtained during the layer-by-layer inspection (see subchapter 6.2.3).
  - Design, development and implementation of a unique, novel and innovative inline QC system for a reliable AM, through the layer-by-layer inspection during the AM PBF processes, using the example of SLS (see subchapter 7.1).



A dissemination of the conducted research work, representing the content of this doctoral thesis, has been carried out in [78] [79] [30] [31] [80] [81] [33].

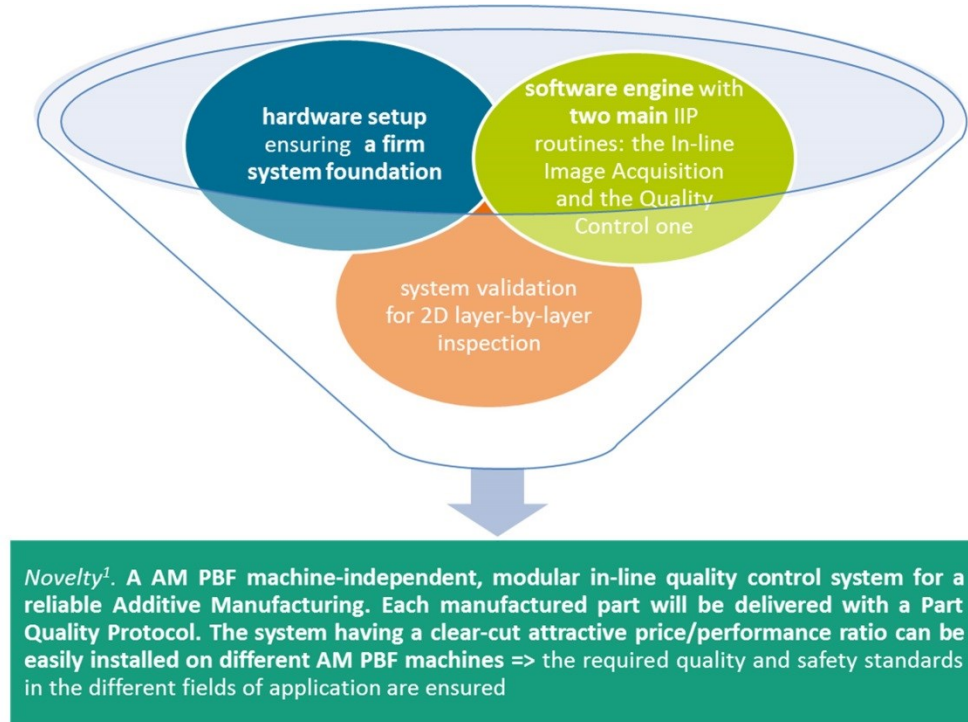


Figure 8.1. Inline QC system for the AM PBF processes: forming the main distinct contribution to the knowledge of the QC in the field of tomorrow's additive production.

In conclusion, the main objective of the thesis was achieved. Through all distinct contributions, culminating in the inline QC system for the AM PBF production (see Figure 8.1), the first enabler of the large AM PBF batch production was realised, namely a system that:

- ensures a machine-independent approach;
- is able to work uninterruptedly, performing 100% inline inspection, hence improved AM product quality, higher yields and lower production costs;
- can detect quality failures during AM PBF production processes;
- creates a quality report of the produced part;
- can indirectly cancel the production process in extreme cases;
- has a clear-cut attractive price/performance ratio.

Nevertheless, the results of this doctoral thesis open up further research work topics:

- Part inspection on 3D level: knowing the layer thickness, a mathematical reconstruction algorithm of the 2D actual layer's images can be implemented. In this way, the 3D actual data of each manufactured part can be obtained; nominal/actual comparisons as well as inspection operations (e.g. part surface inspection) can take place on 3D level.
- Realising a conjunction with production accompanying methods for the operational quality assurance, such as the SPC (Statistical Process Control): exemplary a statistic module concerning the process capability

can be implemented (e.g. plotting the history of a value and calculate its Process Capability Index) and added to the system.

- Direct correlation between each detected failure and its associated production parameters for a future applicable in-situ process optimisation.
- Additional sensors (e.g. microbolometer or multispectral imaging sensors) can be added to the modular HW platform. Corresponding data evaluation algorithms can be implemented in order to cover the overall tasks identified (e.g. applied powder layer thickness variation). In case of a multi sensor HW platform, data fusions algorithms to improve the image data and the system sensitivity, can be added to the modular SW platform of the developed inline QC system for the AM PBF processes.

## ANNEXES

### Annex 1: Hardware components

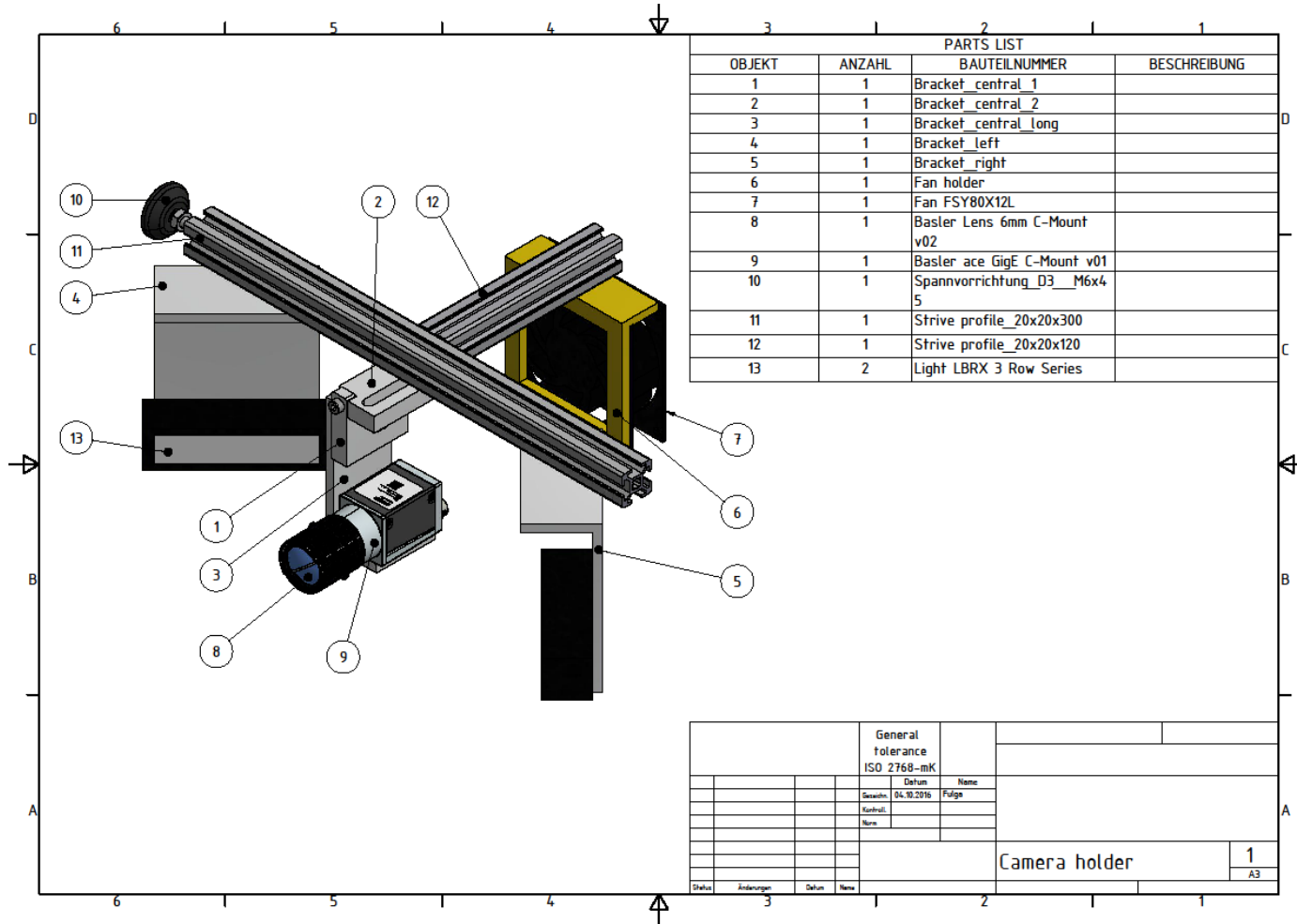
#### Basler Low-Cost camera acA3800-10gm

Basler ace	acA2500-14gm/gc	acA2500-20gm/gc	acA3800-10gm/gc	acA4600-7gc
<b>Camera</b>				
Resolution (H×V pixels)	2592×1944	2590×2048	3856×2764	4608×3288
Sensor	Aptina MT9P031	PYTHON 5000	Aptina MT9J003	Aptina MT9F002
Sensor Size (optical)	1/2.5"	1"	1/2.3"	1/2.3"
Sensor Technology	CMOS, rolling shutter	CMOS, global shutter	CMOS, rolling shutter	CMOS, rolling shutter
Pixel Size [µm²]	2.2×2.2	4,8×4,8	1,67×1,67	1,4×1,4
Frame Rate [fps]*	14	21	10	7
Mono/Color	Mono/Color	Mono/Color	Mono/Color	Color
Video Output Format	Mono (8, 12, 12 Packed), Bayer GB (8, 12, 12 Packed), YUV 4:2:2 (Packed, YUYV Packed)	Mono (8, 10, 10 Packed), Bayer BG (8, 10, 10 Packed), YUV 4:2:2 (Packed, YUYV Packed)	Mono (8, 12, 12 Packed), Bayer BG (8, 12, 12 Packed), YUV 4:2:2 (Packed, YUYV Packed)	Mono 8, Bayer BG (8, 12, 12 Packed), YUV 4:2:2 (Packed, YUYV Packed)
Interface	Fast Ethernet (100 Mbit/s) or Gigabit Ethernet (1000 Mbit/s)			
Synchronization	Via external trigger, via the Ethernet connection or free run			
Exposure Control	Via external trigger or programmable via the camera API			
<b>Mechanical/Electrical</b>				
Housing Size (L × W × H)	42 mm × 29 mm × 29 mm			
Housing Temperature	Up to 50 °C			
Lens Mount	C, CS	C	C	C
Digital I/O	1 opto-isolated input/1 opto-isolated output + 1 GPIO (only acA2500-20gm/gc)			
Power Requirements	Via Power over Ethernet (IEEE 802.3af) or + 12VDC (±10%) via the camera's 6-pin Hirose connector (+24 VDC for acA2500-20gm/gc)			
Power Consumption (PoE/AUX)	2.5W/2.2W	-3.5W	3.5W/3.3W	3.5W/3.3W
Weight (typical)	<90 g			
Conformity	CE, FCC, IP30, RoHS, PoE (IEEE 802.3af), UL (in preparation for acA2500-20gm/gc)			
<b>Software Environment</b>				
Driver	Basler pylon Camera Software Suite or 3rd party GigE Vision Software			

**Basler Lens C125-0618-5M**

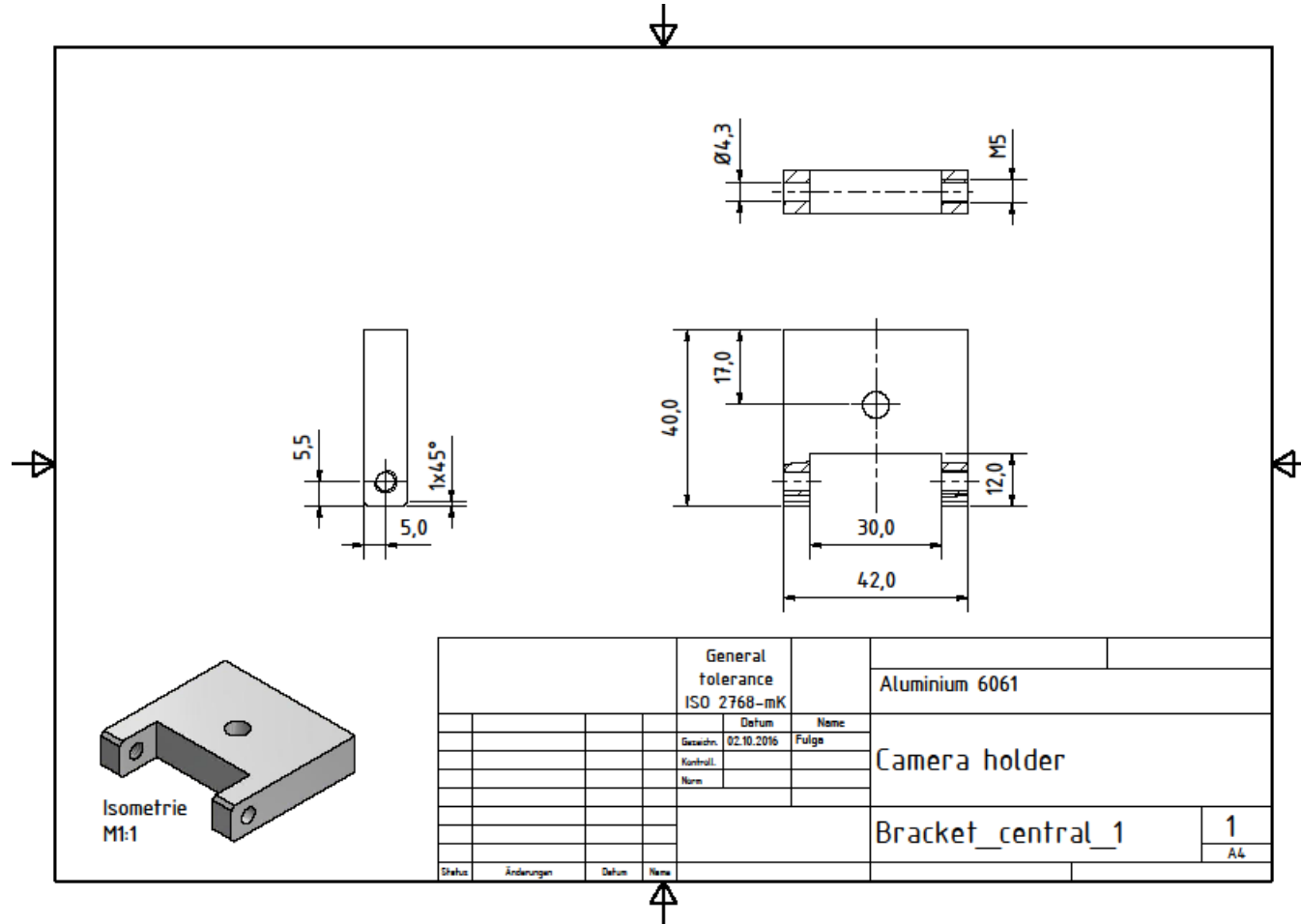
Optical Specifications		
Focal length $f'$	6.05 mm $\pm$ 5 %	
Aperture range	$f/1.8$ to $f/22$	
Image circle	7.3 mm (1/2.5" format)	
Focus range	0.1 m to infinity	
Optimum working distance	0.5 m	
Optimum magnification	0.012	
Relative illumination at full aperture	At least 55 % (see Figure 3)	
Resolution (25 % MTF, center, full aperture)	Designed for 230 LP/mm (2.2 $\mu$ m pixel size, see Figure 6)	
Resolution at 5.8 mm image diagonal (80 % of a Basler acA2500-14xx camera)	At least 50 % of resolution in image center	
Optical distortion	Typical -9.9 % (barrel distortion, see Figure 4)	
Angle of view, 1/2.5" format (using a Basler acA2500-14xx camera)	horizontal	52.4° (@ MOD) to 53.1°
	vertical	39.6° (@ MOD) to 40.1°
Angle of view, 1/3" format (using a Basler acA1300-30xx camera)	horizontal	44.9° (@ MOD) to 45.5°
	vertical	33.6° (@ MOD) to 34.1°
Wavelength range	Visible (400 to 700 nm, see Figure 5)	
Pupil magnification $\beta'_p$	3.95	
Chief ray angle, CRA	8.2°	
Front focal length, $s_F$	12.62 mm	
Back focal length, $s'_F$	9.75 mm	
Principal point separation, HH'	22.45 mm	
Entrance pupil position, $s_{EP}$	14.15 mm	
Exit pupil position, $s'_{AP}$	-14.16 mm	
Overall optical length, $d$	37.42 mm	

System set-up technical drawing: camera holder

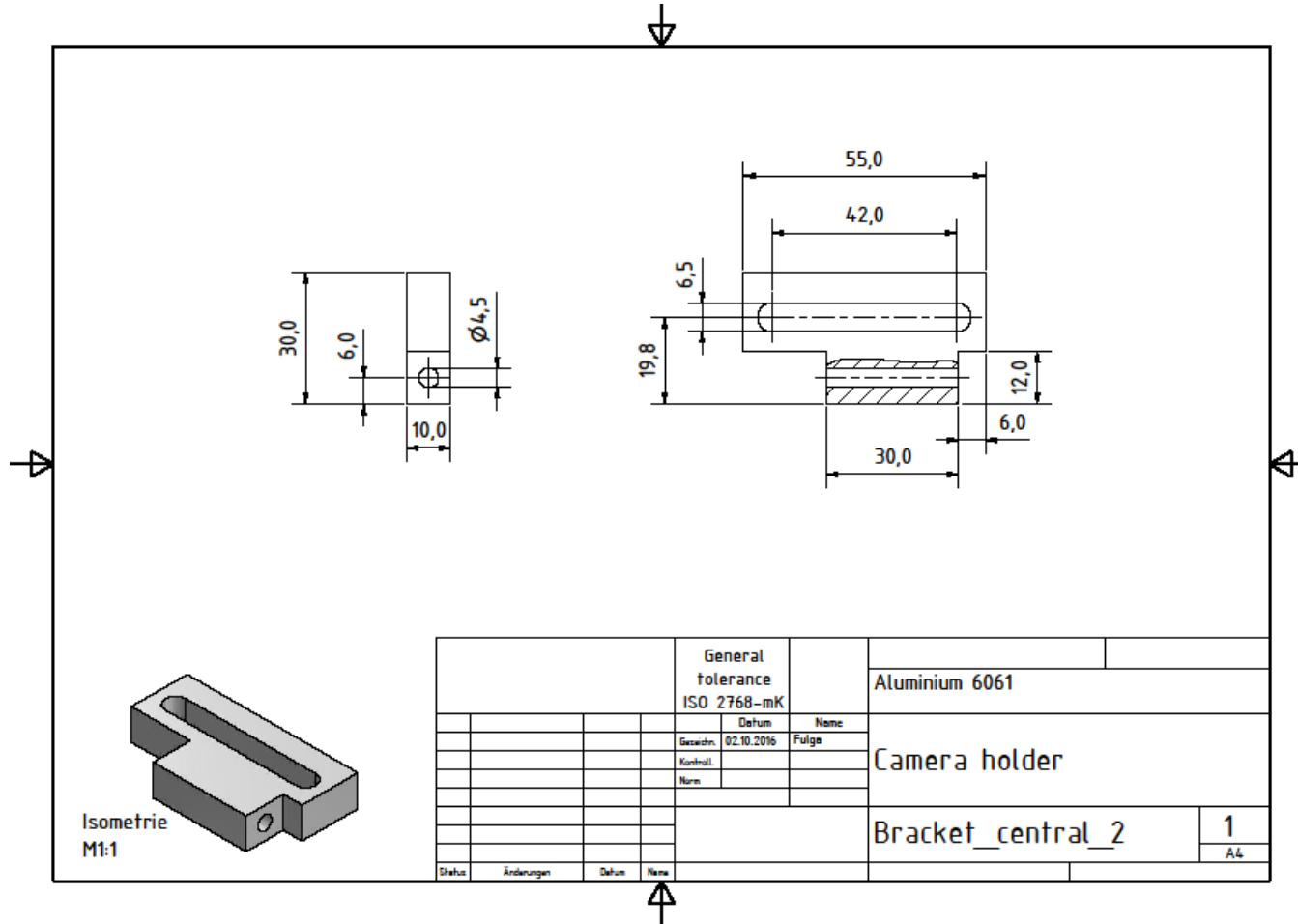


General tolerance ISO 2768-mK			
Datum	Name		
Geschn: 04.10.2016	Folge		
Kontroll:			
Rev:			
		Camera holder	
		1	
		A3	
Status	Änderungen	Datum	Name

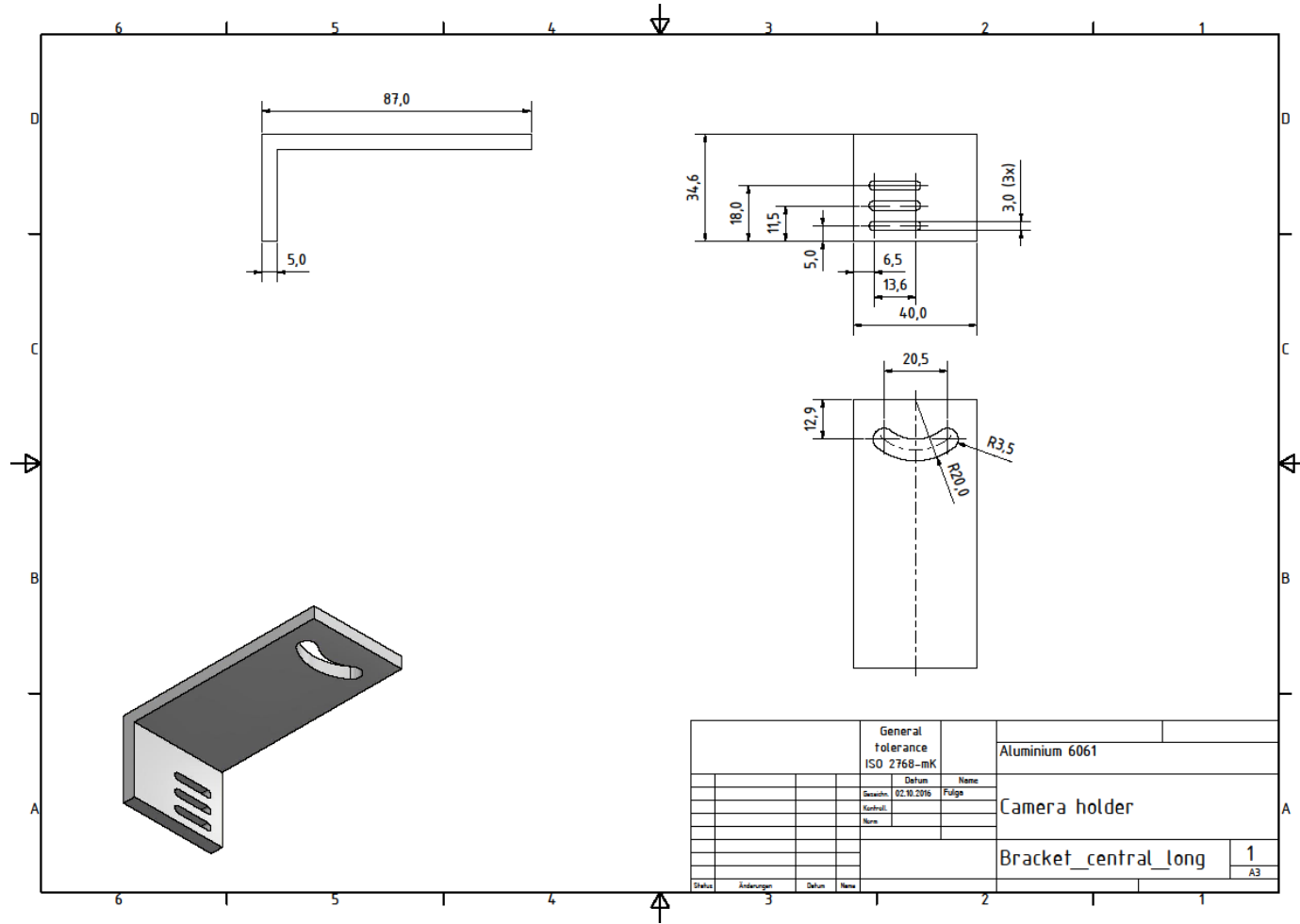
System set-up technical drawing: bracket\_central\_1



System set-up technical drawing: bracket\_central\_2

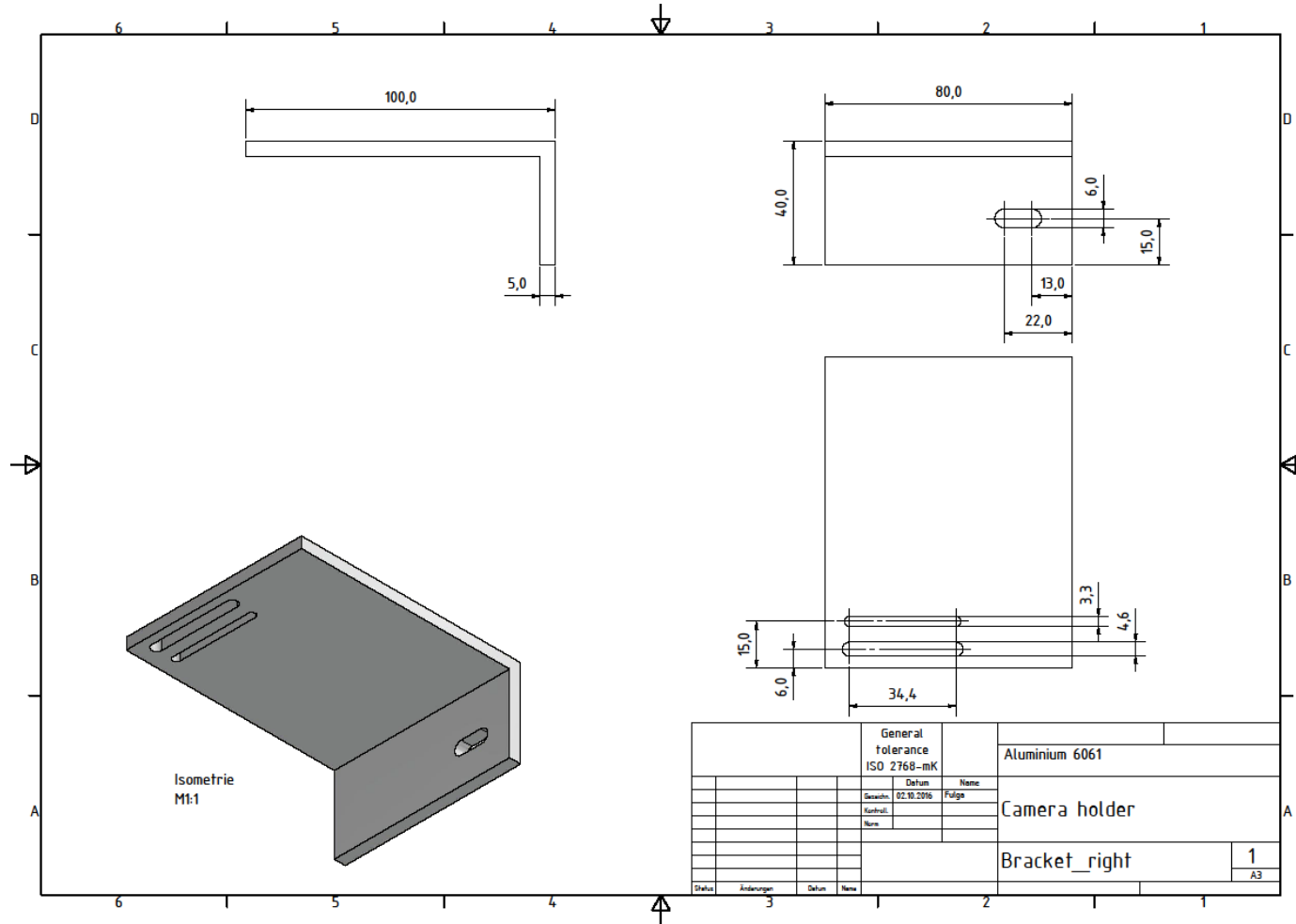


System set-up technical drawing: bracket\_central\_long

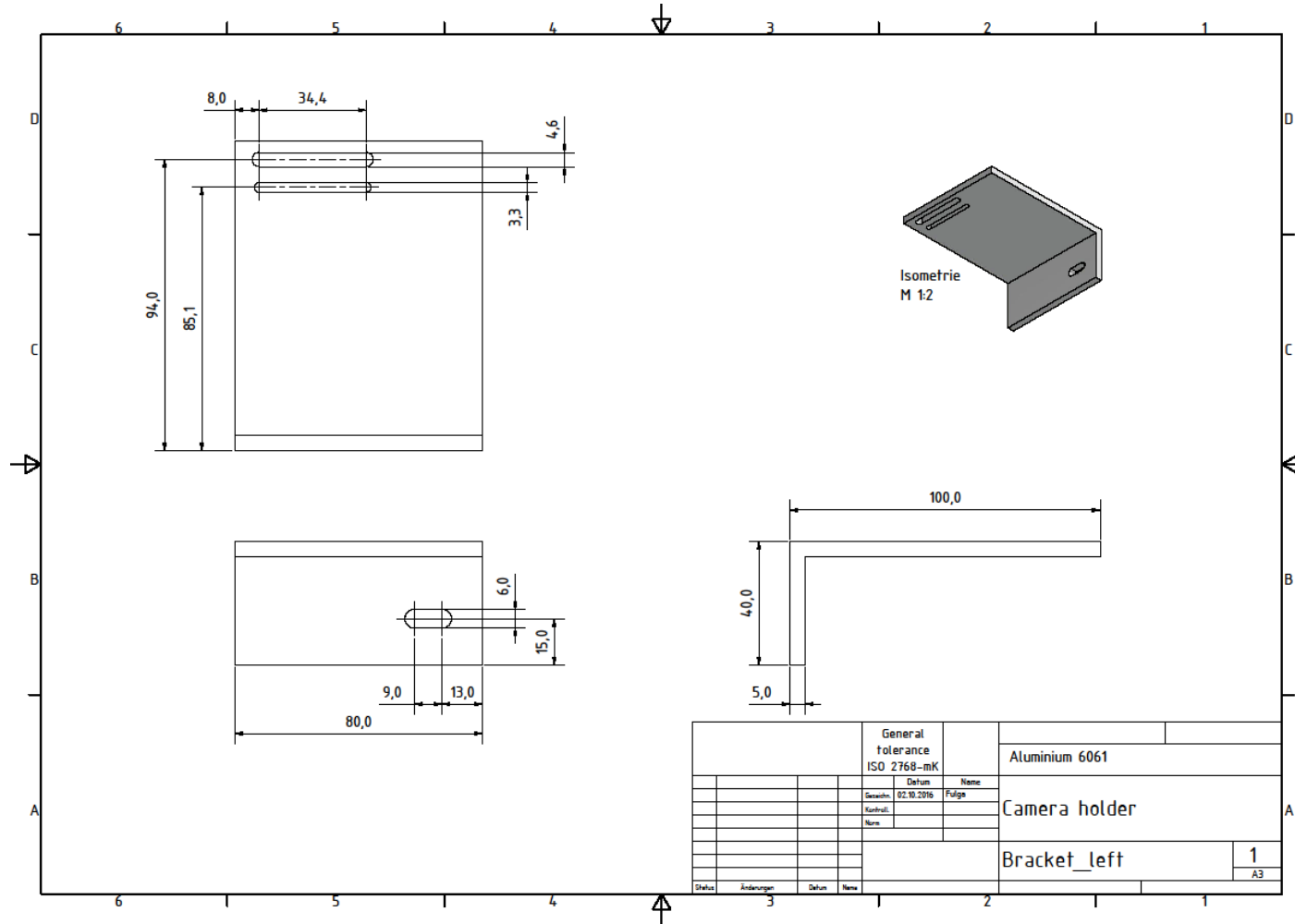




System set-up technical drawing: bracket\_right



System set-up technical drawing: bracket\_left



General tolerance ISO 2768-mK		Aluminium 6061	
Gezeichnet	02.10.2016	Name	
Kontrolliert		Folgt	
		Camera holder	
		Bracket_left	
		1	
		A3	

## Annex 2: Software - Framework

### **IQ4AMInterfacingClass – central interface in order to interact with the framework**

```

namespace AMPBFQualityControl {
IQ4AMInterfacingClass::IQ4AMInterfacingClass()
{
    datastore = DatastorePtr(new Datastore());
}

IQ4AMInterfacingClass::~IQ4AMInterfacingClass()
{
    // no op.
}

std::string IQ4AMInterfacingClass::getLastErrorMessage() const
{
    return lastErrorMessage;
}

IQ4AMInterfacingClass IQ4AMInterfacingClass::getDefaultInstance()
{
    IQ4AMInterfacingClass ndt;
    // Create default command factory.
    std::shared_ptr<CommandFactory> cmdFac =
        std::shared_ptr<CommandFactory>(new ProcessingFactory());
    ndt.registerCommandFactory(cmdFac);
    return ndt;
}

int IQ4AMInterfacingClass::executeJob(const Parameter& params)
{
    // Call with empty observer vector.
    return executeJob(params, std::vector<ObserverPtr>());
}

int IQ4AMInterfacingClass::executeJob(const Parameter& params,
std::vector<ObserverPtr>& observers)
{
    CommandPtr cmd;
    int errorCode = 0;
    try
    {
        cmd = createCommandObj(params);
        // Add all observers to the for
        for(std::vector<ObserverPtr>::iterator it = observers.begin(); it !=
observers.end(); ++it)
        {
            cmd->addObserver(*it);
        }
    }
}

```

```

        cmd->execute();
        // Get the last report.
        report = cmd->getReport();
    } catch(std::exception& ex)
    {
        // Set error code and the error message from the given exception.
        errorCode = 1;
        lastErrorMsg = ex.what();
        // Lets see if this is an instance of one of our own exceptions with
        // a error message. Extract it if possible.
        try
        {
            // since we are trying to cast a reference, failure will not
            // a null pointer but rather in a bad_cast exception. Gonna
            // catch em all!
            parameter_error ndtEx = dynamic_cast<parameter_error
            &>(ex);
            lastErrorMsg = ndtEx.errorMsg;
        } catch(std::bad_cast)
        {
            //no op.
        }
    }
    return errorCode;
}

const ReportNodePtr IQ4AMInterfacingClass::getReport() const
{
    return report;
}

void
IQ4AMInterfacingClass::registerCommandFactory(std::shared_ptr<CommandFactor
y> cmdFactory)
{
    // Change datastore to "our" shared store so its equivalent to all registered
    factories.
    cmdFactory->setDatastore(datastore);
    commandFactories.insert(cmdFactory);
}

CommandPtr IQ4AMInterfacingClass::createCommandObj(const Parameter&
params)
{
    // Create with the use of all processing factories one or more command
    objects
    // which (in case of multiple ones) will be wrapped in a composite
    command.
    std::vector<CommandPtr> commands;
    for(auto f : commandFactories)

```

```

    {
        CommandPtr cmd = f->createCommand(params);
        if(cmd != 0)
        {
            commands.push_back(cmd);
        }
    }

    if(commands.size() == 1)
    {
        // Check if we have only one command in the vector, if so we can
directly
        // return this.
        return commands.front();
    } else if(commands.size() == 0)
    {
        // No command found. Throw a parameter error.
        throw parameter_error("No suitable command found.");
    }
    return CommandPtr(new CompositeCommand(datastore, params,
commands));
}
}

```

### Parameter Class

```

namespace AMPBFQualityControl {
Parameter::Parameter(const std::map<std::string, std::string> &data)
    : data(data)
{
    // no op.
}

Parameter::Parameter()
{
    // no op.
}

Parameter::Parameter(const Parameter &rhs) :
    data(rhs.data)
{
    // no op.
}

std::string Parameter::getString(const std::string& key) const
{
    try {
        return data.at(boost::to_upper_copy(key));
    } catch(...) {
        throw parameter_error("Key was not found (or error during
conversion): "+key);
    }
}

```

```
}

bool Parameter::hasKey(const std::string& key) const
{
    return (data.find(boost::to_upper_copy(key)) != data.end());
}

int Parameter::getInt(const std::string& key) const
{
    // If no conversion could be performed, an invalid_argument exception is
    // thrown.
    // If the value read is out of the range (or not in the map at all) of
    // representable
    // values by an int, an out_of_range exception is thrown.
    try {
        return std::stoi(data.at(boost::to_upper_copy(key)));
    } catch(...) {
        throw parameter_error("Key was not found (or error during
conversion): "+key);
    }
}

double Parameter::getFloat(const std::string& key) const
{
    try {
        return std::atof(data.at(boost::to_upper_copy(key)).c_str());
    } catch(...) {
        throw parameter_error("Key was not found (or error during
conversion): "+key);
    }
}
}
```

### ParameterBuilder Class

```
namespace AMPBFQualityControl {

ParameterBuilder::ParameterBuilder()
{ //no op. }

ParameterBuilder::~ParameterBuilder()
{ //no op.}

ParameterBuilder::ParameterBuilder(const ParameterBuilder& rhs) : data(rhs.data)
{ //no op.}

Parameter ParameterBuilder::build()
{
    return Parameter(data);
}

ParameterBuilder& ParameterBuilder::setValue(std::string key, std::string value)
```

```

{
    convertToUpper(key);
    if(key == "CMD")
    {
        convertToUpper(value);
    }
    data[key] = value;
    return *this;
}

ParameterBuilder& ParameterBuilder::setValue(std::string key, int value)
{
    convertToUpper(key);
    data[key] = std::to_string(value);
    return *this;
}

ParameterBuilder& ParameterBuilder::setValue(std::string key, double value)
{
    convertToUpper(key);
    data[key] = std::to_string(value);
    return *this;
}

void ParameterBuilder::clear()
{
    data.clear();
}

void ParameterBuilder::checkUpperCase(std::string& key, std::string& value) const
{
    if(key == "CMD")
    {
        convertToUpper(value);
    }
}

void ParameterBuilder::convertToUpper(std::string& str) const
{
    std::transform(str.begin(), str.end(), str.begin(), ::toupper);
}

} // end namespace AMPBFQualityControl

DataCommand Class
namespace AMPBFQualityControl {

DataCommand::DataCommand(DatastorePtr datastore, const Parameter& params) :
    datastore(datastore),
    params(params),

```

```
        report(TextNode::create("Report", ""))
    {
        // no op.
    };

DataCommand::~~DataCommand()
{
    //no op.
};

DataCommand::DataCommand(const DataCommand& rhs) :
    params(rhs.params), progress(0)
{
    //no op.
}

const ReportNodePtr DataCommand::getReport() const
{
    return report;
}

float DataCommand::getProgress() const
{
    return progress;
}

void DataCommand::setProgress(float progress)
{
    if(progress < 0 || progress > 100)
    {
        // Not in accepted range. Do nothing.
        return;
    }

    this->progress = progress;
    // Progress of this object has changed. Tell this
    // our observers!
    notifyObserver();
}

void DataCommand::execute()
{
    // Validate if all parameter are existing.
    validateParams();

    // Call the custom workload.
    doWork();
}
}
```



**CommandFactory Class**

```
namespace AMPBFQualityControl {
void CommandFactory::setDatastore(DatastorePtr store)
{
    if(store == 0)
    {
        throw std::invalid_argument("DatastorePtr can not be null.");
    }

    this->store = store;
}
}
```

**Datastore Class**

```
using namespace AMPBFQualityControl;
Datastore::Datastore()
{
}

Datastore::~Datastore()
{
    // no op. Since smart-pointer is used
}

Datastore::Datastore(const Datastore& rhs) :
    storage(rhs.storage)
{
    // no op.
}

void Datastore::clearAll()
{
    storage.clear();
}

ConstDataPtr Datastore::getData(std::string key) const
{
    try {
        return storage.at(key);
    } catch(std::out_of_range ex) {
        return std::shared_ptr<Data>(0);
    }
}

void Datastore::setData(std::string key, DataPtr data)
{
    if(data == 0)
    {
        throw std::invalid_argument("DataPtr can not be null.");
    }
    storage[key] = data;
}
```

```
bool Datastore::hasData(std::string key)
{
    return !(storage.find(key) == storage.end());
}
```

```
void Datastore::clearData(std::string key)
{
    storage.erase(key);
}
```

### FilterCommand Class

```
namespace AMPBFQualityControl {
void FilterCommand::doWork()
{
    using namespace std;
    string keySource = params.getString("DATASOURCE");

    if(!datastore->hasData(keySource))
    {
        throw parameter_error("Datasource is not present. Open the source
first.");
    }
    input = datastore->getData(keySource);
    // Let the child commands setup their data output.
    setupOutputData();
    // Call the filter logic.
    filter();
}

void FilterCommand::validateParams() const
{
    if(params.getString("DATASOURCE").empty())
    {
        throw parameter_error("FilterCommand: DATASOURCE can not be
empty.");
    }
}
}
```

### Annex 3: Software - IIP

#### Median NM

```

void applyMedianNM(IQ4AM_IIP::image Input, IQ4AM_IIP::image Output, long
matrixHeight, long matrixWidth)
{
    uchar *pucPixel, *pucLine, *ucptrP, *ucptrQ;
    long i, j, k, l, left, right;
    long ParWidth, ParHeight, IPixelCounter, INumberElements, IMiddle;
    uchar *ucptrMem, *ucptrMedian;
    unsigned char ucHelp, uc;

    if((ucptrMem=(uchar *)malloc(Input.vpitch*Input.height))==NULL)
    return ;//IQ4AM_ERR_NO_MEMORY;
    INumberElements = matrixWidth*matrixHeight;
    if((ucptrMedian=(uchar *)malloc(INumberElements))==NULL)
    {
        free(ucptrMem);
        return ;//IQ4AM_ERR_NO_MEMORY;
    }
    // reduce window corresponding to matrix size
    ParWidth = Input.width+1-matrixWidth;
    ParHeight = Input.height+1-matrixHeight;
    // Filter operation
    pucLine = Input.st;
    IPixelCounter = 0;
    IMiddle = INumberElements / 2;

    for(i=0; i<ParHeight; i++, IPixelCounter++, pucLine+=Input.vpitch)
    {
        for(j=0, pucPixel=pucLine; j<ParWidth; j++, IPixelCounter++,
pucPixel++)
        {
            // get median of matrix
            // pucPixel points the upper left corner of the matrix
            // fill sorting list
            ucptrP = pucPixel;
            for(k=0; k<matrixWidth; k++, ucptrP++)
            {
                ucptrQ = ucptrP;
                for(l=0; l<matrixHeight; l++,
ucptrQ+=Input.vpitch)
                    ucptrMedian[k*matrixHeight+l] = *ucptrQ;
            }

            // searching the median with SELECT
            left = 0;
            right = INumberElements-1;
            while(right > left)
            {
                uc = ucptrMedian[right];

```

```

        k = left-1;
        l = right;
        do
        {
            for(; ucptrMedian[++k]<uc; );
            for(; ucptrMedian[--l]>uc; );
            ucHelp = ucptrMedian[k];
            ucptrMedian[k] = ucptrMedian[l];
            ucptrMedian[l] = ucHelp;
        } while(l>k);
        ucptrMedian[l] = ucptrMedian[k];
        ucptrMedian[k] = ucptrMedian[right];
        ucptrMedian[right] = ucHelp;
        if(k>=IMiddle) right = k-1;
        if(k<=IMiddle) left = k+1;
    }
    // save new value
    ucptrMem[IPixelCounter] = ucptrMedian[IMiddle];
}
// enter new values to original image
pucLine = Output.st;
IPixelCounter = 0;
// long IOffset = (IHorValue/2)*(IHPitch+IVPitch);// add offset to center
point of matrix
long IOffsetH = (matrixWidth/2);
long IOffsetV = (matrixHeight/2)*(Output.vpitch);

for(i=0; i<ParHeight; i++, IPixelCounter++, pucLine+=Output.vpitch)
{
    for(j=0, pucPixel=pucLine; j<ParWidth; j++, IPixelCounter++,
pucPixel++)
        *(pucPixel+IOffsetH+IOffsetV) =
ucptrMem[IPixelCounter];
}
free (ucptrMem);
free (ucptrMedian);
}

```

### Dilatation

```

void applyDilation(IQ4AM_IIP::image Input, IQ4AM_IIP::image Output, int
noIterations, int behaviour = 0)
{
#ifdef OPEN_CV_C
    IplImage*
cvInput=cvCreateImageHeader(cvSize(Input.vpitch,Input.height),IPL_DEPTH_8U,
1);
    cvSetImageROI(cvInput, cvRect(0, 0, Input.width, Input.height));
    cvInput->imageData=(char *)Input.st;

```

```

IplImage* cvOutput =
cvCreateImageHeader(cvSize(Output.vpitch,Output.height), IPL_DEPTH_8U, 1);
cvSetImageROI(cvOutput, cvRect(0, 0, Input.width,Input.height));
cvOutput->imageData=(char *)Output.st;

if (behaviour == 0)
{ cvDilate(cvInput, cvOutput,NULL, noIterations); }
else
{ cvErode(cvInput, cvOutput,NULL, noIterations); }
#endif
#ifdef OPEN_CV_CPP
int imageType = CV_8UC1;
if(Input.bpp == 3)
    imageType = CV_8UC3;

const long srcWidth = Input.vpitch/Input.bpp;
const long dstWidth = Output.vpitch/Output.bpp;
Mat src(Size(srcWidth, Input.height), imageType, Input.st);
Mat srcROI(src, Rect(0, 0, Input.width, Input.height));
Mat dst(Size(dstWidth, Output.height), imageType, Output.st);
Mat dstROI(dst, Rect(0, 0, Input.width, Input.height));
if (behaviour == 0)
{ dilate(srcROI, dstROI, Mat(), Point(-1, -1), noIterations); }
else
{ erode(srcROI, dstROI, Mat(), Point(-1, -1), noIterations); }
#endif
}

```

### Binarisation

```

void binImg8(IQ4AM_IIP::image Input, IQ4AM_IIP::image Output, long
IForeground, long IBackground, long ILowerThreshold, long IUpperThreshold)
{
    unsigned char* VarIn=Input.st;
    unsigned char* VarOut=Output.st;

    for(int IY=0; IY<Input.height; IY++)
    {
        VarIn = Input.st + IY*Input.vpitch;
        VarOut = Output.st + IY*Output.vpitch;

        for(int IX=0; IX < Input.width; IX++)
        {
            if(VarIn[IX]> IUpperThreshold){
                VarOut[IX]= IForeground;
            }
            else{
                if(VarIn[IX] <= ILowerThreshold){
                    VarOut[IX]= IBackground;
                }
                else{
                    VarOut[IX]=VarIn[IX];
                }
            }
        }
    }
}

```

```
    }  
    }  
}
```

### Erosion

```
void applyErosion(IQ4AM_IIP::image Input, IQ4AM_IIP::image Output, int  
noIterations, int behaviour = 0)  
{  
#ifdef OPEN_CV_C  
    IplImage*  
cvInput=cvCreateImageHeader(cvSize(Input.vpitch,Input.height),IPL_DEPTH_8U,  
1);  
    cvSetImageROI(cvInput, cvRect(0, 0, Input.width,Input.height));  
    cvInput->imageData=(char *)Input.st;  
  
    IplImage* cvOutput =  
cvCreateImageHeader(cvSize(Output.vpitch,Output.height), IPL_DEPTH_8U, 1);  
    cvSetImageROI(cvOutput, cvRect(0, 0, Input.width,Input.height));  
    cvOutput->imageData=(char *)Output.st;  
  
    if (behaviour == 0)  
    { cvErode(cvInput, cvOutput,NULL, noIterations); }  
    else  
    { cvDilate(cvInput, cvOutput,NULL, noIterations); }  
#endif  
#ifdef OPEN_CV_CPP  
    int imageType = CV_8UC1;  
    if(Input.bpp == 3)  
        imageType = CV_8UC3;  
  
    const long srcWidth = Input.vpitch/Input.bpp;  
    const long dstWidth = Output.vpitch/Output.bpp;  
    Mat src(Size(srcWidth, Input.height), imageType, Input.st);  
    Mat srcROI(src, Rect(0, 0, Input.width, Input.height));  
    Mat dst(Size(dstWidth, Output.height), imageType, Output.st);  
    Mat dstROI(dst, Rect(0, 0, Input.width, Input.height));  
    if (behaviour == 0)  
    { erode(srcROI, dstROI, Mat(), Point(-1, -1), noIterations); }  
    else  
    { dilate(srcROI, dstROI, Mat(), Point(-1, -1), noIterations); }  
  
#endif  
}
```

### Gauss filter

```
void applyGauss(IQ4AM_IIP::image Input, IQ4AM_IIP::image Output, int filterSize,  
double stdDeviation)  
{  
    if(filterSize%2 == 0)
```

```

        filterSize++;
    int imageType = CV_8UC1;
    if(Input.bpp == 3)
        imageType = CV_8UC3;
    const long srcWidth = Input.vpitch/Input.bpp;
    const long dstWidth = Output.vpitch/Output.bpp;
    Mat src(Size(srcWidth, Input.height), imageType, Input.st);
    Mat srcROI(src, Rect(0, 0, Input.width, Input.height));
    Mat dst(Size(dstWidth, Output.height), imageType, Output.st);
    Mat dstROI(dst, Rect(0, 0, Input.width, Input.height));
    GaussianBlur(srcROI, dstROI, Size(filterSize, filterSize), stdDeviation);
}

```

### Local binarisation

`void` applyLocalBinarisation(IQ4AM\_IIP::image Input, IQ4AM\_IIP::image Output, `int` w, `double` sensitivity)

```

{
    double k = 0.5 + (sensitivity / 1000.0);
    int imageType = CV_8UC1;
    if ( w % 2 == 0 )
    {
        w +=1;
    }
    const long srcWidth = Input.vpitch/Input.bpp;
    const long srcHeight = Input.height;
    const long dstWidth = Output.vpitch/Output.bpp;
    Mat src(Size(srcWidth, Input.height), imageType, Input.st);
    Mat srcROI(src, Rect(0, 0, Input.width, Input.height));
    Mat dst(Size(dstWidth, Output.height), imageType, Output.st);
    Mat dstROI(dst, Rect(0, 0, Input.width, Input.height));
    adaptiveThreshold(srcROI, dstROI, 255, ADAPTIVE_THRESH_MEAN_C,
    THRESH_BINARY, w, sensitivity);
}

```

### Contrast spread

`void` applyContrastSpread(IQ4AM\_IIP::image Input, IQ4AM\_IIP::image Output, `int` fromMin, `int` fromMax, `int` toMin, `int` toMax)

```

{
    TPixelPointer pSrcLine,pDstLine;
    pSrcLine = Input.st;
    pDstLine = Output.st;
    // do operation
    for(int j=0;j<Input.height;j++)
    {
        TPixelPointer pSrc = pSrcLine;
        TPixelPointer pDst = pDstLine;
        for(int i=0;i<Input.width;i++)
        {
            (*pDst) = f( *pSrc, fromMin, fromMax, toMin, toMax);
            pDst ++;
            pSrc ++;
        }
    }
}

```

```

        }
        pSrcLine += Input.vpitch;
        pDstLine += Output.vpitch;
    }
}

```

```

inline unsigned char f(unsigned char i, int fromMin, int fromMax, int toMin, int
toMax)
{
    int temp = (i-fromMin) * (toMax-toMin)/(fromMax-fromMin) + toMin;
    return (temp < 0)?0:( temp>255)?255:(unsigned char) temp;
}

```

### Ring Radial Mean / Circle fit to a given set of 2D data points.

```

long ApplyFilterRingRadialMean(IQ4AM_IIP::image img,int centerX, int centerY, int
innerRadius, int outerRadius, double dXToYRatio )
{
    double *sumtable = new double[outerRadius + 1]; // 0...innerRadius will be
ignored
    double *cnttable = new double[outerRadius + 1]; // 0...innerRadius will be
ignored
    memset(sumtable,0,(outerRadius + 1) * sizeof(double));
    memset(cnttable,0,(outerRadius + 1) * sizeof(double));
    unsigned char* ucpLine = img.st;
    for ( int j = 0; j< img.height; j++)
    {
        unsigned char* ucpPixel = ucpLine;
        for ( int i = 0; i< img.width; i++)
        {
            int dist = distance ( i,j, centerX, centerY);
            if ( dist <= outerRadius)
            {
                sumtable[dist] += (*ucpPixel);
                cnttable[dist] += 1;
            }
            ucpPixel++;
        }
        ucpLine += img.vpitch;
    }
    for ( int i = 0; i < outerRadius; ++i)
    {
        double dAvg = sumtable[i] / cnttable[i];
        sumtable[i] = dAvg;
    }

    ucpLine = img.st;
    for ( int j = 0; j< img.height; j++)
    {
        unsigned char* ucpPixel = ucpLine;
        for ( int i = 0; i< img.width; i++)

```



```

        {
            int dist = distance ( i,j, centerX, centerY);
            if ( dist <= outerRadius)
            {
                (*ucpPixel) = sumtable[dist];
            }
            ucpPixel++;
        }
        ucpLine += img.vpitch;
    }
    delete [] sumtable;
    delete [] cnttable;
    return IQ4AM_SUCCESS;
}

inline int distance ( int x, int y, int centerX, int centerY)
{
    double dist = sqrt((double)((x-centerX) * (x-centerX) +(y-centerY) * (y-
centerY)));
    return (int) dist;
}

```

**Edge detection by 'minimum contrast' method, calculate local thresholds by 'contrast method' - binary edge detection with local threshold list**

GetEdge (TIPAList \*ptPAList, long ParEdgeCount, long EdgeDirection, long MinContinuity)

```

{
    //TPixelPointer *pucpPixel;
    TPixelPointer ucpLastPixel;
    long *lptrPAList,*lptrPixel;
    long PListCounter = 0;
    long VPitch,PixelSize, ImageHeight, ImageWidth;
    PixelPointer pucImageBase;
    long ActEdgeDirection;
    long ActSegmentLength; // number of last pixels with same colour
    TPixelPointer pucPreEdgePixel,pucPostEdgePixel;
    long lPointReady = 0;
    GeoPoint tPSave;
    long lActEdgeDirectionSave = 0;
    unsigned char ucThresholdSave = 0;
    long lEdgeQualitySave = 0;
    // first reset the last result
    ActNoOfEdges = 0;
    if(ParEdgeCount < 0)
        return IQ4AM_ERR_PARM;
    if(ParEdgeCount > MaxNoOfEdges)
        ParEdgeCount = MaxNoOfEdges;
    // parameter check
    if (TGsiEdgeBinary::IsValid() == 0)
        return IQ4AM_ERROR;
}

```

```

if(ptPAList->GetPListAdr(lpPtrPAList) != IQ4AM_SUCCESS)
    return IQ4AM_ERROR;
if(ptPAList->GetPListLength(PListCounter) != IQ4AM_SUCCESS)
    return IQ4AM_ERROR;
// calculate the lower and upper Image Address Boundaries
if(tGsiFrame->GetVPitch(VPitch) != IQ4AM_SUCCESS)
    return IQ4AM_SUCCESS;
if(tGsiFrame->GetHPitch(PixelSize) != IQ4AM_SUCCESS)
    return IQ4AM_SUCCESS;
if(tGsiFrame->GetLinesPerFrame(ImageHeight) != IQ4AM_SUCCESS)
    return IQ4AM_SUCCESS;
if(tGsiFrame->GetPixelsPerLine(ImageWidth) != IQ4AM_SUCCESS)
    return IQ4AM_SUCCESS;
if(tGsiFrame->GetImageBaseAddress(pucImageBase) != IQ4AM_SUCCESS)
    return IQ4AM_SUCCESS;

ActNoOfEdges = 0;
if(PListCounter <= 0) // if list of plist <=0
    return IQ4AM_SUCCESS; // no edges detected
// get memory for list of thresholds
unsigned char *ucPtrThrList = (unsigned char *)malloc(PListCounter);
unsigned char *ucPtrThr;
ucPtrThr = ucPtrThrList;
GetListOfThresholds(lpPtrPAList,PListCounter,pucImageBase,ucPtrThrList);

// skip to next Pixel
lpPtrPixel = lpPtrPAList;
ucpLastPixel = pucImageBase+*lpPtrPixel;//pucpPixel;
PListCounter--;
ucPtrThr++;
lpPtrPixel++;
ActSegmentLength = 1;
// main loop through the IPAList
while((ActNoOfEdges<ParEdgeCount)&&(PListCounter>0))
{
    // if address is not valid, skip
    if(!(IsValidAddress(pucImageBase+*lpPtrPixel)))
    {
        do
        {
            //ucpLastPixel = *pucpPixel;
            lpPtrPixel++;
            PListCounter--;
            ucPtrThr++;
            //ActSegmentLength = 1;
            if(PListCounter <= 0)
                break;
        }
        while(!(IsValidAddress(pucImageBase+*lpPtrPixel)));
        if(PListCounter <= 0)
            continue;
    }
}

```

```

        // skip one more to have one valid pixel in back
        ucpLastPixel = pucImageBase+*lptrPixel;
        lptrPixel++;
        PListCounter--;
        ucptrThr++;

        ActSegmentLength = 1;
        continue;
    }
    // check whether there is no switch
    if( (*ucpLastPixel <= *ucptrThr)&&*(pucImageBase+(*lptrPixel)) <=
*ucptrThr)|| (*ucpLastPixel >*ucptrThr)&&*(pucImageBase+(*lptrPixel))
>*ucptrThr)|| (*ucptrThr <= 0)||(*ucptrThr >=255))
    {
        ucpLastPixel = pucImageBase+*lptrPixel;
        lptrPixel++;
        PListCounter--;
        ucptrThr++;

        ActSegmentLength++;
        continue;
    }
    // now we have a switch around the threshold
    pucPreEdgePixel = ucpLastPixel;
    pucPostEdgePixel= pucImageBase+*lptrPixel;
    // now check for the correct edge direction
    if(*(pucImageBase+(*lptrPixel)) > *ucptrThr)
    {
        // dark to bright edge candidate
        ActEdgeDirection = DARK_TO_BRIGHT_EDGE;

        // skip if wrong direction
        if(EdgeDirection == BRIGHT_TO_DARK_EDGE) // then skip
        {
            if((lPointReady != 0)&&(MinContinuity <= ActSegmentLength))
            {
                (ListOfEdges[ActNoOfEdges])->SetPoint(tpSave);
                (ListOfEdges[ActNoOfEdges])->
SetEdgeDirection(lActEdgeDirectionSave);
                (ListOfEdges[ActNoOfEdges])->SetGreyValue(ucThresholdSave);
                (ListOfEdges[ActNoOfEdges])->
SetEdgeQuality(lEdgeQualitySave);
                (ListOfEdges[ActNoOfEdges])->SetValid();
                ActNoOfEdges++;
            }
            ucpLastPixel = pucImageBase+*lptrPixel;
            lptrPixel++;
            PListCounter--;
            ucptrThr++;

            ActSegmentLength=1;

```

```

        IPointReady = 0;
        continue;
    }
    // enter edge candidate
    {
        double dX,dY;
        long IOffs,IXRes,IYRes,IThr;
        IThr = (long)*ucptrThr;
        IOffs = (pucPostEdgePixel-pucPreEdgePixel);
        tGsiFrame->
GetSubpCoordinatesToAddress(pucPreEdgePixel,SUBPIXEL_FACTOR_L,
        IOffs,IThr,IXRes,IYRes);
        dX = (double)IXRes;
        dX /= SUBPIXEL_FACTOR_D;
        dY = (double)IYRes;
        dY /= SUBPIXEL_FACTOR_D;

        if(IPointReady != 0)
        {
            (ListOfEdges[ActNoOfEdges])->SetPoint(tPSave);
            (ListOfEdges[ActNoOfEdges])->
SetEdgeDirection(IActEdgeDirectionSave);
            (ListOfEdges[ActNoOfEdges])->
SetGreyValue(ucThresholdSave);
            (ListOfEdges[ActNoOfEdges])->
SetEdgeQuality(IEdgeQualitySave);
            (ListOfEdges[ActNoOfEdges])->SetValid();
            ActNoOfEdges++;
        }
        IPointReady = 1;
        tPSave.Set(dX,dY);
        IActEdgeDirectionSave = ActEdgeDirection;
        ucThresholdSave = *ucptrThr;
        IEdgeQualitySave = 1000;

        ucpLastPixel = pucImageBase+*lptrPixel;
        lptrPixel++;
        PListCounter--;
        ucptrThr++;

        ActSegmentLength=1;
    }
}
else
{
    // dark to bright edge candidate
    ActEdgeDirection = BRIGHT_TO_DARK_EDGE;

    // skip if wrong edge direction
    if(EdgeDirection == DARK_TO_BRIGHT_EDGE) // then skip
    {

```

```

        if((IPointReady != 0)&&(MinContinuity <= ActSegmentLength))
        {
            (ListOfEdges[ActNoOfEdges])->SetPoint(tPSave);
            (ListOfEdges[ActNoOfEdges])->
SetEdgeDirection(IActEdgeDirectionSave);
            (ListOfEdges[ActNoOfEdges])->
SetGreyValue(ucThresholdSave);
            (ListOfEdges[ActNoOfEdges])->
SetEdgeQuality(IEdgeQualitySave);
            (ListOfEdges[ActNoOfEdges])->SetValid();
            ActNoOfEdges++;
        }

        ucpLastPixel = pucImageBase+*lptrPixel;
        lptrPixel++;
        PListCounter--;
        ucptrThr++;

        ActSegmentLength=1;
        IPointReady = 0;
        continue;
    }
    // enter edge candidate
    {
        double dX,dY;
        long IOfs,IXRes,IYRes,IThr;
        IThr = *ucptrThr;
        IOfs = (pucPreEdgePixel-pucPostEdgePixel);
        tGsiFrame->
GetSubpCoordinatesToAddress(pucPostEdgePixel,SUBPIXEL_FACTOR_L,
IOfs,IThr,IXRes,IYRes);

        dX = (double)IXRes;
        dX /= SUBPIXEL_FACTOR_D;
        dY = (double)IYRes;
        dY /= SUBPIXEL_FACTOR_D;

        if(IPointReady != 0)
        {
            (ListOfEdges[ActNoOfEdges])->SetPoint(tPSave);
            (ListOfEdges[ActNoOfEdges])->
SetEdgeDirection(IActEdgeDirectionSave);
            (ListOfEdges[ActNoOfEdges])->
SetGreyValue(ucThresholdSave);
            (ListOfEdges[ActNoOfEdges])->
SetEdgeQuality(IEdgeQualitySave);
            (ListOfEdges[ActNoOfEdges])->SetValid();
            ActNoOfEdges++;
        }
        IPointReady = 1;
        tPSave.Set(dX,dY);
    }

```

```

        IActEdgeDirectionSave = ActEdgeDirection;
        ucThresholdSave = *ucptrThr;
        IEdgeQualitySave = 1000;

        ucpLastPixel = pucImageBase+*lptrPixel;
        lptrPixel++;
        PListCounter--;
        ucptrThr++;

        ActSegmentLength=1;
    }
} // end of case 'dark to bright'
} // end of global 'while()'
if ((PListCounter <= 0)&&(MinContinuity <= ActSegmentLength))
{
    if(lPointReady != 0)
    {
        (ListOfEdges[ActNoOfEdges])->SetPoint(tPSave);
        (ListOfEdges[ActNoOfEdges])->
SetEdgeDirection(IActEdgeDirectionSave);
        (ListOfEdges[ActNoOfEdges])->SetGreyValue(ucThresholdSave);
        (ListOfEdges[ActNoOfEdges])->SetEdgeQuality(IEdgeQualitySave);
        (ListOfEdges[ActNoOfEdges])->SetValid();
        ActNoOfEdges++;
    }
}
free (ucptrThrList);
return IQ4AM_SUCCESS;
}

// local function to calculate list of local thresholds
// determine local threshold as: if pixel is not in region of an edge; else: mean
intensity of all pixels in local edge environment
GetListOfThresholds(long* lptrPAList, long lListLength,TPixelPointer tBase, unsigned
char* ucptrThrListParm)
{
    long i;
    unsigned char *ucptrP;
    unsigned char *ucptrThrList;
    unsigned char ucThr;
    long lIndexBack,lIndexFocus,lIndexFront;
    long lAccu,lAccuCounter,lContrast,lThrIndex;
    TPixelPointer tFront,tBack, tFocus;
    long lLastThr,lActThr, lVal,lMinMedium,lMaxMedium;
    ucptrThrList = ucptrThrListParm;
    // by default: initialize with global threshold
    for(ucptrP=ucptrThrList,i=0;i<lListLength;i++)
        *ucptrP++=0;//(unsigned char)lThreshold;
    // parameter check
    if(lEdgeWidth >= lListLength)
        return;
}

```

```

// setup pointers
lIndexBack = 0;
lIndexFront = lEdgeWidth;
lIndexFocus = lIndexFront/2;
for(i=0;i<lIndexFocus;i++)
    ucptrThrList[i] = 255;          lThrIndex = i-1;
tFront = tBase+lptrPAList[lIndexFront];
tBack = tBase+lptrPAList[lIndexBack];
lContrast = *tFront-*tBack;
if(lContrast < 0)
    lContrast = -lContrast;
// look for first homogenous area
while(lContrast >= lMinimumContrast)
{
    lIndexBack ++;
    lIndexFront++;
    lIndexFocus++;
    if (lIndexFront >= lListLength)
    {
        while(lThrIndex < lListLength)
        {
            ucptrThrList[lThrIndex]=0;//ucThr;
            lThrIndex++;
        }
        return;
    }
    tFront = tBase+lptrPAList[lIndexFront];
tBack = tBase+lptrPAList[lIndexBack];
    lContrast = *tFront-*tBack;
    if(lContrast < 0)
        lContrast = -lContrast;
}
// look for first contrast region
lAccu = 0;
lAccuCounter=0;
while(lContrast < lMinimumContrast)
{
    ucptrThrList[lIndexFocus]=255;
    lIndexBack++;
    lIndexFront++;
    lIndexFocus++;

    if (lIndexFront >= lListLength)
    {
        while(lThrIndex < lListLength)
        {
            ucptrThrList[lThrIndex]=0;
            lThrIndex++;
        }
        return;
    }
}

```

```

        tFront = tBase+lptrPAList[lIndexFront];
        tBack = tBase+lptrPAList[lIndexBack];
        tFocus = tBase+lptrPAList[lIndexFocus];
        lAccu += *tFocus;
        lAccuCounter++;
        lContrast = *tFront-*tBack;
        if(lContrast < 0)
            lContrast = -lContrast;
    }
    if(lAccuCounter > 0)
        lLastThr = lAccu/lAccuCounter;
    else
        lLastThr = 0;
    // now loop until end of list for next homogenous area
    ucThr = 0;
    lThrIndex = lIndexFocus;

    while (1)
    {
        lMinMedium = 255;
        lMaxMedium = 0;
        while(lContrast >= lMinimumContrast)
        {
            tFocus = tBase+lptrPAList[lIndexFocus];
            if(*tFocus < lMinMedium)
                lMinMedium = *tFocus;
            if(*tFocus > lMaxMedium)
                lMaxMedium = *tFocus;
            ucptrThrList[lIndexFocus]=0; // mark the entry as: edge

candidate

            lIndexBack++;
            lIndexFront++;
            lIndexFocus++;
            if (lIndexFront >= lListLength)
            {
                while(lThrIndex < lListLength)
                {
                    if(ucptrThrList[lThrIndex] < 255)
                        ucptrThrList[lThrIndex]=0;
                    lThrIndex++;
                }

                return;
            }
            tFront = tBase+lptrPAList[lIndexFront];
            tBack = tBase+lptrPAList[lIndexBack];
            lContrast = *tFront-*tBack;
            if(lContrast < 0)
                lContrast = -lContrast;
        }
    }

```



```

// look for next contrast region
lAccu = 0;
lAccuCounter=0;
while(lContrast < lMinimumContrast)
{
    ucptrThrList[lIndexFocus]=255; // mark the entry as:
no edge candidate
    lIndexBack++;
    lIndexFront++;
    lIndexFocus++;

    if (lIndexFront >= lListLength)
    {
        for (long k=lIndexBack;k<lListLength;k++)
            ucptrThrList[k]=255;

        if(lMinMedium <= lMaxMedium)
        {
            lVal =
lMinMedium+(lThresholdPercentage*(lMaxMedium-lMinMedium))/100;
            ucThr = (unsigned char)lVal;
        }
        else
            ucThr = 0;

        while(lThrIndex < lListLength)
        {
            if(ucptrThrList[lThrIndex] < 255)
                ucptrThrList[lThrIndex]=ucThr;
            lThrIndex++;
        }

        return;
    }
    tFront = tBase+lptrPAList[lIndexFront];
    tBack = tBase+lptrPAList[lIndexBack];
    tFocus = tBase+lptrPAList[lIndexFocus];
    lAccu += *tFocus;
    lAccuCounter++;
    lContrast = *tFront-*tBack;
    if(lContrast < 0)
        lContrast = -lContrast;
}
if (lAccuCounter <= 0)
    lActThr = lLastThr;
else
    lActThr = lAccu/lAccuCounter;
if(lMinMedium <= lMaxMedium)
{
    lVal = lMinMedium+(lThresholdPercentage*(lMaxMedium-
lMinMedium))/100;

```

```
        ucThr = (unsigned char)lVal;
    }
    else
        ucThr = 0;

    lLastThr = lActThr;
    while(lThrIndex < lIndexFocus)
    {
        if (ucptrThrList[lThrIndex] < 255)
            ucptrThrList[lThrIndex]=ucThr;
        lThrIndex++;
    }
}
```

## REFERENCES

- [1] ASTM International, ISO/ASTM 52900-15, Standard Terminology for Additive Manufacturing – General Principles – Terminology, West Conshohocken: , 2015.
- [2] T. T. Wohlers and T. Caffrey, Wohlers report 2015, Fort Collins, Colo.: Wohlers Associates, 2015, p. 314.
- [3] T. T. Wohlers, T. Caffrey and R. I. Campbell, Wohlers report 2016, Fort Collins, Colo.: Wohlers Associates, 2016, p. 335.
- [4] D. L. Bourell and et al., "A Brief History of Additive Manufacturing and the 2009 Roadmap for Additive Manufacturing: Looking Back and Looking Ahead," in *US-URKEY Workshop On Rapid Technologies*, Istanbul , 2009.
- [5] J. Breuninger, R. Becker, A. Wolf, S. Rommel and A. Verl, Generative Fertigung mit Kunststoffen, Berlin, Heidelberg: Springer, 2013, pp. 1 online resource (Online-Ressource (XII), 257.
- [6] R. Becker, A. Grzesiak and A. Henning, "Rethink assembly design," *Assembly Automation*, vol. 25, no. 4, pp. 262-266, 2005.
- [7] Freedom Of Creation , "Trabecula bench designed by Janne Kyttanen," <https://www.behance.net/gallery/21972999/Trabecula-bench-for-Freedom-of-Creation>, 2007.
- [8] VDI-Gesellschaft Produktion und Logistik, VDI 3405 :2014-12, Additive Fertigungsverfahren. Grundlagen, Begriffe, Verfahrensbeschreibungen = Additive manufacturing processes. basics, definitions, processes., Dezember 2014, Ausg. dt./engl ed., vol. 3405, Berlin: Beuth, 2014, p. 40.
- [9] 3D Systems, <http://www.3dsystems.com/>.
- [10] 3D Systems Corporation, "STEREOLITHOGRAPHY PRINTERS," [http://www.3dsystems.com/sites/www.3dsystems.com/files/sla\\_brochure\\_0216\\_a5\\_us\\_web\\_1.pdf](http://www.3dsystems.com/sites/www.3dsystems.com/files/sla_brochure_0216_a5_us_web_1.pdf), 2016.
- [11] A. Gebhardt, Generative Fertigungsverfahren, 3. Aufl. ed., München: Hanser, 2007, pp. XVI, 499 S..
- [12] Stratasys, <http://www.stratasys.com/de/3d-drucker/dental-series>.
- [13] EnvisionTEC, <http://enviontec.com/>, 2016.
- [14] Solidimension, <http://www.cubictechnologies.com/SD300.htm>.
- [15] EOS GmbH, <http://www.eos.info/en>.
- [16] Blueprinter, <http://blueprinter-powder-3dprinter.co.uk/>, 2016.
- [17] L. Li and J. T. Fourkas, "Multiphoton polymerization," *Materials Today*, vol. 10, no. 6, pp. 30-37, 2007.
- [18] M. Vaezi, H. Seitz and S. Yang, "A review on 3D micro-additive manufacturing technologies," *The International Journal of Advanced Manufacturing Technology*, vol. 67, no. 5-8, pp. 1721-1754, 2013.
- [19] I. Yadroitsev and P. Bertrand, "SELECTIVE LASER MELTING IN MICRO

## 156 References

---

- MANUFACTURING," *Annals of DAAAM & Proceedings of the 21st International DAAAM Symposium*, p. 319, January 2010.
- [20] R. C. Gonzalez and R. E. Woods, *Digital image processing*, 3rd ed ed., Upper Saddle River, N.J: Prentice Hall, 2008, pp. xxii, 954.
- [21] VDI/VDE, "Standard VDI/VDE 2632 Blatt 1, Machine vision - Basics, terms, and definitions," The Association of German Engineers (VDI) , 2010.
- [22] B. Silver, "An Introduction to Digital Image Processing," Cognex Corporation, 2000.
- [23] S. J. Ahn, I. Effenberger and S. Fulga, "Automated Segmentation and Object Recognition in 3D Data Sets," in *Transilvania University of Brasov / Manufacturing Engineering Department / Scientific Research Centre for Computerized Technological Developments: Computing and Solutions in Manufacturing Engineering COSME '04 / CD-ROM*, Brasov, Romania, 2004.
- [24] I. Effenberger, J. Kühnle and A. Verl, "Fast and flexible 3D object recognition solutions for machine vision applications," *SPIE*, 2013, p. 86610N.
- [25] N. Albrecht and N. Bauer, Eds., *Guideline industrial image processing*, vol. 1, Erlangen: Fraunhofer-Allianz Vision, 2003, p. 26.
- [26] G. Bebis, "Intensity Transformations (Chapter 3)," CS474/674, <https://www.cse.unr.edu/~bebis/CS474/>.
- [27] S. Güttler, S. Fulga, A. Grzesiak, O. Refle, R. Bischoff, F. Breitling and V. Stadler, "Electrophotography—An Efficient Technology for Biochip Fabrication," *Journal of Imaging Science and Technology*, vol. 55, no. 4, p. 040306, 2011.
- [28] J. Pannekamp, *Adaptive Verfahren zur Bewertung texturierter Oberflächen*, vol. 414, Heimheim: Jost-Jetter, 2005.
- [29] A. Wegner and G. Witt, "Process Monitoring in Laser Sintering using Thermal Imaging," *Proceedings of the 22nd Annual International Solid Freeform Fabrication Symposium (SSF 2011)*, pp. 405-414, 2011.
- [30] S. Fulga, A. Davidescu and I. Effenberger, "Identification of in-line defects and failures during Additive Manufacturing Powder Bed Fusion processes," in *Paper will be presented at the 4th international conference on computing and solution in manufacturing engineering - CoSME'16*, Brasov, to be published, 2016.
- [31] S. Fulga, A. Davidescu and I. Effenberger, "Conceptual approach for an in-line quality control system in Additive Manufacturing Powder Bed Fusion processes," in *Paper will be presented at the 4th international conference on computing and solution in manufacturing engineering - CoSME'16*, Brasov, to be published, 2016.
- [32] J. Kroll, S. Botta, J. Breuninger and A. Verl, "Adaptive quality assurance of the product development process of Additive Manufacturing with modern 3D data evaluation methods," in *IS&T/SPIE Electronic Imaging*, Burlingame, California, United States, 2013.
- [33] G. Kreiseler, J. Kroll, S. Fulga and I. Effenberger, "Quality assurance for the complete process chain of Additive Manufacturing with modern metrology techniques," *Proceedings of the 14th International Conference of the European Society for Precision Engineering and Nanotechnology*, vol. 1, pp. 121-124, 2014.
- [34] Energetics Inc. for National Institute of Standards and Technology, "Measurement science roadmap for metal-based additive manufacturing,"

- [http://www.nist.gov/el/isd/upload/NISTAdd\\_Mfg\\_Report\\_FINAL-2.pdf](http://www.nist.gov/el/isd/upload/NISTAdd_Mfg_Report_FINAL-2.pdf), 2013.
- [35] M. Mani, B. Lane, A. Donmez, S. Feng, S. Moylan and R. Fesperman, Measurement Science Needs for Real-time Control of Additive Manufacturing Powder Bed Fusion Processes, NIST Interagency/Internal Report (NISTIR) 8036, 2015.
- [36] T. Craeghs, F. Bechmann, S. Berumen and J.-P. Kruth, "Feedback control of Layerwise Laser Melting using optical sensors," *Physics Procedia*, vol. 5, pp. 505-514, 2010.
- [37] T. Craeghs, S. Clijsters and E. e. a. Yasa, "Online quality control of selective laser melting," *Solid Freeform Fabrication Proceedings*, pp. 212-226, 2011.
- [38] T. Craeghs, S. Clijsters, J.-P. Kruth, F. Bechmann and M.-C. Ebert, "Detection of Process Failures in Layerwise Laser Melting with Optical Process Monitoring," *Physics Procedia*, vol. 39, pp. 753-759, 2012.
- [39] K. A. Mumtaz and N. Hopkinson, "Selective Laser Melting of thin wall parts using pulse shaping," *Journal of Materials Processing Technology*, vol. 210, no. 2, pp. 279-287, 2010.
- [40] Y. Ning, Y. S. Wong, J. Fuh and H. T. Loh, "An approach to minimize build errors in direct metal laser sintering," *IEEE Transactions on Automation Science and Engineering*, vol. 3, no. 1, pp. 73-80, 2006.
- [41] A. Simchi, F. Petzoldt and H. Pohl, "On the development of direct metal laser sintering for rapid tooling," *Journal of Materials Processing Technology*, vol. 141, no. 3, pp. 319-328, 2003.
- [42] X. Wang, "Calibration of shrinkage and beam offset in SLS process," *Rapid Prototyping Journal*, vol. 5, no. 3, pp. 129-133, 1999.
- [43] M. Mahesh, Y. S. Wong, Fuh, J. Y. H. and H. T. Loh, "A Six-sigma approach for benchmarking of RP&M processes," *The International Journal of Advanced Manufacturing Technology*, vol. 31, no. 3-4, pp. 374-387, 2006.
- [44] S. Berumen, F. Bechmann, S. Lindner, J.-P. Kruth and T. Craeghs, "Quality control of laser- and powder bed-based Additive Manufacturing (AM) technologies," *Physics Procedia*, vol. 5, pp. 617-622, 2010.
- [45] Y. Chivel and I. Smurov, "On-line temperature monitoring in selective laser sintering/melting," *Physics Procedia*, vol. 5, pp. 515-521, 2010.
- [46] P. Lott, H. Schleifenbaum, W. Meiners, K. Wissenbach, C. Hinke and J. Bültmann, "Design of an Optical system for the In Situ Process Monitoring of Selective Laser Melting (SLM)," *Physics Procedia*, vol. 12, pp. 683-690, 2011.
- [47] I. Yadroitsev, P. Krakhmalev and I. Yadroitsava, "Selective laser melting of Ti6Al4V alloy for biomedical applications: Temperature monitoring and microstructural evolution," *Journal of Alloys and Compounds*, vol. 583, pp. 404-409, 2014.
- [48] R. B. Dinwiddie, R. R. Dehoff, P. D. Lloyd, L. E. Lowe and J. B. Ulrich, "Thermographic in-situ process monitoring of the electron-beam melting technology used in additive manufacturing," SPIE, 2013, p. 87050K.
- [49] S. Price, J. Lydon, K. Cooper and K. Chou, "Experimental temperature analysis of powder-based electron beam additive manufacturing," *Proceedings of the 24th Solid Freeform Fabrication Symposium*, p. 162-173, 2013.
- [50] S. Price, J. Lydon, K. Cooper and K. Chou, "Temperature Measurements in

- Powder-Bed Electron Beam Additive Manufacturing," *Proceedings of the ASME 2014 International Mechanical Engineering Congress & Exposition*, 2014.
- [51] F. M. D. E. C. T. D. M. C. H. E. Rodriguez, "Integration of a Thermal Imaging Feedback Control System in Electron Beam Melting," in *WM Keck Cent. 3D Innov. Univ. Tex. El Paso*, 2012.
- [52] S. P. Santospirito, K. Słyk, B. Luo, R. Łopatka, O. Gilmour and J. Rudlin, "Detection of defects in laser powder deposition (LPD) components by pulsed laser transient thermography," *SPIE*, 2013, p. 87050X.
- [53] H. Krauss, C. Eschey and M. F. Zäh, "Thermography for Monitoring the Selective Laser Melting Process," *Proceedings of the 23rd Annual International Solid Freeform Fabrication Symposium*, p. 999–1014, 2012.
- [54] M. Pavlov, M. Doubenskaia and I. Smurov, "Pyrometric analysis of thermal processes in SLM technology," *Physics Procedia*, vol. 5, pp. 523-531, 2010.
- [55] M. Shiomi, K. Osakada, K. Nakamura, T. Yamashita and F. Abe, "Residual Stress within Metallic Model Made by Selective Laser Melting Process," *CIRP Annals - Manufacturing Technology*, vol. 53, no. 1, pp. 195-198, 2004.
- [56] L. van Belle, G. Vansteenkiste and J. C. Boyer, "Investigation of Residual Stresses Induced during the Selective Laser Melting Process," *Key Engineering Materials*, Vols. 554-557, pp. 1828-1834, 2013.
- [57] M. L. Vlasea, B. M. Lane, L. F. Lopez, S. Mekhontsev and M. A. Donmez, "Development of powder bed fusion additive manufacturing test bed for enhanced real time process control," in *Solid Freeform Fabrication Symposium*, Austin, TX, 2015.
- [58] C. A. L. and M. S. P., "Process Intermittent Measurement for Powder-Bed Based Additive Manufacturing," in *22nd Int. SFF Symp. - Addit. Manuf. Conf.,*, 2011.
- [59] D. B. Pedersen, L. De Chiffre and H. N. Hansen, "Additive Manufacturing: Multi Material Processing and Part Quality Control," 2013.
- [60] S. Kleszczynski, J. zur Jacobsmühlen, J. Sehart and G. Witt, "Error detection in laser beam melting systems by high resolution imaging," in *Proc. 23rd Solid Freeform Fabrication Symposium*, 2012.
- [61] G. Bi, B. Schürmann, A. Gasser, K. Wissenbach and R. Poprawe, "Development and qualification of a novel laser-cladding head with integrated sensors," *International Journal of Machine Tools and Manufacture*, vol. 47, no. 3-4, pp. 555-561, 2007.
- [62] M. Schmid and G. Levy, "Quality Management and Estimation of Quality Costs for Additive Manufacturing with SLS," in *Proc. of Direct Digital Manufacturing Conference 2012*, Berlin, 2012.
- [63] I. Effenberger, T. Felix, A. Frommknecht and S. Fulga, "Machine vision platform for non-destructive testing methods of fibre reinforced plastics," in *Proc. 19th World Conference on Non-Destructive Testing 2016*, Munich/Germany, 2016.
- [64] N. Josuttis, *Objektorientiertes Programmieren in C++*, 1. Aufl ed., Bonn; Paris; Reading, Mass. [u.a.]: Addison-Wesley, 1994, pp. XIII, 553 S. ;.
- [65] S. Moylan, J. Slotwinski, A. Cooke, K. Jurrens and M. A. Donmez, "An Additive Manufacturing Test Artifact," *J. RES. NATL. INST. STAN. (Journal of Research of the National Institute of Standards and Technology)*, vol. 119, p. 429, 2014.

- [66] T. B. Moeslund, Introduction to video and image processing, London; New York: Springer, 2012, pp. 1 online resource (xi, 227).
- [67] L. G. Shapiro and G. C. Stockman, Computer vision, Upper Saddle River, NJ: Prentice Hall, 2001, pp. xx, 580.
- [68] D. Chen and G. Zhang, "A new sub-pixel detector for x-corners in camera calibration targets.," *WSCG Short Papers*, pp. 97-100, 2005.
- [69] D. C. Brown, "Close-range camera calibration.," *Photogrammetric Engineering*, p. 855-866, 1971.
- [70] C. Demant, C. Garnica and B. Streicher-Abel, Industrial image processing, Second revised edition ed., pp. 1 online resource (xvii, 369).
- [71] Haar Romeny, Bart M. ter, Front-end vision and multi-scale image analysis, vol. v. 27, Dordrecht; Boston: Kluwer Academic, 2003, pp. xviii, 464.
- [72] J. Bernsen, "Dynamic thresholding of gray-level images," *Proc. Intl. Conf. Patt. Recog*, p. 1251-1255, 1986.
- [73] B. Jähne, Digitale Bildverarbeitung, 6., überarb. und erw. Aufl. ed., Berlin; Heidelberg; New York: Springer, 2005, pp. XIII, 642 S..
- [74] D. J. Jackson, "Computer Vision & Digital Image Processing: Histogram Processing I," <http://jjackson.eng.ua.edu/courses/ece482/lectures/LECT06-6.pdf>.
- [75] G. Taubin, "Estimation of lanar curves, surfaces and nonplanar space curves defined by implicit equations, with applications to edge and range image segmentation," *IEEE Trans.*, pp. 1115-1138, 1991.
- [76] Werth Messtechnik GmbH, "Werth TomoScope® HV 500," <http://www.werthinc.com/us/our-range/products/multisensor-cmms/for-computer-tomography/werth-tomoscope-hv-500.html>.
- [77] cirp GmbH, 2016. [Online]. Available: [https://www.cirp.de/verfahren/lasersintern\\_EN.php5](https://www.cirp.de/verfahren/lasersintern_EN.php5).
- [78] S. Fulga, A. Davidescu, I. Effenberger and A. Verl, "In-line quality control system for a reliable Additive Manufacturing -layer by layer inspection-," in *Paper will be presented at the International Conference on Additive Technologies - iCAT 2016*, Nürnberg, to be published, 2016.
- [79] S. Fulga, A. Davidescu, I. Effenberger and A. Verl, "Tasks for in-line quality control and in-situ optimisation of Additive Manufacturing Powder Bed Fusion Processes (Professional)," in *Paper will be presented at the International Conference on Additive Technologies - iCAT 2016*, Nürnberg, to be published, 2016.
- [80] S. Fulga, "Qualität 4.0 für die additive Fertigung der Zukunft : Intelligente Qualitätssicherung und Optimierung," in *Vortragsforum:Qualität 4.0 - Neue Herausforderungen an die Mess- und Prüftechnik*, Stuttgart, 2016.
- [81] S. Fulga, "Qualität 4.0 für die additive Fertigung von Kunststoffbauteilen : Intelligente Qualitätssicherung und Optimierung," in *20. Anwenderforum Rapid Product Development*, Stuttgart, 2015.
- [82] D. Silverman, "Doing Qualitative Research," Sage Publications, London, 2009.