Objective surface inspection and semi-automated material removal for metal castings

by

Daniel Wilhelm Schimpf

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Industrial Engineering

Program of Study Committee:
Frank Peters, Major Professor
Michael Dorneich
Matthew Frank
Yan-Bin Jia
Stephen Vardeman

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this dissertation. The Graduate College will ensure this dissertation is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University

Ames, Iowa

2021

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my advisor Prof. Frank Peters for the guidance, motivation, and support throughout my research. I would also like to thank my committee members, Prof. Michael Dorneich, Prof. Matthew Frank, Prof. Yan-Bin Jia, and Prof. Stephen Vardeman, for their guidance and contributions to this study.

Further, I am grateful to my friends, colleagues, the Department of Industrial and Manufacturing Systems engineering faculty, and staff for their time, help, and suggestions. Finally, I would like to thank my family for their support and positive attitudes throughout my life and work. I would especially like to thank my wife, Jesse Schimpf, for her support and patience along this journey.

Daniel

ABSTRACT

This dissertation will present two methods of how 3D point cloud data can be used to significantly advance two important operations within the metalcasting process. The first is the inspection of the casting surface, and the other is an automation method to replace the current reliance on manual grinding.

Currently, surface roughness inspection is performed manually by an operator who compares the surface of a casting with comparators and determines if they are acceptable based on the casting design specifications. The comparators are pictures or physical replicas of representative casting surfaces. As the inspection process is manual, it is very subjective. The low repeatability and reproducibility of the inspection process cause communication problems between foundries and customers as well as within the foundry. This could cause unacceptable castings to be sent to customers, customers falsely identifying acceptable castings as unacceptable, and excessive rework iterations.

The dissertation will present an objective method to inspect castings repeatably and reliably. The method will use 3D scan data in the form of point clouds. The point clouds will be used to determine the underlying geometry of castings and then use the distance between point clouds and the mesh representing the underlying geometry to calculate the surface roughness.

The second operation covered in this dissertation is grinding of casting surfaces in foundries. Much of the steel casting industry is made up of companies that produce a high mix in low production quantities. This environment precludes currently available automation solutions. Hence there is a heavy reliance on manual grinding. Manual grinding is one of the operations in foundries that has the most ergonomic issues as well as most safety incidents. Currently, the automation of grinding operations is mainly done through fixed automation, where the robot

performs the same operation for every part. This requires expensive fixturing for repeatable orientation and programming for every new part. Large production quantities are required to justify the fixturing and process planning tasks.

This dissertation will present a semiautomatic grinding solution. In this method, the operator identifies the excess material that needs to be ground and marks it with colored markers on the casting. The casting is then placed in a robotic cell that only requires the casting to be secured and but not the need for exact fixturing. A 3D scanner with a color camera is used to scan the casting, identify the markings, and segment the surface based on the markings. A removal strategy is automatically determined and executed.

Overall this dissertation will present two methods to utilize 3D information to improve foundry operations.

CHAPTER 1. GENERAL INTRODUCTION

1.1. Introduction

After a casting is poured, solidified, and shaken out of its mold, excess material on the casting needs to be removed. This includes contacts where the risering and gating system was attached. The removed material could be both expected, like the gating system's contacts, and unexpected, such as abnormalities on the surface. Most of the riser and gate material is usually cut or knocked off. However, about 6 mm (1/4 in) of the contact remains on the casting surface, but this value is not consistent among castings. The remaining excess material is often removed through grinding. Depending on factors, such as casting geometry and production volume, the excess material may be removed through manual grinding by an operator, which is most common, or in an automated cell by a robot. After being ground, the casting may go through further processing steps such as machining to fulfill drawing requirements before it goes through inspection.

Castings are typically inspected to ensure that the casting's surface meets the customer's requirements before a casting is delivered to the customers. This casting surface inspection is commonly performed via visual inspection. It determines if the overall surface roughness and the abnormalities are acceptable. Abnormalities include issues such as fusion, porosity, removal marks, and inclusions. The surface specifications are usually set based on comparators, which are either pictures, metal, or plastic replicas of castings with varying surfaces. Commonly these comparators are ranked from smoothest/best surface to roughest/worst. Some comparators only assess surface roughness, while others also asses other typical casting surface characteristics or abnormalities such as inclusions or gas porosity. The operator then determines if the surface is better or equal to the specified comparator for a given casting surface.

The subjectiveness of the inspection creates problems for casting producers and customers [1]. Customers will typically call out the required surface characteristics in the engineering drawings. Once the casting is produced in the foundry, the inspector will determine if it fulfills this requirement through visual inspection. After the casting is delivered to the customer, they may perform their own inspection. Because of the subjective inspection, these two inspections may not align, causing the product to be declined and making rework necessary. Thus the first research question guiding this research is: How can the surface of metal castings be characterized quantitatively, reliably, and with adequate scale using digital surface data?

This dissertation will introduce a digital method for objective surface roughness determination. This method will utilize a 3D scan of a flat area on the casting to calculate an objective roughness value for the scanned surface.

Since many foundries have their own 3D scanners and are using them for dimensional measurements, it would be beneficial to use these 3D scans for the roughness determination instead of a separate inspection step. Thus the second research question guiding this research is: How can the surface of metal castings be characterized quantitatively, reliably, and with adequate scale using digital surface data from the entire 3-dimensional shape?

This dissertation will introduce a method to objectively determine casting's surface roughness utilizing 3D scans of casting already performed at foundries.

The knowledge gained by working with 3D scan data to create a method for determining roughness values and identifying abnormalities from point clouds can also be utilized to improve the manual removal of excess material on castings, by using 3D scan data to determine the location and extend of excess material on castings. This information can then be used to plan a path for a robot to remove the material.

Currently, the excess material is commonly removed through manual grinding. Manual grinding is hard work with ergonomic challenges [2] and one of the operations with the most safety incidents in foundries [3]. Further, in times of low unemployment, foundries experienced high labor turnover rates. All these are reasons to automate the grinding process. For fixed automation, the robot's path plan is determined once and can then be used for all castings of the same kind. To accommodate this automation strategy, an accurate location of the casting in the robotic cell is required, which is often achieved with fixturing. The cost of the programming and the fixture is then amortized over the number of castings produced with that program and fixture. If the number of castings is high, it will not increase each castings cost, but for job shops, this will be too expensive. Also, the program's creation may take longer than manual removal, especially if the foundry does not perform its own robotics programming.

Further, since not all areas where material needs to be removed can be known when predetermining the robot's path plan in fixed automation, manual operations still may be necessary to remove material in these previously unknown areas. Besides, the amount of material that needs to be removed in one location may vary, for example, because of manual gating system cut off. This fixed removal process could compensate for this by planning to remove more material than necessary. This results in air cutting, which causes longer cycle times and increases the automation solution's cost. Overall automated removal process on castings is a challenge, especially in job shops with low production volumes.

The final guiding research question is: How can excess material be removed to blend a surface to the surrounding surface automatically for castings from high variety, low volume production using minimal operator input?

This dissertation will introduce an adaptive automation method that relies on an operator marking the areas where material needs to be removed on the castings. It will analyze the point cloud data and color information gathered by a 3D sensor to detect the markings, determine the size of the abnormality and shape of the surface around it, as well as determine a path plan for removing the material.

1.2. Dissertation Organization

Chapter 1 introduces the casting surface inspection and material removal problem. Chapter 2 provides a literature review of both issues and clarifies the motivation. Chapter 3 introduces a proposed solution to the objective casting surface inspection problem. Chapter 4 improves upon the proposed solution of the previous chapter to make the objective method more accessible and useable to foundries. Chapter 5 presents a novel way to utilize point cloud data to segment a surface based on marking, determine the desired surface, similarly to the underlying geometry detection in Chapters 3 and 4, and remove surface abnormalities. Chapter 6 is providing a conclusion and evaluates the potential for future work.

References

[1] M. M. Stallard (Voelker), C. A. MacKenzie, and F. E. Peters, "A probabilistic model to estimate visual inspection error for metalcastings given different training and judgment types, environmental and human factors, and percent of defects," *J. Manuf. Syst.*, vol. 48, no. July, pp. 97–106, 2018, doi: 10.1016/j.jmsy.2018.07.002.

[2] F. Peters and P. Patterson, "Ergonomic Improvements for Foundries," Iowa State University, Ames, IA, Jun. 2002. doi: 10.2172/796904.

[3] R. Moore, "Steel Foundry Safety Benchmark Study," in *2019 Technical & Operating Conference*, Steel Founders' Society of America, Chicago, IL, 2019.

CHAPTER 2. LITERATURE REVIEW

The first part of this literature review will introduce current visual inspection standards for castings, studies evaluating visual inspection, and objective alternatives to visual inspection because this dissertation will introduce a new method to evaluate the surface roughness of castings. The second part of the literature review will discuss the current methods for material removal on castings for high and low production volumes, because of the material removal method for low production volumes introduced in this dissertation.

2.1. Inspection

The surface characteristics of castings are some of the quality indicators which are identified by inspectors in foundries, and often done via visual inspection. For castings, comparator plates or pictures are utilized to determine the surface roughness of castings as well as the level and kind of abnormalities present on the surface. Some examples of those comparators are the GAR-C9 comparators [1], the MSS SP-55 Visual Method [2], the ACI Surface Indicator Scale [3], and the SCRATA comparator plates. The latter one being referenced by one of the most common surface standards in the USA, ASTM A802 [4].

The GAR-C9 comparator (Figure 2-1) has nine surface roughness levels from 20 to 900 RMS Microinches. How these RMS values were determined is unclear. Out of the four comparators, this comparator has the highest resolution with nine different roughness levels. However, it does not contain any definitions of other surface characteristics such as porosity or cutting marks.

Figure 2-1: GAR C-9

The ACI Surface Indicator Scale has four different levels of surface roughness ranging from SIS-1 to SIS-4 (Figure 2-2). The SIS levels have RMS Microinch equivalents, as seen in Table 2-1.

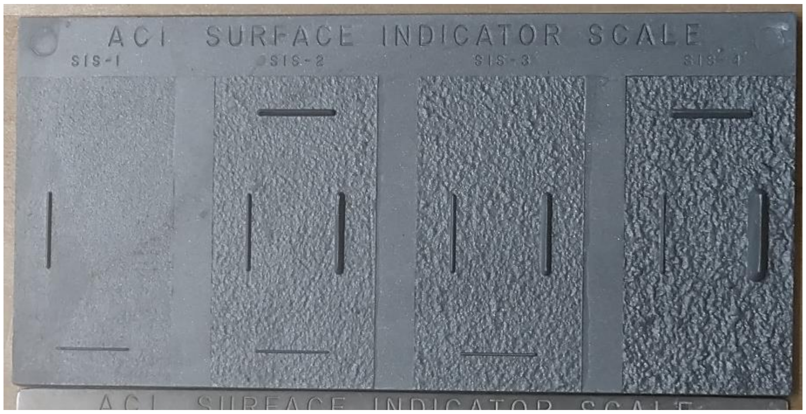

Figure 2-2: ACI Surface Indicator Scale

Table 2-1 Corresponding SIS numbers and RMS average deviations [3]

RMS Microinches	SIS Number
200	1
350	2
500	3
900	4

Unlike the GAR-C9 comparator, the ACI Surface Indicator Scale considers not only surface roughness but also other irregularities. The ACI comparators do not differentiate between the type of irregularities. Instead, the height, area, and number of occurrences are considered. In addition to the four different surface roughness levels, the comparator has 15 different grades or irregularities. These irregularities are differentiated by their permissible height and depth. Table 2-2 shows the 15 different grades with their permissible heights and depths. Each SIS level has four or fewer grades associated with them. Grade 5, for example, would mean surface roughness of SIS-2 and irregularities smaller or equal to 1/32 inch. Starting with SIS number 3, irregularities are sometimes allowed to be deeper than high.

Table 2-2 ACI Surface grade Limits [3]

		Irregularities Permissible – Inch	
Grade	SIS No.	Depth	Height
I	1	None	None
II	1	1/64	1/64
III	1	1/32	1/32
IV	2	1/64	1/64
V	2	1/32	1/32
VI	2	3/64	3/64
VII	2	1/16	1/16
VIII	3	1/32	1/64
IX	3	1/16	1/32
X	3	1/16	1/16
XI	3	1/8	1/16
XII	4	1/16	1/32
XIII	4	1/8	1/16
XIV	4	1/8	1/8
XV	4	¼	1/8

The standard further specifies the number of irregularities allowed for different surface grades and irregularities sizes. Irregularities are considered individual irregularities when they are separated by more than four times their minimum dimension.

The ASTM A802 references the SCRATA plates (Figure 2-3), and unlike ACI, which does not differentiate between different types of abnormalities, ASTM A802 differentiates between nine different surface characteristics, one of which is surface texture, as seen in Table 2-3. Four different levels of surface texture are referenced: SCRATA A1 - A4; an A5 plate exists as well but is not referenced by ASTM, likely because the roughness is beyond what is acceptable. The SCRATA A4 surface itself is a lot rougher than any surface on GAR C9 and ACI. Further, the difference between the different surface levels seems larger between the SCRATA A levels than for GAR C9 and ACI, making it less challenging for an objective method to differentiate between the roughness levels. The actual size of the SCRATA plates (3.8 x 5.8 in) is also a lot larger than the ACI (1.1 x 2.1 in) or GAR-C9 (0.4 x 1.5 in) surfaces.

Table 2-3 Visual Inspection Acceptance Criteria [4]

Surface Feature	Level 1	Level 2	Level 3	Level 4
Surface texture	A1	A2	A3	A4
Nonmetallic inclusions	B1	B2	B4	B5
Gas porosity	C2	C1	C3	C4
Fusion discontinuities		D1	D2	D5
Expansion discontinuities			E3	E5
Inserts			F1	F3
Metal removal marks				
Thermal	G1	G2	G3	G5
Mechanical	H1	H3	H4	H5
Welds	J1	J2	J3	J5

The MSS SP-55 Visual Method does not have physical comparators like GAR-C9, ACI, or SCRATA but works instead with images of acceptable and unacceptable surfaces. Further, it does consider not only surface roughness but also abnormalities. Overall it differentiates for 12 different types of surface characteristics between acceptable and unacceptable surfaces. Thus unlike the other three comparators, it does not differentiate between multiple different levels for each characteristic. Instead, it is a binary decision. The 12 different characteristics are hot tear and cracks, shrinkage, sand inclusion, gas porosity, veining, rat tails, cutting marks, scabs, chaplets, weld repair areas, surface roughness, and wrinkles, laps, folds, and cold shuts. This visual method also has SCRATA equivalents for some of the criteria. This means that for some MSS SP-55 criteria, one can use a physical SCRATA comparator to determine if a surface is acceptable. For example, for surface roughness acceptance criteria of MSS SP-55, the SCRATA A3 or better are considered equivalent and acceptable.

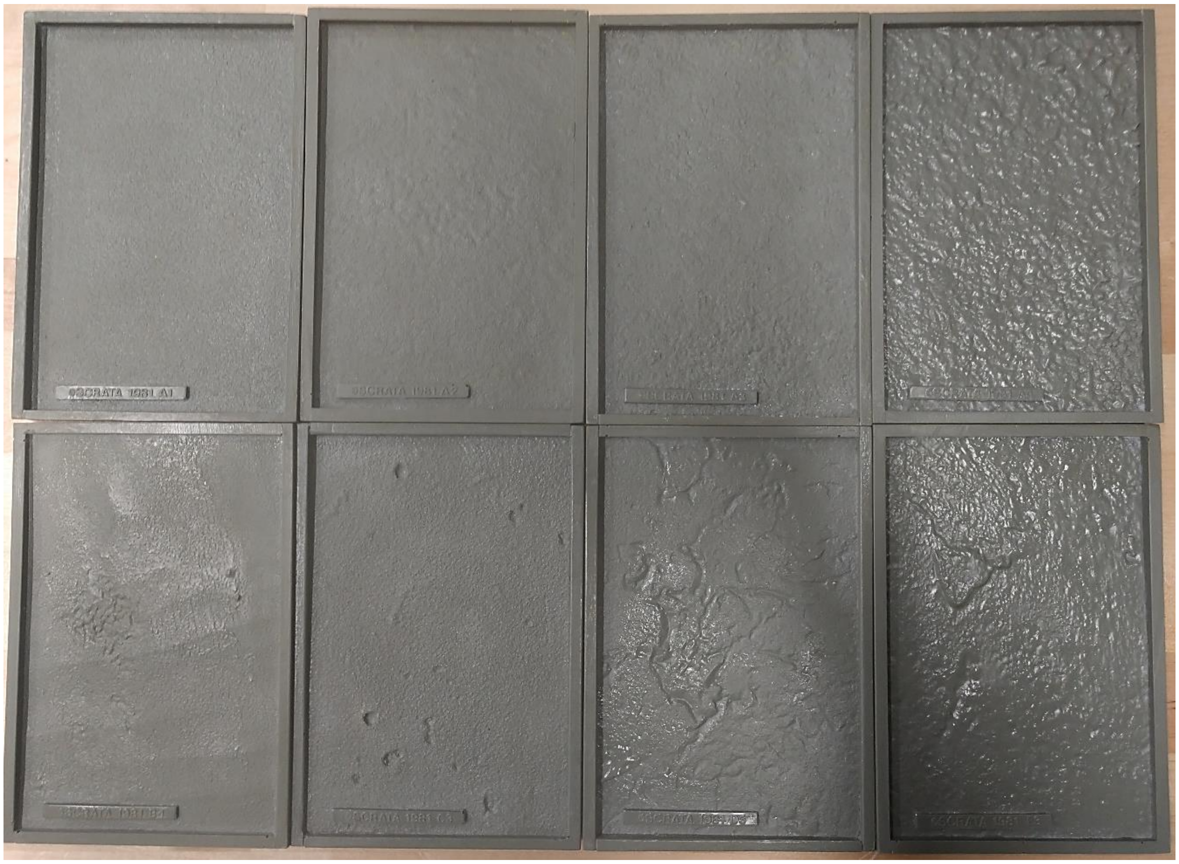

Figure 2-3: Various SCRATA comparators. First row left to right: A1, A2, A3, A4. Second row left to right: B4, C3, D5, E3

Visual inspection is rather subjective, which can lead to errors. Thus multiple studies have investigated the effectiveness of visual inspection. One study [5] explained the risks involved in visual inspection. The author describes two types of errors that can occur during the visual inspection, bad parts can be missed, and good parts can be rejected. Both lead to costs. When a bad part is missed during the inspection, the error is considered customer risk, and when a good part is rejected, the error is considered producer risk. In both cases, the producer will pay for the mistake. In the former, it will be indirect, but in the latter, the producer will pay directly through unnecessary rework or scrap. Overall the results for the final inspection of aluminum castings at the author's facility found an average of 82% effectiveness, ranging from 69% to 92% effective. The author states that with a visual standard, about 80% effectiveness can be expected, further suggests that with improved training and procedures, 96% may be reached. However, the author concludes that without removing the human from the equation, it will not be possible to get any better.

Another study [6] focused more on the visual inspection of anomalous areas on castings. The study worked on a method to enable the measure of repeatability and reproducibility error for visual inspection of anomalous areas on castings. Repeatability error is the inspection variation for the same part and operator over multiple trials. Reproducibility error, on the other hand, is the inspection variation for the same part and multiple operators. The method identifies anomalous areas with circular markers. Through multiple trails and operators, master clusters are generated, showing all areas in which markers were placed. These master clusters are used for repeatability and reproducibility calculations to determine how well the same defect regions are identified, and the results are reported in terms of percent match for these master clusters. Part of this study was a gage R&R study at three different foundries. The average repeatability for the three foundries was only a 64% master cluster match, whereas the reproducibility, on average, showed that two operators only agreed 45% of the time.

Both of the previous two studies show that there is a significant amount of error possible during the visual inspection of casting surfaces. One study [7] worked on improving visual inspection because of the high potential for error. The study presented a test (Matching Familiar Figures Test), which could help foundries determine how much potential an applicant has to become an inspector. Further, they showed that rastering training would improve the inspection results. Rastering is a strategic way to inspect the whole part surface and thus make it less likely to miss abnormal areas.

The subjectiveness of the visual inspection causes issues like low repeatability and reproducibility. Objective methods for surface characterization exist but are not suited for the classification of casting surfaces. Contact methods like stylus profilometer are very common [8] to determine the surface roughness of machined surfaces. The stylus is pulled over a short section of the part while the deflection is recorded. This generates a two-dimensional view of a small part section. Based on these results, surface roughness values like R_a, the arithmetic mean, can be determined. This method has a couple of disadvantages [9]; it is time-consuming to use on big surface areas, limited in the measuring amplitude, and results depend on the measuring direction. On the other hand, it has high accuracy and thus can be used to differentiate even very smooth surfaces. Further, the method is beneficial for machined surfaces because of the regular surface structure. One is able to select a very small area on the machined surface, which would be representative of the whole surface. For castings, this is not a valid assumption because casting surfaces have a random surface structure.

Non-contact methods, in general, are able to acquire a much higher number of data points in a bigger area, but their accuracy is usually lower than for contact methods. A 3D equivalent for the popular 2D parameter R_a is the S_a parameter for the average roughness. It is the 3D equivalent because the calculation is very similar, but the results of S_a and R_a measured on the same surface may still differ considerably. One study [10] found that differences as high as 52% can be found for the same surface if only one R_a profile is compared to the S_a measurement. This error can be reduced by averaging multiple R_a profiles and avoiding profile directions parallel to the main feature direction.

The S_a , arithmetic mean height, surface roughness parameter is commonly available for roughness determinations on microscopes with focus variation where typically only an area of 3 by 3 mm is inspected [11]. This is an alternative to a profilometer for machined surfaces, but not very useable for casting surfaces because of the small area covered.

Objective methods for the casting surface classification have been proposed, which look at larger areas [12]. The method developed in the study considers both abnormalities and the underlying geometry of the surface. It uses point clouds commonly gathered by a non-contact measuring device. Point clouds are a list of points in the three-dimensional space and are also used to calculate parameters such as the earlier mentioned S_a. The method specifies a surface based on three parameters: the baseline roughness, the maximum abnormality level, and the maximum abnormal percentage of the inspected surface. For this specification, the method can evaluate if a surface meets or does not meet the specification.

This dissertation will address the shortcomings of the current methods and provide a digital method to determine surface roughness objectively while considering abnormalities, the underlying geometry, and the spatial relationships of the points in the point cloud.

2.2. Material Removal

Material removal processes on castings are common. After the casting is removed from its mold and undergone an initial blast cleaning, most excess material is removed. This metal includes the risering and gating system inherent to the process as well as potentially excess material at the parting line. This is done via sawing, torch cutting, arc air, water jet cutting, or breaking. To not damage the surface of the casting, the removal process usually leaves up to 6 mm of material above the desired surface. The rest of the material is commonly removed by grinding. For very accessible areas and when much material needs to be removed, grinders with a large grinding wheel diameter are used and mounted in a way that supports most of their weight but lets the operator move the grinder around (swing frame grinder). For less accessible areas, when it is impractical to move the casting into position or when less material needs to be removed, handheld grinders are commonly used by the operators.

Robots can be used to perform this grinding task. Depending on the size of the casting that needs to be ground, either the robot may hold the casting and move it to the grinder, or the robot holds the grinder and moves it to the casting to grind [13]. In general, for low production volumes, robotic grinding is difficult to achieve because programming is necessary for every part number change unless parametric programming was utilized, which is restricted to parts of the same product family. The U.S. steel casting industry produces over 1 million tons of castings each year [14]. However, the majority of this production is for short lot sizes that cannot support the use of currently available automation systems. New products have been developed to reduce the time it takes to program robot paths. Robot producers started to add touchpads to their robots for easier programming, and third parties like Sisu offer smart pendants that enable programming with a panel and handheld controller [15]. Especially the latter is focused on smaller companies by reducing the cost of training and external programmers for robotic automation. However, for complex path plans, these are only of limited use. Another way to avoid expensive programming is by having a human control the robot during the grinding operation. This can either be done directly by guiding the robot by hand [16] or controlling the robot with a controller from outside of the cell [17]. For both applications, the operator can be relieved from most of the grinding forces.

This is important because manual grinding is very strenuous work. Constant pressure has to be exerted on the grinder to remove material while moving around the casting, often in non-ergonomic positions. The grinding can have negative impacts on the operators and productivity.

An industry survey done for a report [18] indicated that hand grinding is the job in steel foundries with the most ergonomic issues. It was further determined that the ergonomic problems are difficult to solve. The report identified multiple possible solutions, such as avoiding unnecessary grinding tasks and the usage of swing grinders to assist the operator with the force application. Two other studies modeled hand grinding tasks to evaluate the ergonomics, and both identified the working height as an important factor in improving ergonomics [19][20]. Furthermore, surface finishing operations in foundries, which grinding is part of, is the second-highest cause of workplace injuries in foundries right after melting [21].

These health-related issues are one of the reasons for increased interest in automating grinding tasks in foundries. Other reasons include price pressures through global competition, low retention rates, and high labor turnover. The latter two are negatively affected by the arduous work and low unemployment rates in recent years. Overall this has been restricting the optimal operation of foundries [22].

Automation would be able to help with these issues, but automation for castings is harder than for machined parts. The higher dimensional variability makes it difficult to fixture workpieces and use the same machine code because of the differing amount of material that needs to be removed and a geometry that does not closely match the CAD file. For low variety and high volume production, these problems are manageable but may cause some inefficiencies, like air cutting because of the unknown amount of material left. Furthermore, for high production volumes, the cost of the fixture to orient the part repeatedly and of the robot programming can be spread out over the number of parts. Thus, robotics can be found in low variety, high volume productions.

Visits of 3 foundries members of the Steel Founders' Society of America that employ robotic automation in their foundries have nonetheless shown that there is some isolated use by some smaller foundries of robotic grinders. One such foundry utilizes parametric programming to save programming costs. They have parts that have the same geometry but different scales. That way, the general program can stay the same; only some sizes need to be changed. Further, the surfaces where the material is removed are restricted to flat, conical, and cylindrical surfaces, which simplifies the path plans. Because the amount of material remaining is unknown, a lot of the processing time is air cutting. The robot was not required to be as economical as a human because the project was still in its development stage and because the labor shortage was significant. Another larger foundry used robotic grinding on some high-volume parts. On these parts, a flat area was ground to the desired dimension. Since the robot picked up the part and moved it to a stationary grinder, the solution required compatible tooling to pick up each part. To switch to a new part number changes in the automation code and new tooling may be required.

The lower precision of the casting process causes issues for robotic grinding because of many unknowns but is also advantageous because, in many areas of the casting, only relatively low precision is required making automation easier. Much robotic surface finishing research focuses on robotic blending or polishing [23]–[27]. For these operations, the focus lies on removing an even amount of material instead of targeted material removal in specific areas. For these blending and polishing operations, a type of force control is often used to maintain contact to an unknown surface and remove an even amount of material. Some previous work [28] focused on automatically remove excess material on castings included a prototype system using a gantry with a small rotary tool. An operator would identify the work area by positioning a contact probe in multiple locations. The system would then decide on locations on which it would sample the surface. These sample points were then used to generate a path plan for the rotary tool. Overall, the method was restricted by its size as well as the possible orientations of the rotary tool.

In this dissertation, a method for grinding automation in a low production environment is presented and will address these challenges.

References

[1] G. Electroforming, “C-9 Cast Microfinish Comparator.” https://garelectroforming.com/index.php?route=product/product&path=59_64&product_id=73 (accessed Mar. 18, 2019).

[2] ANSI/MSS SP 55-2011, “Quality Standard for Steel Castings for Valves, Flanges and Fittings and Other Piping Components (Visual Method),” *Manuf. Stand. Soc. Valve Fitt. Ind.*, 2011.

[3] P. Eubanks, A. Crawford, L. Johnson, G. Larkin, W. Patschke, and E. Schoefer, “Standard for the Visual Inspection of Casting Surfaces,” *Manual. Des Plaines Steel Founders’ Soc. Am. Alloy Cast. Inst.*, 1969.

[4] A. /A802 − 95 1982 (2015), “Standard Practice for Steel Castings, Surface Acceptance Standards, Visual Examination,” *ASTM Int. West Conshohocken, PA*, pp. 3–5, 2015, doi: A0802-95R15.

[5] T. J. Schorn, “Visual Inspection Errors : Measurement and Impact,” *AFS Trans.*, vol. 114, no. 01, pp. 17–23, 2006.

[6] G. Daricilar and F. Peters, “Methodology for Assessing Measurement Error for Casting Surface Inspection,” *Int. J. Met.*, vol. 5, no. 3, pp. 7–15, Jul. 2011, doi: 10.1007/BF03355514.

[7] F. Peters, R. Stone, K. Watts, P. Zhong, and A. Clemons, “Visual Inspection of Casting Surfaces,” *AFS Trans.*, vol. 121, p. 8, Jan. 2013.

[8] D.-H. Lee and N.-G. Cho, “Assessment of surface profile data acquired by a stylus profilometer,” *Meas. Sci. Technol.*, vol. 23, no. 10, p. 105601, Oct. 2012, doi: 10.1088/0957-0233/23/10/105601.

[9] X. D. Yang, J. C. Li, and T. B. Xie, “A Large Measuring Range Profilometer for Three-Dimensional Surface Topography Measurement,” *Key Eng. Mater.*, vol. 364–366, pp. 750–755, Dec. 2007, doi: 10.4028/www.scientific.net/KEM.364-366.750.

[10] H. T. Lancashire, “A simulated comparison between profile and areal surface parameters: Ra as an estimate of Sa,” pp. 1–9, Aug. 2017, Accessed: Oct. 23, 2019. [Online]. Available: http://arxiv.org/abs/1708.02284.

[11] U. C. Nwaogu, N. S. Tiedje, and H. N. Hansen, “A non-contact 3D method to characterize the surface roughness of castings,” *J. Mater. Process. Technol.*, vol. 213, no. 1, pp. 59–68, 2013, doi: 10.1016/j.jmatprotec.2012.08.008.

[12] M. M. Voelker and F. E. Peters, “Development of a digital standard to specify surface requirements of cast metal surfaces,” *Mater. Perform. Charact.*, vol. 6, no. 2, pp. 130–143, 2017, doi: 10.1520/MPC20160014.

[13] S. Meyer, “The next generation of grinding - The Robot Report,” Jul. 25, 2016. https://www.therobotreport.com/the-next-generation-of-grinding/ (accessed Sep. 16, 2020).

[14] Steel Founders’ Society of America, “Introduction to steel castings - Steel Casting Wiki.” https://wiki.sfsa.org/index.php/Introduction_to_steel_castings (accessed Oct. 09, 2020).

[15] E. Cole, “Robotics Industry Insights - Programming Made Easy: How Millennial-Friendly Technologies Are Transforming Robot Programming,” 2020. https://www.robotics.org/content-detail.cfm/Industrial-Robotics-Industry-Insights/Programming-Made-Easy-How-Millennial-Friendly-Technologies-Are-Transforming-Robot-Programming/content_id/8952 (accessed Sep. 16, 2020).

[16] S. Roschi, “Cobot 7A-15 Robotic Arm Lends a Strong Helping Hand,” Jun. 14, 2017. https://www.roboticsbusinessreview.com/supply-chain/robotic-arm-lends-strong-hand/ (accessed Sep. 16, 2020).

[17] Vulcan Engineering Co., “VTS™ – Floor Mounted Model,” 2020. https://www.vulcangroup.com/product/vts-floor-mounted/ (accessed Sep. 16, 2020).

[18] F. Peters and P. Patterson, “Ergonomic Improvements for Foundries,” Iowa State University, Ames, IA, Jun. 2002. doi: 10.2172/796904.

[19] M. Sharma, K. K. Kataria, S. Kant, and N. M. Suri, “Ergonomic Assessment of a Fettling Operation in Foundry Based on Digital Human Modeling and Statistical Analysis,” in *Optimization Methods in Engineering*, M. Tyagi, A. Sachdeva, and V. Sharma, Eds. Singapore: Springer Singapore, 2021, pp. 481–502, doi: 10.1007/978-981-15-4550–4_30.

[20] Y. Sun, J. Chen, Z. Lu, B. Yang, and W. Liu, “Ergonomics Analysis of Hand-Held Grinding Operation Working Posture Based on Jack,” in *Man--Machine--Environment System Engineering*, S. Long and B. S. Dhillon, Eds. Singapore: Springer Singapore, 2020, pp. 733–740.

[21] R. Moore, “Steel Foundry Safety Benchmark Study,” in *2019 Technical & Operating Conference*, Steel Founders’ Society of America, Chicago, IL, 2019.

[22] U. S. I. T. Commission, *Foundry Products: Competitive Conditions in the U.S. Market, Inv. 332-460*. Washington, DC: DIANE Publishing, 2005.

[23] Southwest Research Institute, “Robotic Blending Focused Technical Project Scan-N-Plan Case Study.” https://static1.squarespace.com/static/51df34b1e4b08840dcfd2841/t/55b3d415e4b0b8cfa45a3c19/1437848597029/ROS-I+Scan-N-Plan+Final.pdf (accessed Mar. 04, 2021).

[24] Y. Wen and P. R. Pagilla, “Robotic surface finishing of curved surfaces: Real-Time identification of surface profile and control,” in *ASME 2018 13th International Manufacturing Science and Engineering Conference, MSEC 2018*, 2018, vol. 3, doi: 10.1115/MSEC2018-6659.

[25] F. Nagata, T. Hase, Z. Haga, M. Omoto, and K. Watanabe, “CAD/CAM-based position/force controller for a mold polishing robot,” *Mechatronics*, vol. 17, no. 4–5, pp. 207–216, May 2007, doi: 10.1016/j.mechatronics.2007.01.003.

[26] K. Mohy El Dine, L. Lequievre, J. A. Corrales Ramon, Y. Mezouar, and J.-C. FAUROUX, “Hybrid Position/Force Control with Compliant Wrist for Grinding,” in *MUGV 2018 (Machines et Usinage à Grande Vitesse) & Manufacturing’21*, France, 2018.

[27] Y. Oba and Y. Kakinuma, “Simultaneous tool posture and polishing force control of unknown curved surface using serial-parallel mechanism polishing machine,” *Precis. Eng.*, vol. 49, pp. 24–32, 2017, doi: 10.1016/j.precisioneng.2017.01.006.

[28] D. Wang, F. E. Peters, and M. C. Frank, “A Semiautomatic, Cleaning Room Grinding Method for the Metalcasting Industry,” *J. Manuf. Sci. Eng. Trans. ASME*, vol. 139, no. 12, 2017, doi: 10.1115/1.4037890.

CHAPTER 3. VARIOGRAM ROUGHNESS METHOD FOR CASTING SURFACE CHARACTERIZATION

Daniel Wilhelm Schimpf, Frank Earl Peters

Department of Industrial and Manufacturing Systems Engineering, Iowa State University

Modified from a manuscript published in the *International Journal of Metalcasting*

Schimpf, D.W., Peters, F.E. Variogram Roughness Method for Casting Surface Characterization. Inter Metalcast 15, 17–28 (2021). https://doi.org/10.1007/s40962-020-00451-0

Abstract

Casting surface specifications are set based on aesthetics, functionality, or a combination of both. To classify casting surfaces, visual inspections are performed by an operator who compares the casting surface to pictures or comparator plates (e.g., metal, plastic) that represent a certain roughness level. This inspection process is highly subjective, and disagreements arise on the acceptance of a casting between the casting producer and buyer. To minimize these disagreements and use developments in 3D scanning, this study aims to develop a digital surface characterization method. The method developed and implemented in this study utilizes underlying geometry estimation, abnormality detection, and a new roughness characterization formula based on a variogram to determine a surface roughness value. Tests were done to compare the new roughness characterization formula with existing quantification methods (i.e., Sa, Sq) and to compare the results of the method with human operators. The tests indicated that the variogram roughness was able to differentiate between the roughness levels of the current surface roughness standards GAR-C9 and SCRATA. In addition, the results are repeatable as well as reproducible and agree with operator judgment based on a ranking comparison between the operator and the digital method. Overall, the digital surface roughness method has the potential to improve the

communication between casting suppliers and designers and make the surface roughness classification more reliable and repeatable.

Keywords: variogram roughness, surface roughness, abnormalities, metalcasting

3.1 Statement of Authorship

This chapter was authored by Daniel W. Schimpf and Frank E. Peters. Daniel W. Schimpf conceptualized, reviewed the literature, designed, implemented, and tested the variogram roughness method under the supervision of Frank E. Peters. Finally, the manuscript was drafted by Daniel W. Schimpf and revised by Daniel W. Schimpf and Frank E. Peters.

3.2 Introduction

During the design process of castings, the casting surface is assigned a surface specification, including surface roughness. Once the casting is produced, typically, a human operator determines if a surface is within specifications by comparing the casting surface with comparators which are referred to in the standard. These comparators can be images or comparator plates made from plastic or metal. During an inspection process, the inspector compares the plate or photograph with the inspection surface to determine if the inspection surface is better than the required criteria. When using reference images, only the eyesight of the inspector is utilized to compare the surfaces. Some examples of these would be the ACI Surface Indicator Scale [1] and the MSS SP-55 Visual Method. [2] In the case of physical comparator plates, the inspector has the option to use both vision and tactile sense to compare inspection and reference surface. One of the popular standards in the USA is ASTM A802, [3] which references the plastic comparator plates provided by SCRATA, see Figure 3-1. The SCRATA comparator plates do not only classify surface roughness but also other surface characteristics like inclusions, gas porosity, discontinuities (fusion and expansion), or removal marks (thermal, mechanical, and welds). Most of these categories have four different levels ranging from smooth to rough, little to many inclusions, or shallow to deep removal marks. One of the disadvantages of the SCRATA plates is the lower resolution, meaning the difference between two adjacent surface roughness levels is higher (this will be demonstrated in the result section Figure 3-12), with usually only four levels of surface roughness in comparison with the GAR Microfinish Comparator C9 (Figure 3-2) which has nine. The GAR Microfinish Comparator C9 plate states the roughness in the form of root mean square (RMS) from 20 to 900 microinches, but it is not clear how these values were obtained, as the authors have had difficulty replicating these numerical values. Figure 3-3 shows the results of 5 profilometer measurements on each of the nine roughness levels on the GAR-C9 comparator. The figure shows that the reported RMS value and the measured RMS values do not match up. The measured results are almost always higher than the roughness level specification. For roughness levels 720 and 900, some measurements had to be repeated because the variation of the surface was higher than the profilometer was capable of measuring. This means the true roughness of the 720 and 900 roughness levels is probably even higher than the reported results. The GAR-C9 comparator also does not have any anomaly definitions, such as porosity. [4]

Figure 3-1 A1 through A4 SCRATA plates (left to right).

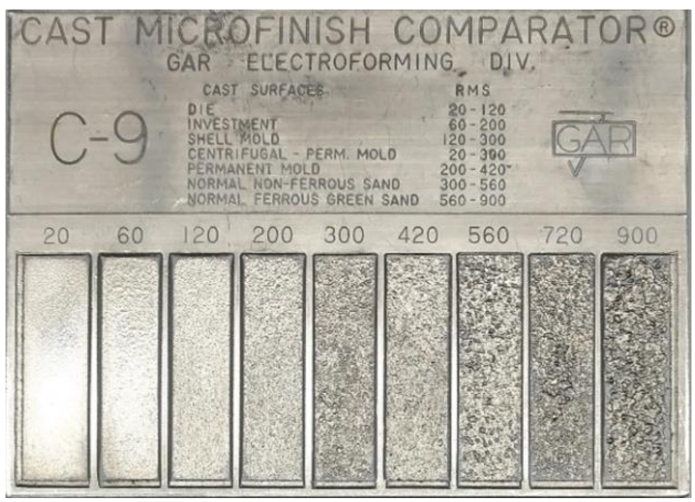

Figure 3-2 GAR microfinish comparator C9.

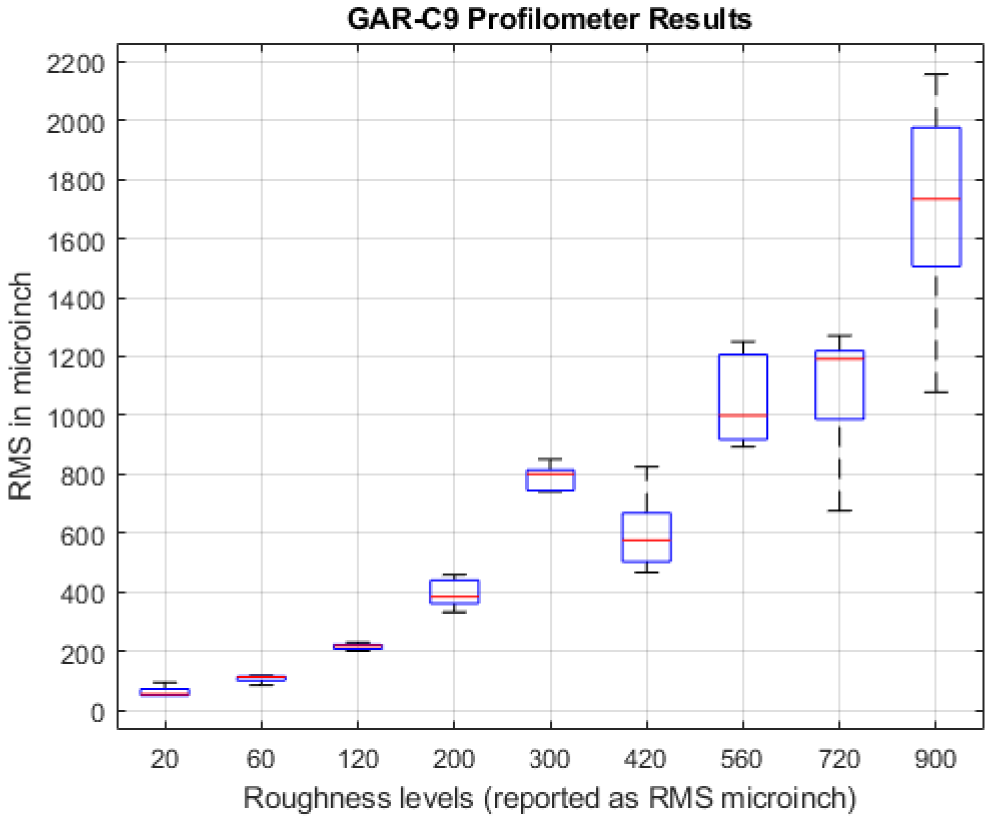

Figure 3-3 Results of 5 profilometer measurements on each roughness level of the GAR-C9 plate.

Utilizing reference images and comparator plates is subjective because different operators may arrive at different classifications. This may cause problems for producers if the casting buyer's inspector classifies the surface as rougher than the producer inspector. Much research has been done to address visual inspection errors and their causes. One such study [5] was conducted to determine the effectiveness of visual inspection using a visual comparative method, the MSS SP-55. It was found that the effectiveness of inspectors during the final visual inspection of aluminum castings ranged from 69 to 92%, with an average of about 82%. The study concluded that with visual standards like the MSS SP-55, effectiveness could only be as high as approximately 80%. A similar study [6] dealt with visual inspection error, and its measurement specifically looked at abnormalities on casting surfaces, which can also be classified using surface comparison specimens. The gage repeatability and reproducibility study looked at three different foundries with two inspectors and concluded that average reproducibility (within inspector deviation) was 45% matching of results, and the average repeatability (between inspectors) was 64% match. Both of these studies show that the classification of surface properties (roughness, abnormalities, etc.) by a human operator is subject to significant error. Another study [7] looked into some methods to improve the visual inspection process. The study showed that the inspection outcome could be improved by training operators to use systematic and thorough search patterns (rastering training).

These studies demonstrate the need for an improvement in the casting surface characterization. Training human operators to improve their repeatability and reproducibility is not enough, because one will never be able to overcome the subjectiveness inherent in the process. One way to improve the reliability of surface roughness characterizations would be to develop an objective digital method to classify casting surfaces. An automated visual inspection process for surface roughness would be problematic as it would need to rely on color or shading information. The inspection would be based on the assumption that deeper groves create darker shadows, but these darker shadows would be hard to differentiate from color changes of the surface itself. Thus, two surfaces with the same surface roughness but with one having a color pattern on its surface would likely get different roughness results.

Using 3D methods for measurement purposes is not a new idea. In the machining industry, objective methods to classify surface roughness are widely used. Some of these are contact methods like a stylus profilometer, but these methods are often time-consuming, limited in their measuring amplitude, mostly line sampling, and may not be able to detect undulations

[8][9] that exist on sand castings. Contact methods are widely used on machined surfaces where the surface often has a cyclic character and generally a lower surface roughness than castings. On the other hand, non-contact methods are able to acquire a large number of three-dimensional surface points in a short time. One popular roughness parameter calculated from 3D data is Sa (average roughness), which is essentially the 3D equivalent of the 2D roughness parameter Ra. When only one Ra profile is compared with the Sa parameter for the same surface, differences as high as 52% have been reported [10]. This difference can be reduced by averaging multiple Ra profiles and further reduced by removing profiles parallel to the main feature direction. One disadvantage of currently common areal surface characterization is the small area inspected. The Sa parameter, for example, is used in combination with microscopes where the area covered is typically less than 3 by 3 mm [11]. For machined surfaces, this small area may be representable for the whole surface, but this is most often not the case for casting surfaces.

Voelker proposed a method for casting characterization that considers the underlying geometry of the surface and possible abnormalities, but spatial information of the points in space was ignored. Voelker utilized a point cloud, which can be acquired with a variety of methods (e.g., time-of-flight, triangulation, structured light), to determine if a surface is as good as a set specification or better. For that, the designer had to define three values: a baseline roughness, an abnormality level, and an abnormality percentage. However, a weakness of this method was the inability to determine a roughness value for a specific surface. [12]

Considering the low reliability of the current visual inspection methods and that current digital methods do not consider spatial relations of 3D data, improvements for the casting surface roughness classification are necessary. The goal of this paper is to propose a method that is able to produce reliable and objective surface classifications, which include measured surface roughness values and abnormality parameters. The method proposed in this paper will consider the spatial information of the points, as well as the underlying geometry and abnormalities, to determine the surface roughness value.

3.3 Methods

The proposed digital method ‘variogram roughness’ (SVR) uses point cloud data, which includes the x, y, and z coordinates of each point. The method of acquisition is irrelevant because the roughness method only needs x, y, z data points, which can be acquired in a variety of ways, but a non-contact method is able to record ample points rapidly and is thus preferred. Ultimately, the accuracy of the final results is depended on the accuracy of the recorded points, which are used as input for the method. The proposed method will objectively quantify casting surface roughness. To do this, we focused on three properties that need to be considered: spatial information, underlying geometry, and abnormalities on the surface.

3.3.1 Spatial Information

Casting surfaces have a random, irregular pattern, whereas machined surfaces often have a cyclic character. This is why it is more important for castings to consider 3D areas instead of just relying on 2D line samples and to consider the spatial relation of the sample points. Spatial information means considering not only the height of a point (commonly z coordinate) but also its x and y position. Equations 1 and 2 represent the roughness average (Sa), and the root mean square (Sq). [13] Both equations only consider the z values of the points recorded without considering the x and y values. Therefore, any surface that has points with the same z values produces the same Sa and Sq values no matter how these points are distributed on an x–y plane. Overall, calculating the Sa or Sq value only utilizes 33% of the information acquired during a surface scan.

$$Sa = \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}\left[z\left(x_{k,}y_{l}\right)\right] \tag{1}$$

$$Sq = \sqrt{\frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}[z(x_k, y_l)]^2} \tag{2}$$

The method presented here considers the x and y coordinates as well as the z coordinate. The final result is derived from a variogram. Previous applications of variograms include the characterization of soil surface roughness. [14–16] Equation 3 shows that a variogram calculates one-half of the mean of the squared height differences for points at a specified distance.

$$\gamma(l) := \frac{1}{2N}\sum_{x}[z(x) - z(x + l)]^2 \tag{3}$$

To create a variogram, the calculation of the semivariance γ has to be done for all lag distances l that one wants to visualize. The number of pairs considered is N, and z(x) is the height at point x. The variogram itself often looks similar to the square root function. Variograms consider spatial information through distance calculations based on x and y values to determine the height differences for a distance bucket. Because roughness values are usually measured in mm and not mm2, the square root of everything on the right side of Eqn. 3 is calculated. Some changes were also made to accommodate the approximate (non-continuous) calculation and arrived at Eqn. 4.

$$v(d) := \sqrt{\frac{1}{2N(d)} \sum_{N(d)} \left[z_i - z_j\right]^2} \tag{4}$$

where N(d) is the number of pairs for the distance d which has a negative tolerance s (step size) and v(d) is the value on the variogram for distance d. Distance d is the Euclidian distance on the x–y plane. Several methods were tested to determine a single quantitative value from a variogram. In one of the study's characterizing soil surfaces, the correlation length was used to classify surfaces and was determined by finding the point of intersection of two lines. [14] One line is tangent to the beginning of the variogram and the other tangent to the end of the variogram. The y value of the point of intersection is then used to classify the surface. The implementation of this metric can be seen in Figure 3-4. The linear dashed lines are tangent to the variogram at the beginning and end. Their intersection (1.3, 0.08) marks where the correlation length is determined. This correlation length was tested for casting surfaces, but it was decided to be not as effective because of its sensitivity to outliers, specific variogram forms, and boundary effects. The method that seems to be most promising calculates the average of distance buckets in the range of the evaluation length, of which 0–5 mm seemed to be the most promising (Eqn. 5).

$$S_{VR}(e) = \frac{\sum_{N(b)} v}{N(b)} \tag{5}$$

where N(b) is the number of distance buckets b within the evaluation length e, v is the result of Eqn. 4 for each bucket, and SVR is the variogram roughness. The result of this method is then able to represent the surface roughness of a specific surface. The vertical solid line in

Figure 3-4 marks the evaluation length, while the horizontal solid line marks the variogram roughness.

3.3.2 Underlying Geometry

Surface characterization includes several parameters, including roughness and waviness. Roughness is made up of high-frequency components, whereas waviness is made of medium frequency components. [17] When calculating the surface roughness of a profile, the waviness is filtered from the recorded data, and only the roughness profile is used to calculate the surface roughness parameters such as Ra or Rq.

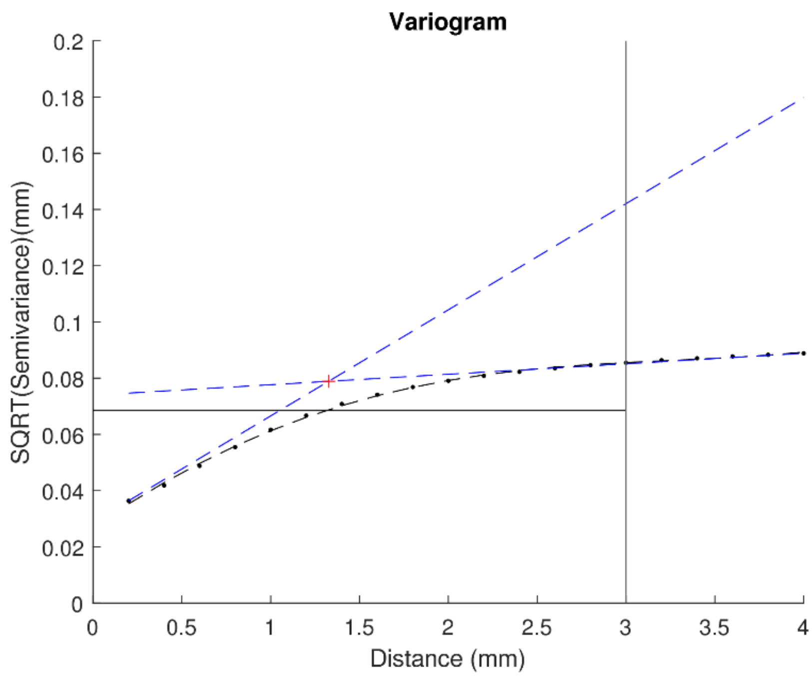

Figure 3-4 Variogram with correlation length and variogram roughness (0-3 mm).

The differentiation between roughness and waviness in two dimensions can be achieved by high and low pass filtering, but this is not easily translatable to three dimensions. What is known as waviness in two dimensions is referred to as underlying geometry in three dimensions. The determination of underlying geometry for castings is more complicated than for machined surfaces because of the inherent dimensional variability. For example, surfaces designed as planes will have some curvature after the casting process because of mold wall movement and differences in shrinkage. Thus, whereas the CAD file may be used to establish the underlying geometry on a machined part, it cannot be used to estimate the underlying geometry of a casting surface. A method is needed to estimate the underlying geometry, which then can be used to calculate a roughness value.

The method chosen to determine the underlying geometry for this study is utilizing a grid of planes, which have sides equal to the sides of the surrounding planes. Figure 3-5 shows how this method works. In the first step of the method, a grid size is set; it was set to 5 mm in both x and y directions. Next, the first set of z values is calculated by determining the mean of the points in the area. Following this, quadratic polynomial functions are then fitted to the points in x and y directions. Subsequently, the functions are used to update the z value of the grid. For every x, y coordinate, the polynomial functions in x and y directions are multiplied by 0.25 and then added to the former z value multiplied by 0.5. By giving the polynomial function and the old z values 50% of the weight, the influence of abnormalities on the grid and the effect on the grid when the quadratic polynomial functions do not resemble the underlying surface can be reduced. The grid is then used to define small rectangular surface patches, as in B in Figure 3-5. For every point, the closest surface patch is determined, and then, the distance from the point to surface is calculated. This calculated distance is used as the z value for the updated point cloud. Figure 3-5 shows the steps of the process starting with a curved point cloud. From this point cloud, the underlying geometry is estimated and used to determine the z values of the points. The color of the points in the point cloud in Figure 3-5 is determined by their difference to the mean z value of the point cloud. Small differences have a green color, higher differences a red color.

This method worked on surfaces that did not contain very complex geometries. For complex shapes, combining the general geometry information from a CAD file and then adjusting it based on this method may produce better results.

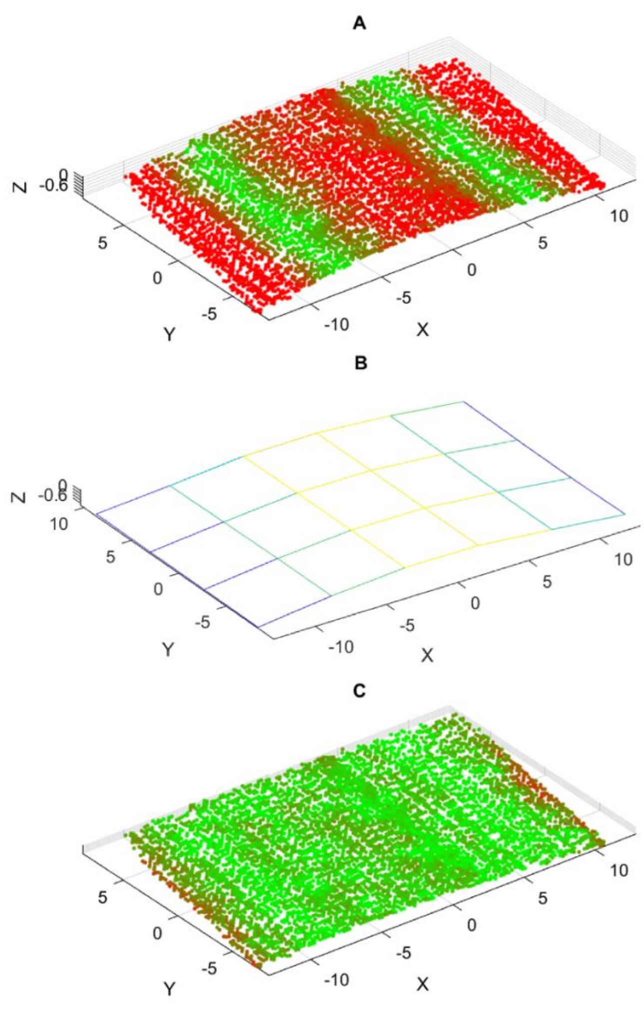

Figure 3-5 Underlying geometry and distance adjustment.

3.3.3 Abnormalities

In this project, surface deviations that deviate from the general surface roughness but are not large enough to be considered underlying geometry are considered an abnormality. On casting surfaces, this includes but is not limited to nonmetallic inclusions, gas porosity, solidification discontinuities, sand expansion discontinuities, and metal inserts. These abnormalities are an important part of a surface classification, but deciding if abnormalities on a surface are acceptable is not intuitive. For example, it is not clear if many small abnormalities are equivalent to a single large one of the same surface area. This discrepancy often occurs when a SCRATA plate is applied, which contains abnormalities (such as the E or B plates). Abnormalities affect the life of castings and their fatigue properties. Thus, they may be another important surface quality characteristic. The ability to more accurately quantify the abnormalities being developed by this method will help better develop the relationship between the surface and properties.

To determine abnormalities on a surface, some parameters have to be defined, including the allowed surface roughness and how much bigger (a multiplier) than the surface roughness a point has to be considered abnormal. A height or depth limit could be used, but it has the disadvantage that it does not relate back to the particular surface roughness. At the beginning of the abnormality detection, the point cloud is converted into a grayscale image (first image Figure 3-6). This is done by laying a grid over the point cloud. Each square in the grid resembles one of the pixels in the grayscale image. The average z value determines the color of the pixel in the corresponding grid square. Depending on the resolution of the scan, some grid squares may not have any points. These points show up as white in the grayscale image and are also recorded (missing pixels). The missing pixels are shown in the second picture in Figure 3-6. Small clusters of missing pixels are removed and then dilated. The resulting image is then used to determine areas in which no abnormality detection can take place because of limited information. Following this, the holes in the grayscale image are filled through interpolation. The grayscale image is then thresholded. For this, the surface roughness and the multiplier are used to determine how far points must be away from the underlying geometry to be considered abnormalities. Good results were achieved with multiplier values 2 or 3. In this step, it can be decided if recessive, protruding structures or both should be considered abnormalities. The sixth picture in Figure 3-6 shows this in which white areas are abnormal areas. Small clusters of white points are filled in black because they could be considered surface roughness. Following this, all white pixels are dilated and black holes within white areas are turned white as well. The eighth picture in Figure 3-6 shows the found abnormalities overlaid in red over the grayscale image. The results will include the position of the abnormalities, a point cloud in which these abnormalities were removed (Figure 3-7) and the abnormality percentage which is determined through the ratio of abnormality surface area over the total surface area.

A variation of the abnormality detection method in which the allowed surface roughness does not need to be known was also developed and tested. In this variation, in an iterative process, the underlying geometry was determined, point cloud updated, surface roughness calculated, and then based on the calculated roughness and multiplier, the abnormalities were determined. The abnormalities were then removed from the point cloud, and the next iteration starts. If abnormalities were detected, the next iteration would likely have a lower surface roughness leading to the next most abnormal points to be removed. However, if the surface roughness did not approach a roughness value higher than zero, all points may be considered abnormalities at the end. Overall, this variation was only sometimes able to produce good results when a low number of iterations were performed.

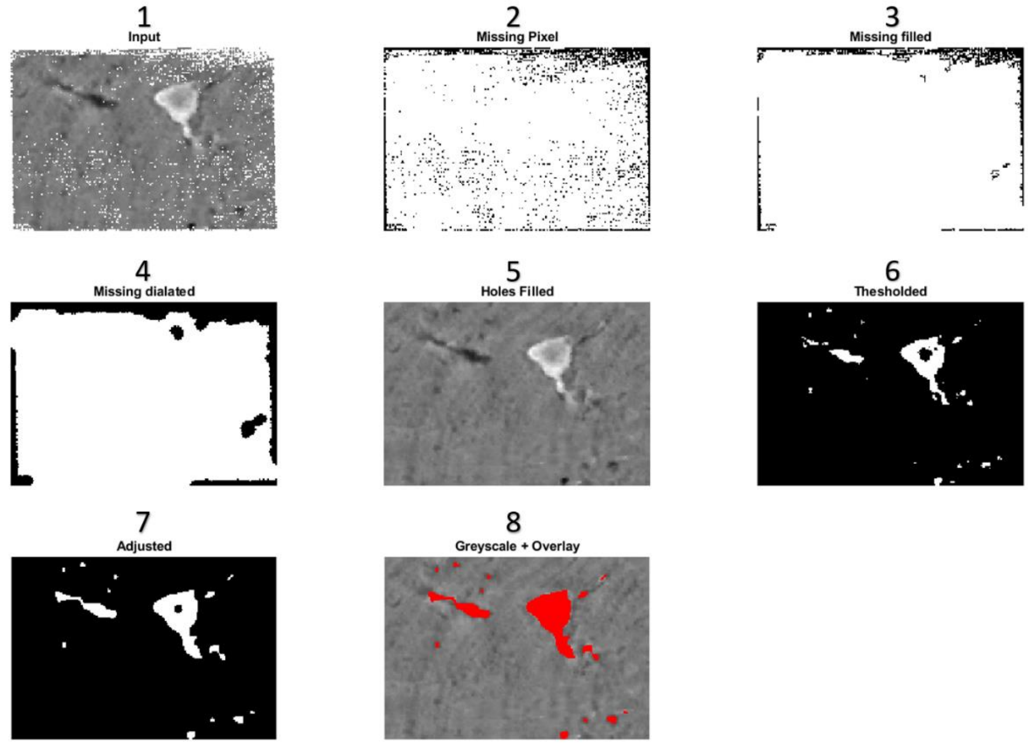

Figure 3-6 Abnormality detection based on image analysis.

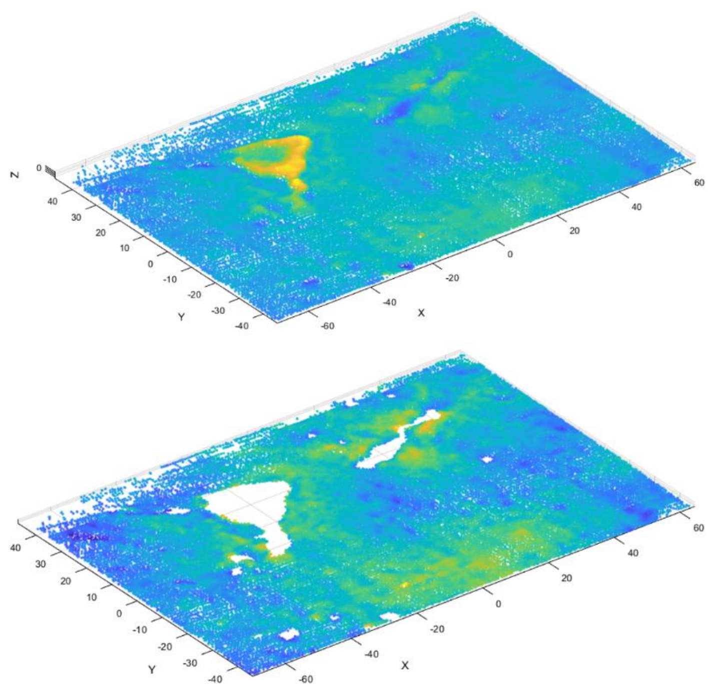

Figure 3-7 SCRATA E3 point cloud before (top) and after (bottom) abnormality removal.

Removing abnormal points from the point cloud is important for the surface roughness calculation for two reasons. First, when determining the surface roughness of a surface with abnormalities present, these abnormalities should not influence the roughness value. Second, the abnormalities on the surface skew the underlying geometry determination, and a worse fit of the underlying geometry causes higher roughness values. Overall, large abnormalities have a greater influence on the determination of the underlying geometry. Thus, excluding them from the abnormal points from the point cloud before determining the underlying geometry improves the fit of the underlying geometry.

3.4 Combining Underlying Geometry, Abnormalities, and Variogram Roughness

This section integrates the pieces described above for a cohesive surface characterization method. The method starts with the acquisition of a point cloud (Figure 3-8). This can be done using, for example, a structured light sensor, laser scanner, or stereo system. This method only uses a patch of the surface, so the patch of a desired size needs to be excised from the larger point cloud. Once the point cloud for just the patch is secured, the cloud is opened with the developed application. The user has the option of changing parameters used in the surface characterization, such as the downsampling value, the evaluation length for the roughness calculation, the grid size for the underlying geometry detection, the expected surface roughness, the abnormality multiplier, and if protruding and recessing abnormalities should be considered. This calculation will start by creating a grid of points. These points are the basis of the mesh, which represents the underlying geometry. Based on this underlying geometry, the z values of the point cloud are updated by calculating the distance for each point from the grid. The updated grid is then used to calculate the surface roughness. Based on expected surface roughness and the multiplier, the abnormalities in the point cloud are determined. Next, the abnormalities are removed from the point cloud. The grid is updated and used to calculate the new z values of the point cloud and finally, a new roughness value.

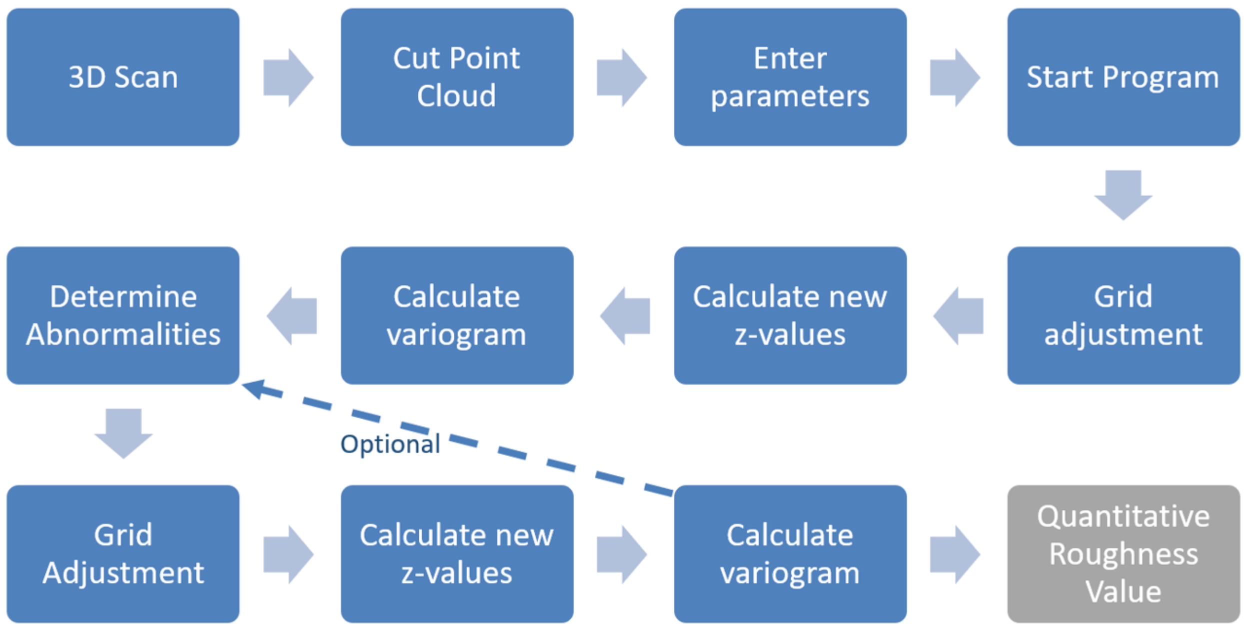

Figure 3-8 Process pipeline.

The chosen parameters influence the determined surface roughness. Picking a smaller grid size will lower the surface roughness because overfitting will likely occur. Similarly, when reducing the evaluation length of the variogram, the surface roughness value will get smaller. This is the case because if only a very small radius around a point is inspected the variations will be small. Reducing the step size of the variogram will also reduce the surface roughness value. The step size of the variogram determines the width of the bucket during the variogram calculation and since there are more points at a higher distance, they will have a bigger effect on the value for each bucket.

3.5 Results

This section presents the results of a series of evaluations that have been performed to validate the method. These studies were conducted in the laboratory as well as at casting producers.

The first study focuses on analyzing the results of the variogram without abnormality removal or underlying geometry detection and comparing them with other popular roughness values. This also shows the difference between a roughness value that considers spatial information and one that does not.

For this study, four test point clouds were created that have very similar root mean square height values. All point clouds have the same amount of points that were spread over the same x–y grid. The four created surfaces, shown in Figure 3-9 were:

1. A cloud with randomly distributed z values.
2. A cloud with a zigzag pattern (z values increase and decrease linear in y direction but stay constant in the x direction).
3. A cloud with points on two planes on different levels (levels have different z values).
4. A cloud with points on a plane that has a slope (z values increase linearly in y direction but stay constant in the x direction).

These point clouds were used as input for the surface roughness calculation, and the values Sa, Sq as well as multiple variogram roughness values with different evaluation lengths were determined (see Figure 3-11). Underlying surface detection and abnormality detection were not active.

The results of the test point cloud in the form of pictures of the variograms are presented in Figure 3-10. When looking at the variogram from the random point cloud, one can see that the mean difference between two points for all distances is about the same. This is expected since points for all distances are random. If one looks at the variogram of the zigzag point cloud one is able to see this zigzag in the variogram as well. For higher distances, this zigzag correlation becomes less visible and that is because for longer distances, more points influence the variogram roughness. The third variogram shows the result for the bi-level example and the fourth the sloped point cloud example. For both point clouds, it is valid that the higher the distance between two points the higher is the distance variation between them. For the bi-level point cloud, there is a drop off in the slope of the variogram at about 5 mm and that is because the point cloud has an x and y size of 10 mm. Figure 3-11 also shows the influence of the evaluation lengths for the variogram roughness. The evaluation length is the range of variogram points that are averaged to determine the variogram roughness. A lower evaluation length causes a smaller roughness value. The sensitivity seems to increase with a higher variogram roughness value.

The results in Figure 3-11 show that the Sa and Sq values are not able to differentiate between these test clouds very well, whereas the variogram roughness values are able to differentiate between these surfaces. One can notice that the Sq value and the variogram roughness value are the same for the ‘random’ point cloud. This is the case because these values are based on the root mean square and in the random cloud, no information is stored in the x, y coordinates. The Sa value is different because the square root of the squared z values is not taken. The difference between the non-spatial values and spatial values becomes bigger when one goes from the ‘zigzag’ point cloud, to ‘levels’ and finally to the ‘slope’ point cloud.

Furthermore, with decreasing evaluation length, the difference between nonspatial and spatial values increases.

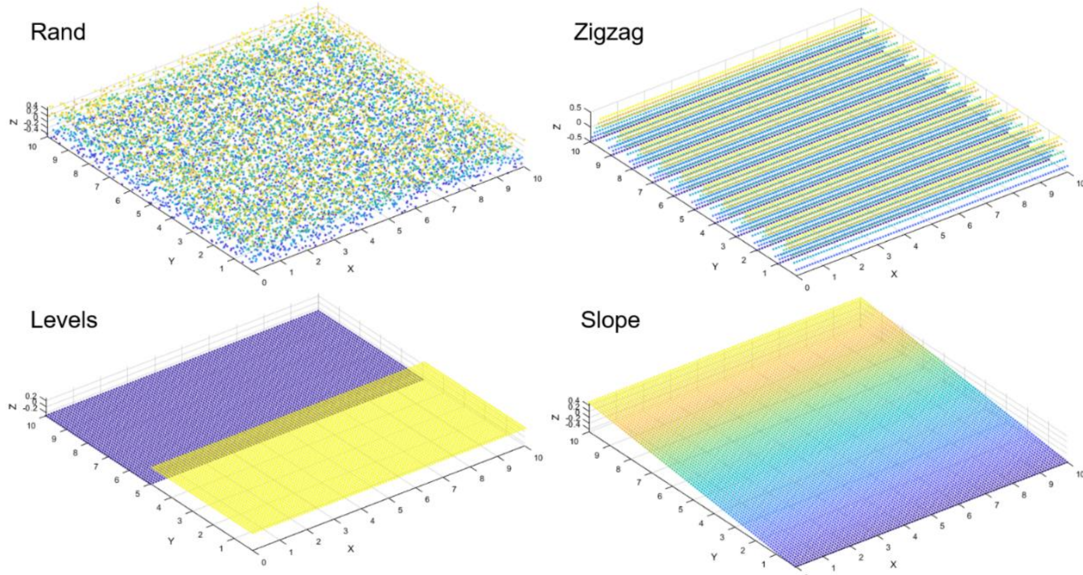

Figure 3-9 Created test point clouds.

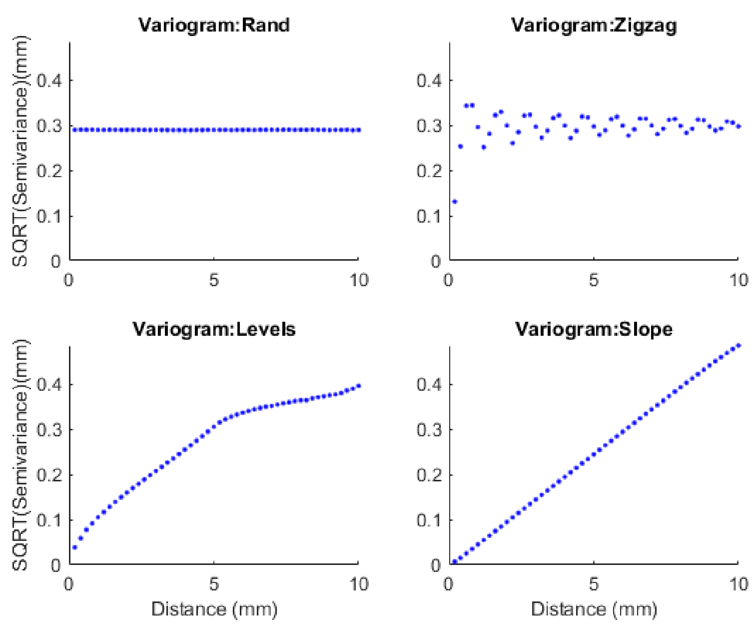

Figure 3-10 Variogram for test point clouds.

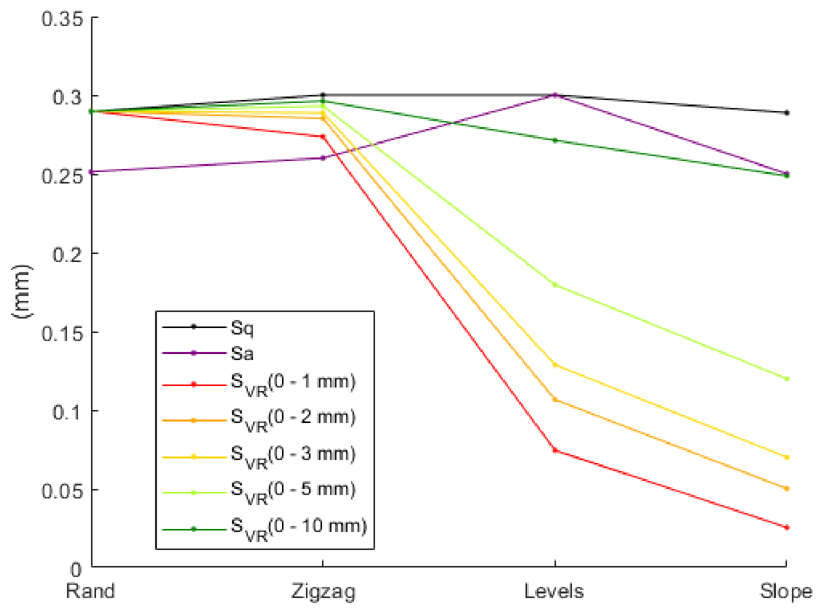

Figure 3-11 Results for test point clouds.

To be able to determine if the variogram roughness value is able to classify casting surfaces satisfactorily, the new method was used to calculate the roughness values of SCRATA and GAR-C9 plates, which are the current standards in surface roughness determination of castings. To create the boxplots in Figure 3-12, eight scans of every surface were taken, and the variogram roughness for an evaluation length of 0–5 mm was calculated. The abnormality detection was not active. One is able to see that the rougher surfaces cause a higher variogram roughness value. For these analyzed surfaces, there was also no overlap of the boxplot antennas for the different surface roughness. This means that a threshold value can be determined to differentiate the levels of surface roughness. For instance, a threshold value for the SCRATA A2 plate could be placed at 0.048 mm and 0.063 mm. Based on these threshold values, surfaces can be matched with a SCRATA roughness level. For example, if a surface has a roughness value of 0.04 mm, it would be considered A1, for 0.055 mm A2. It also shows that the method with the specific scanner is repeatable enough to differentiate between the roughness levels. Because of the accuracy of the scanner [FARO Edge and ScanArm: 3D measuring arm: ±41 µm, laser line probe: ±35 µm)[18]], the smoothest GAR-C9 surface that was analyzed was the 200 microinch RMS. Smoother surfaces were too smooth to be able to differentiate in between based on the laser scan data. If data from a 3D scanner with higher accuracy is available, the method should be able to differentiate between the lower GAR-C9 roughness levels.

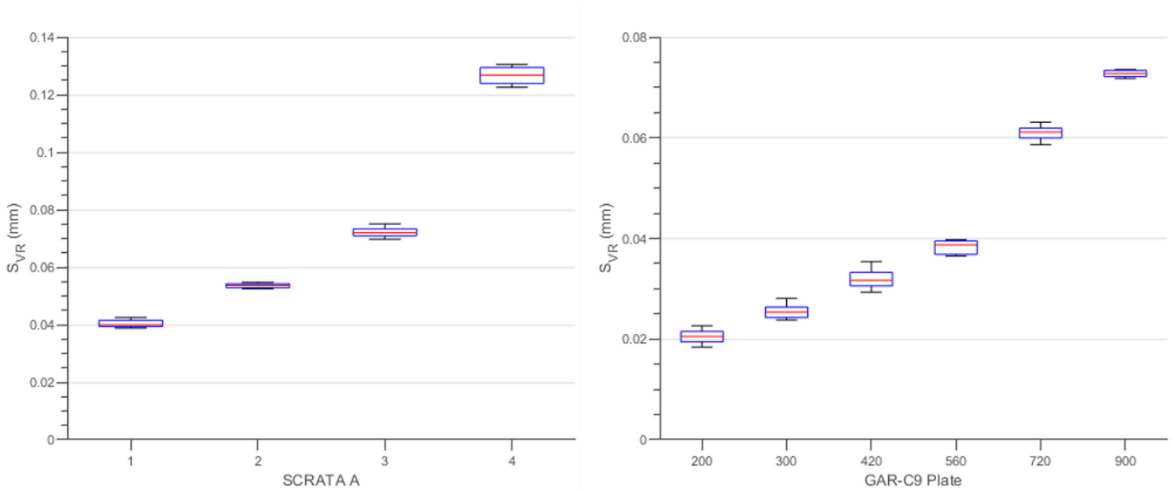

Figure 3-12 Variogram results for SCRATA A1-A4, GAR-C9.

To further analyze if this method would solve the objectivity issues of the surface inspection process, a gage repeatability & reproducibility (Gage R&R) study was performed. For this, four operators scanned each of the four SCRATA A plates three times. From these scans, the surface roughness value was calculated. The roughness value averages for each plate and operator are presented in Figure 3-13. It shows that the roughness value can distinguish between the four different surfaces, A1–A4. Table 3-1 presents the key values repeatability, reproducibility, and their combination.

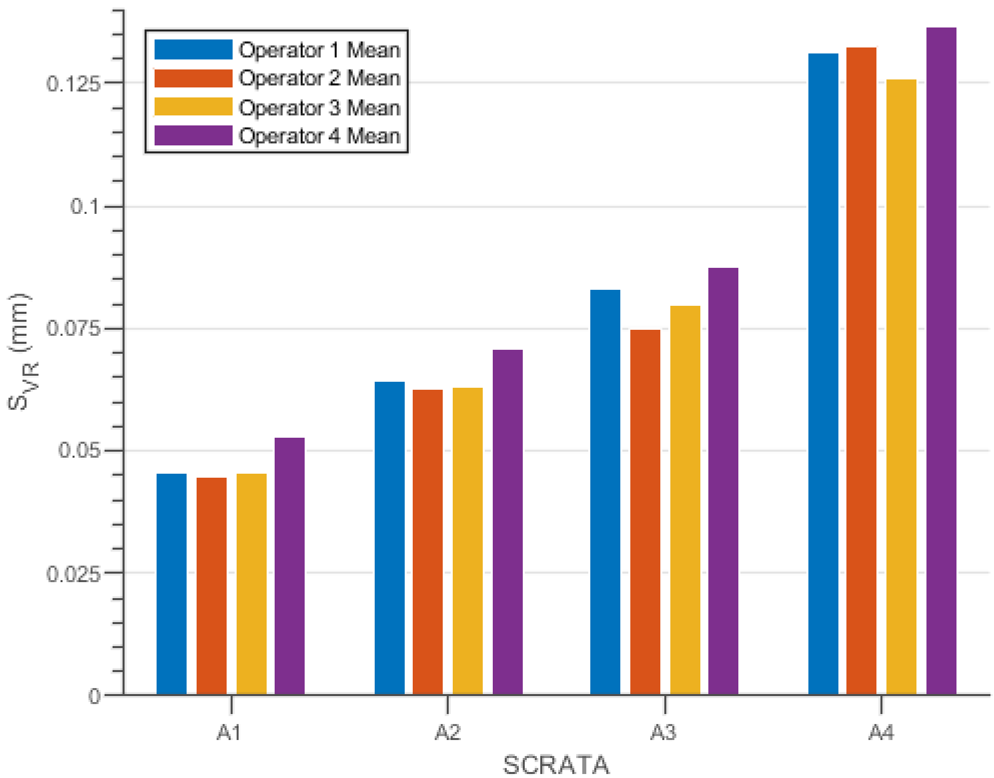

Figure 3-13 Gage R&R results, each bar represents the mean of four results.

Table 3-1 Gage R&R Results

Gage R&R Category	**Error (%)**
Repeatability (Equipment)	9
Reproducibility (Operator)	10
Repeatability & Reproducibility	13

To further investigate if the method produces results that agree with the current standard, a ranking comparison test was performed. Since this is a new method to classify surface roughness, and there is not an established digital method, one cannot just compare the values of the new method with the values of an established method. The current standard is the classification through visual inspection by operators. Human operators are not perfect at determining the surface roughness of a casting, but if the method should be used in the industry, it has to agree in some ways with the human results on surface roughness. It was assumed that human operators should be good at a pairwise comparison between plates and thus should be able to order plates from lowest to highest surface roughness. Nineteen replicas of real casting surfaces were used, and four operators ranked them. Some of these plates had abnormalities on them, which were marked, and the operators were told to not consider them in their surface roughness rankings. The replicas were laser scanned, the abnormalities that the humans were told to ignore were digitally removed, the surface variogram roughness values were calculated, and were ranked by the variogram roughness values. Figures 3-14 and 3-15 present the results. Figure 3-14 shows that for some replicas (e.g., replica 11), the consensus between operator and program is good. For other replicas, such as replica 16, one operator gives it the ranking 3 and another ranking 12. From Figure 3-15, one can conclude that there is some overall consensus. We checked for a correlation between the mean operator ranking and the digital method (program) with the Spearman's rank-order correlation. The test concluded that there is a highly significant correlation ($rs = 0.76$, $P < 0.001$). The agreement between the variogram roughness values and the operator means supports that the method is working similar to an operator. This experiment also showed the disagreement of the operators. Often when there was disagreement between the operator and the methodology, there were abnormalities that the operators were supposed to exclude but likely subconsciously considered in their evaluation.

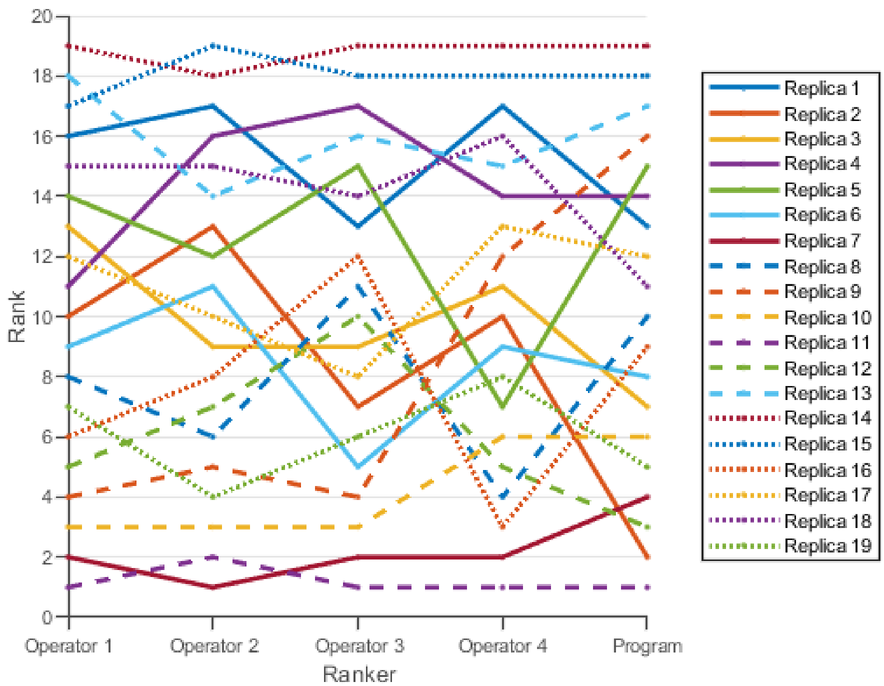

Figure 3-14 Rank comparison: ranker versus rank.

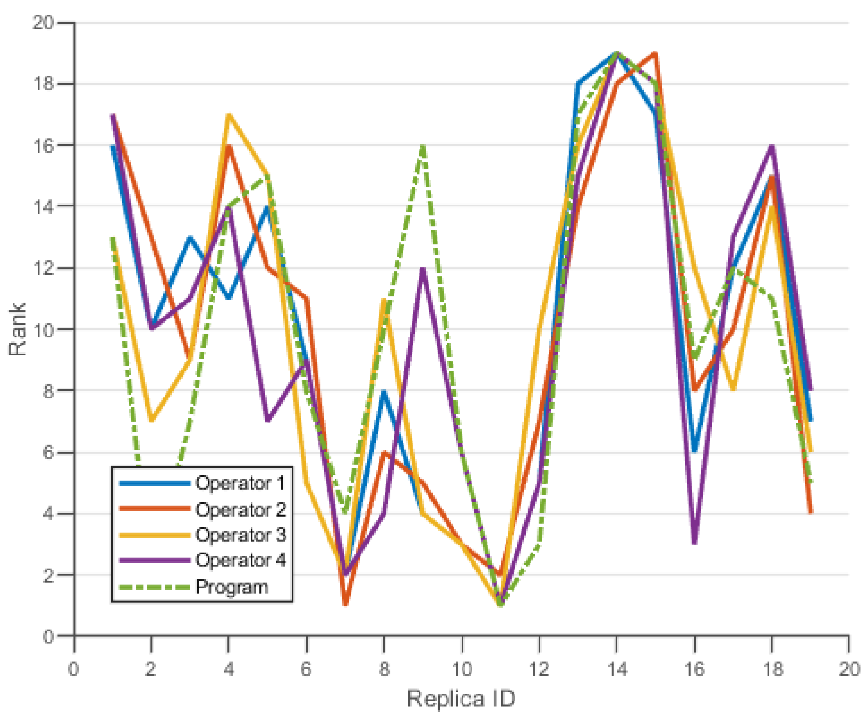

Figure 3-15 Ranking comparison: plate number versus rank.

The method was also tested on casting surface patches at three different foundries. At these foundries, surface area pairs were scanned in which one was considered acceptable (no further surface grinding required) and one that was not acceptable (grinding required to improve the surface) based on surface condition. The results showed that the method usually detected a higher surface roughness for the areas considered not acceptable. However, there were some exceptions for which the possible cause was analyzed. In some instances, the non-acceptable surfaces seemed to have been classified as unacceptable because of abnormalities and not because of high surface roughness. On other occasions, the geometry seemed to have caused higher roughness values for the acceptable patch.

To enable the usage of this roughness determination method in the industry, the effects of different scanning equipment have to be analyzed. Four different 3D scanners have been used to acquire 3D data of the SCRATA A plates for the roughness calculation. The preliminary results have shown that different scanners resulted in different standard deviations for multiple scans of the same plate. This could be explained by different accuracy or smoothing operations during acquisition. Overall, three out of four scanners have been able to differentiate between all SCRATA A1–A4 surface roughness levels. For the scanner that was not able to differentiate between all roughness levels, it is suspected that the accuracy of the scanner may not have been sufficient, because using the scan data it can be differentiated between SCRATA levels A2, A3, and A4. The results for the A1 level were slightly higher than A2. Furthermore, the SVR values for the roughness levels differ depending on the scanner. This is a problem if multiple different scanners are used for the surface roughness characterization. The investigation on the cause of these differences is still ongoing. It may be caused by different accuracies of the scanners because scanners with a higher noise level would cause the surface roughness to increase or it may be caused by the settings used during the acquisition, some scanners may filter outliers from their scans.

3.6 Discussion

Overall, the new method showed promising results in the objective surface classification. Laboratory tests showed its advantages to current digital surface roughness standards which it was able to outperform in differentiating between surfaces. That it was able to differentiate between current comparator plate specifications, means that it can potentially be used to classify surfaces similar to today's specifications. It also showed improvements in the area of repeatability and reproducibility in comparison with the current standard visual inspection, which will be able to improve communication and reduce the disagreement between producer and designer. Furthermore, it showed that the correlation between the mean result of four inspectors and the objective method is highly significant. This indicates agreement with today's standard, a human inspector. During this research, a MATLAB application was developed, which is able to calculate the surface roughness automatically.

The results also showed the importance of the underlying geometry detection. A bad fit of the determined underlying geometry can cause higher roughness values as well as decrease the accuracy of the abnormality detection.

Future work will include further testing in the industry and the following optimization of the standard parameters where necessary. Also, the investigation on how different 3D scanners and their scanning settings affect the calculated surface roughness values will be continued. The knowledge gained from this will go into developing an application that will utilize the scans of castings already performed in foundries today.

Acknowledgements

This American Metalcasting Consortium (AMC) project is sponsored by the Defense Logistics Agency Troop Support, Philadelphia, PA, and the Defense Logistics Agency Information Operations, J62LB, Research and Development, Ft. Belvoir, VA.

Disclaimer

Disclaimer: The publication of this material does not constitute approval by the government of the findings or conclusion herein. Wide distribution or announcement of this material shall not be made without specific approval by the sponsoring government activity.

References

[1] P. Eubanks, A. Crawford, L. Johnson, G. Larkin, W. Patschke, E. Schoefer, *Steel Founders'* (Steel Founders'Society of America. Crystal Lake, Alloy Casting Institute, 1969)

[2] MSS SP-55, *Quality Standard for Steel Castings for Valves, Flanges and Fittings and Other Piping Components (Visual Method)* (Manufacturers Standardization Society of the Valve and Fittings Industry, New York, 2016)

[3] A802/A802M − 95 1982 (2015), *Standard Practice for Steel Castings, Surface Acceptance Standards*, (Visual Examination. ASTM International, West Conshohocken, 2015), pp. 3–5

[4] Gar Electroforming, C-9 Cast Microfinish Comparator, (Gar Electroforming 2019) https://garelectroforming.com/index.php?route=product/product&path=59_64&product_id=73. Accessed 18 Mar 2019

[5] T. J. Schorn, AFS Trans. 114, 17–23, (2006)

[6] G. Daricilar, F. Peters, Int. J. Met. (2011). https://doi.org/10.1007/BF03355514

[7] F. Peters, R. Stone, K. Watts, P. Zhong, A. Clemons, AFS Trans., 121, 8, (2013)

[8] D.-H. Lee, N.-G. Cho, Meas. Sci. Technol. (2012). https://doi.org/10.1088/0957-0233/23/10/105601

[9] X. D. Yang, J. C. Li, and T. B. Xie, Key Eng. Mater. (2007). https://doi.org/10.4028/www.scientific.net/KEM.364-366.750

[10] H. T. Lancashire, A simulated comparison between profile and areal surface parameters: Ra as an estimate of Sa, (Arxiv, 2017), http://arxiv.org/abs/1708.02284. Accessed 23 Oct 2019

[11] U. C. Nwaogu, N. S. Tiedje, H. N. Hansen, J. Mater. Process. Technol. (2013). https://doi.org/10.1016/j.jmatprotec.2012.08.008

[12] M. M. Voelker, F. E. Peters, Mater. Perform. and Charact. (2008). https://doi.org/10.1520/MPC20160014

[13] D. M. Shivanna, M. B. Kiran, S. D. Kavitha, Procedia Mater. Sci. (2014). https://doi.org/10.1016/j.mspro.2014.07.416

[14] F. Darboux, P. Davy, C. Gascuel-Odoux, C. Huang, CATENA, (2002). https://doi.org/10.1016/S0341-8162(01)00162-X

[15] E. C. Kamphorst et al., Soil Sci Soc Am J, 64 (1992), 1749–1758, (2000). https://doi.org/10.2136/sssaj2000.6451749x

[16] T. Belem, F. Homand-Etienne, M. Souley, Rock Mech. Rock Eng. (2000)

[17] ASME 846.1-2009 Section 1, *Surface Texture (Surface Roughness, Waviness and Lay)* (ASME, New York, 2009)

[18] FARO, FARO Edge and ScanArm ES, (FARO, 2013), ftp://ftp4.jgc.gr/Pdf_files/TechSheet_FARO Edge.pdf. Accessed 06 Sep 2018

CHAPTER 4. 3D ANALYSIS OF CASTING SURFACE CHARACTERIZATION BASED ON THE VARIOGRAM ROUGHNESS METHOD

Daniel W. Schimpf, Frank E. Peters

Department of Industrial and Manufacturing Systems Engineering, Iowa State University

Modified from a manuscript to be submitted to the *International Journal of Metalcasting*

Abstract

Casting surface specifications are commonly inspected via visual inspection. Since visual inspection is subjective, the results are not very reliable. Objective methods that could replace visual inspection for simple surface patches and thus increased the reliability of the inspection process have been introduced. This paper continued the work on the variogram roughness method and adopted the method to produce reliable results for 3D scans of complete castings. This enables foundries to utilize existing 3D scanners used for dimensional inspection for surface roughness inspection of their castings and potentially using the same dataset for both needs. This paper will describe the function of this method and how it was adopted to whole castings. Tests will also be performed to validate the method in 3D. To accomplish this, the results from the validated method for flat surfaces were used to determine the surface roughness of flat surface patches on a cube. The cubes were created from plastic copies of the SCRATA comparators. The results showed that comparable results to the previous method were possible. The highest deviations were seen for rough surfaces. Further, the method can also be used as an input for automated surface finishing.

4.1. Statement of Authorship

This chapter was authored by Daniel W. Schimpf and Frank E. Peters. Daniel W. Schimpf conceptualized, reviewed the literature, designed, implemented, and tested the variogram roughness method under the supervision of Frank E. Peters. Finally, the manuscript was drafted by Daniel W. Schimpf and revised by Daniel W. Schimpf and Frank E. Peters.

4.2.Introduction

Surfaces of castings have specifications that determine what surface roughness or abnormalities are acceptable. The inspection taking place to evaluate this specification is typically a visual inspection. A human inspector compares the casting surface visually with surface comparator plates or pictures and determines if a surface is better than the required specification. One common example of the surface comparator plates in the US are the SCRATA plates, which are incorporated in ASTM A802[1]. These specify different levels of surface roughness (as seen in Figure 4-1) as well as other surface quality characteristics (such as gas porosity, inclusions, removal marks (welds, mechanical, and thermal), or discontinuities (fusion and expansion)).

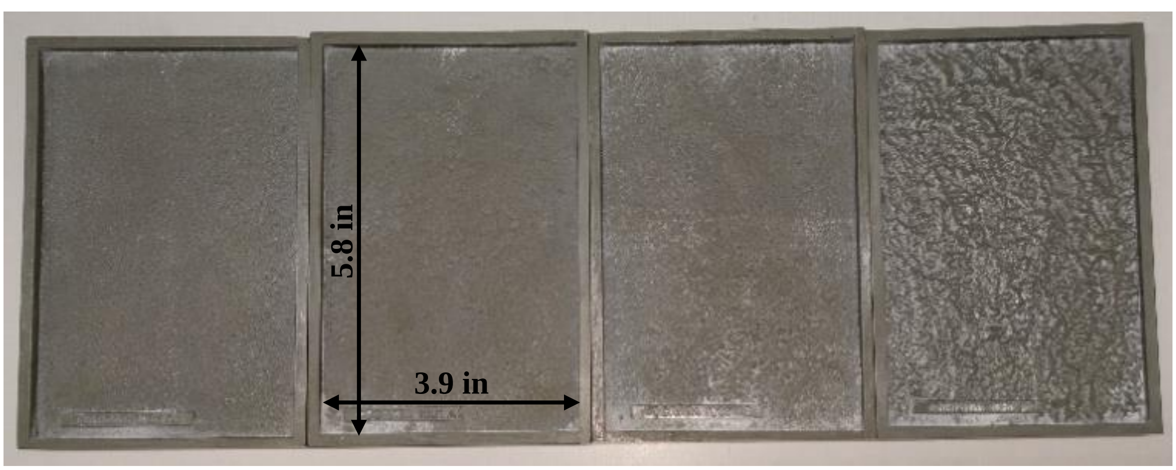

Figure 4-1 SCRATA A plates sorted from smoothest to roughest with dimensions (left to right; A1 - A4)

Multiple studies looked at the visual inspection of surfaces. One of these studies stated that visual standards could achieve effectiveness only as high as 80% [2]. Another study looked at the repeatability and reproducibility of the inspection method in a Gage R&R study. The method identified master clusters, which combines the identified areas from multiple trials, and used these clusters to identify master cluster match for repeatability and reproducibility. The results were 64% master cluster match for average repeatability and 45% master cluster match for average reproducibility [3]. Both of these studies showed that visual inspection could cause misclassification of surfaces. These misclassifications are a problem for the industry because if the inspector of the foundry and the inspector of the customer do not agree on the classifications of a surface, rework may have to take place, causing higher costs and later delivery. One study [4] provides insight into the extent of these costs and states that quality costs are in the range of 5% to 25% of the total sales value. Over 70% of these quality costs are caused by external and internal failure. Examples of external failure are delivering non-conforming products, and examples of internal failure are rework, re-inspection, and scrap cost.

Objective methods to classify surfaces exist and could be a solution to that problem. Contact measurements are one such solution, but they are time-consuming and usually only sample small surface areas [5][6]. Other options include 3D measurements using optical devices, such as laser scanners. The laser scanners return point clouds, which are a list of x,y,z data points that were sampled on a surface. Point clouds can then be assessed based on different methods. A common parameter for the surface classification of 3D data is Sa, which is the 3D version of Ra, the arithmetic mean, a popular parameter for machined surfaces. This, however, does not mean that Sa values can be easily compared with Ra values. Studies have reported differences as high as 52% between the two, depending on the surface structure and the measuring direction [7].

Further, Ra and Sa do not consider the spatial relationship of the points, abnormalities, or underlying geometry. The Voelker Surface Ratio (VSR) method [8] does consider the underlying geometry and possible abnormalities but does not consider the spatial relationships of the points. The underlying geometry is surface variations caused by the part geometry. It is a surface variation on a larger scale than surface roughness and should not be included in the roughness calculation. The points' spatial relationship is important because, for roughness evaluations, only the differences of points in close proximity are important. Further, the VSR method is able to determine if a surface fulfills or does not fulfill a certain surface specification.

To solve these problems and shortcomings, the variogram roughness method [9] was developed by the authors of this paper. This method considers the underlying geometry of the casting, the spatial information of the points, and also abnormalities. The method was shown to have a repeatability error of 9% and a reproducibility error of 10%, which is an improvement in comparison to the visual inspection. The method was further validated by showing its capability of differentiating between different surface roughness comparator plates such as the SCRATA A and GAR-C9 plates. It was also tested at multiple foundries to determine if the method is able to differentiate between acceptable and unacceptable surfaces based on the foundries' definition.

The limitation of the variogram roughness method was that it only worked for surfaces with simple geometry. The idea was that a representative surface patch with simple geometry was selected from a whole casting and then analyzed. The authors originally intended for the variogram roughness method to be implemented on a handheld device and used to classify surfaces. Conversations with multiple foundries resulted in the knowledge that many foundries already have 3D scanners to check the dimensions of their castings. Many of these 3D scanners also had similar accuracies as the 3D scanner used by the authors and thus should be able to be used to determine surface roughness values based on the variogram roughness method. This paper will present a version of the variogram roughness method, which can use scans of complete castings as an input for the surface roughness analysis.

4.3. Methods

The first part of this section will give an overview of the variogram roughness method from [9]. The variogram roughness method considers three main things: the spatial relationship of points, the underlying geometry, and surface abnormalities.

Methods like the three-dimensional arithmetic mean, Sa, do not consider the spatial relationship of points, which means that if one has a list of points with z values, it does not matter how they are spread over the surface; the Sa roughness value will always be the same. The variogram roughness method does consider the spatial relationship since it is based on a variogram. A variogram plots the height differences between points for different distances that they are apart. The variogram method calculates a variogram for a surface patch and then takes the average of the variogram for an evaluation length to determine one roughness value. This method has been shown to determine differing roughness values for surfaces with the same z values but arranged differently in space. [9]

The underlying geometry needs to be considered for roughness calculations of castings, especially because castings deviate more from the shapes specified in the design than machined surfaces. If the underlying geometry is not considered, then the deviations for the surface roughness calculation will be determined from a mean plane, and the geometry of the part will have a big influence on the calculated surface roughness value. For two parts with the same theoretical surface roughness, the one with more complex geometry will generate higher surface roughness values because the deviations from the mean plane will be higher since a plane does not describe complex geometries well, and thus result in higher surface roughness values. The underlying geometry is similar to the waviness, which is removed from profilometer measurements through frequency filters. In the variogram roughness method, the underlying geometry is considered by creating a surface mesh that resembles the underlying geometry. For each point in the point cloud, the closest surface patch is determined, and the distance from the point to mesh is calculated. These values are then used as the new z-values for the roughness calculation.

Abnormalities are surface structures that are not considered underlying geometry but are sometimes considered as part surface roughness. Some examples of what may be considered abnormalities are removal marks, gas porosity, and inclusions. The SCRATA comparator plates, for example, specify these types of surface characteristics, but they do not answer the question if one big abnormality is worse than many small ones. Because abnormalities can be considered as apart from the surface roughness, they need to be identified and removed from the calculations for an accurate surface roughness calculation. Further, since the abnormalities are not considered underlying geometry, it is important that they are not used for the underlying geometry detection. Not doing so would cause the underlying geometry to be skewed towards the abnormality because the underlying geometry would try to follow the abnormality, which in turn would cause higher roughness values for the surface around the abnormalities. Abnormalities are determined based on their deviation from the underlying geometry. They are widened and next filtered by size to remove small outliers. Abnormalities are removed from the surface before the final surface roughness calculation.

For the present study, changes had to be made to all the parts of the previous method to enable foundries to analyze complete castings and not just representative surface patches. Figure

4-2 shows an overview of the new method. In the beginning, a 3D scan of a part will be performed by an operator. An operator then starts the program, and all following steps will be automatic. The resulting point cloud will be downsampled, and its edges will be detected and removed from the point cloud. This is done because the underlying geometry detection does not work very well around edges and produces errors. After the edges are removed, clusters are extracted. For each cluster, the underlying geometry (in the form of a mesh) is estimated, and then the point to mesh distances are determined. Once the point to mesh distances for all clusters are calculated, the distances are used to determine the variogram roughness of the scanned surface. In the last step, abnormalities are detected based on threshold values and the local roughness value.

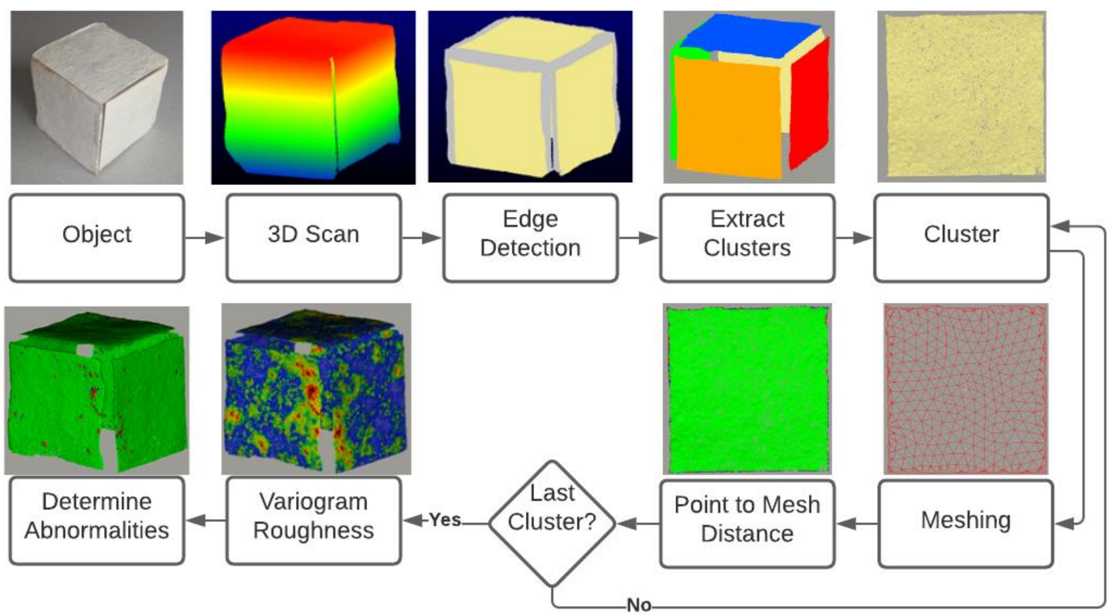

Figure 4-2: Method overview

As seen above, the first step is to acquire a point cloud of a casting. To achieve this, the casting is scanned by a laser scanner or other 3D sensor. Sparse point clouds will reduce the accuracy of the surface roughness calculations, so a more dense cloud is captured, which is later downsampled. Figure 4-3 shows a picture of a casting and a 3D scan of that casting.

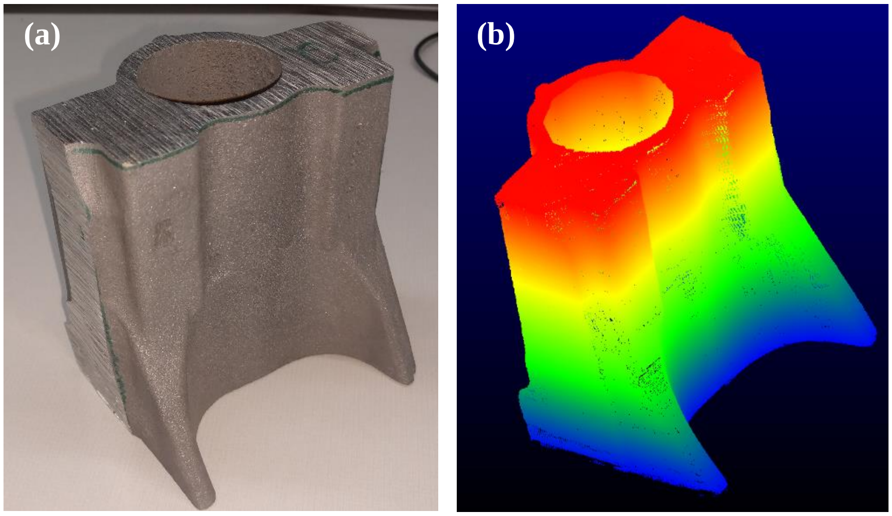

Figure 4-3: Picture of a sample casting **(a)** and scanned point cloud colored from blue to red based on the z-value **(b)**

For the implementation of the surface roughness method, the Point Cloud Library (PCL) and the Computational Geometry Algorithms Library (CGAL) were used. As the names may suggest, the PCL contains many algorithms for the manipulation and analysis of points clouds [10], and CGAL focuses more on geometry and surfaces [11]. The new method follows Algorithm 1, seen below. The following will describe each step in the algorithm.

In the beginning, the dense input point cloud is downsampled with a voxel grid filter. The voxel splits space into many voxels, which are small volumetric cubes. For each voxel, all points within the voxel have approximated by the centroid. For this application, the point cloud was downsampled with a voxel edge length of 0.2 mm. After the point cloud is downsampled, the edges of the geometry are determined. This is done because, without removal of the edges, especially around the corners, the points are further away from the mesh, as seen in Figure 4-4. This shows that the underlying geometry detection is not very accurate around the corners. It

Algorithm 1 Surface Roughness Determination Algorithm

procedure SURFACE ROUGHNESS DETERMINATION
Downsample point cloud
Determine & delete edges
Extract *Clusters*
for all the *Clusters* **do**
    Optional: Grid filter point cloud
    Downsample mesh source point cloud
    Calculate normals of mesh point cloud
    Orient normals
    Orient normals away from point cloud center
    Delaunay-based surface reconstruction
    Dense isotropic remeshing
    Distance calculation between mesh and point cloud
    Remove points on mesh corners
    Calculate Variogram Roughness
end for
Combine Variogram roughness results of *Clusters*
Determine Abnormalities
end procedure

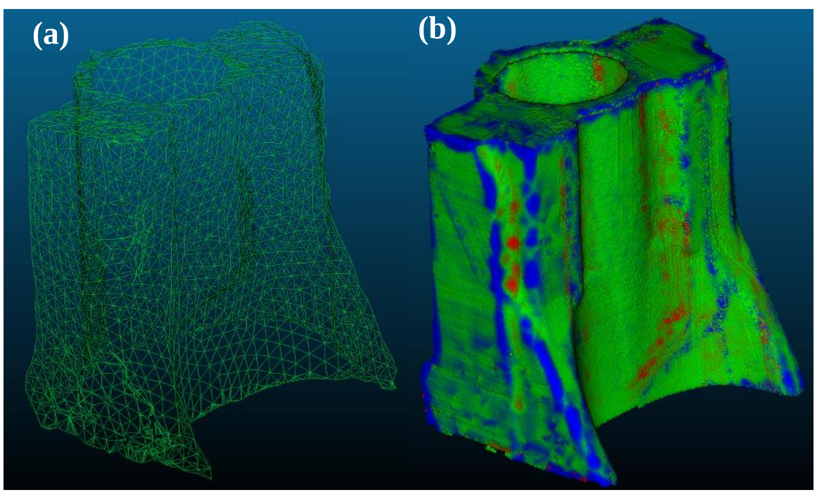

Figure 4-4 Surface mesh representing the underlying geometry **(a)** and colored point cloud based on the distance to the mesh **(b)**. Green close to the mesh; blue and red far from the mesh; red outside of the mesh, blue on the inside

smoothes the sharp edges and thus results in higher distances between points and the mesh around the edges. For an accurate surface roughness calculation, accurate point to underlying geometry distances are necessary. Therefore, the points around the edges need to be avoided for the roughness calculation.

To be able to avoid these inaccuracies, one needs to identify what points are around the edges. The Canny edge detection algorithm [12] is a widely used method to identify edges in images. An implementation of the Canny edge detection for PCL exists [13] as long as the point cloud is organized and contains color information. An organized point cloud stores each point in a matrix similar to an image and is often the result of a 3D sensor like the Kinect. The point clouds used as an input for this application are mostly unorganized point clouds. Thus, instead of a canny edge detection algorithm, an algorithm for unorganized point clouds was used. This algorithm utilizes the eigenvalues of the covariance matrix based on the k-nearest neighbors to detect sharp edges [14]. Since the point cloud distribution is not necessarily consistent, the algorithm was slightly altered to use a radius search instead of a k-nearest neighbor search. The results of this edge detection can be seen in Figure 4-5. The grey points are points that are considered edge points and all other points are yellow and can be used for the surface roughness calculation.

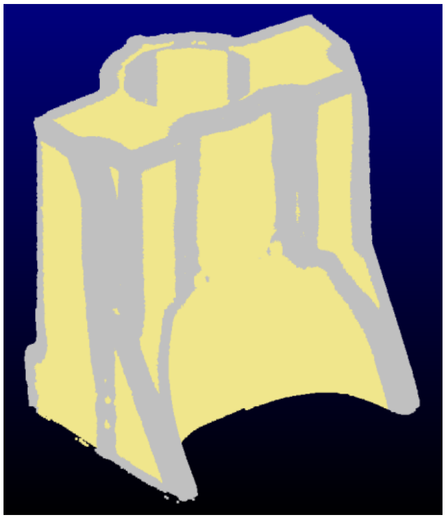

Figure 4-5 Edge points colored grey, and all others colored yellow

Once the edges are determined, they are removed from the point cloud. As can be seen in Figure 4-5, the edges split the point cloud into multiple smaller areas. Thus, the point cloud is next separated into multiple clusters, which all fulfill a separation (d = 10 mm) and minimum size (n = 5000) criteria. Next, for each cluster, a couple of calculations take place. Since the variogram roughness method is sensitive to noise, optionally grid filtering can be performed. This grid filter can remove more noise from the point cloud data than the voxel grid filter with voxels of equal edge lengths. This grid filter aligns the cluster with the best fitting plane and then utilizes a voxel grid filter where the x and y edges are equal to the downsampling value while the z value is much higher. This way, all points within one x-y grid are combined to one point. This filter works well as long as the point cloud clusters size in the two main dimensions (x & y) is much larger than in the third dimension (z).

Next, a point cloud for the mesh is downsampled even more for the upcoming meshing. The point cloud's normals are then calculated and oriented in the following step utilizing the orient normals method from CGAL [15]. Utilizing a greedy Delaunay-based surface reconstruction algorithm [16], a dense mesh is created of the point cloud with oriented normals. This densely meshed surface is remeshed to create a uniform mesh surface with a goal edge length of 5 mm. The remeshing is performed by utilizing the isotropic remeshing algorithm [17][18] implemented in CGAL. The reconstructed surface will be considered the underlying geometry of the casting and represented as a mesh. In the next step, for each point the closest point on the mesh and the distance to the closest point on the mesh is calculated. This is one of the computationally most expensive operations. The mesh may not cover all points in the point cloud, resulting in high distance results for these points. Thus, these points are removed from the point cloud.

To be able to determine the variogram surface roughness value, both the distance from point to underlying geometry and distance from point to point are important. The distance from point to underlying geometry was previously determined. The points for the point to point distance will be the corresponding closest points on the mesh determined in the previous step. The distance from point to point is necessary to consider the spatial relationships. The variogram, which is used for this method, plots the height difference between points for different distances. If the points are closer together, the height difference is usually lower. There are two common ways the distance between the points can be determined. The easier of the two methods is just determining the cartesian distance between the two points. To calculate the cartesian distance, only the location of both points is necessary. The second method is using the shortest distance along the mesh to connect the two points. This is more accurate for our purpose because, for the relationship between the points, the surface distance has more meaning than the cartesian distance.

Both methods were implemented for the variogram roughness calculation. The surface distance measurement is the more accurate one, but on a computer with a six-core processor (i7-8850H), the calculations took over an hour for the example part and are thus too long to be feasible in practice. The calculation of the surface distance is that much more computationally expensive than the cartesian distance because a search algorithm has to determine the closest path between the two points on the surface. The point clouds for the surface roughness calculation can quickly have millions of points. For each one of those points, the distance to thousands of points has to be calculated. Thus, the computational time for this distance calculation is high.

In practice, the cartesian distance will be used to evaluate the distance between two points on the surface. The biggest inaccuracies for the cartesian distance calculation occur around edges. By not using edge points for the roughness calculation, the cartesian distance is a good compromise between calculation time and accuracy.

Once the distances between points on the surface as well as the point to surface distances are determined, the variogram roughness value can be calculated for each point. These variogram roughness values can color the point cloud and present the local roughness on the point cloud. One can see examples of this in the result section. This can help identify areas on the surface of different roughness. After the variogram roughness values for all clusters are calculated, they are combined to determine the scanned surface's final roughness value without its edges.

The described method would provide one overall roughness value, but a casting might have a variety of different surface roughness's spread over the surface. Further, the underlying geometry detection is not perfect; thus, more accurate values will likely be achieved if analyzing a simple surface like a plane. For both cases, it may be advantageous to cut out a smaller section and determine the smaller section's surface roughness.

After the local roughness values were determined, the information can be used to determine abnormalities on its surfaces. This can be done by comparing the local surface roughness values with the overall roughness value. If, for example, a local surface roughness is twice as large as the final roughness value, it could be considered abnormal. The method differentiates between abnormalities where material is in excess or missing by utilizing the relationship between the oriented normal vectors and the center of the point cloud.

4.4. Results

The SCRATA comparator plates were used to validate the implementation of the method to analyze three-dimensional castings rather than just flat surfaces. Rubber molds (3 x 3 in) of the SCRATA plate sections were used to create plastic copies of the SCRATA plate sections, Figure 4-6. These squares are used to create epoxy copies of the SCRATA plate molds.

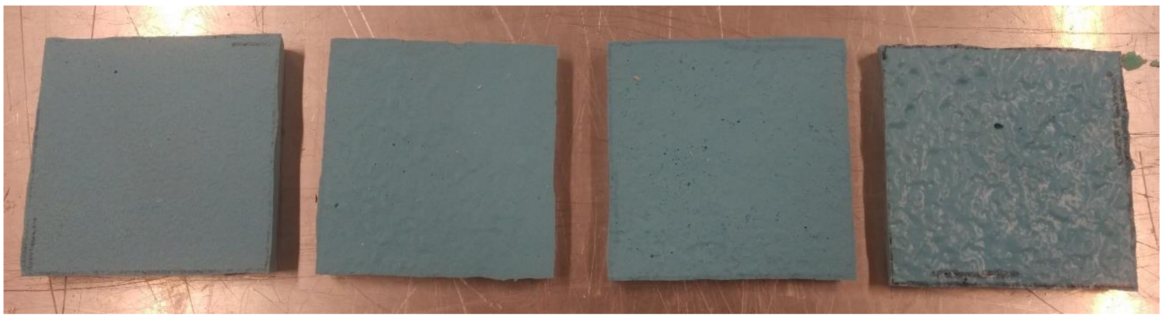

Figure 4-6 Rubber molds of SCRATA A1-A4 comparators

Five of the plastic copies are assembled to create a three-dimensional cube where all sides are one of the four SCRATA surface roughness levels (A1, A2, A3, and A4); see Figure 4-7 & Figure 4-8. The cubes will have discontinuities/gaps around the corners, but the edge points are not used for the roughness calculation and therefore, should not impact the results. Since only a square section of the SCRATA plates is used, and inaccuracies in the copies exist, the roughness results are not expected to be exactly the same as for a SCRATA plate but similar.

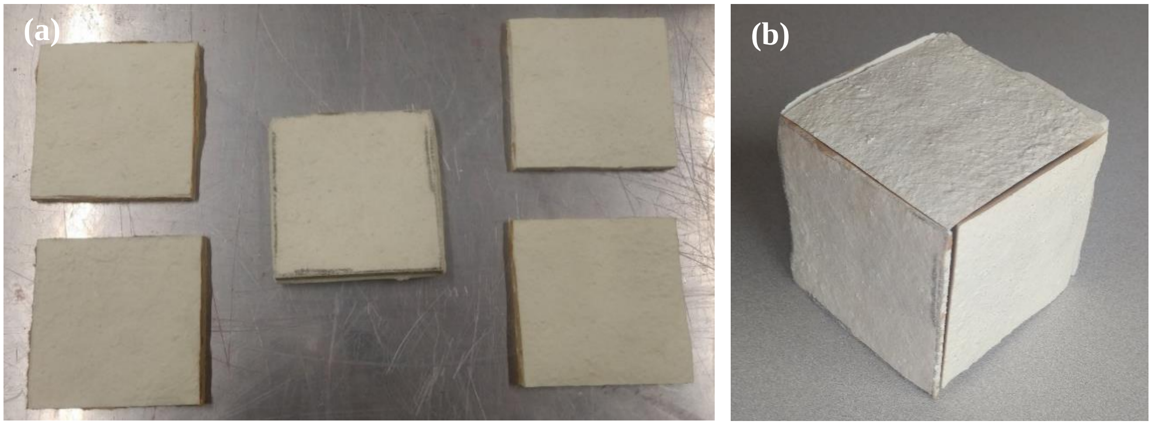

Figure 4-7: 5 SCRATA A3 copies **(a)** and SCRATA A3 cube **(b)**

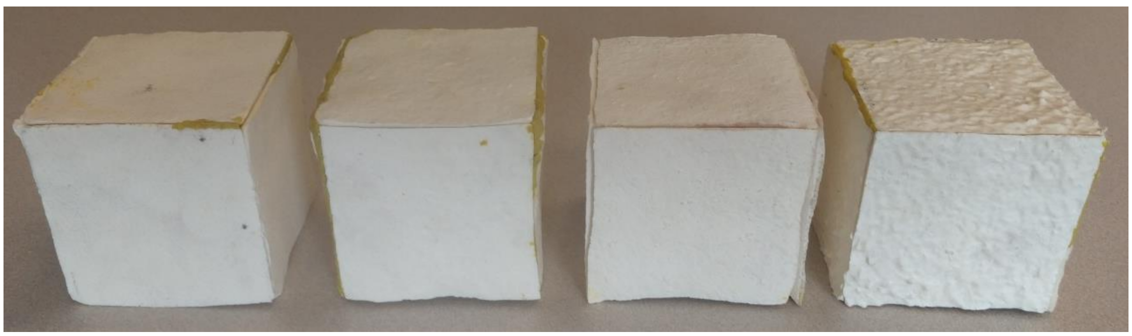

Figure 4-8: SCRATA cubes A1- through A4

To determine if the old validated method and the new method are able to produce similar results, the SCRATA cubes were scanned. These scans were gathered with two different 3D scanners, one of which had very dense but nosier results. In contrast, the other scanner's results were defined by low noise but a sparse point cloud with a non-uniform point distribution similar to a meshed surface. For the first part of the test, each scan of a SCRATA cube was cut into five smaller scans. Each smaller scan is one side of the cube. These 20 scans were then analyzed using the old MATLAB method and the new method implemented in C++. For all following tests for both methods, the point clouds were downsampled to 0.2 mm, and an evaluation length of 5 mm was used for the variogram roughness calculation. Further, for the underlying geometry detection, a surface patch size of 5 mm was chosen. For the old method, this means a grid size of

5 x 5 mm, and for the new method, a goal edge length of 5 mm for the triangles. This test is supposed to investigate the variation between the results if both methods receive the same input. The results can be seen in Figure 4-9 **(a)**, where A1-1 through A1-5 are scans of the five sides of the SCRATA cube A1. For each scan, two results are displayed. The filled circle represents the new method, while the unfilled circle represents the old method. One can see that the results are very similar. Most of the time, the old method produces slightly rougher results. **(b)** presents the deviation between the old and new method. A positive deviation corresponds to the result of the old method being larger. **(b)** further shows that the deviations all are smaller than 0.008 mm, all except one even smaller than 0.005 mm.

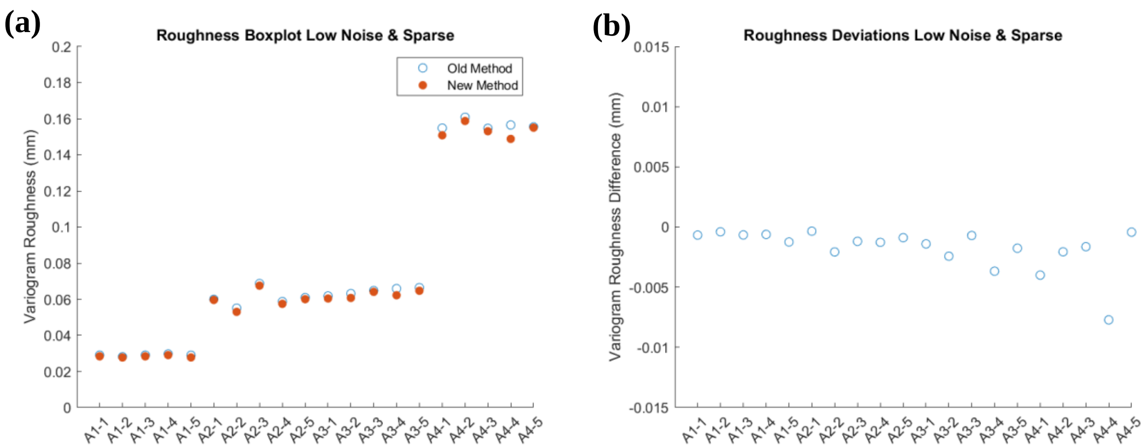

Figure 4-9: Roughness results for the sides of SCRATA cubes analyzed separately (Low Noise & Sparse)

Figure 4-10 presents the same data as Figure 4-9 but for the scans with the dense but rather noisy data. One can see that the differences between the old and new method are larger in this case. Since the point cloud is noisy, the new method utilizes the grid filter. One can see that the variations are larger than for the point cloud with low noise. The maximum deviation is now closer to 0.013 mm.

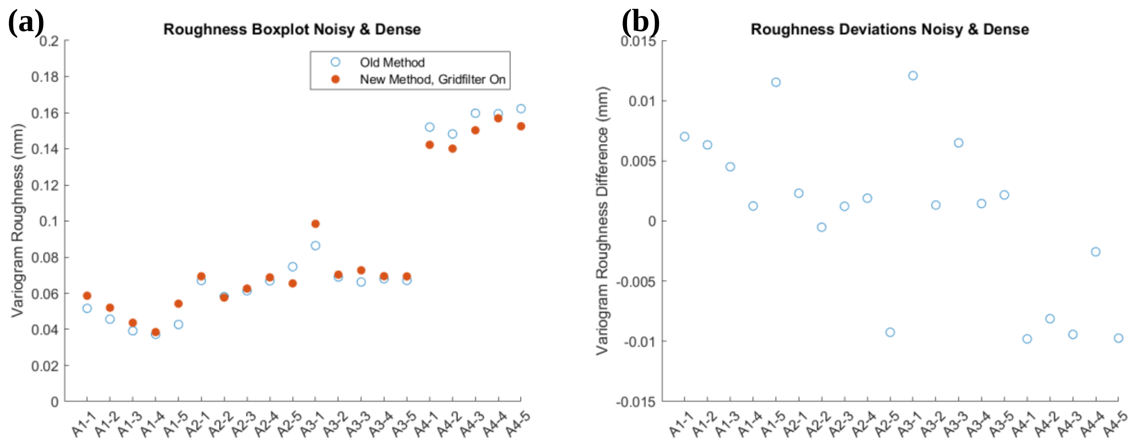

Figure 4-10: Roughness results for the sides of SCRATA cubes analyzed separately (Noisy & Dense)

While Figure 4-9 showed the results of the SCRATA cube sides analyzed individually for both methods, Figure 4-11 **(b)** shows results where a scan of the whole cube was used as an input for the new method. The boxplots in Figure 4-11 **(a)** present the results of the four different SCRATA cubes where the five sides are analyzed separately by the old method. The new method uses the whole scan and then removes the edges automatically, while for the old method, the edges are removed manually so that they can be analyzed separately. Since the input will differ, some variation is to be expected. Comparing **(a)** and **(b)**, one can see that the results are similar, but especially the results for the SCRATA A4 cube are different. This can be explained by issues in the edge detection. The edge detection uses the curvature in an area to find edges. While this works very well for A1-A3 surfaces, A4 surfaces themselves have high curvature, which causes the new method to classify part of the A4 surface as edges and not use them for the roughness calculation. Since the areas with high curvature are also areas where more considerable height differences between points are to be expected, this overall reduces the roughness reported by the new method for the SCRATA A4 cubes.

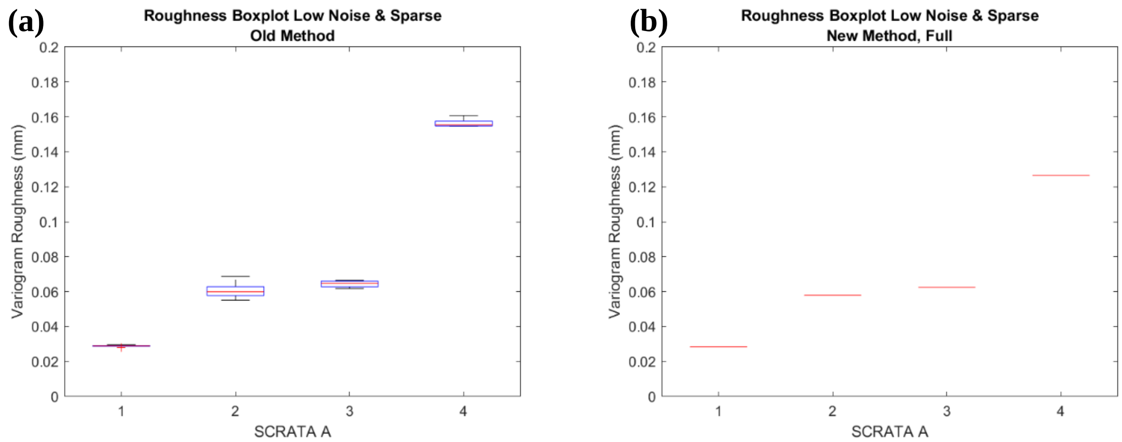

Figure 4-11: Low Noise & Sparse: Roughness result comparison between the old **(a)** (five results for each roughness level) and new **(b)** method (one results for each roughness level)

Again Figure 4-12 shows the same data as Figure 4-11, but for the scans with the dense and noisy data. Comparing the results, the variation between the five sides of the cubes is a little higher for the noisy data, and the results for the lowest roughness level A1 are higher in general.

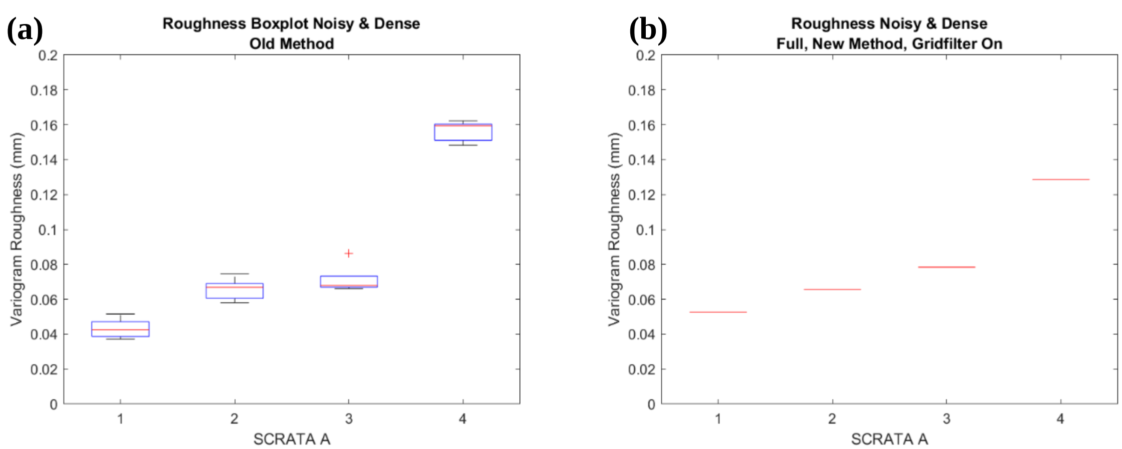

Figure 4-12: Noisy & Dense: Roughness result comparison between the old **(a)** (five results for each roughness level) and new **(b)** method (one results for each roughness level)

In the results presented above, the cubes' edges were removed either manually or with automatic edge detection. Figure 4-13, on the other hand, shows the roughness results achieved by the new method without removing the edges to check what kind of error is introduced if the edges are not removed. **(a)** presents the results for the scans with low noise but sparse data while **(b)** shows the results of noisy but dense data. The results are not close to the old method results, as seen in Figure 4-11, which shows that the edge removal is necessary because of the meshing operation's inaccuracies.

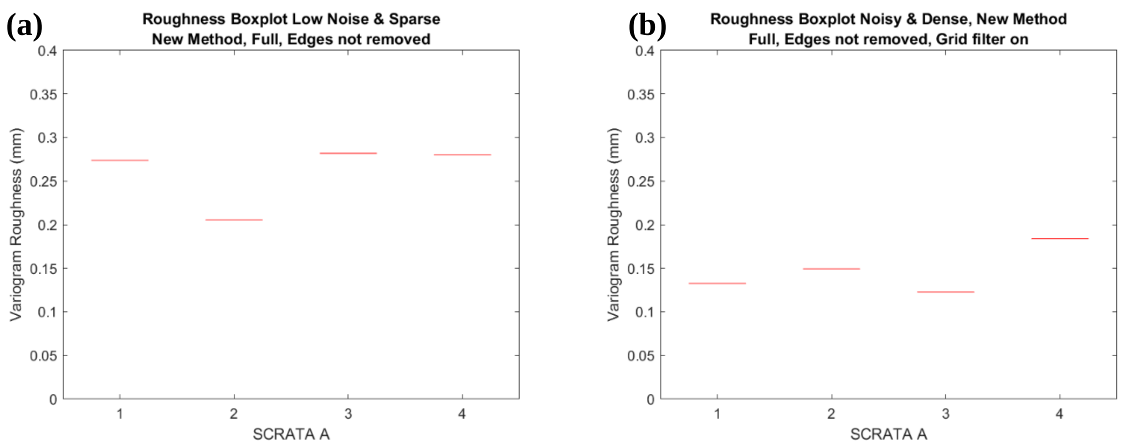

Figure 4-13: Roughness result without edge removal Low Noise & Sparse **(a)** and Noisy & Dense **(b)**

Similar to how scans are compared to CAD files for dimensional analysis, the point clouds can be colored based on the local variogram value to visualize roughness variation of the surface. This can be useful to identify areas in which subsequent finishing operations like grinding are necessary. Figure 4-14 presents the variogram colored point clouds for the SCRATA A1 through A4 cubes. The coloring scheme can be seen in Table 4-1 and is based on previous results acquired with the old method.

Table 4-1 Variogram color scheme. Between values, colors are interpolated

Local Variogram Roughness Value	Color	SCRATA Equivalent
0 mm	White	-
0.04 mm	Blue	A1
0.055 mm	Green	A2
0.072 mm	Yellow	A3
0.13 mm	Red	A4
0.2 mm	Black	-

From Figure 4-14, one can see the differences between the roughness levels. While A1 and A4 are very different, the color differences between A2 and A3 are much smaller. This matches the roughness results shown in the previous figures.

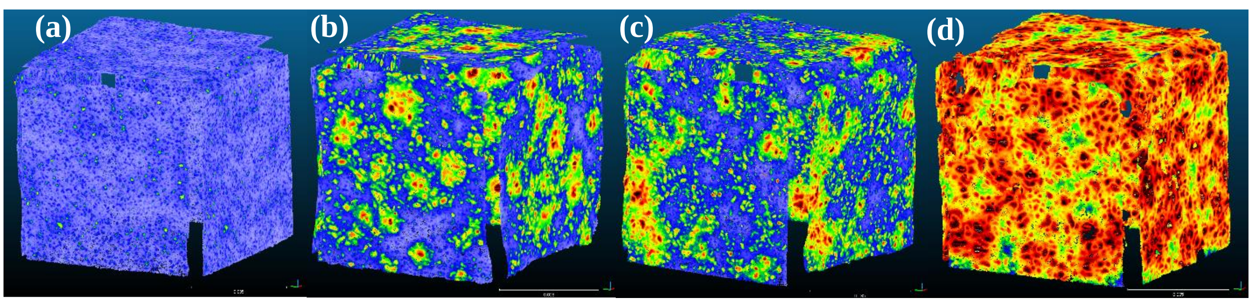

Figure 4-14: Variogram colored SCRATA A1 **(a)** through A4 **(d)** cubes (Low Noise & Sparse)

Figure 4-15 shows an example where a cube where the sides do not have the same surface roughness. One can see a clear difference between the three facing sides. Utilizing Figure 4-14 or Table 4-1, one can determine that the left side's surface roughness is close to A1, while the top's surface roughness is close to A3 and the right side's surface roughness is close to A4.

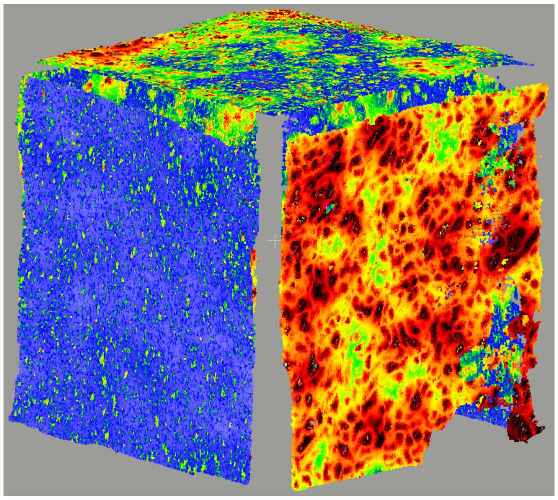

Figure 4-15: Variogram colored cube with different sides. Left-facing side A1; right-facing side A4; top A3

The method also includes an abnormality detection that, similarly to the point cloud's variogram coloring, can assist in identifying abnormalities on the casting. In this case, abnormalities are defined as having two times the variogram roughness than the average variogram roughness of the casting. Excess material is marked in red while missing material is marked in blue. Figure 4-16 shows an example of abnormality detection on the SCRATA cubes. Since they do not have any big abnormal areas, only small spots are marked.

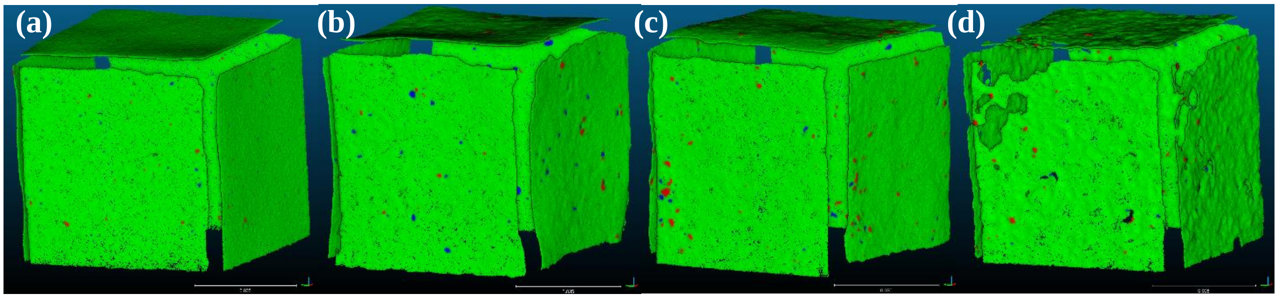

Figure 4-16: Detected abnormalities on SCRATA A1 **(a)** through A4 **(d)** cubes (Low Noise & Sparse)

4.5. Discussion

The tests showed that the method is able to produce comparable results for SCRATA A1-A3 cubes. The roughness results for the SCRATA A4 cube were comparable if the edges were removed manually. However, the automatic edge removal currently removed too much of the surface for A4 surfaces, which leads to lower roughness results than the previous method. Overall, this should be acceptable because castings with surfaces as rough as the SCRATA A4 plate are not very common. Nonetheless, one way around this would be to compare the scanned surface data with the castings CAD model. While this presents other issues as the casting does not match the CAD model perfectly, the variogram roughness method may be able to filter out the deviations between the CAD and actual casting geometry and still produce good roughness results.

This method's limitations further include that the method does not evaluate the whole casting since the edge surfaces are not evaluated. For most castings, this should nonetheless enable a roughness evaluation of most surface areas.

The roughness results from the two different inputs, one noisy and dense, the other sparse but less noisy, showed the importance of good scan data. Overall the scan data with less noise was able to produce more consistent results. The method's ideal input data would be a dense point cloud, with uniform point distribution and low noise. The low noise scan data available to us did not have a uniform point distribution because of the smoothing taking place in the scanner's software.

Tests have also been performed with underlying geometry detection based on Bezier curves, but the results were, in general, worse than the underlying geometry detection presented above. Nonetheless, it would be advantageous to test more methods for the underlying geometry estimation and their influence on roughness results.

Overall this surface roughness measurement method should make the technique more accessible for foundries since now scans of whole castings can be analyzed. It should be possible for the foundries to include the method to the geometric analysis of castings already taking place utilizing the same 3D scan. The method could also be used to identify areas on surfaces where a subsequent automatic finishing operation reduces the surface roughness or removes abnormalities.

Acknowledgments

This research is sponsored by the DLA-Troop Support, Philadelphia, PA and the Defense Logistics Agency Information Operations, J68, Research & Development, Ft. Belvoir, VA.

Disclaimer:

The publication of this material does not constitute approval by the government of the findings or conclusion herein. Wide distribution or announcement of this material shall not be made without specific approval by the sponsoring government activity.

References

[1] A. /A802 − 95 1982 (2015), "Standard Practice for Steel Castings, Surface Acceptance Standards, Visual Examination," *ASTM Int. West Conshohocken, PA*, pp. 3–5, 2015, doi: A0802-95R15.

[2] T. J. Schorn, "Visual Inspection Errors : Measurement and Impact," *AFS Trans.*, vol. 114, no. 01, pp. 17–23, 2006.

[3] G. Daricilar and F. Peters, "Methodology for Assessing Measurement Error for Casting Surface Inspection," *Int. J. Met.*, vol. 5, no. 3, pp. 7–15, Jul. 2011, doi: 10.1007/BF03355514.

[4] D. Zimwara, L. Mugwagwa, D. Maringa, A. Mnkandla, L. Mugwagwa, and T. T. Ngwarati, "Cost of Quality as a Driver for Continuous Improvement - Case Study – Company X," *Int. J. Innov. Technol. Explor. Eng.*, vol. 2, no. 2, pp. 132–139, 2013.

[5] D.-H. Lee and N.-G. Cho, "Assessment of surface profile data acquired by a stylus profilometer," *Meas. Sci. Technol.*, vol. 23, no. 10, p. 105601, Oct. 2012, doi: 10.1088/0957-0233/23/10/105601.

[6] X. D. Yang, J. C. Li, and T. B. Xie, "A Large Measuring Range Profilometer for Three-Dimensional Surface Topography Measurement," *Key Eng. Mater.*, vol. 364–366, pp. 750–755, Dec. 2007, doi: 10.4028/www.scientific.net/KEM.364-366.750.

[7] H. T. Lancashire, "A simulated comparison between profile and areal surface parameters: Ra as an estimate of Sa," pp. 1–9, Aug. 2017, Accessed: Oct. 23, 2019. [Online]. Available: http://arxiv.org/abs/1708.02284.

[8] M. M. Voelker and F. E. Peters, "Development of a digital standard to specify surface requirements of cast metal surfaces," *Mater. Perform. Charact.*, vol. 6, no. 2, pp. 130–143, 2017, doi: 10.1520/MPC20160014.

[9] D. W. Schimpf and F. E. Peters, "Variogram Roughness Method for Casting Surface Characterization," *Int. J. Met.*, vol. 15, no. 1, pp. 17–28, Jan. 2021, doi: 10.1007/s40962-020-00451-0.

[10] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 1–4, doi: 10.1109/ICRA.2011.5980567.

[11] P. Alliez and A. Fabri, "CGAL: the computational geometry algorithms library," in *ACM SIGGRAPH 2016 Courses on - SIGGRAPH '16*, New York, New York, USA, 2016, pp. 1–8, doi: 10.1145/2897826.2927362.

[12] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986, doi: 10.1109/TPAMI.1986.4767851.

[13] Willow Garage, "Point Cloud Library (PCL): pcl/features/organized_edge_detection.h Source File." https://pointclouds.org/documentation/organized__edge__detection_8h_source.html (accessed Feb. 26, 2021).

[14] D. Bazazian, J. R. Casas, and J. Ruiz-Hidalgo, "Fast and Robust Edge Extraction in Unorganized Point Clouds," *2015 Int. Conf. Digit. Image Comput. Tech. Appl. DICTA 2015*, 2015, doi: 10.1109/DICTA.2015.7371262.

[15] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," *Comput. Graph.*, vol. 26, no. 2, pp. 71–78, 1992, doi: 10.1145/142920.134011.

[16] D. Cohen-Steiner, F. Da, and F. A. Da, "A Greedy Delaunay Based Surface Reconstruction Algorithm," *RR- 4564, INRIA.*, 2002, Accessed: Feb. 15, 2021. [Online]. Available: https://hal.inria.fr/inria-00072024.

[17] M. Botsch and L. Kobbelt, "A Remeshing Approach to Multiresolution Modeling," in *Eurographics/ACM SIGGRAPH symposium on Geometry processing*, 2004, pp. 185–192.

[18] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Lévy, *Polygon mesh processing*. CRC press, 2010.

CHAPTER 5. SECTIONING METHOD FOR AUTOMATIC PATH PLANNING: ROBOTIC GRINDING FOR HIGH VARIETY, LOW VOLUME PRODUCTION

Daniel W. Schimpf, Matthew C. Frank, Frank E. Peters

Department of Industrial and Manufacturing Systems Engineering, Iowa State University

Modified from a manuscript to be submitted to the *Journal of Manufacturing Science and Engineering*

Abstract

Most casting processes are near net shape, meaning that the initial component shape and size are close to the finished product, but post-processing is necessary. This includes the removal of parts of the gating system or abnormalities. Most of the gating system is usually cut or broken off, but often around 6 mm (0.25 in) remain. A common method of removal is manual hand grinding, which is arduous work and has been connected to some health issues. These issues have only been made worse by recent labor shortages, and the need for automation of the manual grinding process is only increasing. In low variety, high volume environments, automated solutions are already deployed because the programming and fixture cost can be distributed over a large number of castings. Since this is not the case for high variety, low volume foundries, they need a more flexible solution. This paper presents a sectioning method for automatic path planning, which uses drawn boundaries to determine areas that need to be ground and thus enabling high flexibility. By utilizing ROS and MoveIt, the method is highly transferable to a variety of robots and 3D sensors and so making it accessible to many foundries. The method was tested on three test surfaces, showing that it is able to remove excess material based on drawn boundaries automatically.

5.1 Statement of Authorship

This chapter was authored by Daniel W. Schimpf, Matt C. Frank, and Frank E. Peters. Daniel W. Schimpf conceptualized, reviewed the literature, designed, implemented, and tested the segmentation method under the supervision of Matt C. Frank and Frank E. Peters. Finally, the manuscript was drafted by Daniel W. Schimpf and revised by Daniel W. Schimpf, Matt C. Frank, and Frank E. Peters.

5.2 Introduction

Grinding is an important part of the casting process. After the metal solidifies, the solidified geometry is not the final desired geometry but includes additional geometry that is necessary for the casting process, like gating systems. Usually, most of the material is removed by cutting off the gating system at around 6 mm (0.25 in) from the surface (Figure 5-1). The rest of the material is commonly removed by grinding. While fully automated grinding and finishing cells are common for cast iron and aluminum castings, the steel casting industry has a wide variety of short-run production parts.

Figure 5-1 Examples of castings with excess material at the riser contacts on both flat surfaces **(a) & (b)** and on a curved surface **(c)**

This grinding is commonly done by an operator with a handheld grinder or a mounted grinder, which is guided by the operator like a swing frame grinder. For small castings, stationary grinders are used, and the casting is pushed against the grinder by the operator (Figure 5-2). Commercial products exist to handle grinding very large castings. Some have grinders on industrial robots, which are controlled by the operator from outside the cell [1]. All of these approaches make use of the operator's skill and flexibility to determine where and how much material needs to be removed, which changes from casting to casting. The changes are caused by variation of the casting itself (the parting line may be more or less pronounced, abnormalities may be present or not), or variation in the previous process, which removes most of the gating system (e.g., torch, water jet, arc air, knock-off hammer). The short lot size also does not support dedicated fixturing; hence the orientation of the casting is not known a priori.

Figure 5-2 Examples of manual grinding in foundries: **(a)** Hand grinder, **(b)** Stationary grinder

The grinding work is very strenuous, especially if a hand grinder is used. Workers bend over and move around the casting to remove the excess material while keeping high pressure on the grinder. An industry survey and report [2] indicated that hand grinding is the job in steel foundries with the most ergonomic issues, and there were limited solutions. The report identified multiple possible solutions, such as avoiding unnecessary grinding tasks and the use of swing grinders to assist the operator with the force application. Two other studies modeled hand grinding tasks to evaluate the ergonomics, and both identified the working height as an important factor in improving ergonomics [3][4]. Further grinding was the second-highest cause for workplace injury at foundries with 16%, immediately following melting, which caused a little over 16% of workplace injuries. These safety incidents come with a high cost (on average, $40,000 in workmen's compensation) [5].

Besides the safety-related issue, there are additional economic justifications for an automated grinding process. The steel foundry industry in the United States has an ongoing problem of high labor turnover and low retention rates. The reasons for this are the low unemployment rate in recent years, the arduous nature of the work, and the greater danger than in other jobs. The number of available workers has been restricting the successful operation of foundries [6]. In 2002 china surpassed the USA as the largest producer of foundry products [6], and by 2012, China had four times the market share in comparison to the USA [7]. Reasons for the foreign production of castings, for example for steel valve castings, were the same quality as American casting at consistently lower prices. By 2017 roughly 40% of the foundry industry's production was lost to imports [8]. To address the challenges of increasing price pressure and workforce issues, the demand for automation in the steel casting industry is increasing. For foundries with longer production parts, as in the cast iron and aluminum castings, the implementation of automation is more feasible due to economies of scale. For smaller steel casting companies, especially ones that are situated in a high variety, low volume environment, automation solutions are often more expensive than manual labor costs.

The material removal process on castings can be characterized by the following unknowns:

- The orientation of the part
- The locations where material needs to be removed
- The desired surface after the grinding
- The volume of material that needs to be removed

These unknowns are part of the reason why the material removal process on castings is challenging to automate. There are few isolated examples of foundries implementing robotic grinding for high variety, low volume productions. These foundries have utilized parametric programming for castings of the same product families. These product families had very similar shapes but different dimensions. This made it easier to amortize the programming costs across multiple castings within a product family. In addition, because the amount of material that needed to be removed was unknown, significant air cutting took place, increasing the cycle times. The foundries did not require the robot to be as economical as a human because of the safety and labor advantages.

However, there are characteristics of material removal, which are advantageous for automation. The tolerances that the grinding operation needs to meet are the same as that of the casting, which is a much different operation than precision grinding used on machined components.

The final results achieved will be affected by the robot control strategy. Simple motion control only controls the position and velocity of the robot. This can cause issues if the robot has to be in contact with the surface because the knowledge about the environment is likely not perfect. In this situation of constrained motion, a simple control to only position and velocity may lead to the following errors. High contact forces could lead to a deviation from the planned trajectory as well as emergency stops of the robot or damage on the tool or the surface. Compliant behavior is able to help avoid these issues. This can either be achieved passively by adding a compliant mechanical component between tool and robot or actively by utilizing direct force control.[9]

A study [10] stated that passive compliance has some advantages over active compliance since it guarantees overall stability, is relatively inexpensive, and has a fast response rate. The extent of the last two advantages has likely reduced in the last two decades with decreased sensor prices and faster computers. A hybrid position/force control of a robot enables the application of a constant force while tracking a surface, for example, in polishing applications [11]. A grinding application that utilizes hybrid position/force control as well as impedance joint control to create a compliant wrist was able to produce low impact forces and acceptable performance for force and position tracking [12]. Other robotic surface finishing operations also use force control methods [13]–[15].

Previous work developed a prototype system that included a gantry with a small rotary tool, where an operator identified a work area by moving a probe with a joystick and selected the desired geometry [16]. This method was restricted in the orientation and geometry of the workpiece as well as size, something that the method introduced in this paper will address. The goal of the proposed method is to create a material removal process that will work with a variety of industrial robots, casting shapes, and orientations. The process will not be fully automatic because it will rely on a human operator to mark areas on the casting surface for removal. Automated identification of the areas to be ground is not practical, given the low volume of parts and the unknowns that exist on each casting that were discussed above.

5.3 Methods

The long-term goal is to make grinding automation feasible for high variety, low volume casting producers. The method will remove material in areas specified by the user and blend the surface to the surrounding surface. These areas will often be where the gating and risering system was connected to the casting and has been since cut off, often within 6 mm (0.25 in) of the surface. Since it is mostly a task for aesthetics, the precision requirements are relatively low. Figure 5-3 provides an overview of the step-by-step process of scanning, planning, and automated grinding process. After the risering and gating were cut off, an operator is required to draw boundaries around areas for removal. The intent is that this would be done with markers or paint. A second outer boundary will also be marked on the casting and identifies the area whose surface will be used to determine the desired geometry in the area to be ground. Once the areas are marked, the casting can be moved to a robotic cell, where a 3D scanner with an integrated color sensor will scan the casting and detect the markings. The marked boundaries will be used to segment the point cloud into desired and undesired geometry, as well as estimating the desired geometry underneath the undesired geometry. After the desired surface is determined, a path plan for the robot is generated. Utilizing position and velocity control, this path plan is executed. The grinder will remove the excess material layer by layer, while sensors will provide feedback for the position and velocity control. Once the grinding is done, the robot will move away, and the part will be finished. The following section will provide an overview of the setup and methods.

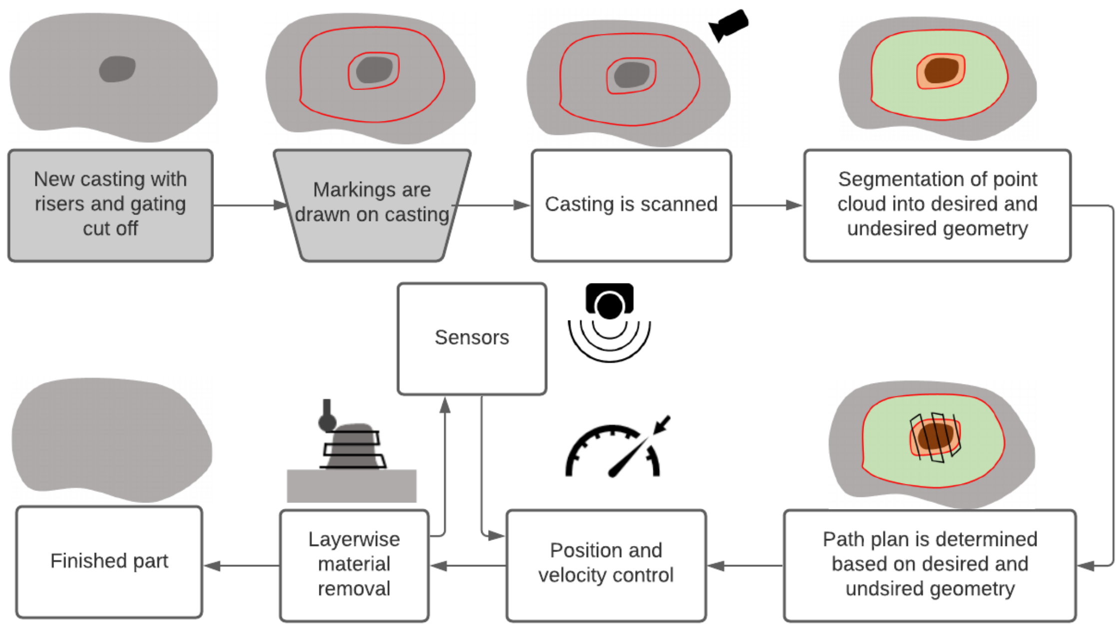

Figure 5-3: Overview of the material removal process. Grey boxes are manual operations, and white are automatic.

5.3.1 Setup

The setup is separated into the physical and virtual setup. Both will be introduced in the next two chapters.

5.3.1.1 Physical Setup

In the robotic cell, a collaborative robot (Universal Robot 10e) was used during the early development stage. This robot has a payload of 10 kg and also a force sensor integrated into the flange. While this payload is too small for an industrial application, it is sufficient to demonstrate feasibility and robust enough to test the methodology. The robot is mounted in a cell to protect the surroundings and workers from grinding debris. A variable speed angle grinder is mounted to the robot flange; the grinder was converted to accommodate a 101 x 25 mm (4 x 1 in) grinding wheel and is activated by the controller via a relay during operation. There is a need for a sensor that can collect a depth map as well as color information across that map. For initial testing, an Intel Realsense 435 was attached to the robot arm (Figure 5-4).

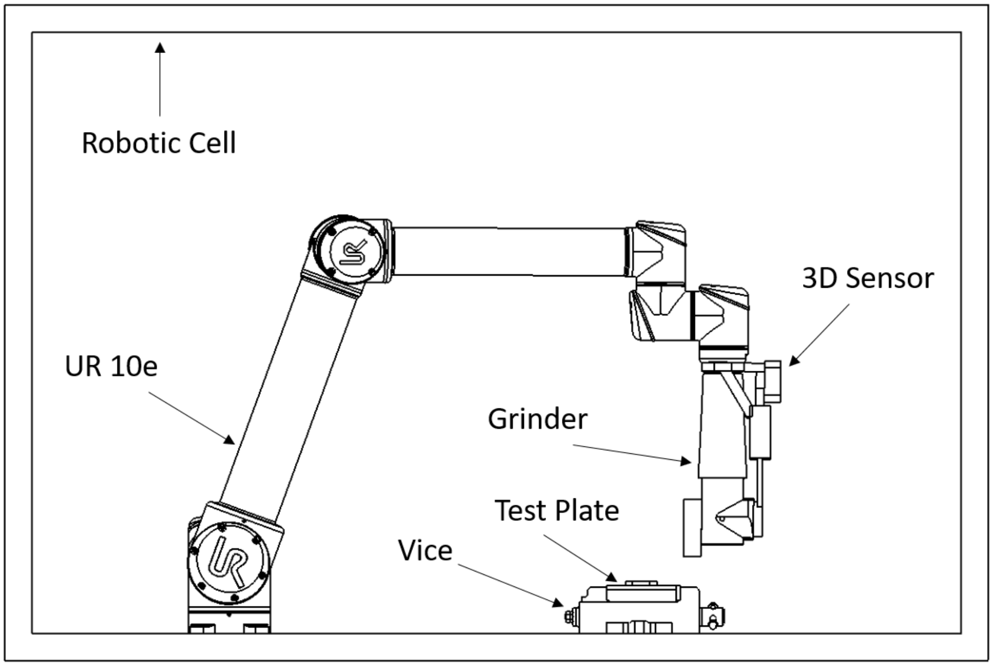

Figure 5-4 UR 10e with the mounted angle grinder and 3D sensor

A passive compliance module will be developed for the end of arm tooling to attach the grinding wheel, with a goal to protect from errors caused by inaccurate path plans, unreliable data, and/or unexpected anomalies outside of the scanning range. This compliance module will cause a contraction along one axis if the forces on the grinder exceed a threshold (Figure 5-5). The robot's path plan is created based on information from the 3D sensor, which can be flawed. In addition to the sensor's inaccuracy, further inaccuracies or missing information can be caused by reflective surfaces or the influence of external light. The compliance device will reduce the forces exerted on the grinding wheel and protect it from causing emergency stops or even shattering of the wheel. Because of the incremental removal of material, the compliance compression stroke needs to account for both the grinding depth and the inaccuracies of the 3D sensor. The compliance module also contains a position sensor that will record the contraction of the compliance module.

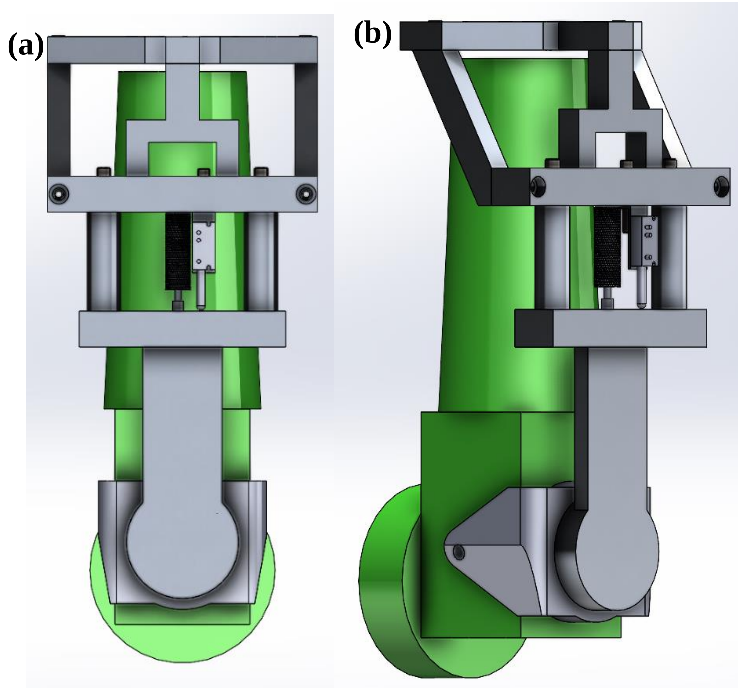

Figure 5-5: Passive compliance mount (grey) for the grinder (green). The movement of contraction is up.

The castings with excess material will be placed in the robotic cell. While the orientation can be chosen freely as long as the robot can access the area, it is imperative that the part does not move once the 3D scan has been captured. Heavy parts may be placed in the cell without any workholding, but lighter parts likely need to be held in place to avoid movement during the grinding operation. The need for the workholding only to secure the casting and not accurately locate it is a significant advantage of this approach.

5.3.1.2 Virtual Setup: Robot-Operating-System (ROS) & MoveIt

The proposed method uses the Robot-Operating-System (ROS)[17] and the MoveIt motion planning framework [18]. ROS is an open-source system that was developed for code reuse across robots of different manufacturers. The code-reuse will allow our method to be used independently of the hardware used and thus make it easier to transfer it to different setups. MoveIt is an open-source framework for motion planning with integrated collision detection and will enable the determination of whether a collision occurs for a given robot in a given position and environment.

Within ROS and MoveIt, a virtual setup of our robot and its environment was created. The virtual setup consists of the four walls, ground & ceiling, as well as 3D models of the robot, 3D sensor, and grinder. For the robot, grinder, and 3D sensor, both a visual and a collision mesh exist. The collision mesh is a simplified version of the visual mesh and reduces the number of facets.

5.3.2 Segmentation and Removal Method

The ultimate goal is for the automation system to identify what needs to be ground and not rely on an operator. However, in the high variety, low volume environment characteristic of the steel casting industry, this is not feasible in at least the short or medium term. This is especially not reasonable as a first step because it would have a high error rate resulting in removing material where it is not necessary or not removing enough material. The most challenging component in this totally automated system was assessed to be the decision-maker, which decides in which areas material needs to be removed. This component is also one that does not require much time and is not physically demanding. Given the combination of the difficulty to automate, the risk of error, and the low time requirement, it was decided to rely on the operator. The operator could communicate this decision in multiple ways. It was considered tasking the operator with marking the areas where material needs to be removed on a 3D scan of the casting but has a couple of disadvantages: computation centers must be accessible, finding abnormalities solely in the 3D scan is often hard so the information would have to be transferred from the real casting to the 3D scan of the casting and the marking process is likely more time-consuming on the 3D scan. Having the operator communicate with the system by just making marks on the casting is a low-cost method that can be made quickly, anywhere in the foundry.

Section 5.3.2.1 will explain that two boundaries are used to enable the segmentation of scan data. It will cover the test plates used to evaluate the method and discuss the steps required to segment the scan and determine the desired surface under the excess material. Section 5.2.3.2 will present how multiple parallel surfaces are created to enable layerwise removal of the material**.** This information is then used to determine a path plan for the robot to remove the excess material. Section 5.3.2.3 will present the execution of the path plan and how collisions are avoided.

5.3.2.1 Segmentation of Point Cloud

This chapter will introduce steps 2 through 4 from Figure 5-3. It will introduce what markings are necessary to enable automatic segmentation, how the segmentation is performed and how the desired surface underneath the excess material is determined.

To enable the automatic removal of excess material, the surface is segmented based on boundaries marked by operators. Figure 5-6 shows an example of how these boundaries can be used to segment surfaces. The example shows two boundaries, an inner and outer boundary. Everything within the inner boundary will be considered abnormal (orange and green area in Figure 5-6), and everything outside of the inner boundary but within the outer boundary will be considered the desired surface (blue area in Figure 5-6). Thus the interference geometry will not be used to interpolate the surface underneath the riser contact. If the example surface is a cylindrical surface with a rectangular extrusion (interference geometry) on this surface, then including the rectangle in the interpolation surface would cause the skewing of the following interpolation because both the extruded rectangle and the cylindrical surface would be used for the interpolation.

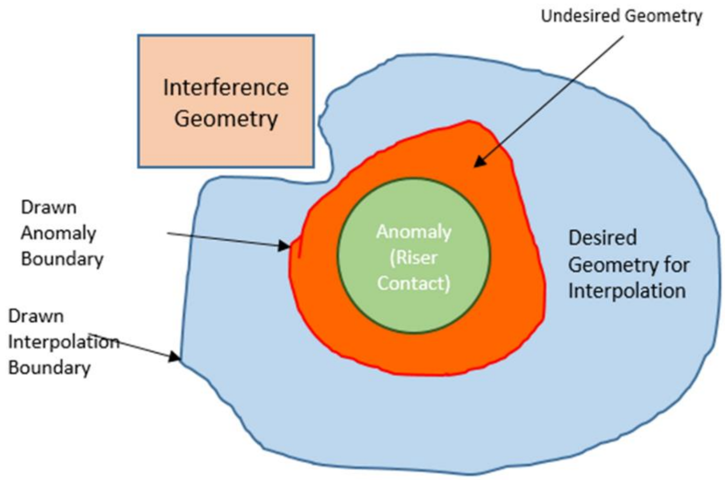

Figure 5-6 Anomaly and interpolation boundary

Figure 5-7 **(a)** displays an example of what this could look like on a casting. The figure shows a cylindrical casting with a flat surface on the top. On that flat surface, a couple of riser contacts remain that need to be removed. To remove one of these two, boundaries were drawn around it. The boundaries mark the abnormal area, which includes the riser entirely, and the desired surface, which is the flat surrounding surface. A second example is a casting with two boundaries drawn in red **(b)**. Like in **(a)**, the inner boundary marks the excess material on the casting, while the outer boundary marks the interpolation surface.

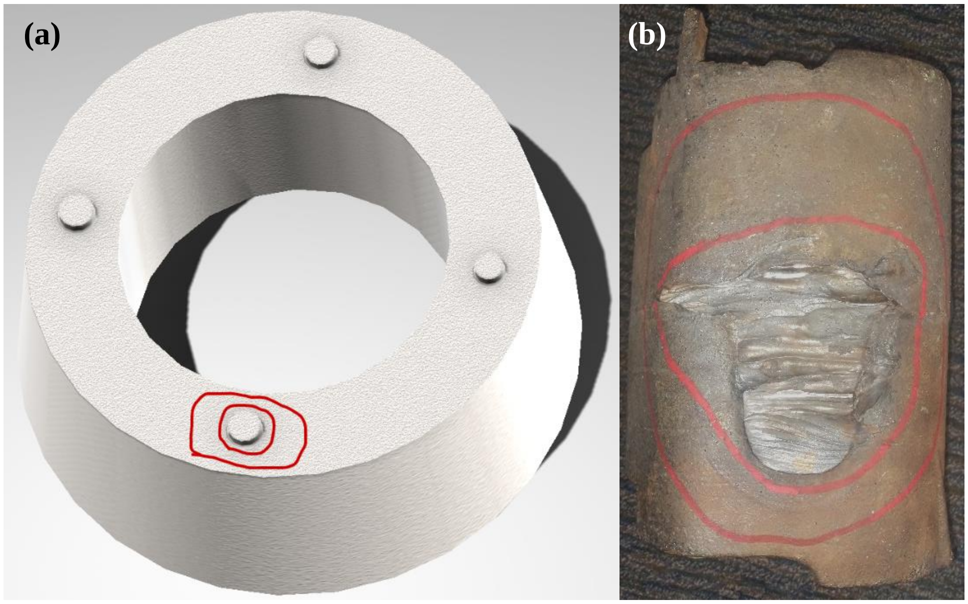

Figure 5-7: **(a)** Example of casting with risers that need to be removed on a flat surface. Around one riser, red boundaries are drawn to enable the segmentation of the surface. **(b)** Example of casting with excess material and drawn boundaries.

To test the method, three test plates (Figure 5-8) were designed and created to represent potential casting surfaces. The test plates include simple surface geometries: flat, cylindrical, and spherical. Each of the test plates has undesired geometry, an extruded 'I' on the flat surface, an extruded rectangle with a circle in the middle on the cylindrical surface, and freeform geometry on the spherical surface. The test plates are made of plastic but can nonetheless be used to evaluate the success of the method. To enable switching to harder materials, a stronger robot and grinder may be necessary.

In Figure 5-8, one can see the test surfaces, with boundaries and captured by the 3D sensor. For the flat and cylindrical interpolation surface, the boundary was drawn at the edge of the plates, and the anomaly boundary was drawn closely around the undesired geometry. In the case of the round surface, the interpolation boundary was drawn at the intersection of the flat and round surface. If the flat surface was included within the interpolation boundary, the system would use this information to determine, erroneously, what the final desired surface was.

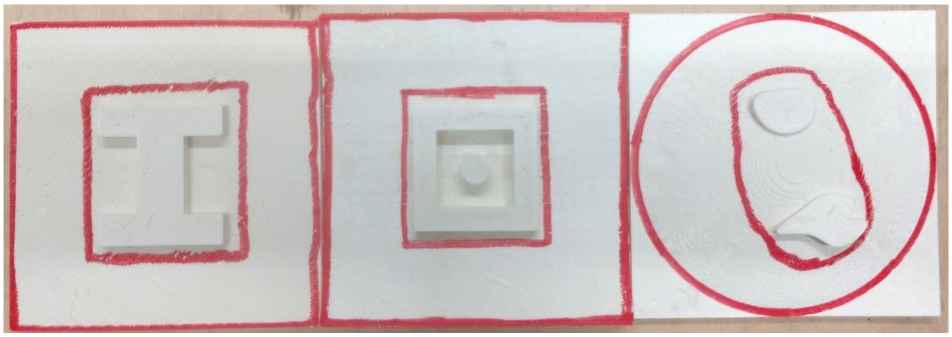

Figure 5-8: Three test surfaces with markings applied: spherical with freeform abnormalities (left), cylindrical with square abnormality (middle), and flat with 'I'-shaped abnormality (right)

When the application is started, the robot will move into a previously specified safe position to take multiple 3D scans of the object. These multiple 3D scans will be merged into a single point cloud. The advantage of combining multiple scans is generating a denser point cloud, which can be downsampled by averaging points within a voxel, which is a volumetric cube in space. The implementation utilized the Point Cloud Library [19] for the point cloud manipulation.

Once the point cloud was recorded and downsampled, the points in the point cloud need to be segmented. To segment the point cloud, the boundaries first need to be found by thresholding the point cloud based on their color values. The point cloud contains for each point the x,y,z coordinate, and its r,g,b values. Each color red (R), green (G), and blue (B) is saved as an integer value from 0 to 255. A really strong red for example would be R = 255, G = 0 and B =

0. In this color space, it is not as easy to select specific colors while allowing different brightness. That is why the colors were converted to the HSV color space. Instead of describing each color by the intensity of each main color (red, green, and blue), the HSV color space specifies hue, saturation, and value. The hue describes the type of color (e.g., red, yellow, green, blue) and commonly ranges from 0-360, representing the 360 degrees of a color circle. Saturation and value range from 0 to 1 and represent the tint and mixture with black and white for each color. The advantage of the HSV color space for this application is that it is less dependent on the brightness differences of the colors. This is important because, in the robotic cell, the brightness may vary. Thresholding for both color spaces RGB and HSV were implemented, but HSV thresholding was more reliable.

The RGB values are converted to HSV based on the algorithm seen below (1)-(6) [20]. Often RGB values are in the range of [0-255]; if so they are divided by 255 in the first step, as shown in (1).

$$r' = \frac{R}{255} \quad g' = \frac{G}{255} \quad b' = \frac{B}{255} \tag{1}$$

$$Max = max(r', g', b'); \;\; Min = min(r''', g', b'); \;\; delta = Max - Min \tag{2}$$

$$V = Max \tag{3}$$

$$S = \begin{cases} 0, & Max = 0 \\ \dfrac{delta}{Max}, & Max \neq 0 \end{cases} \tag{4}$$

$$h' = \begin{cases} undefined, & S = 0 \\ 60 \times \dfrac{g' - b'}{delta}, & r' = Max \\ 60 \times \left(\dfrac{b' - r'}{delta} + 2\right), & g' = Max \\ 60 \times \left(\dfrac{r' - g'}{delta} + 4\right), & b' = Max \end{cases} \tag{5}$$

$$H = \begin{cases} h' + 360, & h' < 0 \\ h', & h' \geq 0 \end{cases} \tag{6}$$

Based on the color space, the appropriate thresholds have to be chosen. Since the environment will vary, the user will need to determine the best threshold values for the specific application.

In Figure 5-9, one can see the original point cloud before thresholding **(a)** and the point cloud after thresholding **(b)**, with the found boundaries marked in red. One can see that the boundaries are not perfect, but small gaps between sections of the boundary lines are acceptable.

In the next step, the found boundary points are grouped based on the vicinity. In general, the vicinity threshold has to be smaller than the minimum distance between the outer and inner boundaries; otherwise, the outer and inner boundary may be grouped as a single boundary. Also, the vicinity threshold has to be bigger than the gaps in the detected boundary; else, a single boundary may be grouped as multiple boundaries and not a single one. Once there are groups of points, each group representing a specific boundary, the groups corresponding to inner and outer boundaries have to be paired up. This is done by checking if any boundary is contained within another boundary, which means that all points of the inner boundary are within the outer boundary. These paired groups are then used to determine which areas are abnormal and which are good. For each group of points, the points are sorted, and boundary polygons are created. The

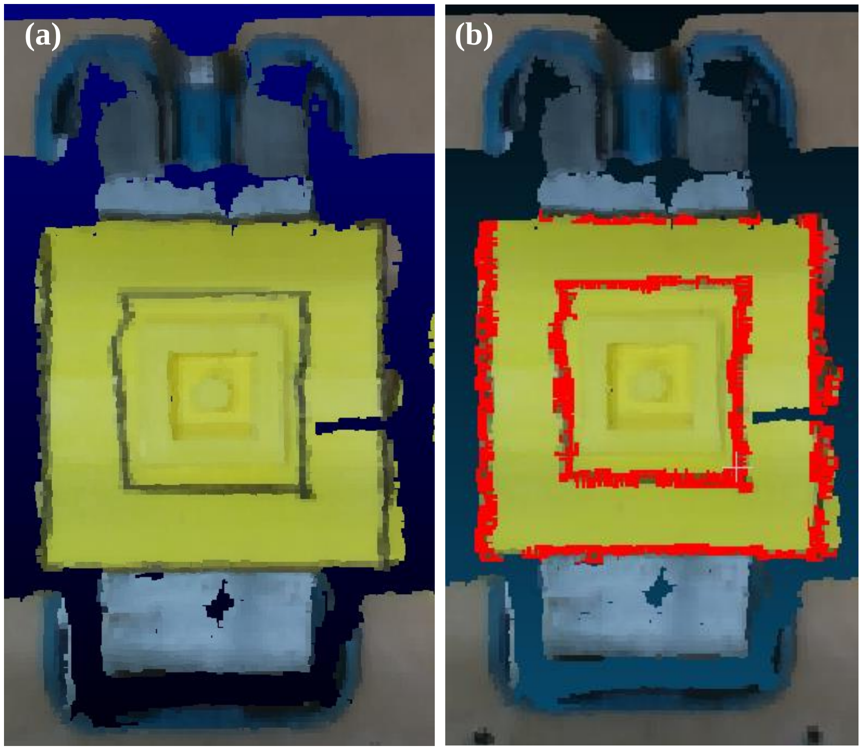

Figure 5-9: Point cloud recorded **(a)** and detected boundaries in red **(b)**; the black stripe on the right of the yellow surface is an area where no points were returned from the 3D sensor.

polygons can then be used to evaluate if points are lying within or outside of them. This is marking all points in the scene that are within the inner boundary as abnormal. In the next step, all points within the outer boundary, but not within the inner boundary, are determined to be part of the good surface, which is used for the interpolation. In Figure 5-10 **(a)**, one can see the points that were determined to be part of the boundaries. These points were then grouped into two groups: the outer and inner boundary. Figure 5-10 **(b)** shows the segmentation of the point cloud. The blue points were found to be the abnormal surface to be ground, and the green points comprise the good surface used for the interpolation.

Following the point cloud segmentation, the desired surface points are matched with three geometric models to determine the best path plan strategy, such as grinding direction. The three geometric models are flat, cylindrical, and spherical. For each model, the coefficients are returned as well as the number of points that are considered inliers. These inliers are points that are within a distance threshold to the model. The model with the most inliers is considered the best match.

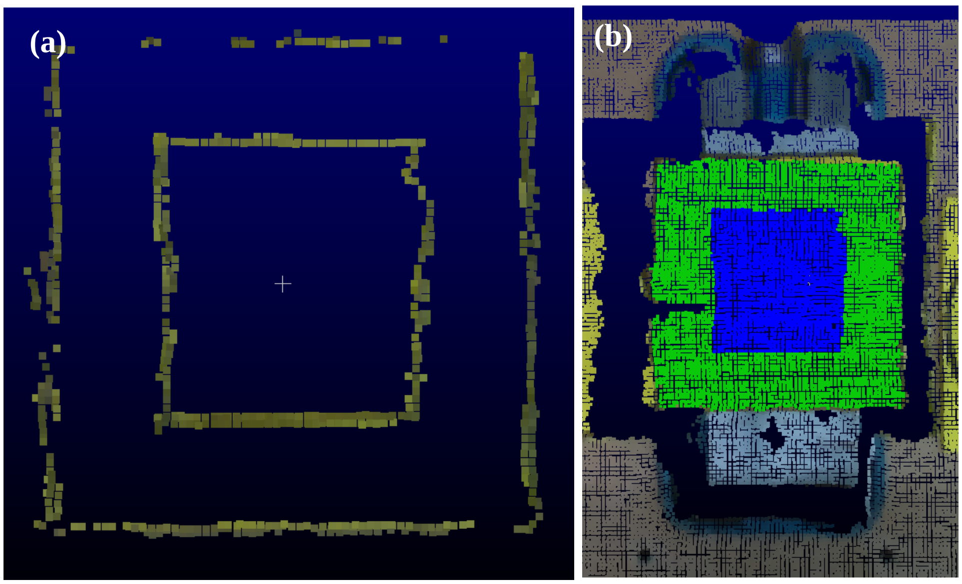

Figure 5-10: Determined boundaries **(a)** and Segmented point cloud: blue (undesired geometry) and green (interpolation surface) **(b)**

Next, the points determined to be part of the good surface can be used to interpolate the desired surface. During the interpolation, the points are smoothed with moving least squares, holes are filled, and then the point cloud is smoothed again. The hole filling is assisted by the knowledge of the best fitting model. For the cylindrical surface, instead of filling the hole based on the surrounding points, it is filled by considering neighbors along multiple crosssections parallel to the cylinder axis.

Figure 5-11 **(a)** is the segmented point cloud, which contains the points within the outer but not within the inner boundary. This point cloud is then smoothed and interpolated through the moving least squares method to produce the desired surface and will be the basis for our path planning (Figure 5-11**(b)**).

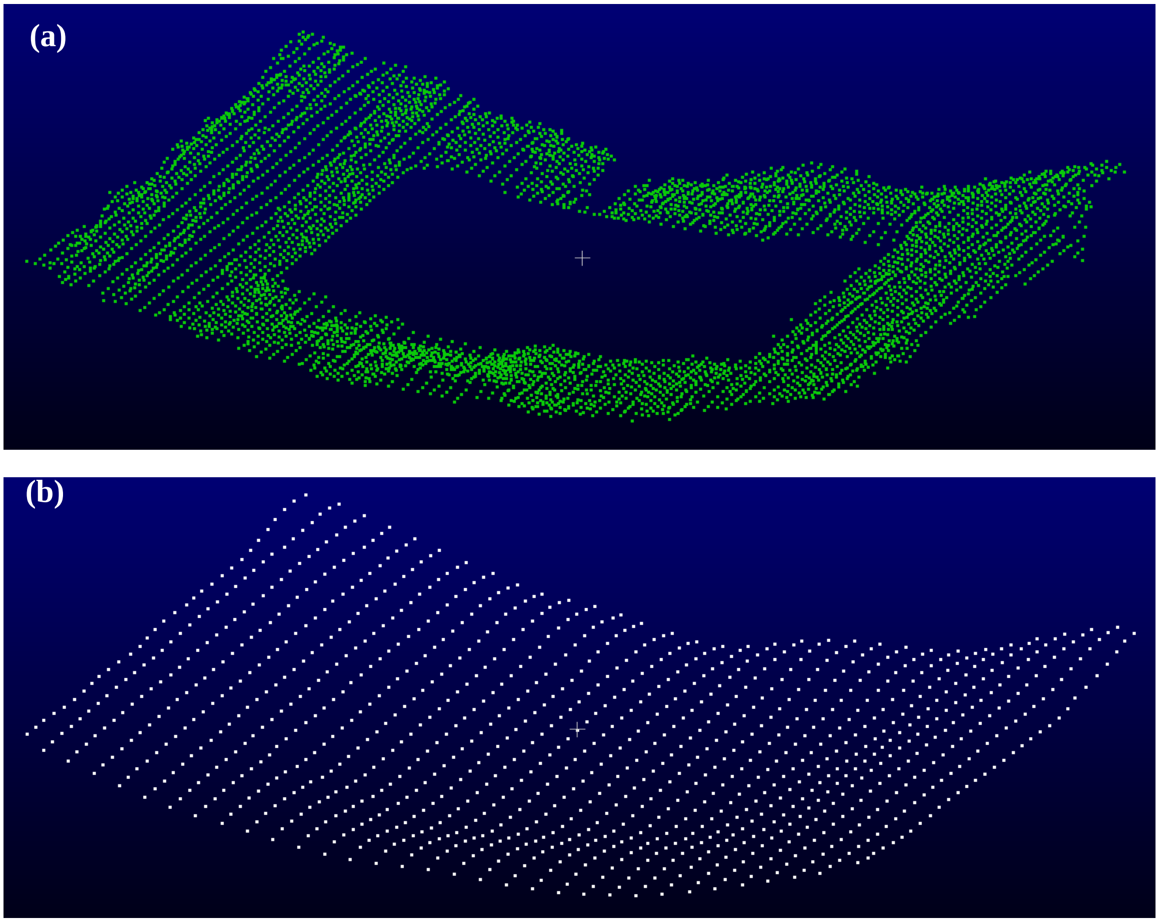

Figure 5-11: The desired surface based on the segmented point cloud **(a)** is smoothed and interpolated to determine the desired geometry around and below the excess material **(b)**

Figure 5-12 presents a summarizing overview of this section. An operator has to mark two boundaries on a casting **(a)**, an inner that defines the abnormal area and an outer which determines the interpolation boundary. These boundaries were drawn on three test plates **(b)**. The test plates are then scanned, color thresholded, and segmented **(c)**. The segmented point cloud is then used to estimate the desired surface under the excess material based on the good area that surrounds the abnormal area **(d)**.

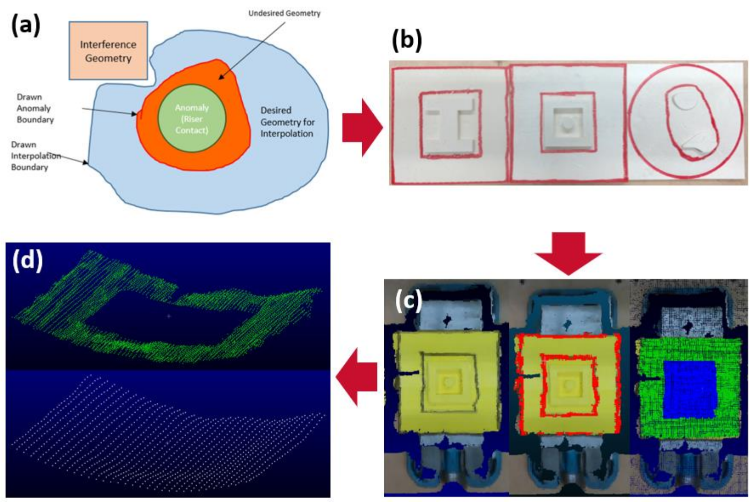

Figure 5-12: *Overview of the segmentation*: **(a)** boundaries, **(b)** test plates, **(c)** segmentation of point cloud, **(d)** determination of the desired surface.

Algorithm 1 summarizes the steps taken to segment the point cloud, starting with the color thresholding, followed by segmentation of the point cloud into desired and undesired geometry, and finishing with interpolating the desired surface to estimate the desired surface below the excess material.

Algorithm 1 Point Cloud Segmentation

```
procedure Analyze Point Cloud
Gather and combine 3D scans
Color thresholding to extract markings
Extract clusters
Sort points to create boundary polygon
Find inner and outer boundary pairs
Segment original point cloud based on polygon pairs (leftover, desired-surface &
abnormality point cloud)
Identify the best matching geometric model (plane, cylinder, sphere) for the desired-
surface and extract model coefficients
Smooth desired-surface point cloud with moving least squares
Fill holes based on surrounding points averages
Smooth filled in point cloud with moving least squares
Report leftover, desired-surface & abnormal point cloud as well as model coefficients
end procedure
```

5.3.2.2 Determination of Path Plan

This chapter will present how a multi-level path plan is created based on the information gathered in the previous chapter. The desired geometry and excess material are used to create multiple parallel surfaces to remove the excess material layer by layer. Once these parallel surfaces points are sampled along intersecting planes and approach and retreat positions are added to create a path plan for the robot.

Now that the desired surface is established, the normal vector for each point in the point cloud is calculated. The normal vectors will be used to determine the pose of the robot and enable users to create different desired surface levels for our path planning. The normal vectors are calculated for each point by estimating a plane from the points in the proximity, and the normal vector of that plane is defined as the normal vector of the point. Next, offset surfaces based on the grinding depth can be created. These offset surfaces are needed because the excess material will not be removed in a single pass but incrementally. Figure 5-13 shows the grinding wheel during the grinding process and the incremental removal of material. It shows that to remove the excess material (from the original surface to the final layer), multiple layers are removed incrementally.

The offset surfaces are created by iterating through the point cloud, and each point is offset by the product of the normalized normal vector and the chosen grinding depth. Figure 5-14 **(a)** shows the abnormal surface. The color spectrum is based on the height of the points and the desired surface in white. Figure 5-14 **(b)** shows, in addition to the abnormal and desired surface, multiple offset desired surfaces. These offset desired surfaces were created as described above and are needed since all the excess material is not removed in one pass; instead, it is removed incrementally, one layer after the other.

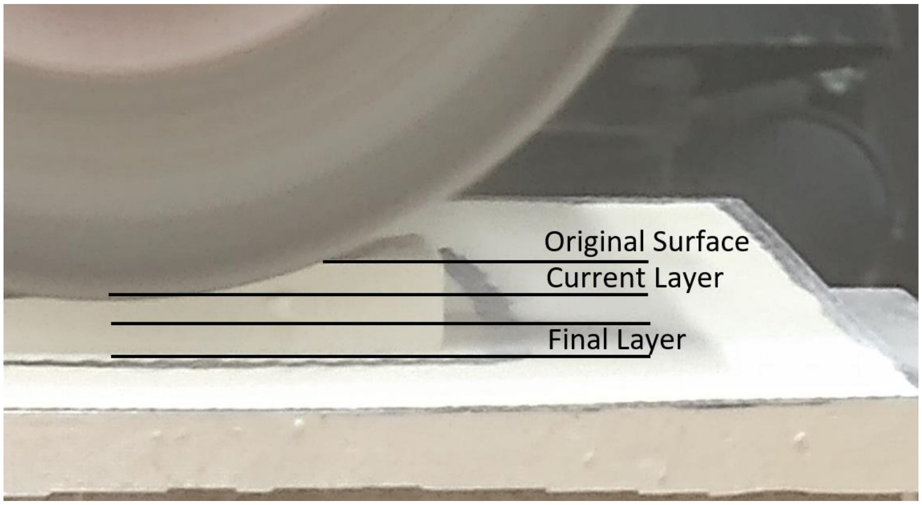

Figure 5-13: Incremental/Layerwise removal of excess material.

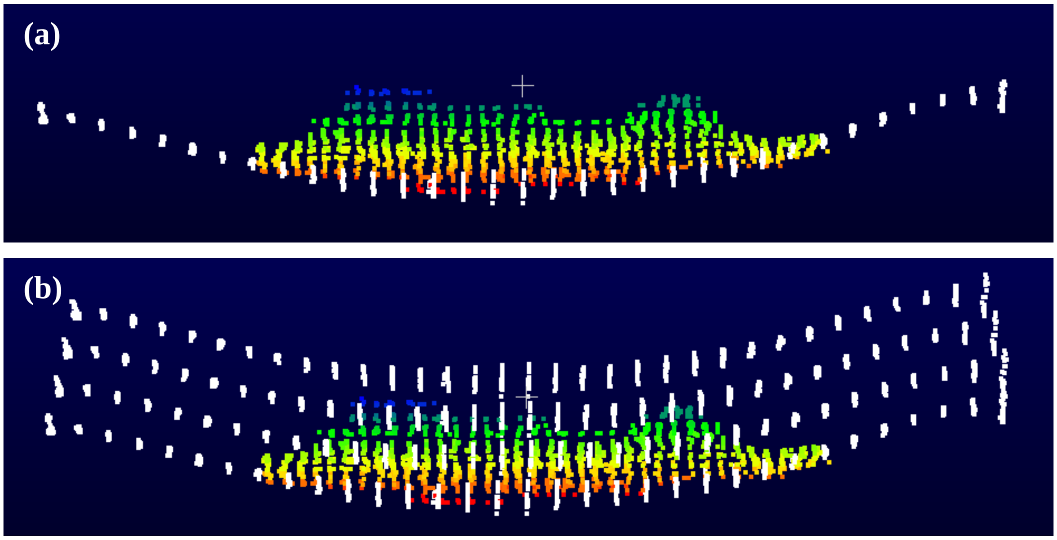

Figure 5-14: Side view of an abnormal surface (rainbow-colored) with the desired surface (white) **(a).** To remove the excess material, grinding on multiple layers is necessary. **(b)** displays multiple desired offset surfaces (white) with the desired (white) and abnormal surface (rainbow-colored)

The number of desired offset surfaces should not be chosen arbitrarily, but for an efficient operation, it should be determined based on the height of the abnormal surface. This can be done by creating an offset surface and checking if abnormal points are above. If so, a new offset surface is created. This continues until no abnormal points are above the offset surface.

The offset surface before that final offset surface, with no points above, will be the first layer for the incremental removal of material. For a given point of the abnormal point cloud, the closest surface patch of the offset desired surface is found. Utilizing the direction of the normal vector from the surface patch to the abnormal point, it can be determined if the abnormal point is above or below the surface patch. Figure 5-15 shows an example of points that were determined to be above and below for a given layer.

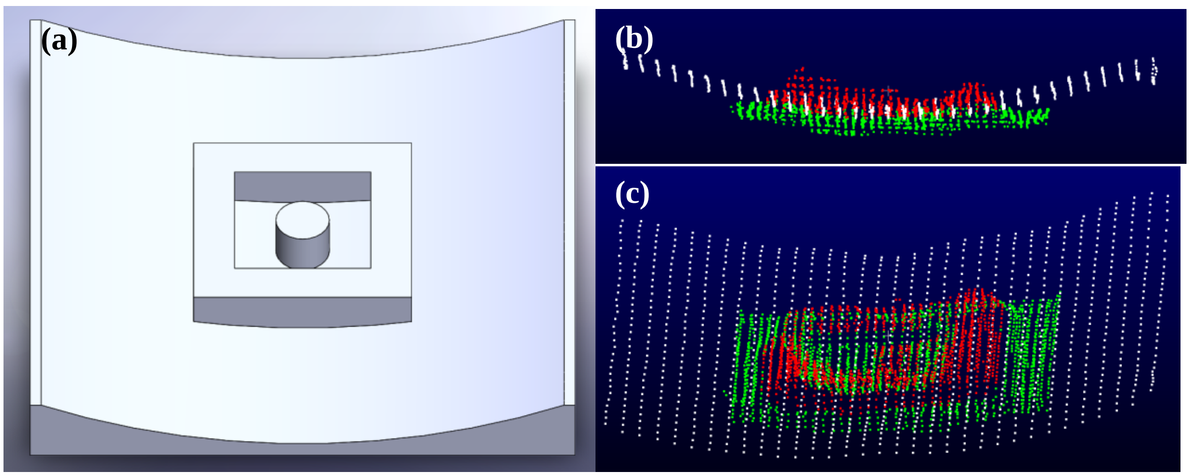

Figure 5-15: Curved test plate **(a),** desired surface layer (white), and excess material (green and red) **(b)** & **(c).** Excess material points above desired surface layer are colored red. That excess material needs to be considered in the removal strategy to create that surface layer.

The information on what points are above the offset surface is not only used to determine the number of grinding layers necessary but also to determine the section of the offset surface where grinding will be necessary. This is done by utilizing the location of all points that are above the offset surface and determining the convex hull that encompasses all these points. Next, the edges of the encompassing convex hull are moved the approach distance *A* away to make sure grinding happens only during forward feed motion and not during down movements.

Equation (1) (based on [21]) shows how the approach distance *A* is calculated *d* is the depth of cut, and *D* is the diameter of the grinding wheel.

$$A = \sqrt{d(D - d)} \tag{7}$$

Overall, air cutting can be avoided by both reducing the size of each surface level and determining the correct number of surface levels.

The path plan is created by sampling points on the surface of each surface layer. The type of sampling depends on the type of surface, which was determined in a previous step. If the desired surface type is flat, sampling on parallel planes is performed. Similarly, if the desired surface type is cylindrical, the sampling is performed on parallel planes, but these planes are perpendicular to the axis of the cylinder. Finally, if the desired surface type is spherical, a radial toolpath is created. The number of radial toolpaths is dependent on the crosssection of the abnormal area to ensure coverage of the whole area.

The path plan for the robot is a list of x,y,z point that the tool center point (TCP) of the robot has to follow, orientations that restrict the grinding wheel to be normal to the surface, and binary information determining if it is rapid move or not. To determine the orientation for a specific point, the normal vector of that point was used, which restricts the movement around the first two axes, and the machining direction sets the rotation around the third axis. There are multiple different ways to describe the orientation in 3D. In this application, the rotation about the three axes is used to determine a quaternion for each point.

The path plan starts with the highest layer. At the starting and endpoint of that layer, a starting and end position was added above the surface. These are the points the robot can traverse to at rapid speeds. At the end of every straight path, when the robot moves over to the next straight path, points were added above the endpoints to make the robot move up, over to the next path, and back down into the material. Once one layer of the surface is done, the robot will move to the end position above the surface and then move to the start position of the next lower level, which is also above the surface. An example of such a trajectory can be seen in Figure 5-16. The green lines are the normal vectors originating at the goal points in the trajectory. Each goal point has an x,y,z coordinate, and orientation associated with them. The blue lines are the trajectories in between each goal point and are straight trajectories.

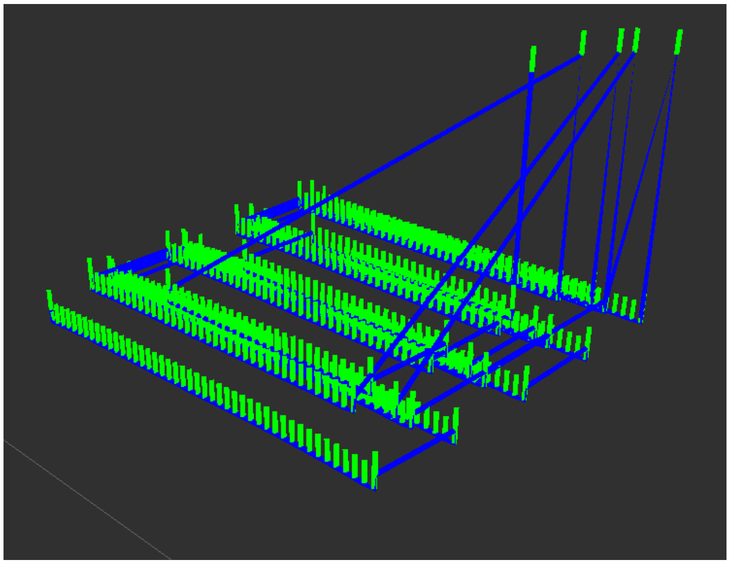

Figure 5-16: Example of a path plan for a planar surface with two layers. Green is the normal vector of that point, which determines the orientation of the robot, and blue is the path from point to point, which the TCP will follow.

Figure 5-17 presents a summarizing overview of this section. In the beginning, the interpolated surface is offset multiple times to enable incremental removal **(a)**. For each offset surface, the excess material above the surface is determined to determine how many layers are needed and in what areas material needs to be removed **(b)**. Finally, points are sampled on the surfaces, approach and retreat positions are added to create the path plan for the robot.

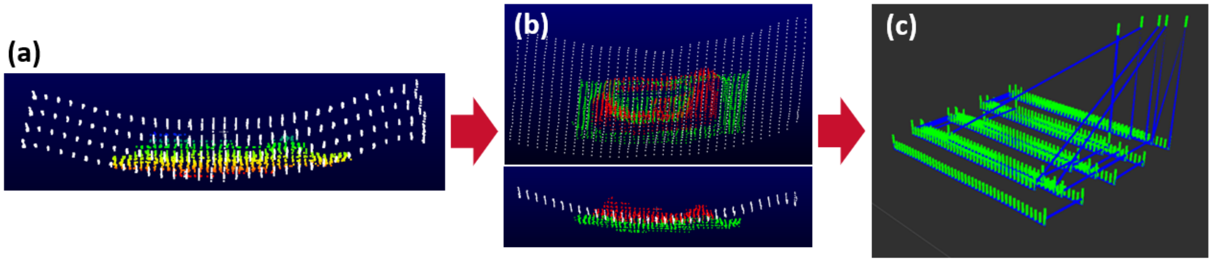

Figure 5-17: Determination of path plan: **(a)** Creation layers for incremental removal, **(b)** Determination of what points are above and below, **(c)** final path plan with approach and retreat positions

Algorithm 2 summarizes the steps taken to generate the path plan. In a loop, multiple offset surfaces are created by utilizing normal vectors until all the excess material is covered. All the excess material is covered if no abnormal points are above the offset surface. Next, the path plan is created by sampling points along lines on the different surface layers as well as adding approach and retreat positions for the robot.

Algorithm 2 Path Plan Determination

procedure Path Plan Determination
Receive *desired-surface* & *abnormality* point cloud as well as *model coefficients*
while *abnormal points* above *desired surface layer* **do**
    if 1st loop do
        Segment *desired-surface* based on *abnormality* point cloud creating the first *desired surface layer*
    end if
    Estimate normals of the *desired surface layer*
    Move *desired surface layer* stepsize length in the normal direction
    Determine if *abnormal* points are above the *desired surface layer*
end while
for number of *desired surface layers* **do**
    Estimate normals of *desired surface layer*
    Sample points along parallel lines for planar and cylindrical geometries, along radial lines for spherical and add to *path plan*
    Add approach and retreat positions to *path plan*
end for
Report *path plan*
end procedure

5.3.2.3 Execution of Path Plan

This section will present how the robotic path plan, created in the previous section, is executed. The path plan is split into multiple fast and slow trajectories, which will be sent to the robot individually. Each trajectory is evaluated for potential collisions. During the grinding operation, sensors are monitored to adjust the feed rate or repeat layers.

The robotic application starts with moving the robot in position over the part to perform a 3D scan. This 3D scan is used to segment the point cloud and determine the desired surface underneath the excess material (Chapter 5.3.2.1). This information is then used to create a multilayer path plan to remove the excess material incrementally(Chapter 5.3.2.2).

After a path plan with goal points, orientations and velocities is generated, it is transformed to the robot's coordinate system. Then it is split into multiple fast and slow trajectories. The path plan is split into smaller trajectories so that for each trajectory, a different feed rate can be set. For moves where the grinder is in contact with the surface, the velocity will be reduced, while all other movements can be faster.

The orientation of the grinder is determined by the normal vector associated with each path plan point as well as the direction of travel for each trajectory. The trajectories are sent to the MoveIt motion planner individually, which determines the angles of the robotic joints along the way and determines if it is possible to execute this path plan without colliding with any of the collision geometry. There are two different types of collision geometry. The static collision environment contains components such as the robot cell and the robot with all its attachments. The nonstatic collision environment consists of an octomap which is created from the leftover point cloud. This leftover point cloud is the data gathered by the 3D sensor with all points outside of the outer boundary. Figure 5-18 visualizes the octomap representing the nonstatic collision environment. The octomap are the cubes colored from light blue to purple.

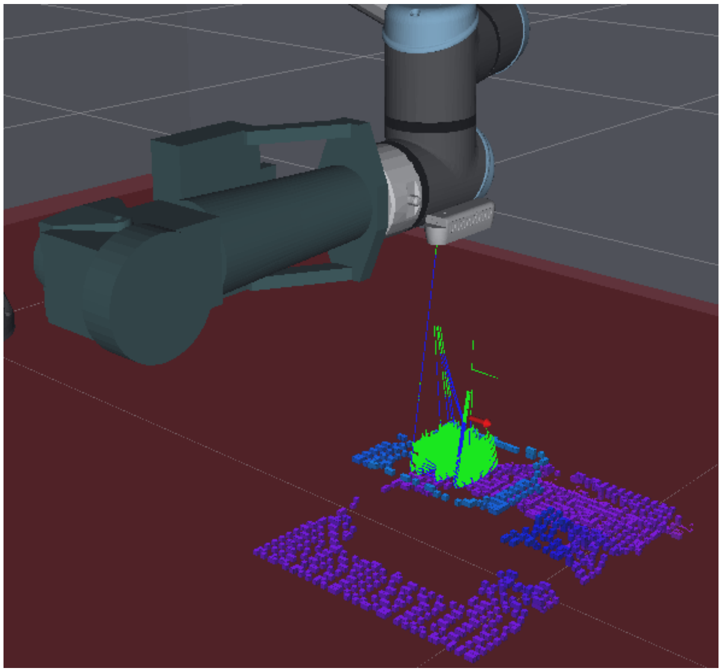

Figure 5-18: Visualized path plan with static and nonstatic collision environment.

If no solution without collision is found, the path plan is not executed. Common reasons for a path plan to fail are collisions between the robot and itself or the surrounding collision objects.

When the robot moves in position above the part before the grinder comes in contact with the surface, the grinder and durst removal system is turned on by using one of the robot controllers' I/O ports. During the execution of a trajectory where the grinder comes in contact with the surface, the robot's position error as reported by the robot, the forces gathered by the force sensor, and the compression of the compliance module as measured by the position sensor are recorded. When a surface layer is finished, the position errors, forces, and compression are compared with threshold values. If the measured values are greater than the threshold values, this would indicate that less material than required to achieve the intermediate was removed. This will cause the layer to be repeated and enable the robot to follow the desired path more closely. During the trajectory execution, the compliance module's compression is monitored and regulated via proportional control (based on [22]) of the robot's velocity. To achieve this, the speed scaling function of the robot is utilized. The speed scaling value can range from zero (the robot's velocity is zero, the robot is stopped) to one (the robot moves with set maximum velocity).

$$P_{out} = K_p(C_D - C_A) + P_{in} \tag{8}$$

P_{out} is the new speed scaling value which will be sent to the robot while P_{in} is the previous speed scaling value. C_D is the setpoint, the desired compression, while C_A is the process variable, the actual compression. K_p is the proportional gain and determines the extent of the response for a given difference between desired and actual compression. Since the speed scaling value that the robot accepts has to be between zero and one, P_{out} is also bounded to be between zero and one. If the compression is higher than desired, the regulations effect will slow the robot's motion until the compression reaches its setpoint or the limit zero is reached. The reduced speed will decrease the grinding forces causing a lower depression. Once the depression is less than desired, the speed will increase again up to its max or when the depression increases again.

During the path plan execution, a constant force during the grinding is not attempted as may be the case for many polishing or blending operations [23]. Instead, the robot follows the predetermined path plan to alter the shape of the part. The execution of the path plan may be terminated by an operator at any time, by stoping the program or pushing the emergency stop.

Algorithm 3 summarizes the steps necessary for the execution of the robotic grinding. The robot will first move into position over the part to scan the part. After the path plan is determined, it is transformed to the robot's coordinate system along with the points outside of the outer segmentation boundary. The path plan is then split into multiple smaller fast and slow trajectories. Next, the orientation of the grinder is determined, the robot moves over the part, and the grinder, as well as the dust removal system, is started. Now the execution of the trajectory starts in a loop. Within each loop, the grinder's orientation is determined, trajectory points are sent to MoveIt, and the robot path is computed. If the trajectory is collision-free, the trajectory's velocity is set as desired with Iterative Parabolic Time Parameterization. This sets appropriate goal times for each point. Next, the trajectory is executed, and the forces, position error, and contraction are recorded. Based on the contraction, the execution speed is adjusted during the execution. Once a trajectory is done, the loop moves to the next trajectory. At the beginning of each loop, if a layer was finished and the thresholds for position error and forces are exceeded, the previous layer is repeated. This means that the first trajectory of the previous layer is set as the next trajectory. Once all layers are finished, the robot moves away from the part, and the program ends.

Finally, Figure 5-19 summarizes the process flow of this robotics application. The shapes of the process flow have two different colors to differentiate between human tasks and automatic processes. One can see that the human is mostly needed at the beginning and end of the operation. At the beginning to perform the setup, including drawing markings and clamping the casting. After the casting was scanned and a strategy to remove the excess material was determined, the operator determines if the path plan is acceptable. This is a precaution to ensure correct execution. In the future, this step may be avoided when the confidence in the automatic removal strategy increases.

Furthermore, during the whole execution of the application, the user has the option to terminate it at any time. In the end, the worker needs to remove the casting from the grinding cell. Finally, the post grinding inspection is currently manually, but since the robotic cell has a 3D scanner, this may be automated in the future.

Algorithm 3 Robotic Grinding

```
procedure Robotic Grinding
Move the robot to the scanning position
Start Point Cloud Segmentation and wait for results
Start Path Plan Determination and wait for results
Transform path plan to robot coordinate system
Transform leftover point cloud to robot coordinate system and convert to octomap for
nonstatic collision avoidance
Split path plan into many smaller fast and slow trajectories
Determine grinder orientation for the first movement
Move the robot to position over part
Turn on grinder and dust removal system
for number of trajectories do
        if trajectory is part of a new layer do
                Check if forces and position error from the previous layer are
acceptable
                if forces, position error or contraction above threshold do
                        Repeat previous layer
                end if
        end if
        Determine orientation of grinder
        Add points trajectory points with orientations to robot trajectory
        Compute & test trajectory
        if trajectory is collision-free do
                Use Iterative Parabolic Time Parameterization to adjust the speed for
                slow and fast movements
                Execute robot path
                while trajectory is executing do
                        Record max position error, forces & contraction
                        Adjust robot velocity based on contraction value
                end while
        end if
end for
Turn off grinder and dust removal system
end procedure
```

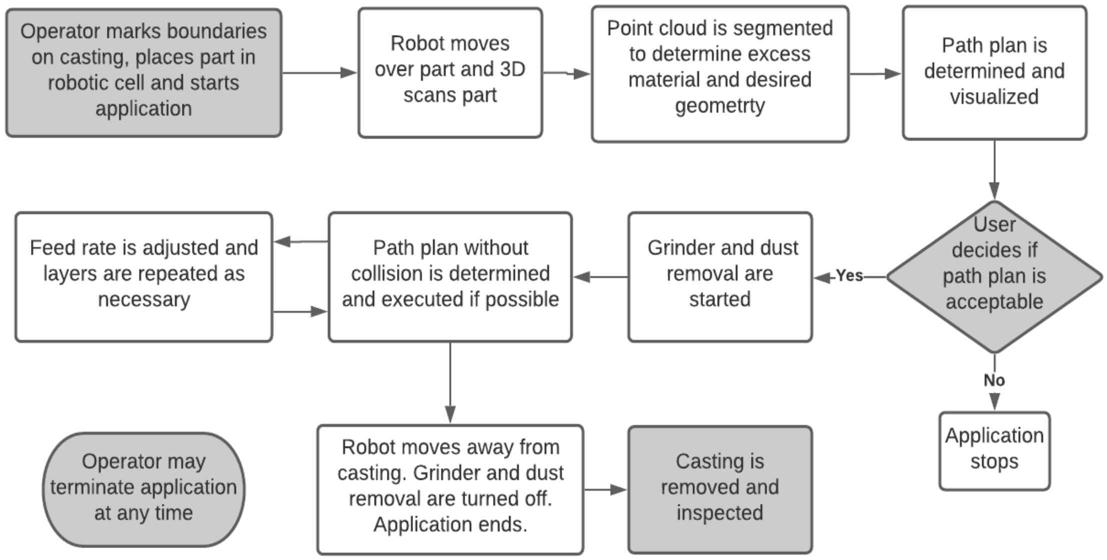

Figure 5-19: Detailed application process flow. Grey shapes represent tasks performed by operators.

5.4 Results

To evaluate the proposed method's effectiveness, tests were run on three different plastic surfaces. To enable this, the boundaries were drawn on the test plates by an operator. The operator then secured the parts in the robotic cell and executed the program. Once a path plan was determined, the operator confirmed the start of the material removal. The tests were performed without the grinder holder with compliance and thus without the velocity control, but should nonetheless indicate if the segmentation method enables the automatic removal of excess material. The grinder holder with compliance will become more important during tests with harder materials. These test surfaces had a variety of geometries. Figure 5-20 through Figure 5-22 present the results from testing the application on three different sample plates with different geometry. In **(a)** and **(c)**, a scan from the before and after grinding is compared with the CAD model. One is able to see that the scan before the grinding (not considering the abnormality) deviates from the CAD model.

Figure 5-20 presents the results on the flat sample plate. The flat sample plate with its abnormality before its ground can be seen in **(a)** and **(b)**. Figure 5-20 **(c)** and **(d)** both show the plate after it has been ground. One can see that the abnormality was removed completely, but a little more material than necessary has been removed. The maximum deviation was around 3 mm. In comparison, the accuracy of the 3D sensor (Intel RealSense D435) used to plan the grinding operation is ≤ 2% up to a two meter distance [24]. The test plates were scanned from a distance of 25 cm resulting in an accuracy of ≤ 5 mm. This means the maximum deviation in the test was smaller than the maximum deviation from the scanner for this distance. Thus, a large part of the deviation may be the result of the accuracy of the 3D sensor.

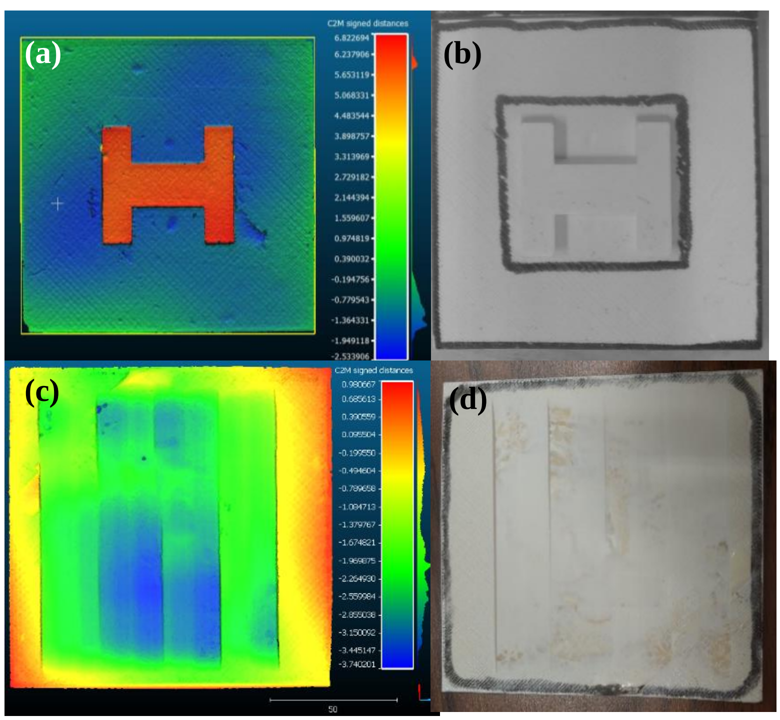

Figure 5-20: Results flat plate: **(a)** Deviations before removal (-2.5 (blue) – 6.8 mm (red)), **(b)** Picture before removal, **(c)** Deviations after Removal (-3.7 (blue) – 1 mm (red)), **(d)** Picture after removal

Figure 5-21 **(a)** and **(b)** show a cylindrical plate with its abnormality, while **(c)** and **(d)** show the test plate after the grinding took place. One can see that the abnormality was not completely removed, but the vast majority of it was. The maximum deviations are around 1 mm in both the positive and negative direction. One can also see that some over- and under-grinding is taking place.

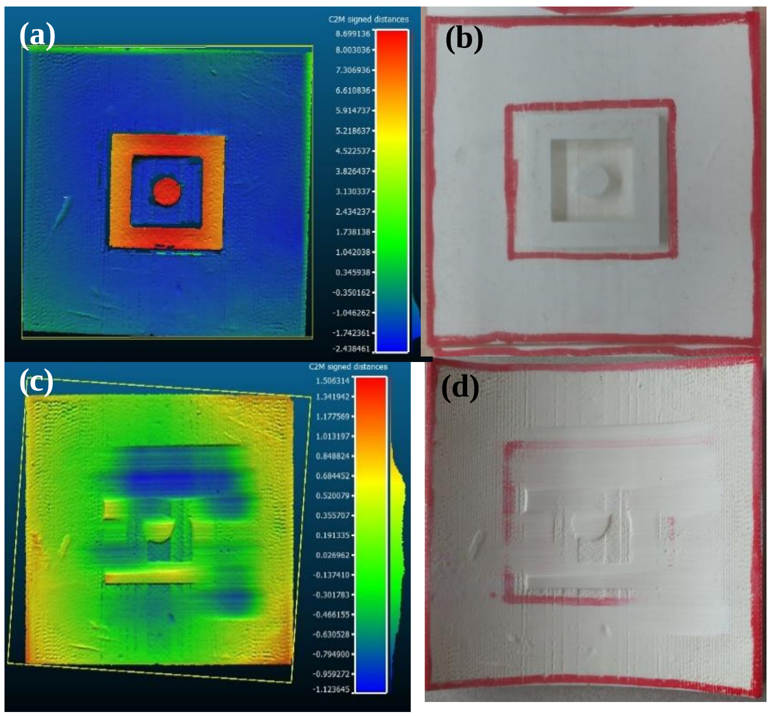

Figure 5-21: Results cylindrical plate: **(a)** Deviations before removal (-2.4 (blue) – 8.7 mm (red)), **(b)** Picture before removal, **(c)** Deviations after Removal (-1.1 (blue) – 1.5 mm (red)), **(d)** Picture after removal

Similarly, Figure 5-22 **(a)** and **(b)** present the spherical plate with abnormalities before it is ground in the robotic cell. Figure 5-22 **(c)** and **(d)** show the results after the grinding operation took place. From **(c)**, one can see that the abnormalities were removed completely, but a little over-grinding took place as well. The maximum deviation was around 3 mm, not considering the porosity.

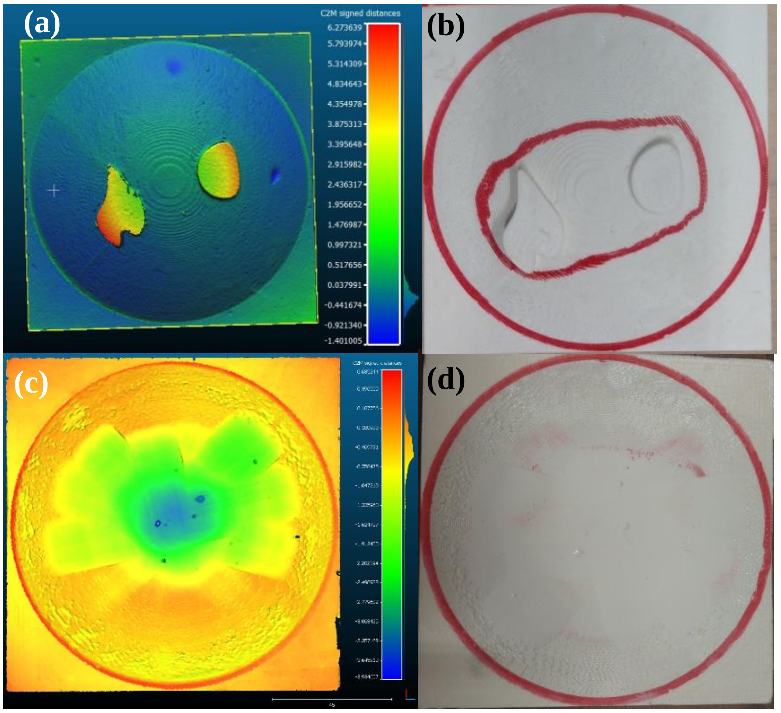

Figure 5-22: Results spherical plate: **(a)** Deviations before removal (-1.4 (blue) – 6.3 mm (red)), **(b)** Picture before removal, **(c)** Deviations after Removal (-3.9 (blue) – 0.7 mm (red)), **(d)** Picture after removal

Overall the majority of the abnormality volume was removed during the tests. The deviations are within the range of the scanner accuracy. The shape of the new surfaces matches the shape of the surrounding surfaces. The transitions between the ground and original surface can still be improved, especially in the areas where the edge of the grinding wheel created the transition.

5.5 Discussion

While there is much research on robotic blending applications, this research focused on a segmentation method for targeted material removal to remove excess material on castings. The segmentation method can not only be used for robotic grinding but can also be used as a basis for the automatic path planning of other operations, such as robotic welding or arc air.

The results showed that the excess material could be identified automatically by using markings on the casting and then automatically execute a removal operation. However, results also showed that while the final surface accuracy is better than the 3D sensor, which is about 5 mm, the surface blending in the transition areas still needs to be improved. Overall, the radial toolpath transitions seem to be best because the circumference of the grinding wheel softened the transition area between the original surface and ground surface. One way of further improving the transitions would be by utilizing a more precise sensor. In addition, to further improve the material removal process results, methods from robotic blending applications may be applied [23]. A blending operation could take place within the outer boundary. A grinding tool better suited for blending could also be used but would require a tool change.

A current limitation of the method is that it is currently only implemented for roughly convex boundary markings. This is caused by the way the points are sorted for the polygon construction for the point cloud segmentation. The method further currently uses only a 3D scan from a single position, thus limiting the surfaces which can be ground. In the future, this is planned to be extended by multiple scans on a sphere around the casting. In addition, the current velocity control is not robot agnostic since not all robot controllers allow such control.

Because of the labor shortages and the high competitiveness of foreign markets, it is more important than ever for smaller shops to be able to use automation in their foundries. The proposed method would increase the financial feasibility of robotic solutions. We expect it to decrease the amount of manual grinding necessary in foundries. In addition, technology adoption often increases a companies reputation, whereas "low technology" companies are often viewed with disdain [25]. The increased use of automation in foundries may thus be able to increase the attractiveness of the foundry for new talent, thus potentially lessening labor shortage problems further.

Overall, the tests showed that boundaries drawn by operators, their detection with a 3D scanner, and following segmentation can be used to automatically remove excess material with a grinding robot. Future work will include tests with different robots and 3D cameras to validate the robot and 3D scanner independence and develop a scanning strategy to enable grinding of more surfaces.

Acknowledgments

This research is sponsored by the DLA-Troop Support, Philadelphia, PA and the Defense Logistics Agency Information Operations, J68, Research & Development, Ft. Belvoir, VA.

Disclaimer

The publication of this material does not constitute approval by the government of the findings or conclusion herein. Wide distribution or announcement of this material shall not be made without specific approval by the sponsoring government activity.

References

[1] Vulcan Engineering Co., “VTS™ – Floor Mounted Model,” 2020. [Online]. Available: https://www.vulcangroup.com/product/vts-floor-mounted/. [Accessed: 16-Sep-2020].

[2] F. Peters and P. Patterson, “Ergonomic Improvements for Foundries,” Iowa State University, Ames, IA, Jun. 2002.

[3] M. Sharma, K. K. Kataria, S. Kant, and N. M. Suri, “Ergonomic Assessment of a Fettling Operation in Foundry Based on Digital Human Modeling and Statistical Analysis,” in *Optimization Methods in Engineering*, M. Tyagi, A. Sachdeva, and V. Sharma, Eds. Singapore: Springer Singapore, 2021, pp. 481–502, doi: 10.1007/978-981-15-4550–4_30.

[4] Y. Sun, J. Chen, Z. Lu, B. Yang, and W. Liu, "Ergonomics Analysis of Hand-Held Grinding Operation Working Posture Based on Jack," in *Man--Machine--Environment System Engineering*, S. Long and B. S. Dhillon, Eds. Singapore: Springer Singapore, 2020, pp. 733–740.

[5] R. Moore, "Steel Foundry Safety Benchmark Study," in *2019 Technical & Operating Conference*, 2019.

[6] U. S. I. T. Commission, *Foundry Products: Competitive Conditions in the U.S. Market, Inv. 332-460*. Washington, DC: DIANE Publishing, 2005.

[7] Cvetkovski S, "Wolrd Casting Industry-Preview," *Mach. Technol. Mater.*, vol. 9, no. 4, pp. 3--7, 2015.

[8] Magazine Modern Casting, "Reshoring and Metalcasting: Why It Matters | Modern Casting," *Modern Casting Magazine*, 2017. [Online]. Available: https://www.moderncasting.com/articles/2017/07/01/reshoring-and-metalcasting-why-it-matters. [Accessed: 09-Feb-2021].

[9] B. Siciliano and L. Villani, *Robot Force Control*. Boston: Springer Science & Business Media, 2012.

[10] W. Wang, R. N. K. Loh, and E. Y. Gu, "Passive compliance versus active compliance in robot-based automated assembly systems," *Ind. Robot An Int. J.*, 1998.

[11] M. H. Raibert and J. J. Craig, "Hybrid Position/Force Control of Manipulators," *J. Dyn. Syst. Meas. Control*, vol. 103, no. 2, pp. 126–133, 1981.

[12] K. Mohy El Dine, L. Lequievre, J. A. Corrales Ramon, Y. Mezouar, and J.-C. FAUROUX, "Hybrid Position/Force Control with Compliant Wrist for Grinding," in *MUGV 2018 (Machines et Usinage à Grande Vitesse) & Manufacturing'21*, 2018.

[13] Y. Oba and Y. Kakinuma, "Simultaneous tool posture and polishing force control of unknown curved surface using serial-parallel mechanism polishing machine," *Precis. Eng.*, vol. 49, pp. 24–32, 2017.

[14] F. Nagata, T. Hase, Z. Haga, M. Omoto, and K. Watanabe, "CAD/CAM-based position/force controller for a mold polishing robot," *Mechatronics*, vol. 17, no. 4–5, pp. 207–216, May 2007.

[15] Z. Pan and H. Zhang, "Robotic machining from programming to process control: A complete solution by force control," *Ind. Rob.*, vol. 35, no. 5, pp. 400–409, 2008.

[16] D. Wang, F. E. Peters, and M. C. Frank, "A Semiautomatic, Cleaning Room Grinding Method for the Metalcasting Industry," *J. Manuf. Sci. Eng. Trans. ASME,* vol. 139, no. 12, 2017.

[17] M. Quigley *et al.*, "ROS: an open-source Robot Operating System," in *ICRA workshop on open source software*, 2009, vol. 3, no. 3.2, p. 5.

[18] S. Chitta, "MoveIt!: An Introduction," in *Robot Operating System (ROS). Studies in Computational Intelligence*, vol. 625, A. Koubaa, Ed. Cham: Springer, 2016, pp. 3–27, https://doi.org/10.1007/978-3-319-26054–9_1.

[19] R. B. Rusu and S. Cousins, “3D is here: Point Cloud Library (PCL),” in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 1–4.

[20] M. K. Agoston, *Computer Graphics and Geometric Modeling: Implementation and Algorithms*. London: Springer-Verlag London, 2005.

[21] M. P. Groover, *Fundamentals of modern manufacturing: materials, processes, and systems*, 6th ed. John Wiley & Sons, 2015.

[22] B. W. Bequette, *Process control : modeling, design, and simulation / Wayne B. Bequette.*, 1st edition. Upper Saddle River, N.J.: Prentice Hall PTR, 2003.

[23] Y. Wen and P. R. Pagilla, “Robotic surface finishing of curved surfaces: Real-Time identification of surface profile and control,” in *ASME 2018 13th International Manufacturing Science and Engineering Conference, MSEC 2018*, 2018, vol. 3.

[24] Intel, “Intel® RealSense™ Camera D400 series Product Family Datasheet Rev. 01/2019,” 2019. [Online]. Available: https://www.intel.com/content/dam/support/us/en/documents/emerging-technologies/intel-realsense-technology/Intel-RealSense-D400-Series-Datasheet.pdf. [Accessed: 29-Mar-2021].

[25] M. E. Porter, “Technology and competitive advantage,” *J. Bus. Strategy*, vol. 5, no. 3, p. 60, 1985.

CHAPTER 6. GENERAL CONCLUSION

6.1 Conclusion

Visual inspection, the current method for surface roughness determination, has been shown to be unreliable because of its low reproducibility and repeatability. The poor reproducibility causes issues in the communication between foundry and customers. The in this dissertation introduced variogram method was developed to address this problem. It uses point cloud data as objective input to determine roughness values and highlight abnormalities. Many foundries are already gathering these points clouds for dimensional measurements. The method uses an underlying geometry estimation to filter the underlying geometry from the roughness calculation. The variogram is used to consider the spatial relationship between points on the surface. For complex geometries, edge detection identified the edge points and removed them from the roughness calculation. This is done because the underlying geometry detection is inaccurate around edges. The method is agnostic of the measurement device. Important is that accurate and dense point cloud data is used as the input. The method has been shown to be able to differentiate between common roughness comparators in the steel casting industry.

The second problem addressed in this dissertation a material removal method for castings. The material removal for low-volume products prevalent in the steel casting industry is widely performed through unergonomic hand grinding. This dissertation introduces a grinding method that utilizes a grinder and a 3D-RGB sensor mounted on a robot. By utilizing ROS, the method is robot and sensor agnostic so that sensors and robots can be switched without much effort. The method requires markings on the casting, which are captured by the 3D-RGB sensor and later identified and interpreted in the point cloud. Based on the markings, the desired surface and the excess material are identified. Both are used to determine a path plan for the material removal process. Before executing the path plan, the path plan is checked to be collision-free for both static and non-static collision environments. For the material removal of different castings, no code has to be changed. A worker has to place a casting, with boundaries marked, securely in the robotic work cell and execute the program. Once the path plan was determined, the operator can allow the execution of the path plan. After the material was removed, the part can be removed from the work cell and inspected by the operator. The method has shown to be able to remove material from previous unknown surfaces without the need for programming, only requiring marking of the surface to identify the excess material.

6.2 Future work

Many further research opportunities exist for both the inspection and finishing projects.

1) Investigate if it is possible to utilize the CAD to scan distance calculations done by geometric inspection software as an input for the variogram roughness method. This could reduce the need to remove the edges of a part before the roughness calculation.
2) Investigate other surface reconstruction methods for the underlying geometry detection.
3) Perform further tests of the variogram method in foundries and on different scanners.
4) Automatic grinding wheel diameter update to account for wear on the grinding wheel.
5) Test grinding method on metal surfaces.
6) Test grinding method with an industrial robot and more accurate 3D sensor.

7) Design and implement a method to find markings on casting surfaces and handle occlusion.
8) Implement post grinding inspection, potentially including a roughness determination, with a feedback loop.