

Montana Tech Library

Digital Commons @ Montana Tech

---

Graduate Theses & Non-Theses

Student Scholarship

---

Fall 2020

## PARTICLE TRACKING OF A SIMULATED MELT POOL OF SELECTIVE LASER MELTING

Prakash Gautam

Follow this and additional works at: [https://digitalcommons.mtech.edu/grad\\_rsch](https://digitalcommons.mtech.edu/grad_rsch)



Part of the [Materials Science and Engineering Commons](#)

---

PARTICLE TRACKING OF A SIMULATED MELT POOL OF  
SELECTIVE LASER MELTING

by  
Prakash Gautam

A thesis submitted in partial fulfillment of the  
requirements for the degree of

Master of Science in General Engineering

Montana Tech

2020



## Abstract

Metal additive manufacturing (AM) produces parts by addition as compared to subtraction of material. Selective laser melting (SLM) is an AM technique that prints objects layer-by-layer, selectively melting powders using a laser. The mechanical properties of SLM parts are affected by processing parameters and powder characteristics, both of which alter the molten melt pool flow field. Marangoni convection (M-flow) is a thermo-capillary mass transfer from a region of lower surface tension to a region of higher surface tension, referred to as a radially outward flow for SLM. However, in the presence of surface-active elements such as oxides and sulfides, the melt pool surface flow direction may shift from radially outward flow to radially inward flow (inverse M-flow). Balling and pores, the most common defects in SLM, have been correlated to the presence of inverse M-flow but the relation has yet to be quantified.

In order to quantify the surface flow using high-speed melt pool videos, numerical simulations and algorithms for melt pool flow were developed. The surface flow of a melt pool was simulated as binary images created using a MATLAB script. A particle tracking algorithm developed in MATLAB, using various functions from the MATLAB Image Processing Toolbox, was used to track the surface oxide particles in the simulated binary images. Various factors that may affect the particle tracking algorithm, such as 1) the melt pool size, 2) the image pixel size, 3) the size and the number of surface oxides, 4) the flow type, and 5) the particle velocity, were varied in the simulated binary images. Experiments were designed and results analyzed against the numerical error measured to generate a model using Design Expert 12, a statistical design analysis software package. Design Expert 12 was used to determine the significance of each factor within the model. The ANOVA results demonstrated that particle velocity and flow type have significant influence on the error of measured displacement. The error increased with decreasing displacement of the particles being tracked in the melt pool simulation. Melt pool algorithm error has been quantified and validated against simulated data, therefore it can be used to analyze actual melt pool data with known confidence.

Keywords: AM, SLM, Marangoni convection, Particle tracking

## **Dedication**

This thesis is in the memory of my late father, who believed in me and supported me to achieve my goals. This is a tribute to all the sacrifices that he made so that I could have a better future. This thesis is wholeheartedly dedicated to my beloved mother, who supported me throughout and taught me discipline and patience that are required to achieve goals.

To my two elder sisters who encouraged and supported me when I needed motivation to complete this work.

Finally, I dedicate this thesis to all my relatives, well-wishers, mentors, friends, and classmates for their advice and wisdom.

## Acknowledgements

I want to express my sincere gratitude to Dr. Peter Lucon, my advisor and the chair of my committee, for his encouragement and support during my Master's program at Montana Technological University, and for his guidance and motivation for the successful completion of my Master's thesis. The guidance of other members of my committee: Mr. Scott Coguill (Member), Dr. K. V. Sudhakar (Graduate School Representative), and Dr. Jack Skinner (Member) was crucial to complete my thesis. Dr. Richard M. LaDouceur mentored and supported me to perform experimental designs and data analysis. His statistical expertise made data analysis enjoyable. A special thanks to Dr. Cristina Stefanescu for her efforts in reviewing my thesis and providing valuable comments to improve it.

Working with the Resonance Technology Research (RTR) group led by Dr. Peter Lucon at Montana Technological University was wonderful. The work performed by Benjamin Rathman, Daniel Jacintho, Ingvar Kulseng-Hansen, Riley McNabb, and Taylor Winsor, fellow members of the RTR group, on the metal Additive Manufacturing (AM) projects were invaluable to create a foundation for this research. Other members of the RTR group, Garon Knudson and Marshall Metcalf, aided in various aspects of the research. The Center for Advanced Mineral, Metallurgical and Materials Processing (CAMP) at Montana Technological University helped me whenever I needed assistance.

Research was sponsored by the Combat Capabilities Development Command Army Research Laboratory and was accomplished under Cooperative Agreement No. W911NF-15-2-0020. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Combat Capabilities Development Command Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

Copyright ©2019 Chapter 1, Chapter 2, and Chapter 3.2.2.1: Reprinted with permission from "2019 Advances in Additive Manufacturing with Powder Metallurgy" Metal Powder Industries Federation, Princeton, New Jersey, USA, 2019.

## Table of Contents

<b>ABSTRACT .....</b>	<b>II</b>
<b>DEDICATION .....</b>	<b>III</b>
<b>ACKNOWLEDGEMENTS .....</b>	<b>IV</b>
<b>LIST OF TABLES .....</b>	<b>VIII</b>
<b>LIST OF FIGURES.....</b>	<b>IX</b>
<b>LIST OF EQUATIONS .....</b>	<b>XVIII</b>
<b>GLOSSARY OF ABBREVIATIONS.....</b>	<b>XIX</b>
1. INTRODUCTION .....	1
2. BACKGROUND.....	4
2.1. <i>AM process flow</i> .....	4
2.2. <i>Methods of AM</i> .....	4
2.3. <i>Selective laser melting</i> .....	5
2.3.1. Lasers used in SLM .....	7
2.3.2. Processing parameters of SLM .....	8
2.3.3. Shape and size of powder .....	9
2.3.4. Atmosphere:.....	9
2.4. <i>Melt pool convective flow</i> .....	10
2.4.1. Marangoni convection .....	11
2.5. <i>State of the art</i> .....	16
2.5.1. Selective laser melting.....	16
2.5.2. Marangoni convection .....	19
2.5.3. Flow visualization .....	20
2.6. <i>Objectives</i> .....	23

3. METHODS .....	25
3.1. <i>Binary imaging</i> .....	25
3.1.1. High-speed images conversion to binary images .....	27
3.2. <i>Particle tracking</i> .....	28
3.2.1. <i>Imfindcircles</i> .....	28
3.2.2. <i>Normxcorr2</i> .....	29
3.2.2.1. Particle image velocimetry (PIV) .....	31
3.2.3. Particle tracking algorithm .....	33
3.2.4. Accuracy of <i>imfindcircles</i> .....	35
3.2.4.1. Effect of the particle size.....	35
3.2.4.2. Effect of the number of particles .....	38
3.3. <i>Test plan</i> .....	40
3.3.1. Scaling of the images.....	42
3.3.2. Optimization of experiments.....	43
3.3.3. Independent studies.....	44
4. RESULTS AND DISCUSSION .....	46
4.1. <i>Binary image simulation</i> .....	47
4.2. <i>Particle tracking</i> .....	50
4.2.1. Vector plots .....	50
4.2.2. Displacement measurement .....	52
4.2.2.1. 2,000 $\mu\text{m}$ melt pool.....	53
4.2.2.2. 200 $\mu\text{m}$ melt pool.....	55
4.2.3. Effects of various factors on the displacement measured by the particle tracking algorithm .....	56
4.2.3.1. Effect of the size of melt pool .....	59
4.2.3.2. Effect of the image pixel size.....	66
4.2.3.3. Effect of the number of surface oxides .....	67
4.2.3.4. Effect of the flow path .....	73
4.2.3.5. Model validation .....	77
4.2.3.6. Independent studies .....	82

4.2.4.	Error analysis .....	88
4.2.4.1.	Linear motion .....	89
4.2.4.2.	Circular motion .....	91
5.	SUMMARY AND CONCLUSIONS.....	95
5.1.	<i>Executive summary</i> .....	95
5.2.	<i>Conclusions</i> .....	96
5.3.	<i>Recommendations for future work</i> .....	98
6.	REFERENCES .....	99
7.	APPENDIX A.....	103
7.1.	<i>Particle tracking algorithm</i> .....	103
7.2.	<i>Error analysis of circular motion</i> .....	105
8.	APPENDIX B.....	110
8.1.	<i>Particle tracking MATLAB script</i> .....	110
8.1.1.	Linear motion .....	110
8.1.2.	Circular motion.....	113
9.	APPENDIX C.....	116



## List of Tables

Table I. Factors that were varied in the simulated binary images of the melt pool. ....	41
Table II. Scaling of the simulated binary images. ....	43
Table III. Factors that were varied for both the sections of the experimental design. ....	44
Table IV. Probability that the error of measurement is greater than or equal to 20% for a 2,000 µm melt pool. ....	54
Table V. Probability that the error of measurement is greater than or equal to 20% for a 200 µm melt pool. ....	56
Table VI. ANOVA for reduced 2FI model (historical model). ....	58
Table VII. ANOVA fit statistics for the historical model. ....	58
Table VIII. Validation test plan. ....	79
Table IX. Result of the validation tests. Probability that the error of displacement measurement was greater than or equal to 20% during three different flow types. ....	79
Table X. Results to quantify the effects of the velocity of surface particles and the size of surface oxides on the displacement measured using the particle tracking algorithm. ....	83
Table XI. Results to quantify the effects of the size of surface oxides and the number of surface oxides on the displacement measured using the particle tracking algorithm. ....	84
Table XII. ANOVA for reduced quadratic model of the first independent study. ....	86
Table XIII. ANOVA fit statistics for the model of the first independent study. ....	86
Table XIV. ANOVA for linear model of the second independent study. ....	87
Table XV. ANOVA fit statistics for the model of the second independent study. ....	87

## List of Figures

- Figure 1. Process flow diagram of AM. First a part is made in a virtual CAD environment. The 3D CAD file is exported to a .STL file, which is then sliced into thin layers. The layer thickness varies with different AM methods. The sliced files are then transformed into printer specific print files (g-code). The parts are then printed layer-by-layer. The AM parts may or may not require post processing to achieve the desired physical part.4
- Figure 2. Schematics of a SLM process illustrating major components. The SLM machine is composed of a high-power laser source, an optical system to focus the laser onto the sample powder, a powder bed, and a build plate mechanism that moves as per the design requirement. ....6
- Figure 3. The percent absorptivity of light energy of various materials at incident wavelengths from 0.1  $\mu\text{m}$  to 10  $\mu\text{m}$ . ....7
- Figure 4. Processing parameters of SLM. ....8
- Figure 5. The natural convective flow of a melt pool during SLM. The density gradient present within the melt pool propels natural convection. The natural convective flow at the surface of the melt pool is from the center of the melt pool, where the laser light is incident and is at the highest temperature  $T_{\text{high}}$ , towards the edges (red dashed arrows). The flow continues downwards along the melt pool boundary (blue dash-dot arrows) and reaches the bottom where it is heated by the laser. The heated molten metal becomes more buoyant and rises to the top of the melt pool (black solid arrows), and the process repeats. ....10
- Figure 6. In a melt pool, the surface forces are from surface tension. The surface tension varies with temperature and in this example, the surface tension increases with decreased

temperature. The surface tension gradient drives flow in the direction of the cooler temperature in this example. ....11

Figure 7. A typical melt pool surface tension gradient without surface active elements drives

flow referred as radially outward flow (a). Radially outward flow occurs when  $\gamma_{Thigh} < \gamma_{Tlow}$ . As the concentration of surface-active elements increase, the intensity/magnitude of radially outward flow decreases and may reverse direction. The reversed flow direction is referred as radially inward flow (b) Radially inward flow occurs when  $\gamma_{Thigh} > \gamma_{Tlow}$  and the force due to surface tension gradient is large .....14

Figure 8. Simulated binary images representing surface flow of a melt pool during SLM. Three

binary images that have a defined time difference (dt) in between each image are considered. Melt pool center, melt pool boundary, and surface oxides are identified as bright spots. The velocity vectors show the direction of flow of surface oxides. The flow directions of the surface particles in the melt pool are displayed by a separate flow velocity image. The global flow direction of the melt pool is radially outward....26

Figure 9. Simulated binary images representing surface flow of a melt pool during SLM. Three

binary images that have a defined time difference (dt) in between each image are considered. Melt pool center, melt pool boundary, and surface oxides are identified as bright spots. The velocity vectors show the direction of flow of surface oxides. The flow directions of the surface particles in the melt pool are displayed by a separate flow velocity image. The global flow direction of the melt pool is radially inward.....27

Figure 10. An example in the documentation of the `normxcorr2` function in MATLAB to locate the position of ‘onion’ image in the ‘peppers’ image. The red circular region denotes the *xoffset* and *yoffset* values. ....31

Figure 11. Cross-correlation PIV. Multiple binary images are spaced at a defined time step  $dt$ . A cross-correlation PIV algorithm is applied between the images that examine many interrogation windows (nine interrogation windows were chosen to describe the process, but the number chosen is application specific) to determine the overall velocity direction and magnitude for each interrogation window. (a) An interrogation window example is displayed in grey and when the algorithm is implanted, a resulting velocity direction and magnitude is assigned for that interrogation window. (b) A velocity map from a cross-correlation PIV.....32

Figure 12. Tracking of an example particle. The circular particle in first image is detected by using the `imfindcircles` function. An interrogation window is then set up about the center of detected circle in both the image pairs. The size of the window created in the first image is twice the diameter of the detected circle. The size of window in the second image is twice that in the first image in this example to allow for a bigger search area in the second image. `Normxcorr2` function computes cross correlation between the two windows and the point of maximum correlation coefficient can be extracted. ....34

Figure 13. Absolute percentage error in the radius measurement of circular particles by using the `imfindcircles` function in MATLAB as a function of the particle size. High percentage error was encountered when small circles were detected. ....36

Figure 14. Absolute percentage error in the detection of x-coordinates of the centers of circular particles by using the <code>imfindcircles</code> function in MATLAB as a function of the particle size. ....	37
Figure 15. Absolute percentage error in the detection of y-coordinates of the centers of circular particles by using the <code>imfindcircles</code> function in MATLAB as a function of the particle size. ....	37
Figure 16. Absolute error of radius detection as a function of the number and the size of circular particles. The absolute error was mostly unaffected by the changes in the number of particles. Error bars are 3 SD from the mean error. ....	38
Figure 17. Absolute error in detection of x-coordinates of the center of detected circles as a function of the number and the size of circles. The absolute error was unaffected by the changes in the number of particles. Error bar is 3 SD from the mean error. ....	39
Figure 18. Absolute error in detection of y-coordinates of the center of detected circles as a function of the number and the size of circles. The absolute error was unaffected by the changes in the number of particles. Error bar is 3 SD from the mean error. ....	40
Figure 19. Hierarchical image representing the flow of work for the project. ....	47
Figure 20. A pair of binary images that were simulated to represent a linear radial outward motion .....	48
Figure 22. A pair of binary images that were simulated to represent a circular motion....	49
Figure 21. A pair of binary images that were simulated to represent a linear radial inward motion .....	49

- Figure 23. Result from the particle tracking algorithm: radial outward motion displayed as a vector plot. The particles had a constant linear displacement in a radial outward direction. ....50
- Figure 24. Result from the particle tracking algorithm: radial inward motion displayed as a vector plot. The particles had a constant linear displacement in a radial inward direction. ....51
- Figure 25. Result from the particle tracking algorithm: circular motion displayed as a vector plot. The particles had a varying linear displacement in a counterclockwise direction. The linear displacements of particles were larger when the particles were away from the center of rotation. ....52
- Figure 26. Effect of melt pool size on the error of measurement of a linear radial outward motion when the number of particles = 13. The probability of error decreased with an increase in the melt pool size. ....59
- Figure 27. Effect of melt pool size on the error of measurement of a linear radial outward motion when the number of particles = 5. The 95% CI was higher for smaller melt pool size when the number of surface oxides was decreased to 5. ....60
- Figure 28. Effect of melt pool size on the error of measurement of a linear radial outward motion when the number of particles = 20. The 95% CI was higher for smaller melt pool size when the number of surface oxides was increased to 20. ....61
- Figure 29. Effect of melt pool size on the error of measurement of a linear radial inward motion when the number of particles = 13. The probability of error was not affected by the changes in the melt pool size. ....62

- Figure 30. Effect of melt pool size on the error of measurement of a linear radial inward motion when the number of particles = 20. The mean probability of error increased slightly with a broader 95% CI as the number of particles was increased to 20. ....63
- Figure 31. Effect of melt pool size on the error of measurement of a circular motion when the number of particles = 13. The probability of error was unaffected by the changes in melt pool sizes but the 95% CI was broader when the melt pool size was in the extremes of the range.....64
- Figure 32. Effect of melt pool size on the error of measurement of a circular motion when the number of particles = 5. The 95% CI became broader as the number of particles was reduced to 5.....65
- Figure 33. Effect of melt pool size on the error of measurement of a circular motion when the number of particles = 20. The mean probability of error increased slightly and the 95% CI became broader as the number of particles was increased to 20. ....66
- Figure 34. Effect of the number of surface oxides on the error of measurement of a linear radial outward motion when the melt pool size = 1, 100  $\mu\text{m}$ . Mean probability of error was unaffected by the changes in the number of surface oxides but the 95% CI became broader when the number of particles approached the extremes of the range.....67
- Figure 35. Effect of the number of surface oxides on the error of measurement of a linear radial outward motion when the melt pool size = 200  $\mu\text{m}$ . The probability of error increased significantly with a broader 95% CI when the melt pool size was decreased. ....68
- Figure 36. Effect of the number of surface oxides on the error of measurement of a linear radial outward motion when the melt pool size = 2,000  $\mu\text{m}$ . The probability of error diminished to about zero with a narrower 95% CI when the melt pool size was increased.....69

Figure 37. Effect of the number of surface oxides on the error of measurement of a linear radial inward motion when the melt pool size = 1,100  $\mu\text{m}$ . Mean probability of error was not affected by the changes in the number of particles but the 95% CI became broader as the number of particles approached 20. ....70

Figure 38. Effect of the number of surface oxides on the error of measurement of a linear radial inward motion when the melt pool size = 200  $\mu\text{m}$ . Probability of error was not much affected when the number of particles were low. However, the probability of error slightly increased with a broader 95% CI as there were higher number of particles.71

Figure 39. Effect of the number of surface oxides on the error of measurement of circular motion when the melt pool size = 1,100  $\mu\text{m}$ . Probability of error increased slightly with the increased in the number of particles and the 95% CI became broader as the number of surface oxides approached the extremes of the range.....72

Figure 40. Effect of the number of surface oxides on the error of measurement of circular motion when the melt pool size = 200  $\mu\text{m}$ . Both the probability of error and the width of its 95% CI increased as the melt pool size decreased. ....73

Figure 41. Effect of the flow paths on the error of measurement when the melt pool size = 1,100  $\mu\text{m}$  and number of surface oxides = 13. Probability of error is low during linear motion and high during circular motion.....74

Figure 42. Effect of the flow paths on the error of measurement when the melt pool size = 200  $\mu\text{m}$  and number of surface oxides = 13. Probabilities of error during a linear radial outward motion and circular motion was significantly increased with a broader 95% CI whereas the probability of error during a linear radial inward motion was unaffected. ....75



- Figure 43. Effect of the flow paths on the error of measurement when the melt pool size = 1,100  $\mu\text{m}$  and number of surface oxides = 5. Mean probability of error was unaffected but the 95% CI became broader as the number of particles approached the lower-limit of the range.....76
- Figure 44. Effect of the flow paths on the error of measurement when the melt pool size = 1,100  $\mu\text{m}$  and number of surface oxides = 20. Mean probability of error was unaffected but the 95% CI became broader as the number of particles approached the upper-limit of the range.....77
- Figure 45. Effect of the velocity of particles on the error of measurement. Probability of error decreased with the increase in the particle velocity.....85
- Figure 46. The detection of a new location of the particle using the normxcorr2 function in MATLAB. The peak of the cross-correlation lies at the integer pixel location where there is maximum correlation coefficient between the matrix template and the image. The peak of the cross-correlation is used to calculate the displacement of particle in both the direction. ....90
- Figure 47. Pythagoras theorem used to calculate the displacement of particles in between a pair of images.....90
- Figure 48. An example of circular motion of two particles with same angular displacement of  $\alpha$ . The bigger particle is near the center of rotation while the smaller particle is away from the center. ....93
- Figure 49. Creation of interrogation windows in an image pair by the particle tracking algorithm to track a circular motion of particles. The interrogation windows in both the images are created around the center of detected circles in in image 1. The size of the interrogation

window in image 1 is equal to the diameter of detected circle whereas the size of the interrogation window in image 2 depend upon the value or  $r$  and the size of the window in image 1. The value of  $r$  in this example is 2, which creates interrogation window in image 2 whose size is five time the size of the window in image 1. ....94

**List of Equations**

Equation 1 .....	9
Equation 2 .....	9
Equation 3 .....	9
Equation 4 .....	12
Equation 5 .....	12
Equation 6 .....	15
Equation 7 .....	15
Equation 8 .....	30
Equation 9 .....	30
Equation 10 .....	103
Equation 11 .....	103
Equation 12 .....	104
Equation 13 .....	104

## Glossary of Abbreviations

<b>Term</b>	<b>Definition</b>
AM	Additive Manufacturing
SLM	Selective Laser Melting
CAD	Computer Aided Design
STL	Stereolithography or Standard Tessellation Language
SLS	Selective Laser Sintering
Nd-YAG	Neodymium-Doped Yttrium Aluminum Garnet
Yb-fiber	Ytterbium fiber
LED	Linear Energy Density
SED	Surface Energy Density
VED	Volumetric Energy Density
M-flow	Marangoni Convection
TIG	Tungsten Inert Gas
CCD	Charge-Coupled Device (high-speed camera)
GTA	Gas Tungsten Arc
CMOS	Complementary Metal-Oxide Semiconductor
PIV	Particle Image Velocimetry
PTV	Particle Tracking Velocimetry
SS	Stainless Steel
ANOVA	Analysis of Variance
CCD	Central Composite Design (experimental design)
CI	Confidence Interval

## 1. Introduction

Additive manufacturing (AM) is a manufacturing technique that builds an object layer-by-layer using a computer aided design (CAD) model. Additive manufacturing, also known as three-dimensional (3D) printing or rapid prototyping, is named after the method of manufacturing where components are built from the addition of materials as opposed to the removal of undesirable material performed by conventional machining techniques. Stereolithography was the first AM method developed by Charles Hull in 1986. The term ‘additive manufacturing’ first emerged in 1987 [1], [2] and is the subject of continuous research in the modern manufacturing world because AM has changed the way products/components are manufactured. It is now possible by using AM to manufacture components such as lattice or cell structures of interconnected struts that were perceived to be impossible (or nearly) to obtain using conventional manufacturing techniques [2], [3].

AM offers manufacturing abilities that conventional manufacturing is unable to provide. AM accelerates the rate of production of involved parts, or parts requiring assembly because complex shapes and some sub-assemblies can be directly 3D printed. The cost of mass production, energy usage, and waste are minimized when products are manufactured using AM methods. AM is a sustainable and environment friendly manufacturing method that makes proper utilization of resources and offers design flexibility [4]–[6]. AM offers mass customization of products, which means different customized products can be mass-produced to reduce the cost [6]. Even though AM offers numerous merits over its counterpart, there are certain limitations that need to be addressed for AM to replace the traditional manufacturing methods. The major hurdles in the development of AM are high initial cost and limited number of currently available materials for printing [4], [6]. Anisotropic microstructure and inferior mechanical properties of

the parts are two of the major drawbacks of AM [6]. AM is not plausible for large components because of inadequate material strength of 3D printed parts and large amount of time to manufacture [4].

Metal AM is a developing technology. There are defects related to metal AM process that need to be addressed in order for the method to replace the conventional machining techniques [7]. One such defect is related to the surface flow field within a molten melt pool created during selective laser melting (SLM) AM process for metals. The surface flow in a melt pool is governed by surface tension forces and buoyancy forces. Marangoni convection (M-flow) is a flow that occurs due to the surface tension gradient existing in the surface of a liquid. In a melt pool, M-flow takes place from a region of lower surface tension (*i.e.*, the center of melt pool) to a region of higher surface tension (*i.e.*, the edges of melt pool) that is referred to as radially outward flow. The flow direction can shift from radially outward to radially inward with the increase in surface-active oxides and sulfides [2], [8], [9]. This shifting is known as inverse M-flow for a melt pool. Defects in SLM parts such as balling and pores have been correlated to the presence of inverse M-flow [2], [10], [11], but the relationship between the defects in the SLM parts and the presence of inverse M-flow has not been quantified.

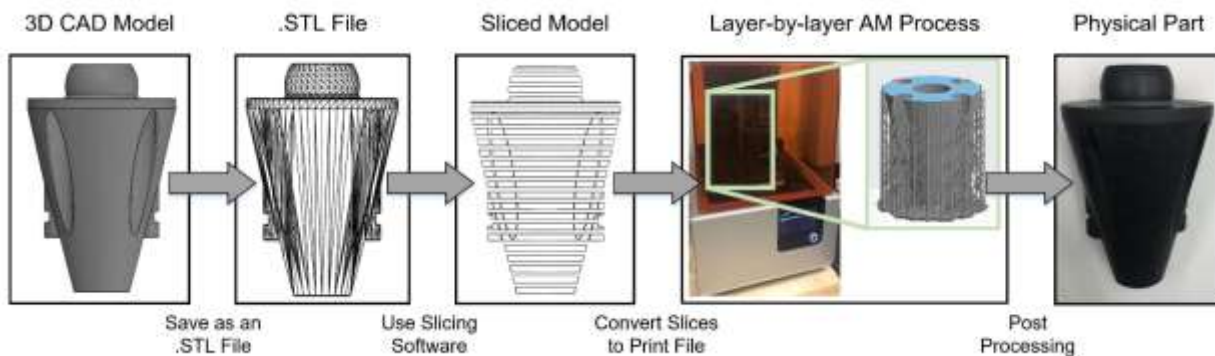
A high-speed camera can be used to visualize and study the behavior of the melt pool surface flow. The melt pool surface flow can be traced by tracking the surface oxide particles in the high-speed images of the melt pool. Therefore, a particle tracking algorithm was developed to track the surface oxide elements and determine their displacements in magnitude and direction. MATLAB was used to simulate the surface flow of a melt pool as binary images and create a particle tracking algorithm by utilizing various functions from the MATLAB Image Processing Toolbox. Binary images were simulated with various levels of each factor identified

to affect the particle tracking algorithm, such as 1) the melt pool size, 2) the image pixel size, 3) the number of surface oxides, 4) the flow type, and 5) the size of surface oxides. Design Expert 12, a statistical design analysis software package, was used to design the experiments, analyze the results against the numerical error of measurement, and generate a model to quantify the effects of various factors on the error of displacement measured by using the particle tracking algorithm. Analysis of variance (ANOVA) was performed on the model to identify the significant factors. Some of the factors that were initially thought to be significant were insignificant in the model, and particle velocity and flow type emerged as significant factors. High error of measurement was encountered when small linear displacements of the particles in the melt pool were measured.

## 2. Background

### 2.1. AM process flow

A flow diagram, Figure 1 [2], [12], demonstrates the AM process used to create a 3D object from the CAD model. The CAD model is saved as a STL (Stereolithography or Standard Tessellation Language) file and then sliced into thin cross-sectional layers using a slicing software package. STL is a file format that defines a 3D design by surface only. Various AM processes can be used to create a 3D object one layer at a time.



**Figure 1. Process flow diagram of AM. First a part is made in a virtual CAD environment. The 3D CAD file is exported to a .STL file, which is then sliced into thin layers. The layer thickness varies with different AM methods. The sliced files are then transformed into printer specific print files (g-code). The parts are then printed layer-by-layer. The AM parts may or may not require post processing to achieve the desired physical part.**

### 2.2. Methods of AM

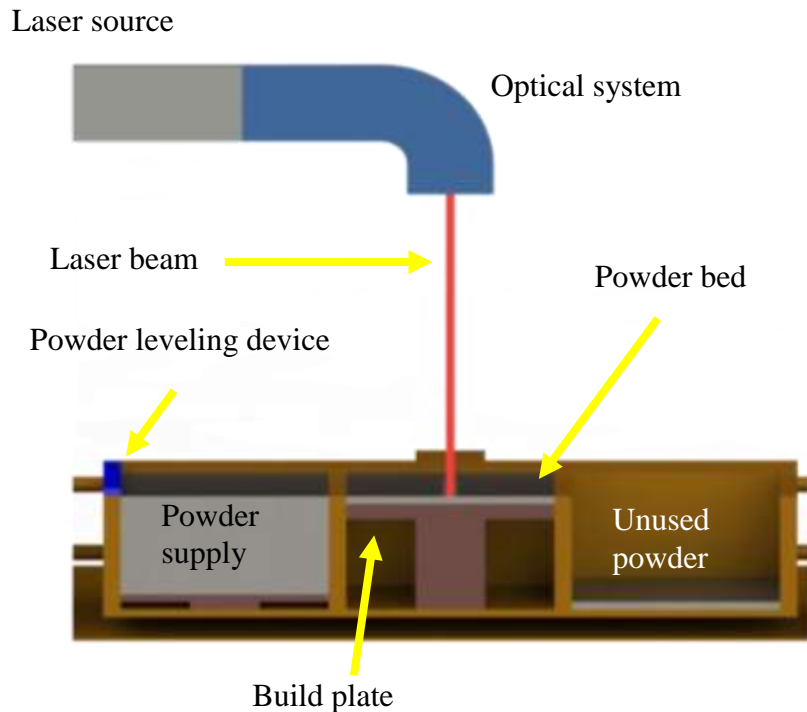
There are different methods of AM depending upon the manufacturing technique, type of materials used, and the application. AM consists of the following methods of fabrication for both metals and polymers according to ASTM standard [13]: 1) powder bed fusion; 2) material extrusion; 3) vat photopolymerization; 4) directed energy deposition (DED); 5) binder jetting or 3D printing; 6) material jetting; and 7) sheet lamination. Powder bed fusion (PBF) is an AM method where layers of powders are fused together by using heat from a laser source or an electron beam [6], [14]. The shape and size of the powder particles are important factors that



determine the density of printed parts. Selective laser sintering (SLS) and selective laser melting (SLM) are the most popular powder bed fusion manufacturing techniques that have identical setup but differ in fusion method [6]. SLS does not fully melt the powders and the layers are fused at molecular level because of elevated temperature. SLS is capable of printing variety of metals, alloys, and polymers. However, SLM uses a higher laser power compared to SLS to melt the powders fully before fusion, but is limited to select metal powders [2], [6], [15], [16]. Laser power and scanning speed are important processing parameters affecting the final properties of the parts printed by the PBF method. The PBF method is suitable for printing complex parts and offers better mechanical properties and relative density compared to DED and other metal AM processes, but it is a slow and costly process [6].

### **2.3. Selective laser melting**

SLM is a powder bed fusion AM technique, where a component is manufactured by melting the base powder using the energy given by a laser source. SLM is used to create complex parts by using a high-power laser source, an optical system to focus the laser onto the sample powder, a powder bed, and a build plate mechanism that moves as per the design requirement, displayed in Figure 2 [2], [6], [14], [17]. SLM process is conducted in an environmental chamber with an inert atmosphere [2], [10], [15], [17]–[20]. Unlike SLS, SLM melts the powder fully before fusion and hence offers better consolidation and mechanical properties [2], [6], [15], [16].



**Figure 2. Schematics of a SLM process illustrating major components. The SLM machine is composed of a high-power laser source, an optical system to focus the laser onto the sample powder, a powder bed, and a build plate mechanism that moves as per the design requirement.**

The AM method for SLM starts with a CAD model that is sliced into cross-sectional layers. The 3D object is created layer wise. Each layer is a 2D representation of a slice in the 3D object. A laser is used to melt the 2D representation of the layer and after each layer is completed, the powder bed is lowered by a distance equal to the thickness of each layer. Powder is distributed for the next printed layer and the aforementioned process repeats.

### 2.3.1. Lasers used in SLM

The type of lasers used in SLM depends upon the material being printed. Materials absorb light energy differently at various wavelengths. The light absorptivity of a material is used to select the type of laser for a specific application. Most metals have high absorptivity of radiation when the incident wavelength is short and hence neodymium-doped yttrium aluminum garnet (Nd-YAG) and

ytterbium fiber (Yb-fiber)

lasers producing beams at

around 1,060 – 1,080 nm

are typically chosen for

SLM of iron, titanium,

cobalt-chromium, and

aluminum-based powders

[2], [15], [16], [18], [21].

However, polymers and

other organic materials have

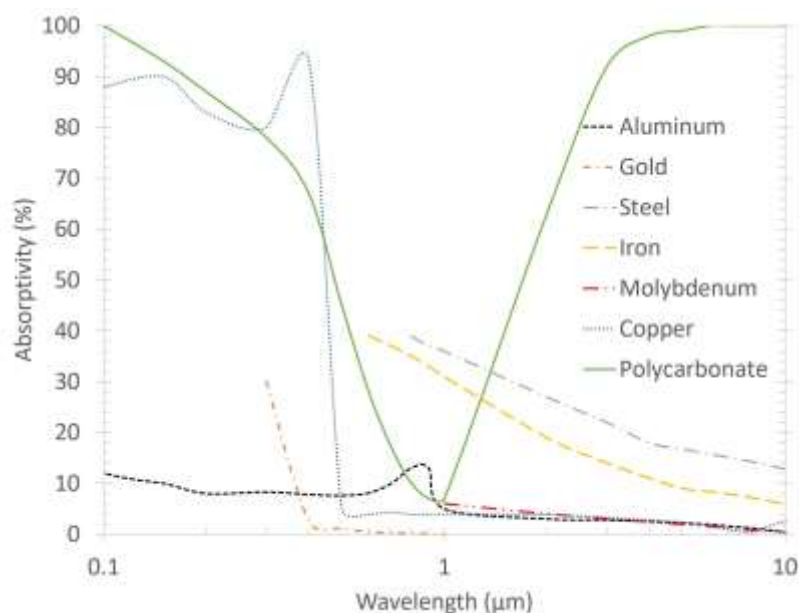
high absorptivity of light radiation at longer wavelengths. The wavelength typically used for

organic materials is 10.06  $\mu\text{m}$ , which is commonly produced by CO<sub>2</sub> lasers. Figure 3 [2], [16]

illustrates the absorptivity of various materials at wavelengths from 0.1 to 10  $\mu\text{m}$ . A Nd-YAG

laser is chosen over a CO<sub>2</sub> laser for steels because the absorptivity at an incident wavelength of

10  $\mu\text{m}$  is about 15 % while the absorptivity is roughly 35% at a wavelength of 1  $\mu\text{m}$  [2], [16].



**Figure 3. The percent absorptivity of light energy of various materials at incident wavelengths from 0.1  $\mu\text{m}$  to 10  $\mu\text{m}$ .**

### 2.3.2. Processing parameters of SLM

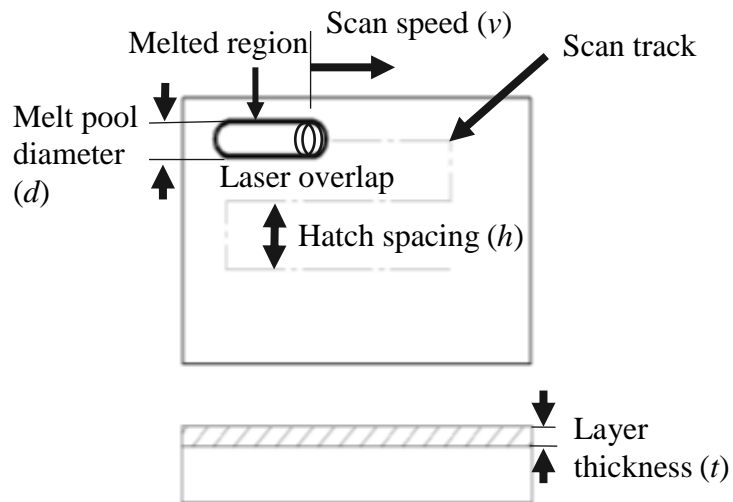
The mechanical properties, microstructure, and relative density of parts manufactured by SLM depend on the various processing parameters [2], [10], [15], [17]–[19], [22]. Figure 4 depicts the processing parameters of SLM: 1) laser power,  $P$ , is the power of the incident laser, 2) scan speed,  $v$ , is the rate at which the laser beam moves across the powder bed, 3) hatch spacing,  $h$ , is the distance between two scan tracks, 4) layer thickness,  $t$ , is the distance the powder bed is lowered each time

a new layer of powder is added, 5) laser beam diameter,  $\phi$ , is the diameter of incident laser beam, and 6) melt pool diameter,  $d$ , is the width of the melt pool created. Melt pool diameter is

greater than the laser beam

diameter because heat transfer by conduction, convection, and radiation transports energy from the molten pool to the surrounding powders. The transported energy melts the surrounding powders. The melt pool diameter increases with increased laser power and decreased scan speed.

Three types of energy densities are defined for SLM, including linear energy density (LED), surface energy density (SED), and volumetric energy density (VED). LED describes a single line scan and is the ratio between laser power and scan speed, while SED and VED define 2D and 3D operations, respectively. The relationships between LED, SED, and VED, and laser power  $P$ , scan speed  $v$ , hatch distance  $h$ , and layer thickness  $t$  are presented in Equation 1, Equation 2, and Equation 3, respectively [2], [21].



**Figure 4. Processing parameters of SLM.**

$$\mathbf{LED} = \frac{P}{v} \quad \mathbf{Equation\ 1}$$

$$\mathbf{SED} = \frac{P}{v * h} \quad \mathbf{Equation\ 2}$$

$$\mathbf{VED} = \frac{P}{v * h * t} \quad \mathbf{Equation\ 3}$$

### 2.3.3. Shape and size of powder

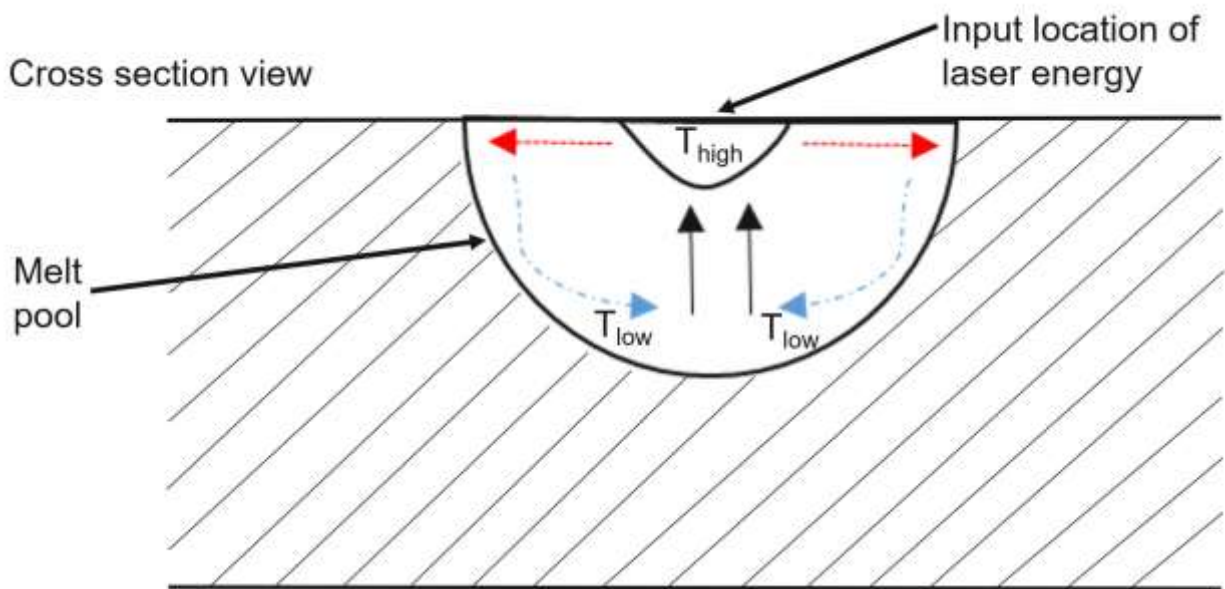
Spherical particles have a higher flow index than irregular shaped particles and have been demonstrated to produce smooth and consistent powder bed layers. The maximum size of the powder should not be greater than the layer thickness [2], [23]. The particle size range of 45 – 75  $\mu\text{m}$  promotes a well distributed melt pool with minimization of balling while larger particles are found to aggravate balling for aluminum-based powders [2], [22]. Generally, in industry production and research applications, spherical particles 15 – 50  $\mu\text{m}$  are preferred for SLM of 316L stainless steel (SS) [2], [18], [19], [21].

### 2.3.4. Atmosphere:

The presence of oxygen in the building environment induces oxidation at elevated temperatures and can lead to defects in the SLM fabricated part [2], [10], [20]. Therefore, the atmosphere during printing is a critical processing parameter and an inert environment of argon (Ar) or nitrogen (N) is preferred to prevent the formation of defects during SLM.

## 2.4. Melt pool convective flow

Laser radiation is absorbed by the material, converting the light into heat energy. When the heat energy is adequate, a melt pool is formed. In absence of any external forces, the melt pool flow is driven by natural convection as displayed in Figure 5 [2]. The density gradient present within the melt pool propels natural convection. The natural convective flow at the surface of the melt pool spreads out from the center of the melt pool, where the laser light is incident and at the highest temperature, towards the edges, red dash arrows in Figure 5. The flow continues downwards along the melt pool boundary (blue dash-dot arrows in Figure 5) and reaches the bottom where it is heated by the laser. The heated molten metal becomes more buoyant and rises to the top of the melt pool (black solid arrows in Figure 5) and the process repeats [2].



**Figure 5. The natural convective flow of a melt pool during SLM. The density gradient present within the melt pool propels natural convection. The natural convective flow at the surface of the melt pool is from the center of the melt pool, where the laser light is incident and is at the highest temperature  $T_{high}$ , towards the edges (red dashed arrows). The flow continues downwards along the melt pool boundary (blue dash-dot arrows) and reaches the bottom where it is heated by the laser. The heated molten metal becomes more buoyant and rises to the top of the melt pool (black solid arrows), and the process repeats.**

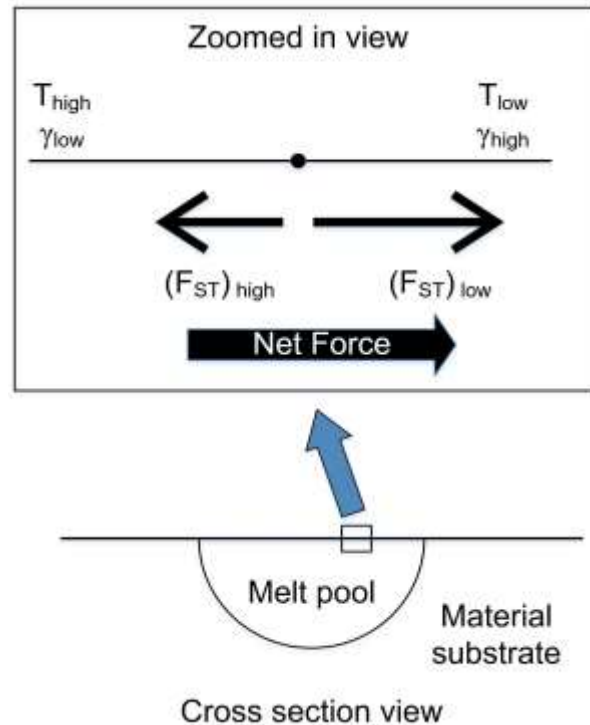
### 2.4.1. Marangoni convection

Marangoni convection (M-flow) is a thermo-capillary mass transfer driven by the surface tension gradient within a liquid. The surface region with higher surface tension pulls on the surface region with lower surface tension and a net force acts on the surface [2].

Therefore, the force on the surface acts from a region of higher surface tension to a region of lower surface tension as displayed in Figure 6 [2], [24]. The lower surface tension region becomes stretched and the flow occurs from a region of lower surface tension to a region of higher surface tension, which is referred to as the Marangoni effect.

Surface tension depends on the temperature of fluid and the concentration of surface-active elements. Equation 4 [2], [25] is used

to calculate the surface tension of a binary system at a given temperature  $T$ . The right side of the equation has three terms. The first term  $\gamma_m$  is the surface tension of pure metal at melting point. The second term  $A(T - T_m)$  is the temperature dependence of surface tension and the third term  $RT\Gamma_s \ln(1 + K\alpha_i)$  is the relation of concentration of surface-active elements to the surface tension of the system.



**Figure 6. In a melt pool, the surface forces are from surface tension. The surface tension varies with temperature and in this example, the surface tension increases with decreased temperature. The surface tension gradient drives flow in the direction of the cooler temperature in this example.**

$$\gamma = \gamma_m - A(T - T_m) - RT\Gamma_s \ln(1 + Ka_i) \quad \text{Equation 4}$$

$\gamma_m$  = Surface tension of pure metal at melting point

$A$  = Negative of temperature coefficient of surface tension of pure metal

$T_m$  = Melting point of the material

$R$  = Universal gas constant

$\Gamma_s$  = surface excess concentration at saturation

$K$  = Adsorption coefficient =  $k_1 e^{\left(\frac{-\Delta H^0}{RT}\right)}$

$k_1$  = Entropy factor

$\Delta H^0$  = Standard heat of adsorption

$a_i$  = activity of surface active element  $i$  in the solution

Temperature coefficient of surface tension  $\frac{\partial \gamma}{\partial T}$  is calculated by taking the partial derivative of Equation 4 with respect to temperature, Equation 5. The temperature coefficient of surface tension is dependent on both the temperature of the fluid and the concentration of surface-active elements [2], [25].

$$\frac{\partial \gamma}{\partial T} = -A - R\Gamma_s \ln(1 + Ka_i) - \left(\frac{Ka_i}{1+Ka_i}\right)\left(\frac{\Gamma_s \Delta H^0}{T}\right) \quad \text{Equation 5}$$

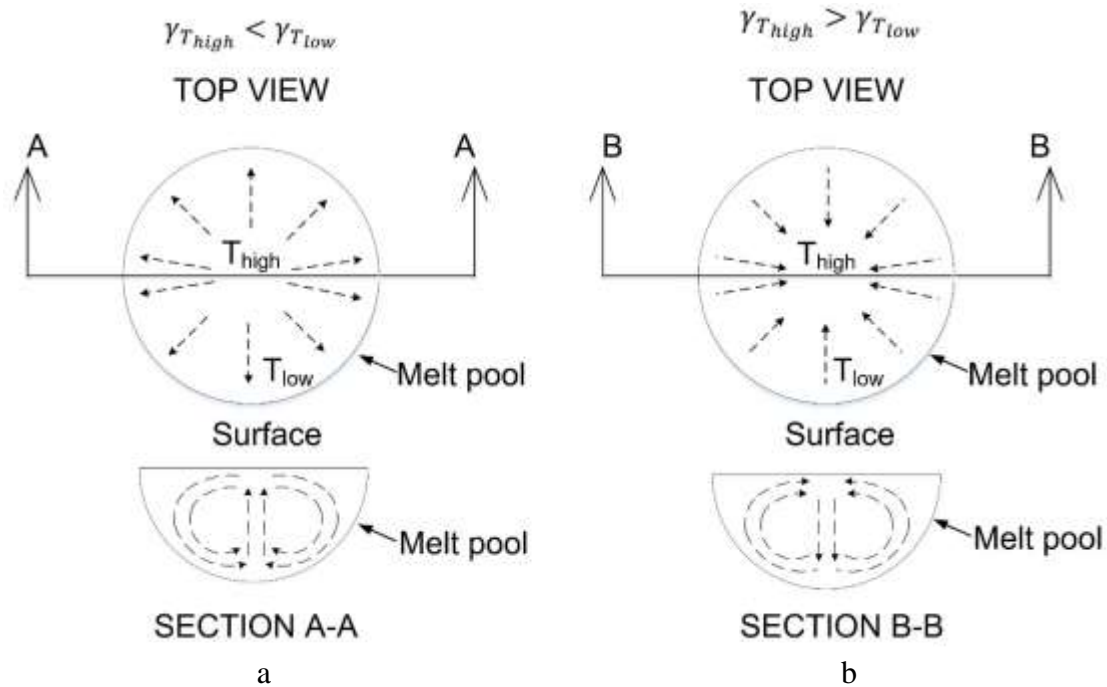
The temperature coefficient of surface tension of most molten metals is negative. The negative temperature coefficient decreases surface tension with an increase in temperature. Therefore, during SLM, the center of the melt pool, where the laser is incident, will have a higher temperature and a lower surface tension than at the melt pool edges. In this case, M-flow



at the melt pool surface is from the center to the edges and is defined as radially outward flow, Figure 5 and Figure 7 (a) [2], [9]. Radially outward flow results in a wide and shallow melt pool, which has been correlated to a reduced balling effect [2], [10], [11]. Balling effect refers to the formation of molten metal balls by melt pool instability that are ejected from the melt pool. If the melt pool is assumed to be a molten metal cylinder of length  $L$  and diameter  $D$ , instability of the melt pool is predicted by Plateau when the ratio between length to diameter of the melt pool exceeds  $\pi$  ( $L/D > \pi$ ) [2], [10], [18], [21]. The melt pool instability, also known as Plateau-Rayleigh instability, forms balls of molten metal to reduce the surface energy through minimization of the surface area. A sphere has the lowest surface area to volume ratio for any 3D shape and consequently the molten metal tends to form spherical balls.

However, the presence of surface-active elements like oxygen or sulfur in the melt pool can not only change the surface tension of the melt pool significantly, but also alter the direction and magnitude of surface flow of melt pool by inverting the surface tension gradient within the molten metal [2], [8], [9]. With surface-active elements, the surface tension at the center of the melt pool may become higher than at the edges. The  $R\Gamma_s \ln(1 + Ka_i)$  term related to the surface-active elements in Equation 5 may become dominant with the increase in concentration of surface-active elements as a result of oxidation at elevated temperatures. Therefore, the temperature coefficient of surface tension reverses and becomes positive. The center of the melt pool surface will have a higher surface tension than the surface edges, creating a net force counter to having no surface-active elements present. When the inverse M-flow resulting forces become greater than the buoyant forces, the global surface melt pool flow travels from the edges to the center and downwards from the center, defined as radially inward flow as displayed in

Figure 7 (b) [2], [9]. The radially inward flow results in a narrow and deep melt pool that is more susceptible to balling because of melt pool Rayleigh instability [2], [10], [11].



**Figure 7. A typical melt pool surface tension gradient without surface active elements drives flow referred as radially outward flow (a). Radially outward flow occurs when  $\gamma_{T_{high}} < \gamma_{T_{low}}$ . As the concentration of surface-active elements increase, the intensity/magnitude of radially outward flow decreases and may reverse direction. The reversed flow direction is referred as radially inward flow (b) Radially inward flow occurs when  $\gamma_{T_{high}} > \gamma_{T_{low}}$  and the force due to surface tension gradient is large**

Marangoni number,  $M_a$ , is a dimensionless number to measure the intensity of M-flow, given by the ratio between the surface tension force and viscous force, Equation 6. The Marangoni number is used to distinguish the direction of M-flow. A negative temperature coefficient of surface tension results in a positive Marangoni number that corresponds to M-flow. However, a positive temperature coefficient of surface tension results in a negative Marangoni number that results in inverse M-flow. The global surface flow continues to be radially outward

until the inverse M-flow becomes dominate over the buoyancy flow. A non-dimensional relation known as the Dynamic bond number,  $B_o$ , is the ratio of relative strengths of the natural or buoyancy convection and the Marangoni convection, Equation 7. The dynamic bond number is used to define the dominant force for driving the melt pool flow during M-flow. If the bond number is less than one, the flow is dominated by Marangoni effect [2], [26].

$$M_a = \frac{-\frac{\partial\gamma}{\partial T}(\Delta T)L}{\mu\alpha} \quad \text{Equation 6}$$

$M_a$  = Marangoni number

$\frac{\partial\gamma}{\partial T}$  = Temperature coefficient of surface tension

$\Delta T$  = Temperature difference between the center and the edge of melt pool

$L$  = Characteristic length

$\mu$  = Dynamic viscosity of the molten metal

$\alpha$  = Thermal diffusivity of molten metal

$$B_o = \frac{\beta\rho gL^2}{-\frac{\partial\gamma}{\partial T}} \quad \text{Equation 7}$$

$B_o$  = Dynamic bond number

$\beta$  = Coefficient of thermal expansion of molten metal

$\rho$  = Density of the molten metal

$g$  = Acceleration due to gravity

$L$  = Characteristic length, radius of melt pool

## 2.5. State of the art

The SLM process has evolved since its inception in 1995 [27]–[29]. New developments on how processing parameters affect the outcome of SLM parts have been explored [10], [15], [17]–[19], [21], [22]. However, a fundamental understanding of the mechanisms that form defects in SLM parts is still unclear. Select SLM, Marangoni convection [9], [11], [26], [30], [31] and visualization of weld/melt pool [32]–[37] research publication highlights and relevant results are described below demonstrating the state of the art in SLM research [2].

### 2.5.1. Selective laser melting

Balling has been a concern of SLM processes due to the creation of rough surface, inhibition in the deposition of a smooth layer, and reduction in the density of parts produced by SLM [15]. The addition of phosphorous in iron-based powder lowers the surface tension of the melt pool and the balling tendency. Kruth *et al.* conducted SLM of iron-based powders and found out that the vaporization effect generated low recoil pressure, made the melt pool flat, and helped mitigate balling. Reported results indicate that the relative density and microstructure of parts improved if the laser is operated in a pulsed mode [15].

Rombouts *et al.* studied the effects of various ingredients of alloyed steel powders on the single line scan and on the 3D object manufactured by SLM. Presence of oxygen in the build atmosphere increased the volume of melt pool of iron-based powders due to the exothermic oxidation of iron (Fe). Presence of silicon in the sample also increased the melt pool volume because the heat released during oxidation of silicon (Si) is considerably higher than that of Fe. Presence of Si often resulted in balling because the increase in length was found to be higher than the increase in width of the melt pool that consequently resulted in balling due to melt pool

instability. Presence of copper decreased the density of parts due to high reflectivity of copper for Nd:YAG lasers [10].

The effect of processing parameters and environment variables on the mechanical properties of 316L SS parts manufactured by SLM was studied by Zhang *et al.* They reported that a smooth substrate surface and high layer thickness caused balling and weak consolidation. In the experiments conducted, a laser power of 100 W, a scan speed of 0.3 m/s, and a layer thickness of 50  $\mu\text{m}$  were determined to be the best combination of processing parameters for SLM of 316L SS. Helium has a higher ionization energy than that of argon or nitrogen. The use of helium as protective gas increased porosity while decreasing relative density of parts because of plasma shielding that reduces incident laser energy onto the metal powder. Preheating the substrate (above 150° C) showed improvements in both the relative density and the dimensional accuracy [18].

Olakanmi explored the selective laser sintering (SLS) and SLM of aluminum-based powders, and reported the effects of processing parameters and powder properties on the microstructure and density of the parts. He created processing windows for a single line scan of Al-based powders, where the region with high power and low scan speed was observed as the region with excessive balling. He reported that for Al-based powders, an energy density greater than 30  $\text{J}/\text{mm}^2$  created high temperature and low viscosity of melt pool, which intensified M-flow and caused balling because of Rayleigh instability [22].

Yakout *et al.* quantified the relationship among various SLM processing parameters and the relative density of Invar 36 parts. Raising the laser energy density (VED) increased the relative density of Invar 36 parts printed by SLM until molten metal started spilling out due to over melting. A laser energy density of 60 – 75  $\text{J}/\text{mm}^3$  produced parts with maximum relative

density of the combinations of processing parameters explored. Correlation and analysis of variance (ANOVA) tests indicated that the laser power, scan speed, and hatch spacing have a statistically significant influence on the relative density of Invar 36 SLM parts [17].

The limitations of VED as a design parameter in SLM was elucidated by Scipioni Bertoli *et al.* Occurrence of continuous tracks, irregular tracks, or balling were observed on the single line scans of 316L SS with constant VED. Constant VED was used with various combinations of laser power and scan speed, and the results demonstrated that the VED cannot control the occurrence of balling. The scanning speed was found to be more accountable than laser power for the instability of melt pool and eventual balling. Balling was reported to occur during a SLM process of 316L SS when using too low VED (below  $100 \text{ J/mm}^3$ ) or by using too high VED (above  $242 \text{ J/mm}^3$ ). Higher laser energy resulted in keyhole melting regime. Keyhole melting regime is a result of a deep and narrow melt pool, where the melt pool has a high depth to width ratio and takes the shape of a keyhole. The keyhole melting mode results in evaporation of molten metal and often produces keyhole pores in manufactured parts [21].

Liverani *et al.* studied the effects of SLM processing parameters on the microstructure and mechanical properties of 316L SS printed by SLM. The reported conclusions stated that the laser power had the highest influence on the density of 316L SS parts of the processing parameters investigated. The experimental results concluded that SLM of 316L SS parts with nearly full density have better mechanical properties than those parts made by conventional manufacturing techniques. Liverani reported that binding defects, gas pores and voids due to residual stresses are the three major solidification defects found in SLM parts. A binding defect is incomplete melting as a result of low power. Gas pores are formed due to inclusion of gases in the melt pool. High rates of cooling induce residual stresses that can create voids or cracks [19].

### 2.5.2. Marangoni convection

Limmaneevichitr and Kou simulated a weld pool of sodium nitrate ( $\text{NaNO}_3$ ) to visualize the Marangoni convection. The transparent melt of  $\text{NaNO}_3$  was placed in glass vessels and a laser light-cut technique was used to visualize the flow using aluminum as the tracer element. The tests were done with a hypothesis that radially outward flow will occur in a weld pool in absence of surface-active elements. The appearance of velocity profile of two cells rotating in opposite direction was reported, which resembles radially outward flow [26]. Subsequently, a surface-active element potassium propionate ( $\text{C}_2\text{H}_5\text{COOK}$ ) was added to the weld pool of  $\text{NaNO}_3$  to visualize the effects of surface-active elements in the flow of the weld pool. The visualization results determined that the direction of flow was reversed inducing radially inward flow. Deeper weld pools were also observed when the direction of flow was reversed [9].

The effect of oxygen concentration on the Marangoni flow during laser spot welding was studied by Zhao *et al.* A defocused Nd:YAG laser emitting laser beam of 1,064 nm wavelength was used to create spot welds. A high-speed camera was used to examine the effects of the changes in  $\text{O}_2$  concentration on the Marangoni flow. A Phantom V.5 high-speed camera was operated at 10,000 frames per second (fps) to visualize the weld pool, and the oxygen level in the chamber was varied from 0.0059 vol. % (59 ppm) to 20.9 vol. %. Inward flow caused by a positive surface tension gradient and an outward flow caused by a negative surface tension gradient were observed. The transition from outward flow to inward flow was observed as a result of increase in the concentration of  $\text{O}_2$  [30].

The balling phenomena during SLM of tungsten was examined by Zhou *et al.* A laser remelting procedure was also performed on the samples to improve density and surface quality. The samples were printed in a chamber where the oxygen content was reduced below 600 ppm.

The exposure time was varied in this experiment and the effect of scan speed on the balling effect was analyzed by using a scanning electron microscope (SEM). The authors claimed that tungsten is vulnerable to balling because of slow wetting and spreading speeds. This behavior is the result of high surface tension and high viscosity of tungsten. The two types of Marangoni convections were discussed and the transition to inverse flow was attributed to the unwanted oxidation during the melting process. The balling effect was found to be more severe when the exposure time was high and the remelting procedure appeared to improve the density of SLM samples [11].

Pariona *et al.* analyzed the effects of scan speed on the Marangoni convection and eventual microstructure and morphology during laser surface remelting (LSR) of casted Al-1.5 wt.% Fe alloy. A finite element method (FEM) model was developed to simulate the distribution of temperature and fluid flow based on the established conditions. The microstructures of the samples were studied under an optical microscope and SEM to verify the findings from the model and to study Marangoni convection. The Marangoni effect was determined to be more significant during lower scan speeds than at higher speeds, which was also further verified from analysis of the SEM images and the results of the simulation [31].

### **2.5.3. Flow visualization**

The flow of molten metal is difficult to observe because molten metal is optically opaque. Various techniques have been used to visualize the flow in the opaque weld/melt pool as reported by Mizutani *et al.*, Henrikson, Zhao *et al.*, Zhao, and Gunenthiram *et al.* A common method to visualize the surface flow of the melt pool is to use a high-speed camera. However, X-ray transmission techniques have been used to envision the internal flow of a metal melt pool as reported by Mizutani *et al.* and Zhao [2].



Mizutani *et al.* used a high-speed video camera and X-ray transmission method to visualize keyhole phenomenon and spattering during laser welding of tin and zinc using a Nd:YAG laser. The spattering images were taken at 13,500 frames per second to observe the high-speed phenomena while illuminating the molten pool with a xenon arc lamp during welding by using a Nd:YAG laser operated in a pulsed mode. The keyhole phenomena were quantified using an X-ray transmission method. The X-ray transmission method was used to visualize interior molten metal at 1,000 frames per second during continuous wave laser welding [32].

Henrikson visualized weld pool surface flow during tungsten inert gas (TIG) welding on 316L SS using a charge-coupled device (CCD) camera, ultraviolet (UV) lights, and a narrow band pass filter. He reported that images taken in the near infrared wavelength range of 780 – 1200 nm and visible wavelength range of 380 – 780 nm may contain reflected radiation and noise from the arc. Therefore, the UV wavelength range was chosen to visualize the flows because of reduced thermal radiations generated in that wavelength range [33].

The surface flows during gas tungsten arc (GTA) welding of 316L SS were analyzed by tracking the movement of natural oxides by Zhao *et al.* The surface velocity was estimated using the high-speed images captured by a complementary metal-oxide semiconductor (CMOS) high-speed camera and particle image velocimetry (PIV). An optical band pass filter (wavelength = 632 nm) was used with Nikon 70-300 mm camera lens to reduce interference from the arc light and to improve image contrast. A laser light (10 mW He-Ne 632 nm) was used to illuminate the weld pool. The camera and laser lights were at 60° to the plate normal when visualization was carried out. Images were taken at 500 Hz with a frame size of 512 x 512 pixels [34].

Zhao *et al.* also used high-speed imaging to observe the various surface flows during laser welding of 304L SS. A Nd:YAG laser with a maximum power output of 3 kW was used to

weld in an oxygen environment without any shielding gas. An optical system consisting of a CMOS high-speed camera (Phantom V5), Nikon 70 – 300 mm optical lens, 62 mm close up lenses, and adapter rings was used for the visualization of surface flows in the weld pool. The images were captured at a speed of 10,000 frames per second with a frame size of 256 x 256 and an exposure time of 14 microseconds [35]. Zhao used an X-ray transmission technique that consists of a micro focused X-ray tube and a high-speed camera system to visualize the internal flow [36].

A Photron Fastcam SA2 high-speed camera was used to visualize melt pool and spatter ejection of 316L SS and Al-12Si powders during SLM by Gunenthiram *et al.* The melt pool was visualized at 12,500 frames per second at 60° angle and the spatter ejection was visualized at 4,000 frames per second at an angle of 0°. A pulsed laser diode emitting rays of 0.83 μm was used to illuminate the melt pool [37].

## 2.6. Objectives

SLM is an important AM technique because of its ability to print complex geometries. However, undesirable defects exist in SLM printed parts. The relation between the processing parameters and the melt pool flow is not clearly understood. The relation between melt pool flow and various SLM defects has not yet been quantified.

The SLM melt pool surface flow can be visualized by using a high-speed camera and an illuminating source. The magnitude and direction of the surface flow of a melt pool produced during SLM can be determined by tracking the motion of the surface oxides by using a particle tracking algorithm. In an attempt to understand the melt pool surface flow, the first step would be to develop a particle tracking algorithm and calibrate it at various levels of each factor that might affect the algorithm. A particle tracking algorithm that uses cross-correlation technique to determine the displacement of objects in between a pair of images was calibrated by using simulated binary images with predetermined displacements of particles in between each image pairs.

The objectives of the thesis are divided into two sections that are as follows:

- I. Write MATLAB scripts to simulate binary images representing a melt pool of SLM with five different factors that are:
  1. melt pool size
  2. number of surface oxide particles
  3. total pixel size of the simulated image
  4. flow paths of surface oxide particles
  5. size of surface oxide particles

- II. Develop and calibrate the MATLAB script for particle tracking by comparing the measured displacements with the model displacements of the particles in the simulated binary images.

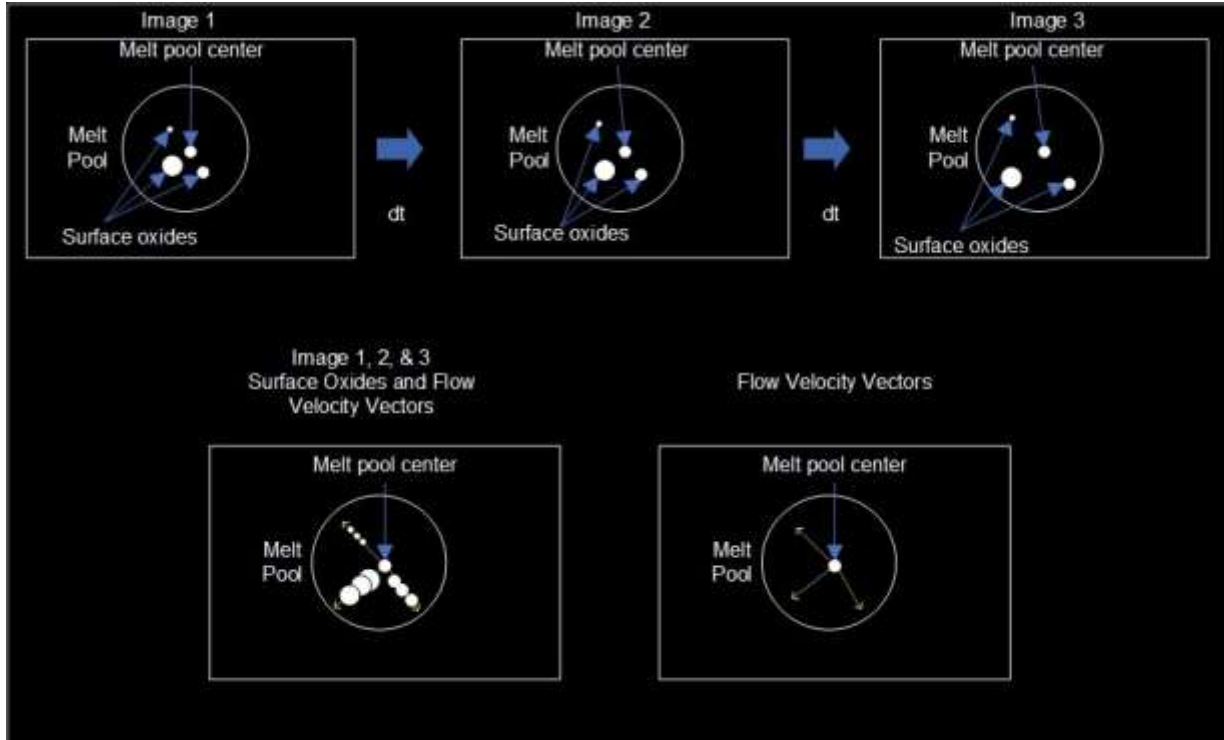
### 3. Methods

The project has two major goals and MATLAB was used to complete both the goals. The first goal is to generate binary images of simulated melt pools to represent actual melt pools during SLM. The second goal is to develop a particle tracking algorithm and use it to analyze those simulated images to calibrate the code. A test plan was designed and optimized using the Design Expert 12 package. The test plan outlined the production of binary images that are sufficient to calibrate the particle tracking code. The discussions of binary imaging, particle tracking, and test plan are presented in the following sections.

#### 3.1. Binary imaging

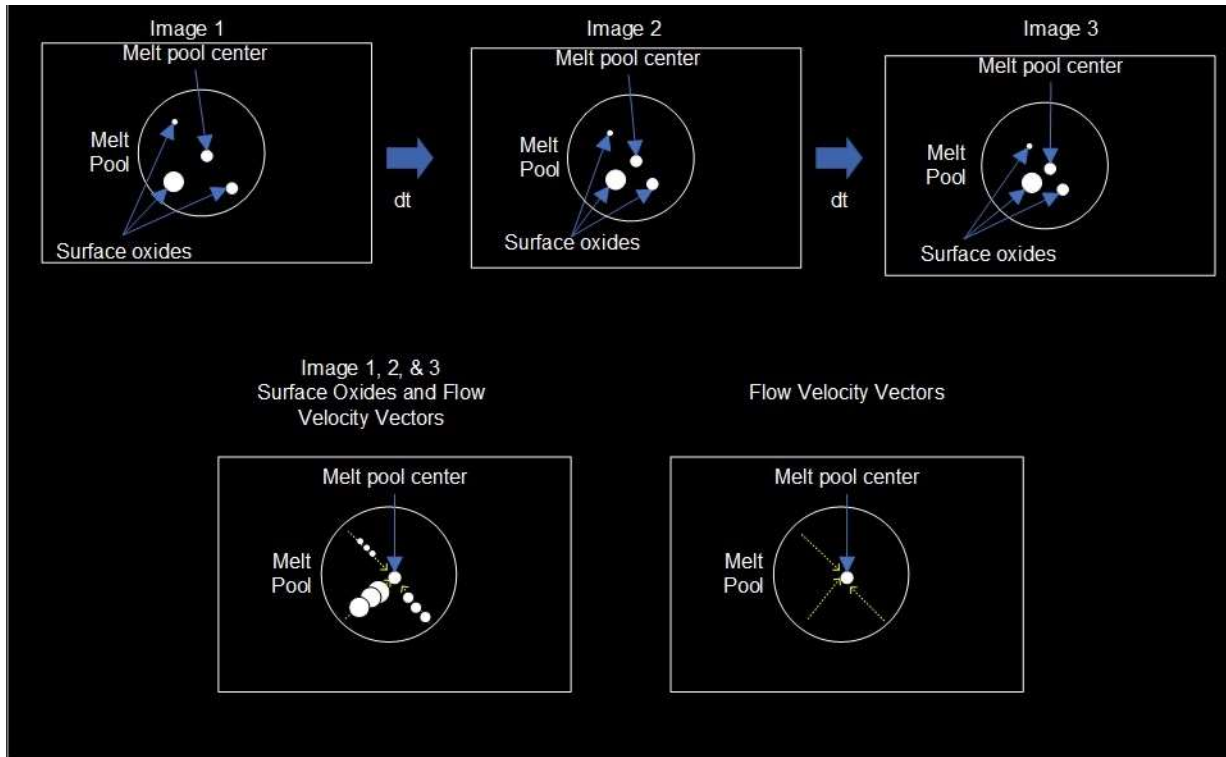
A binary image is defined using only two colors, black and white. There is only one bit per pixel in binary images, differing from grayscale images that contain multiple bits per pixel. Each pixel of a binary image is stored as a single bit, *i.e.*, either 0 (black) or 1 (white).

Binary images for this project were simulated by writing MATLAB scripts. The melt pool boundary, surface oxides, and the center of the melt pool were considered as bright spots for the image simulation, and had an intensity value of 1. The rest of the image pixels contained 0 intensity value and were dark. Figure 8 [2] displays simulated binary images during SLM revealing a radially outward flow due to Marangoni effect. The surface-active elements (oxides) move from near the center to the edges of the melt pool. These surface oxides moved from their location in image 1 of Figure 8 to a new location in image 3 of Figure 8 and generated a flow velocity vector in radially outward direction from the center of the melt pool towards the edges. The surface oxides in the simulated binary images are of different sizes to represent an actual melt pool with multiple sized surface oxides [2].



**Figure 8. Simulated binary images representing surface flow of a melt pool during SLM. Three binary images that have a defined time difference ( $dt$ ) in between each image are considered. Melt pool center, melt pool boundary, and surface oxides are identified as bright spots. The velocity vectors show the direction of flow of surface oxides. The flow directions of the surface particles in the melt pool are displayed by a separate flow velocity image. The global flow direction of the melt pool is radially outward.**

Figure 9 [2] displays simulated binary images of a radially inward flow during SLM. As the melting process continues, high amounts of incident energy will increase the temperature sharply. Rapid oxidation starts to take place at elevated temperature that reverses the M-flow by inverting the temperature coefficient of surface tension. The surface-active elements move from the edge of the melt pool to the center [2].



**Figure 9. Simulated binary images representing surface flow of a melt pool during SLM. Three binary images that have a defined time difference ( $dt$ ) in between each image are considered. Melt pool center, melt pool boundary, and surface oxides are identified as bright spots. The velocity vectors show the direction of flow of surface oxides. The flow directions of the surface particles in the melt pool are displayed by a separate flow velocity image. The global flow direction of the melt pool is radially inward.**

The high-speed images of the melt pool recorded during SLM process are converted into binary images similar to the ones displayed in Figures 8 and Figure 9. These binary images are analyzed using a particle tracking algorithm and explained in a later section.

### 3.1.1. High-speed images conversion to binary images

There are two functions in MATLAB that can be used to convert an image to binary image i.e. `im2bw` and `imbinarize`. The newer versions of MATLAB do not recommend the `im2bw` function because of improved binarization and thresholding functionality in the `imbinarize` family of functions. Therefore, the `imbinarize` function should be used to convert the high-speed images of the melt pool to binary images. All pixels in the image that have intensity

value greater than the threshold value will have their intensity value replaced by 1 (bright spots) while all other pixels will have their intensity value replaced by 0 (dark spots). If the threshold value of intensity is not specified, `imbinarize` function uses Otsu's method to set a globally determined threshold value that minimizes the intraclass variance of the thresholded black and white pixels [38].

The threshold value can be specified as a numeric scalar or as a numeric array with a value between 0 and 1. Numeric scalar is used with 'global' method to binarize images using global image threshold calculated by Otsu's method. Whereas, numeric array is used with 'adaptive' method that uses locally adaptive image threshold to binarize the image. Sensitivity can only be changed for adaptive thresholding that has a default value of 0.5 [39].

### **3.2. Particle tracking**

The tracking of the movement of surface oxides in the binary images was performed by a MATLAB script (script title and reference in Appendix B) that used various functions from the MATLAB Image Processing Toolbox (MIPT), an addon for MATLAB. The functions used for particle tracking are `imfindcircles` and `normxcorr2`. The `imfindcircles` function is found under the Analyzing and Enhancing Images section and the `normxcorr2` function is under the Image Registration section of the MIPT documentation.

#### **3.2.1. Imfindcircles**

The `imfindcircles` function in MATLAB detects circles in an image using circular Hough transform. It is a function in the Analyzing and Enhancing Images section of the MIPT documentation [39]. The Hough transform was invented by Paul Hough in 1962 for extracting edge features from an image. The Hough transform maps a point from the image space into a line or curve in Hough space to detect a group of pixels sharing common features such as being on



the same line or crossing lines. The various objects in an image are then identified after the detected lines are drawn [40]. A circular Hough transform is used to extract three parameters required to represent a circle, which are the cartesian co-ordinates of the center and the radius. The radius range is a user provided variable in the `imfindcircles` function. The edges in the image are detected, and circles with radii that are within the given radius range are drawn centered to those detected edge points. An accumulator matrix collects a vote for every coordinate that overlaps within the predicted circumference of the drawn circles. The pixel cells with the highest number of votes are identified as the coordinates of the centers [41].

`Imfindcircles` function offers various input arguments that can be perturbed to alter the circle detection results, such as object polarity, method, sensitivity, and edge threshold. Object polarity must be specified as 'bright' to detect brighter circles. The default method used in the circle detection is phase code method but can be changed to a two-stage circular Hough Transform. The default sensitivity is set at 0.85. Increasing the sensitivity will allow `imfindcircles` function to detect weak and partially obscured circles. Finally, edge threshold establishes gradient threshold during edge detection. Setting edge threshold to a lower value allows detection of circles with weak edges [39]. The edge threshold can also be set to a lower value while detecting circles on images that have high noise.

### **3.2.2. Normxcorr2**

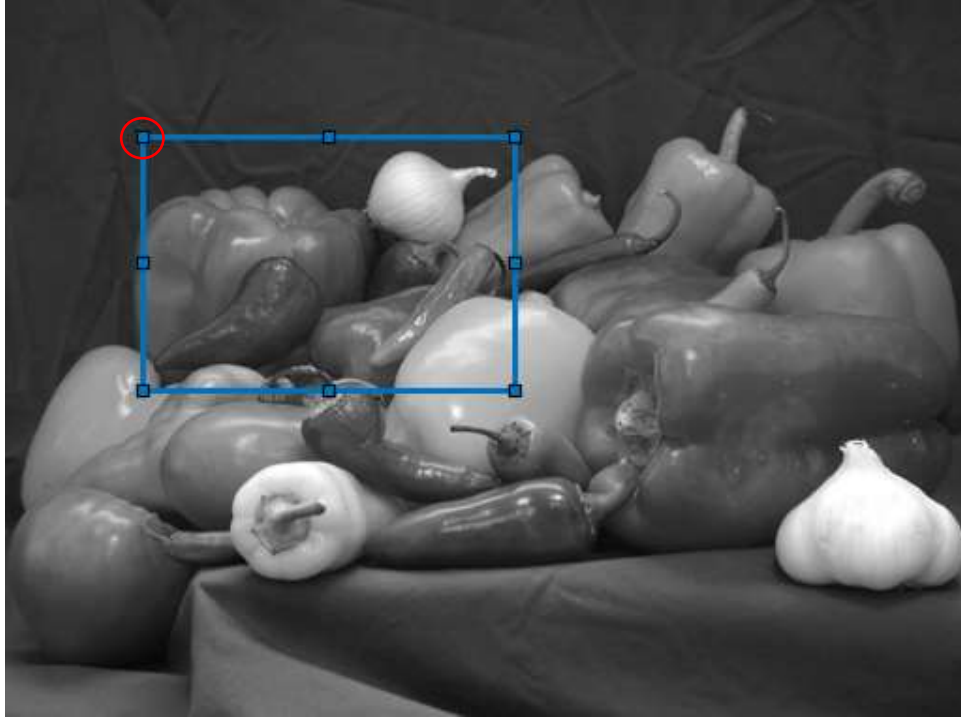
`Normxcorr2` function in MATLAB performs normalized 2D cross-correlation between a matrix template and an image. The matrix template is a window extracted from the first image of the image pair being investigated while the other image is a window extracted from the second image. `Normxcorr2` function is included with the Geometric Transformation and Image Registration section of MIPT, an addon for MATLAB [39]. The template is cross-correlated to

the image and a correlation matrix records the correlation coefficients with values in range of [-1,1] at each position of template in the image. Pixel location with the maximum correlation coefficient is assumed to be the new location of the template whose movement is being tracked. The documentation for `normxcorr2` defines  $x_{offset}$  and  $y_{offset}$  using Equation 8 and Equation 9 [39].

$$x_{offset} = x_{peak} - \mathbf{size}(\mathbf{template}, 1) \quad \text{Equation 8}$$

$$y_{offset} = y_{peak} - \mathbf{size}(\mathbf{template}, 2) \quad \text{Equation 9}$$

The  $x_{offset}$  and  $y_{offset}$  specify the cartesian coordinates of the upper left edge of the template on the image that matches the pixel information of the image. When two images are taken for cross-correlation, the matrix template is extracted from the first image whereas the image is extracted from the second image. The  $x_{offset}$  and  $y_{offset}$  in this case would provide the location of the upper left corner of the template in the image where the template would have the highest correlation with the image. The new location of template can be used to calculate the displacement of objects in between an image pair. Figure 10 [39] displays an example in the documentation of the `normxcorr2` function in MATLAB to locate the position of ‘onion’ image in the ‘peppers’ image.



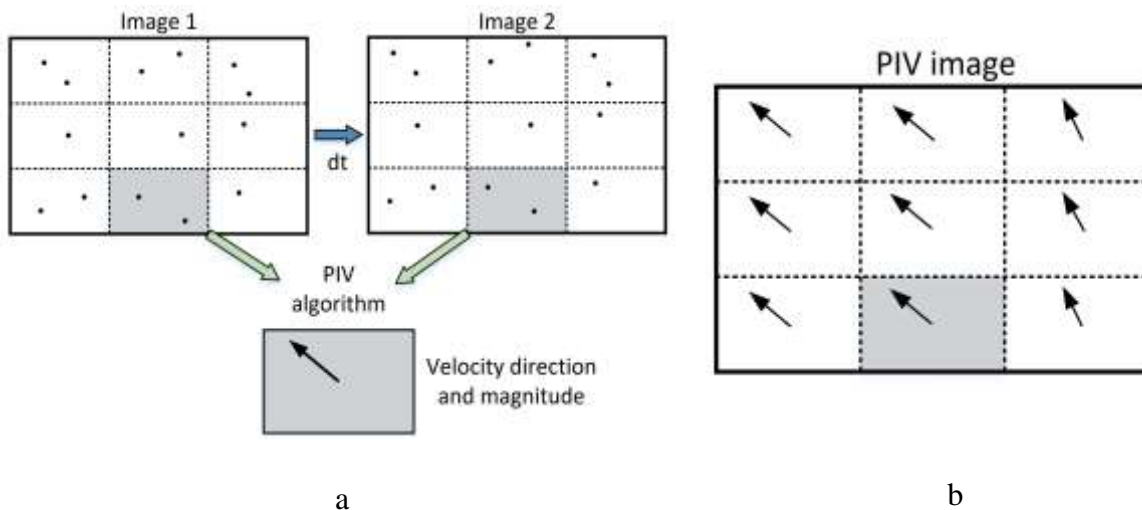
**Figure 10.** An example in the documentation of the `normxcorr2` function in MATLAB to locate the position of ‘onion’ image in the ‘peppers’ image. The red circular region denotes the  $x_{offset}$  and  $y_{offset}$  values.

### 3.2.2.1. Particle image velocimetry (PIV)

Particle image velocimetry (PIV) is an instantaneous velocity measurement technique performed by tracking the surface particles or tracers in a flow field [2]. PIV is used to track a large number of particles in an interrogation area, also known as interrogation window [42]–[44], and differs from particle trace velocimetry (PTV). PTV measures the movement of the individual particles and performs well when the concentration of tracers or surface particles are low in number and it is plausible to individually track each particles [42], [45], [46]. PIV is an Eulerian approach to velocity estimation whereas PTV is a Lagrangian method [44]. PIV compares images at successive intervals of time to generate displacement vectors of particles. Velocities in any direction can then be determined by taking the displacement in that particular direction

divided by the time between the two images. PIV can be used to estimate one-dimensional, two-dimensional, or even three-dimensional velocities of a flow [45]–[47].

PIV with cross-correlation can be used to determine the velocity of surface oxides in a melt pool. The high-speed images are converted into binary images and divided into multiple interrogation windows as illustrated in Figure 11 [2], [34]. Corresponding windows in each image were correlated, and the point of maximum correlation specifies the final average displacement of that window [2]. The interrogation windows can be cross-correlated using ‘normxcorr2’ function in MATLAB that computes normalized cross-correlation between a template and an image that is bigger than the template. Displacement vectors are generated as a result of cross-correlation between the two successive binary images, which are expressed in terms of pixels. These pixel displacement values can then be converted into spatial displacement values by taking into account the area covered by each image [2].

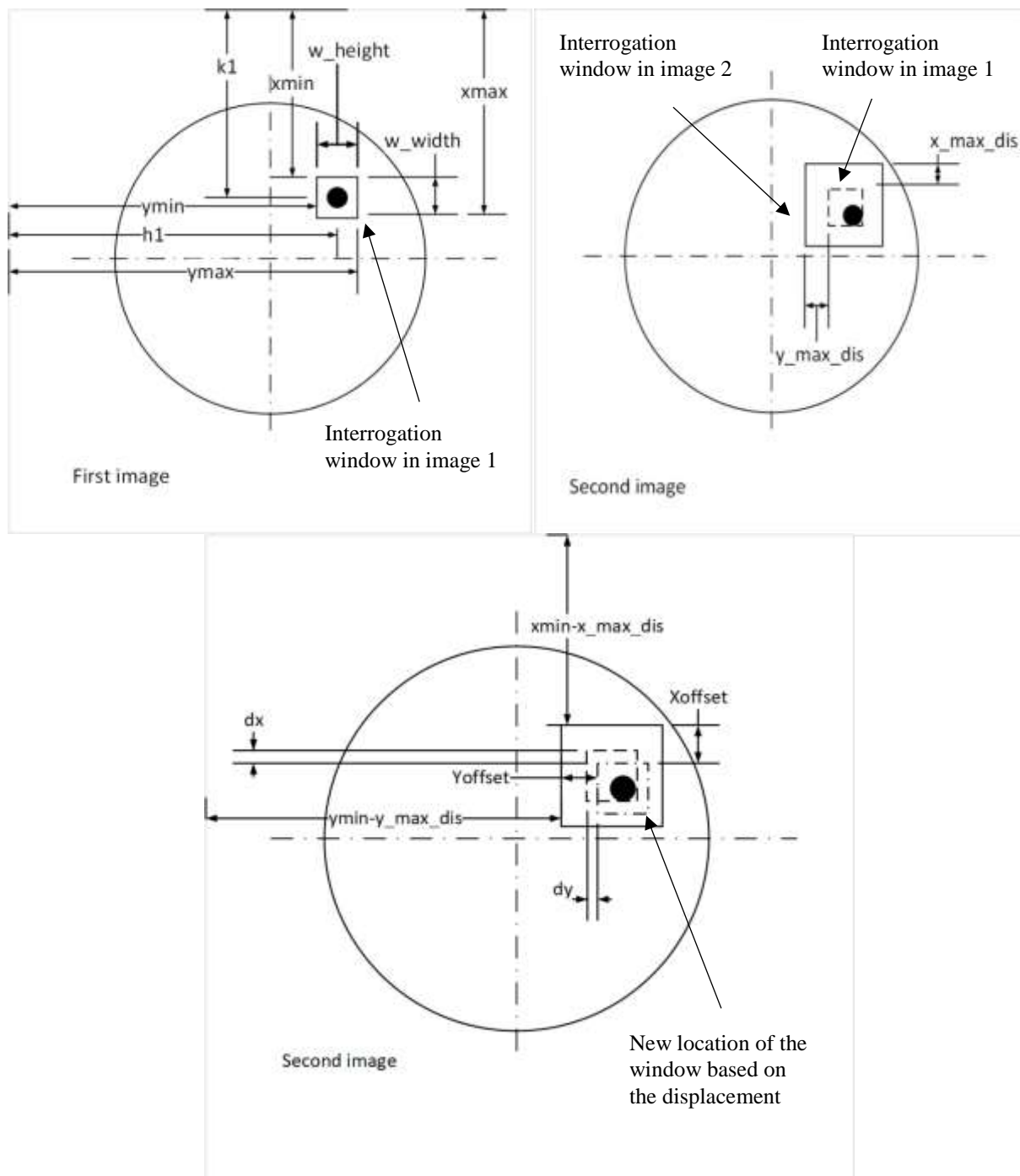


**Figure 11. Cross-correlation PIV.** Multiple binary images are spaced at a defined time step  $dt$ . A cross-correlation PIV algorithm is applied between the images that examine many interrogation windows (nine interrogation windows were chosen to describe the process, but the number chosen is application specific) to determine the overall velocity direction and magnitude for each interrogation window. (a) An interrogation window example is displayed in grey and when the algorithm is implanted, a resulting velocity direction and magnitude is assigned for that interrogation window. (b) A velocity map from a cross-correlation PIV.

### 3.2.3. Particle tracking algorithm

A particle tracking algorithm for the current project was developed by utilizing `imfindcircles` and `normxcorr2` functions. A MATLAB script was written that reads a pair of images with a known time gap between the recorded images. The script performs normalized 2D cross-correlation to track the movement of particles between the images. First, the MATLAB script uses `imfindcircles` function to detect the circular particles and then creates interrogation windows centered around the centers of those detected circles to track the movement of particles by using the `normxcorr2` function.

Figure 12 depicts the detection of a circular particle in the binary images and the tracking of its movement. The circular particle in the first image is detected by using the `imfindcircles` function. An interrogation window is then set up about the center of detected circle in both the image pairs. The size of the window created in the first image is twice the diameter of the detected circle in this example, but is equal to the diameter of the detected circle in the actual code. To allow for a multiple search area, size of the interrogation window in the second image is a variable. The size of the window in the second image is twice that in first image in the current example. Cross-correlation between the two windows of the two images is computed using the `normxcorr2` function.  $x_{peak}$  and  $y_{peak}$  values can be extracted as the coordinates of pixel point in the window of second image where there is maximum correlation coefficient. These  $x_{peak}$  and  $y_{peak}$  can then be used to calculate  $x_{offset}$  and  $y_{offset}$  values by using Equation 8 and Equation 9. A detailed explanation of the calculation of displacement in both the x and y directions is presented in the appendix A.



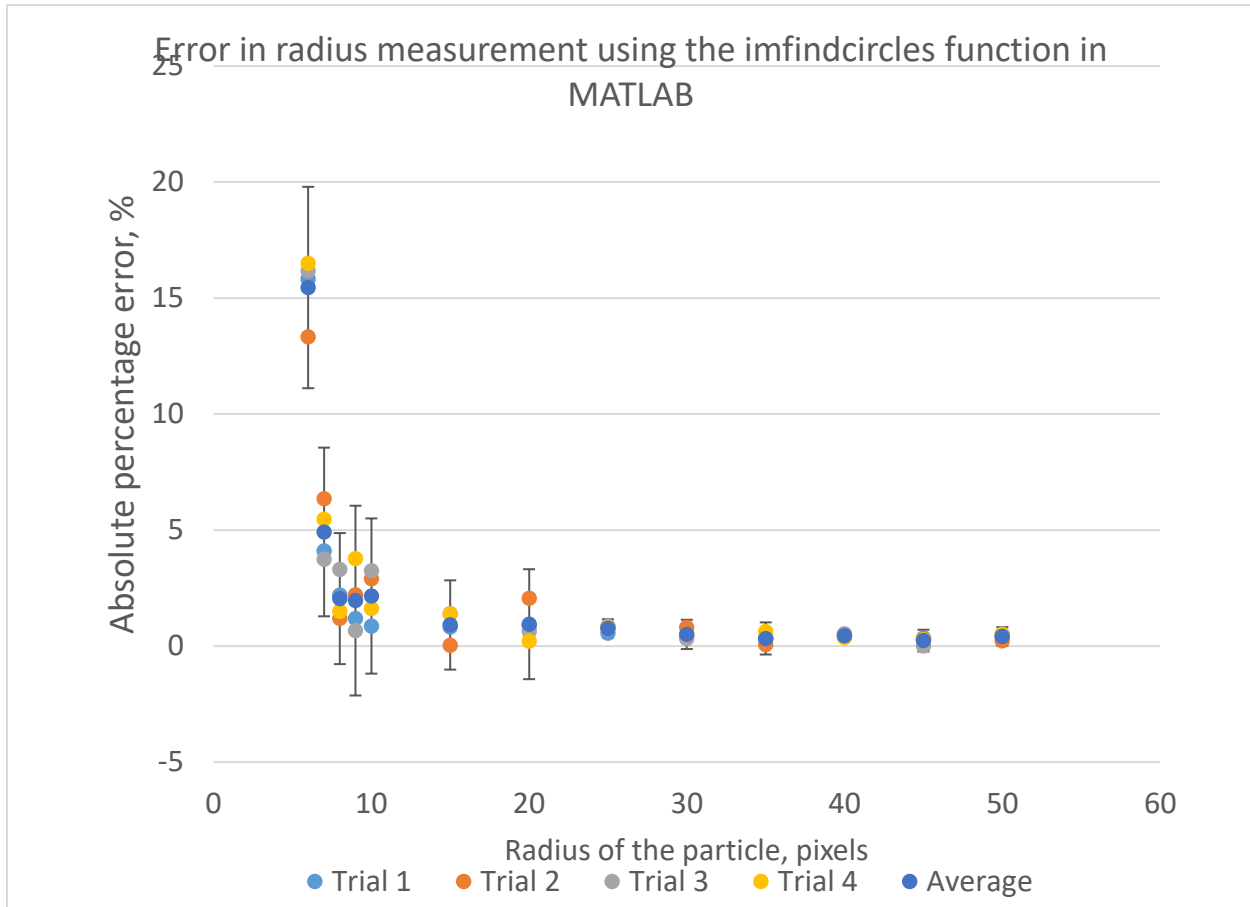
**Figure 12. Tracking of an example particle. The circular particle in first image is detected by using the `imfindcircles` function. An interrogation window is then set up about the center of detected circle in both the image pairs. The size of the window created in the first image is twice the diameter of the detected circle. The size of window in the second image is twice that in the first image in this example to allow for a bigger search area in the second image. `Normxcorr2` function computes cross correlation between the two windows and the point of maximum correlation coefficient can be extracted.**

### **3.2.4. Accuracy of imfindcircles**

A study to quantify the errors in the detection of radii and cartesian coordinates of center of circular objects in a binary image by using the imfindcircles function was conducted. The model coordinates of the center and radii of circular particles were compared to the cartesian coordinates of the centers and radii that were detected by using imfindcircles function to find the error of detection. The measurement error was determined for the variations in particle size and number of particles analyzed per melt pool. Particles were modeled as circles for this analysis.

#### **3.2.4.1. Effect of the particle size**

The error in the detection of radius of a circle particle decreases as the size of the particle is increased. A circle must have a radius greater than 6 pixels to be detected by the imfindcircles function. The absolute percentage error in the radius detection of a circle dropped from about 15% (on average) when the radius of circle was 6 pixels to about 0 – 0.5% when the radius of the circle was increased to 50 pixels, as displayed in Figure 13. The absolute values of the average percentage error were plotted against the radius of each particle in Figure 13 to examine the effect of changes in size of the particles that are detected by using imfindcircles function on the measurement error. Error bars of 3 standard deviations (SD) from the mean were used in the plot. All the observations lie within the error bar.



**Figure 13. Absolute percentage error in the radius measurement of circular particles by using the imfindcircles function in MATLAB as a function of the particle size. High percentage error was encountered when small circles were detected.**

The absolute percentage error in the detection of x and y coordinates of the centers of circular particles do not change significantly as a function of the size of the particles, depicted in Figure 14 and Figure 15, respectively. Therefore, average value of the absolute percentage error remained almost constant even though some observations had high error of measurement. Error bars of 3 SD from the mean error were also used for the plots of Figure 14 and Figure 15, indicating that all the observations fall within that limit. The absolute percentage errors in the detection of x and y coordinates of the centers of circles using imfindcircles function were not higher than  $\pm 1\%$  for all the cases that were studied.



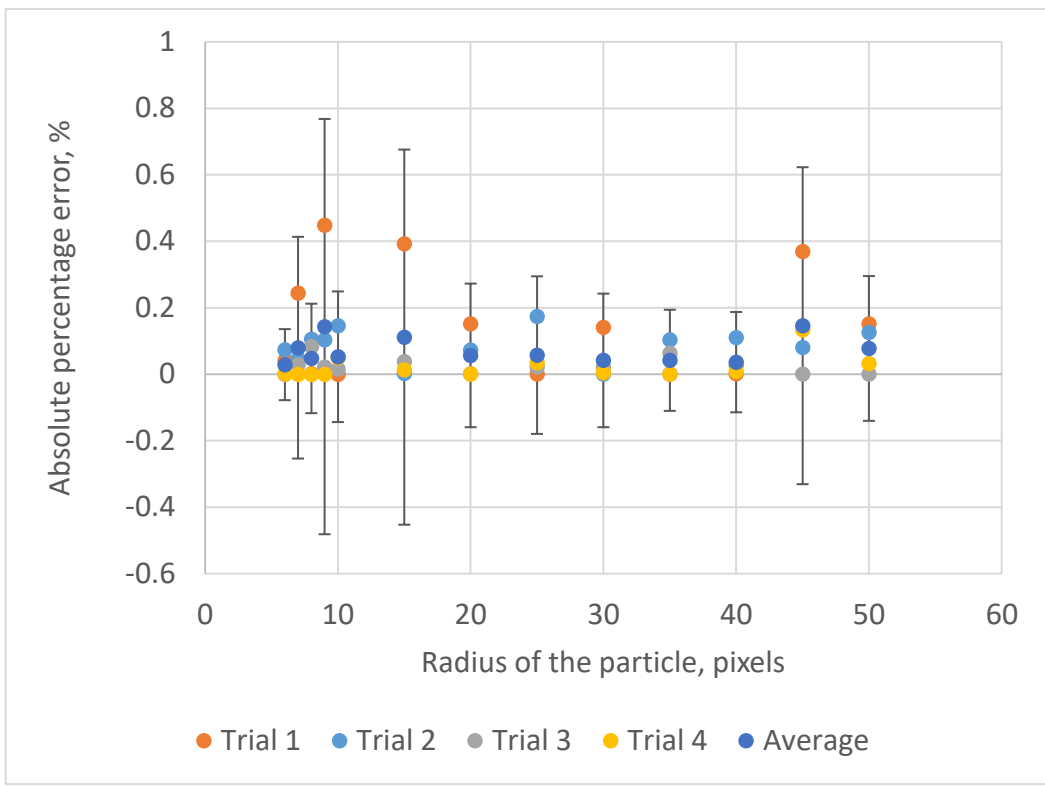


Figure 14. Absolute percentage error in the detection of x-coordinates of the centers of circular particles by using the `imfindcircles` function in MATLAB as a function of the particle size.

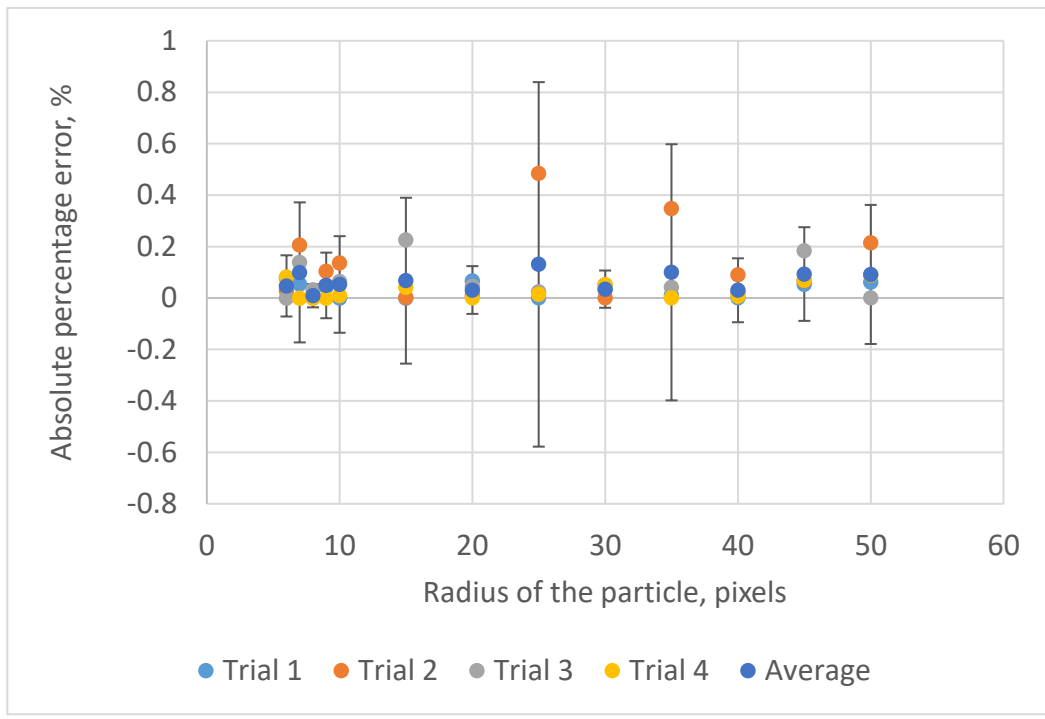
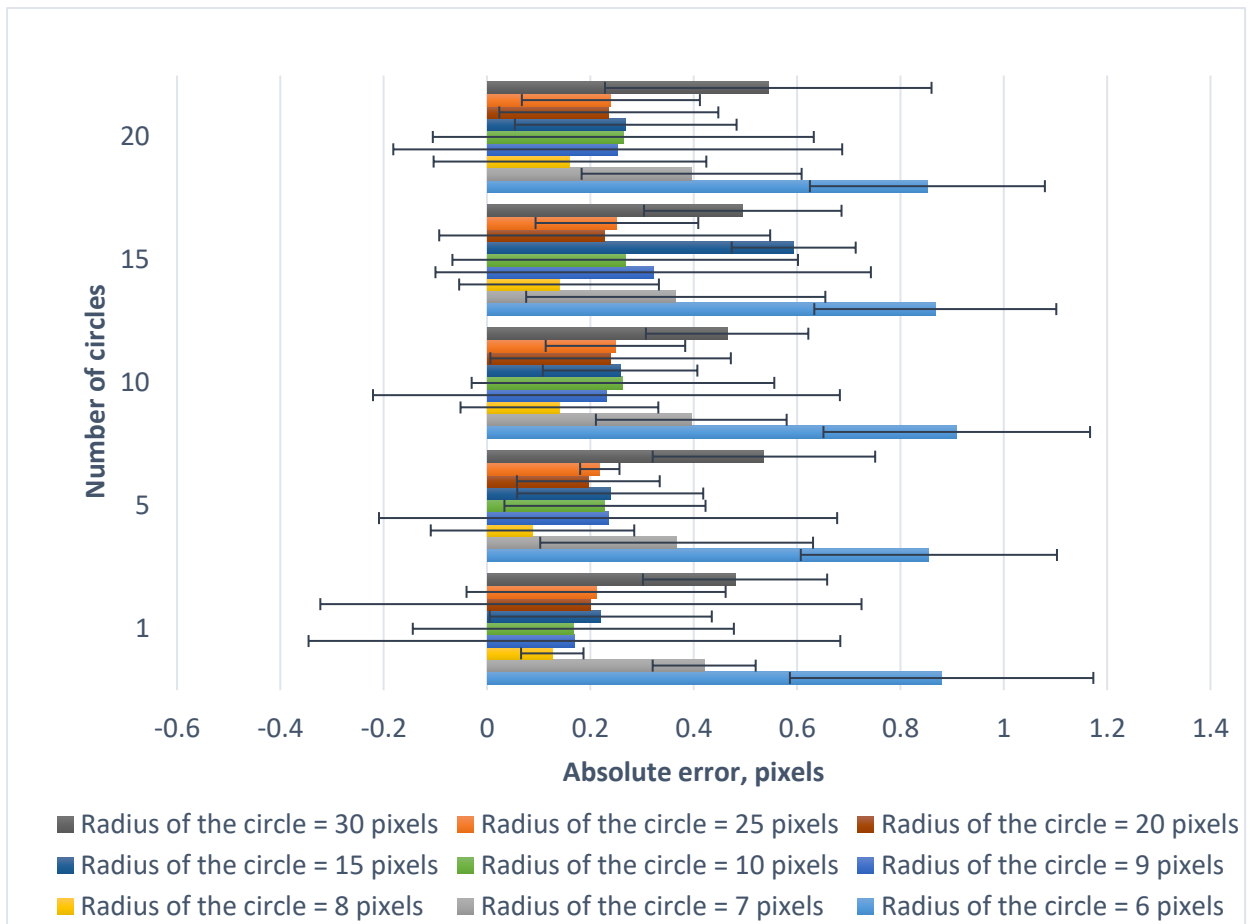


Figure 15. Absolute percentage error in the detection of y-coordinates of the centers of circular particles by using the `imfindcircles` function in MATLAB as a function of the particle size.

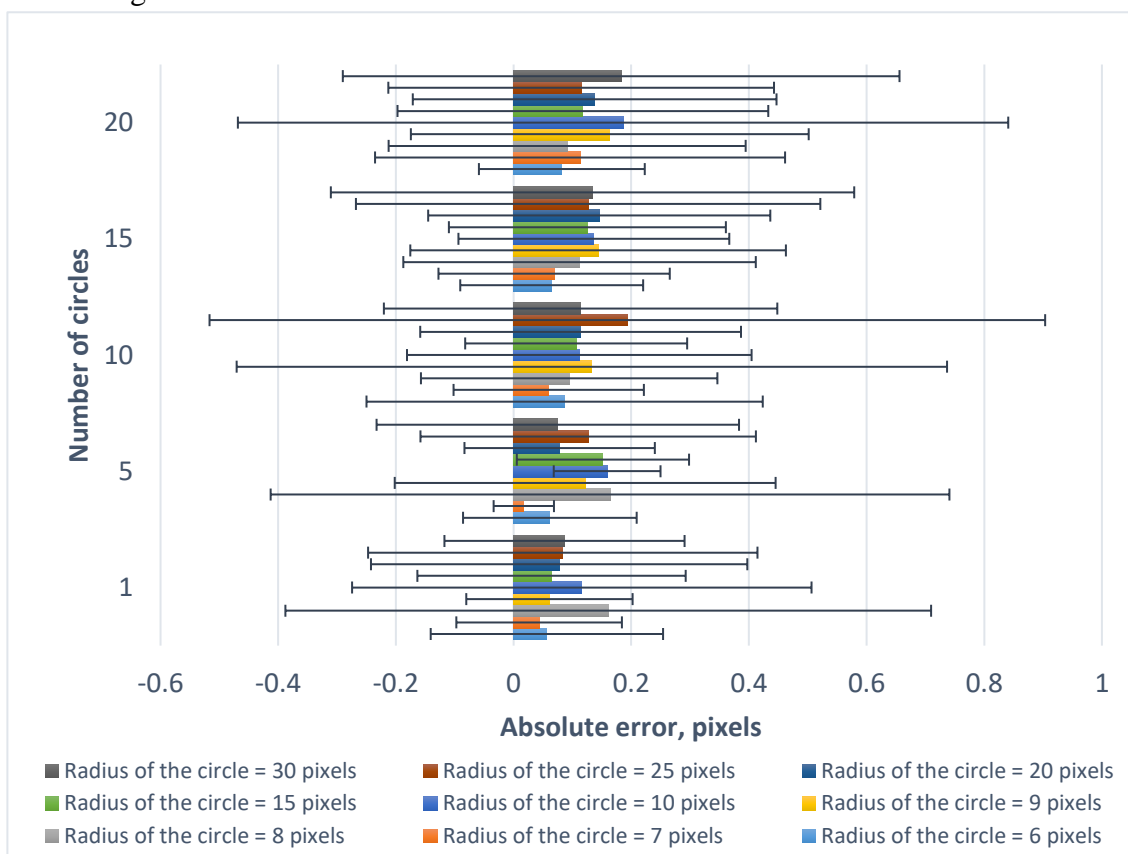
### 3.2.4.2. Effect of the number of particles

The number of circular particles had no apparent effect on the error of radius detection of circular particles when using `imfindcircles` function in MATLAB. Figure 16 displays a bar graph of absolute error of radius measurement of various sized circular particles as a function of the number of detected circles. The bar graph demonstrates that the absolute error of radius detection remains almost constant with the changes in the number of detected circles. However, the detection of circles of 30 pixels in radius yielded high absolute error of radius measurement in spite of having a low absolute percentage error as discussed in the previous section.

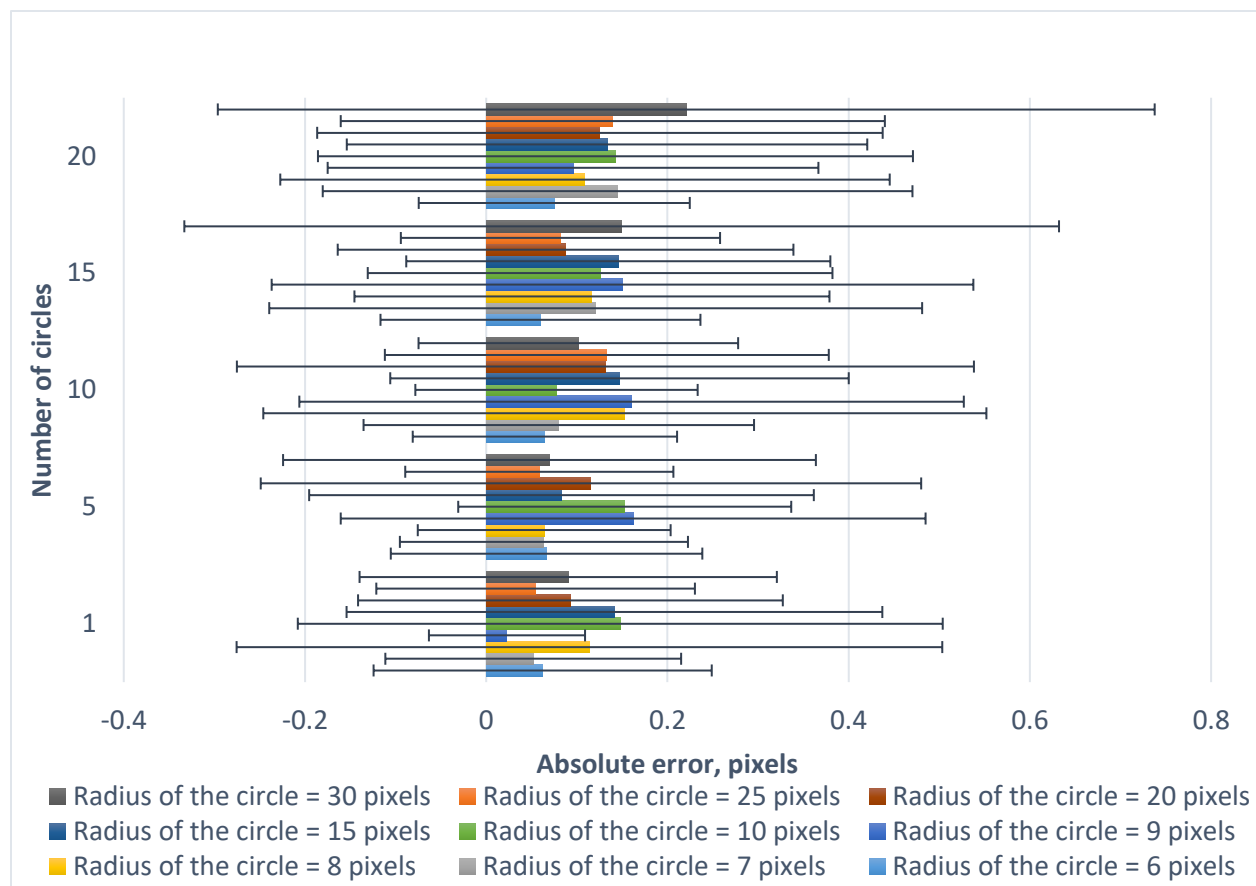


**Figure 16. Absolute error of radius detection as a function of the number and the size of circular particles. The absolute error was mostly unaffected by the changes in the number of particles. Error bars are 3 SD from the mean error.**

The number of circles did not have a significant influence on the error of detection of x and y coordinates of the centers of circles. However, the absolute errors in the detection of x and y coordinates of the center seem to be increasing slightly when the number of detected circles was increased from 1 to 20. Figures 17 and Figure 18 display bar graphs that plot the absolute error in the detection of x and y coordinates of the center of the circles as a function of the number and size of detected circles, respectively. Figures 17 and Figure 18 also depict the average absolute error to be less than 1 pixels in the detection of x and y coordinates when using the `imfindcircles` function. Therefore, the `imfindcircles` function can be used to accurately detect the location of circular surface oxide particles in a binary image of simulated melt pool. Correct detection of the coordinates of the center is necessary as the interrogation window to track these particles using cross-correlation is centered around those detected coordinates.



**Figure 17. Absolute error in detection of x-coordinates of the center of detected circles as a function of the number and the size of circles. The absolute error was unaffected by the changes in the number of particles. Error bar is 3 SD from the mean error.**



**Figure 18. Absolute error in detection of y-coordinates of the center of detected circles as a function of the number and the size of circles. The absolute error was unaffected by the changes in the number of particles. Error bar is 3 SD from the mean error.**

### 3.3. Test plan

The binary images of the melt pool of SLM were simulated under various levels of each factor predicted to affect the displacement measurement of the particle tracking algorithm. Multiple images were simulated for the same condition and compiled into a video representing high-speed visualization of a melt pool during SLM. These images were analyzed using a MATLAB script for particle tracking to determine the displacement of particles in between the two successive images in a video. The goal was to calibrate the particle tracking code by analyzing the difference between the actual displacement in the binary image model and the displacement measured by the MATLAB code.

Binary images of the melt pool were simulated using MATLAB scripts. The images were created with different levels of each factor that might affect the particle tracking algorithm. Size of the melt pool, size of surface oxides, image pixel size, number of surface oxides, and flow paths were varied in different levels during the binary image simulation, as depicted in Table I. Size of the melt pool is a critical parameter because other parameters such as the size and the number of surface oxides will change with the size of the melt pool. The SLM setup had the laser running without a galvo (galvanometer) to control its motion. A focusing lens was used to focus laser beams of 5,000  $\mu\text{m}$  diameter to less than 2,000  $\mu\text{m}$  onto the metallic powder to create a melt pool with a diameter of about 2,000  $\mu\text{m}$ . The visualization experiments were performed at an image scale of 8.9  $\mu\text{m}/\text{pixels}$ . The inhouse-built SLM machine is being equipped with a rotating mirror galvanometer to control the size and motion of the laser beams. The melt pool size will be shrunk to about 200  $\mu\text{m}$  because of a smaller incident laser beam and an image scale of 2.2  $\mu\text{m}/\text{pixel}$  will be used for the high-speed imaging. Therefore, only two melt pool sizes of 200  $\mu\text{m}$  and 2,000  $\mu\text{m}$  were used in the image simulation.

**Table I. Factors that were varied in the simulated binary images of the melt pool.**

<b>Factors</b>	<b>Levels</b>
Size of the melt pool (diameter)	200 and 2,000 $\mu\text{m}$
Size of surface oxides (diameter)	12– 24 $\mu\text{m}$
Image pixel size (matrix size)	100 x 100 – 1000 x 1000
Number of oxide particles	5 – 20
Flow paths	Linear and Circular

The surface oxides in a melt pool can be of different sizes. Therefore, the size of surface oxides is not a constant value in the image simulation rather a random variable between the minimum and maximum size. Random numbers were used to generate various radii of the circular particles in the simulated binary image. The radii of surface oxide particles were varied from 6 to 12 pixels when the size of melt pool was 2,000  $\mu\text{m}$ , and from 6 to 9 pixels when the size of melt pool was 200  $\mu\text{m}$ . The radius range had to be decreased for the 200  $\mu\text{m}$  melt pool as all the particles clustered in the center due to a smaller melt pool size and created undesirable error in the particle tracking.

### **3.3.1. Scaling of the images**

The relative sizes of melt pool, image pixels, and surface oxides were using scaling factors that would represent the actual melt pool during SLM. A scale of 8.9  $\mu\text{m}$  /pixels for a 2,000  $\mu\text{m}$  melt pool and a scale 2.2  $\mu\text{m}$ /pixels for a 200  $\mu\text{m}$  melt pool were used to determine the relative size range of all other factors. The radius of a circle correctly detected by the ‘`imfindcircles`’ function in MATLAB must be greater than 6 pixels. Therefore, the minimum diameter of the surface oxides in the simulated binary images was set to 12 pixels. Image pixel size range was varied based on the size of melt pool in pixels. Table II shows the scaling calculation performed for the test plan that is divided into two sections based on the melt pool size.

**Table II. Scaling of the simulated binary images.**

Melt pool size, $\mu\text{m}$	Melt pool size, pixels	Image pixel size	Size of surface oxides, $\mu\text{m}$ (diameter)		Size of surface oxides, pixels (diameter)		Number of surface oxides
			Min.	Max.	Min.	Max.	
2,000	298.5	300 x 300 - 1,000 x 1,000	106.8	213.6	12	24	5 - 20
200	90.9	100 x 100 - 500 x 500	26.4	39.6	12	18	5 - 20

The test plan was optimized using an experimental software design package “Design Experts 12”. The package offers various design of experiments modules to maximize the resolution of experimental design space with minimum number of experiments.

### 3.3.2. Optimization of experiments

The experiments were divided into two sections for each of the melt pool sizes (200  $\mu\text{m}$  and 2,000  $\mu\text{m}$ ) to simplify the optimization. A central composite design (CCD) in Design Experts 12 that maximizes the resolution of experimental design space and minimizes the number of experiments was used for the optimization. The first section of experiments was designed to evaluate the effects of the image pixel size, the number of surface oxides, and the flow path on the displacements measured by the particle tracking algorithm for a melt pool size of 2,000  $\mu\text{m}$ . Five different levels of the image pixel size and the number of surface oxides were varied for 2,000  $\mu\text{m}$  melt pool during two different flow paths, as shown in Table III. A 2-factor rotatable CCD was used to design the experiments that contained 13 tests with 5 repetitions for each flow pattern resulting in a total of 26 tests for the first section.

The second section of experiments was designed to analyze the effects of the image pixel size, the number of surface oxides, and flow paths on the displacement measured by the particle tracking algorithm for a melt pool size of 200  $\mu\text{m}$ . Five different levels of the image pixel size and the number of surface oxides were used for 200  $\mu\text{m}$  melt pool during two different flow paths, as seen in Table III. A 2-factor rotatable CCD was used to design the experiments that contained 13 tests with 5 repetitions for each flow pattern resulting in a total of 26 tests for the second section. The total number of experiments needed to be performed equals to 52 for both sections.

**Table III. Factors that were varied for both the sections of the experimental design.**

<b>Section</b>	<b>Size of the melt pool, <math>\mu\text{m}</math></b>	<b>Size range of surface oxides, pixels</b>	<b>Image pixel size</b>	<b>Number of surface oxides</b>	<b>Flow paths</b>
First	2,000	12 – 24	300 x 300 – 1,000 x 1,000	5 – 20	Linear and circular
Second	200	12 – 18	100 x 100 – 500 x 500	5 – 20	Linear and circular

### **3.3.3. Independent studies**

The size of surface oxides was not a constant factor in the experiments previously discussed and therefore its effect on the particle tracking algorithm can't be assessed based on the previous tests. To analyze the effect of particle size, a separate independent study was conducted using optimized parameters from the initial study. Furthermore, the experiments previously designed didn't have velocity of particles in between each image pair as a factor in



the study. Results from the model analysis suggested that velocity of the particles might have some influence on the displacements measured using the particle tracking algorithm. Therefore, the optimum melt pool size, image pixel size, flow type, and number of particles were used in a 2-factor face centered CCD experimental design to analyze the effects of the size of surface oxides and the velocity of particles in between each image pair on the displacement measured using the particle tracking algorithm. The size of the surface particles and the velocity of the particles in between an image pair were varied on three levels. Low-range indicated that the radii of the surface oxides varied from 6 – 8 pixels, mid-range indicated that the radii of the surface oxides varied from 8 – 10 pixels, and high-range had radii of the surface oxides from 10 – 12 pixels. The velocities of surface oxides were chosen to be 2, 6, and 10 pixels/frame in the simulated images.

Another independent study was necessary to make conclusions from the results of all the previously conducted experiments. A 2-factor face centered CCD was used to design experiments and to analyze the effects of the size and the number of surface oxides on the displacement measured using the particle tracking algorithm at a constant velocity. Three different levels of both the particles size and the number of surface oxides were used in the experimental design. The size of the surface oxides was varied in low-range, mid-range, and high-range as previously described. The numbers of surface oxides used in the experiments were 5, 13, and 20.

## 4. Results and discussion

The simulated binary images were analyzed using a particle tracking algorithm written as a MATLAB script using the 'imfindcircles' and the 'normxcorr2' functions, as displayed in Figure 19. The MATLAB script for particle tracking reads two images to be analyzed, detects circular objects in the images that have a certain user defined radius range, and then creates interrogation windows for both the images. The cartesian coordinates of the center and the radius of each circular object were detected by the 'imfindcircle' function in MATLAB. The center of the detected circles in the first image is the center of windows in both the images. The size of the window in the first image of the image pair is equal to the diameter of each detected circle whereas the size of the window in the second image is a variable to allow for multiple search area in the second image.

The 'normxcorr2' function then computes the normalized 2D cross-correlation between the two windows of the image pair. The peak of cross-correlation was used to calculate the displacement of the particles in between the image pair. Then, the quiver function in MATLAB was used to create a vector plot depicting the magnitude and direction of displacements of each object in the binary image pair. The displacements generated by the execution of the MATLAB script were compared to the actual displacements used in the model to simulate those binary images.

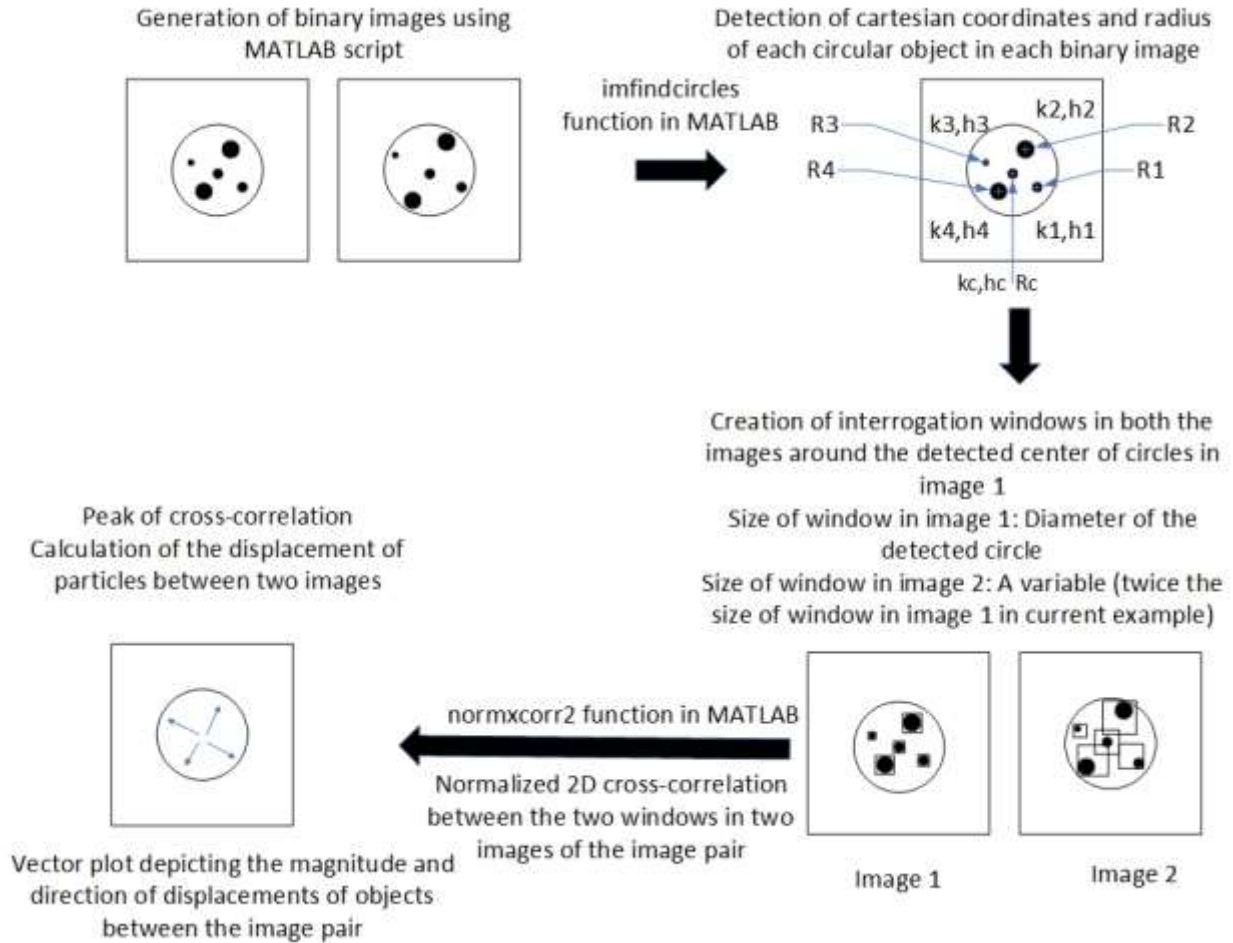
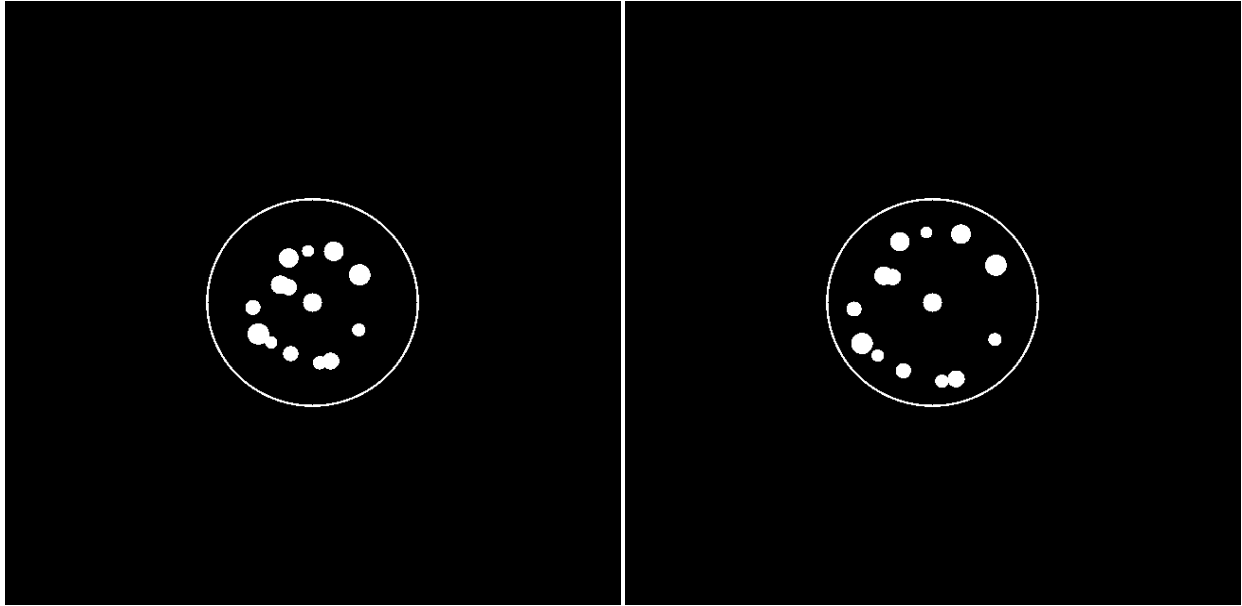


Figure 19. Hierarchical image representing the flow of work for the project.

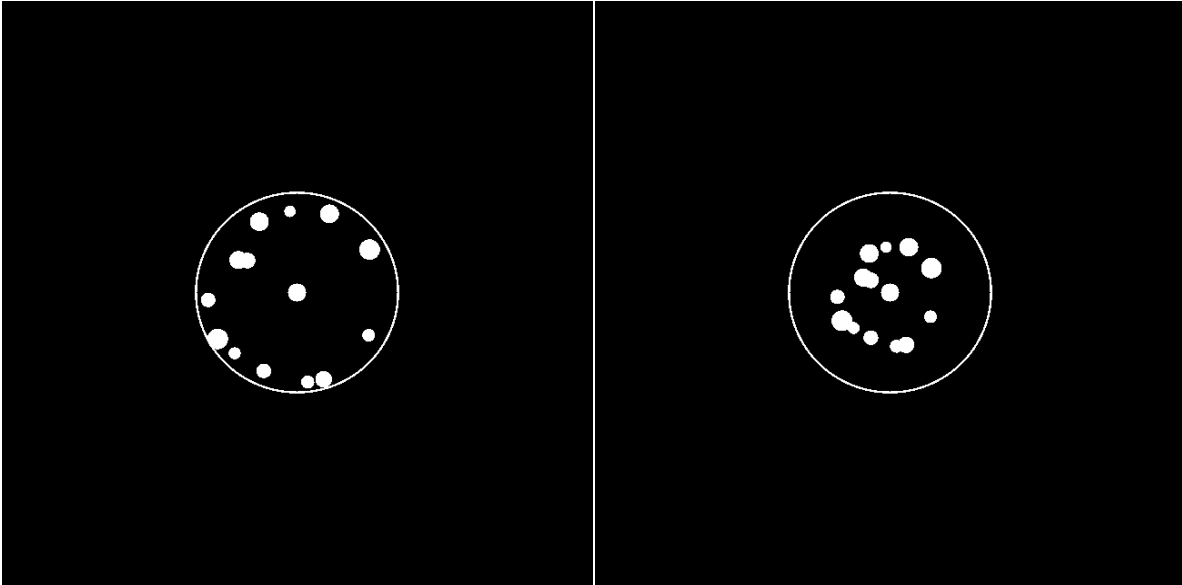
#### 4.1. Binary image simulation

Binary images were simulated using MATLAB scripts to represent a surface flow during SLM. Multiple images were compiled to form a movie portraying a linear or a circular motion of surface particles in a melt pool. A total of 16 images were simulated and then compiled into a video to illustrate various surface flows in a melt pool of SLM. Figure 20 displays a pair of images where the particles have a linear outward displacement among each other. The velocities of surface particles used in the binary image model for radial outward motion were 5 pixels/frame and 2 pixels/frame for the melt pool size of 2,000  $\mu\text{m}$  and 200  $\mu\text{m}$ , respectively.



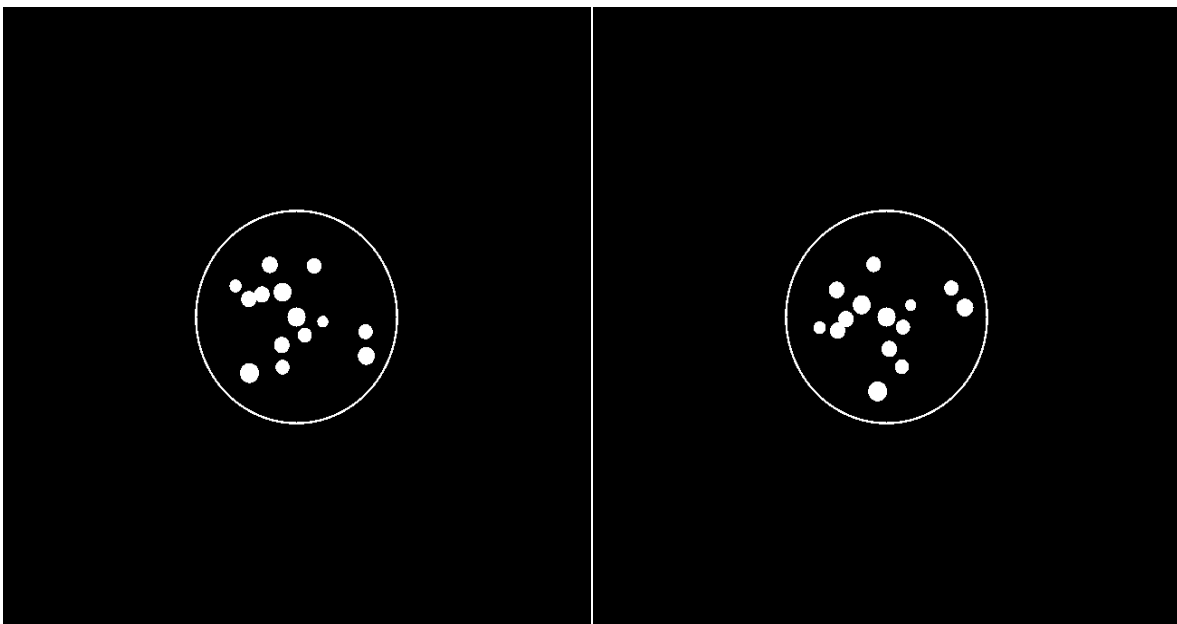
**Figure 20. A pair of binary images that were simulated to represent a linear radial outward motion**

In an actual melt pool, the surface flow starts with a radial outward flow and then the flow transitions to a radial inward flow. Therefore, out of the 15 pairs of consecutive simulated binary images for a linear flow, 12 of them had a radial outward flow of particles whereas the last 3 pairs depicted a radial inward flow of particles. The velocity of radial inward flow can be about five times the velocity of radial outward flow [30]. The model velocities of surface particles during radial inward motion were 20 pixels/frame and 8 pixels/frame for the melt pool size of 2,000  $\mu\text{m}$  and 200  $\mu\text{m}$ , respectively. A radial inward motion is depicted in Figure 21 by using a pair of simulated binary images.



**Figure 22. A pair of binary images that were simulated to represent a linear radial inward motion**

Binary images were also simulated with a circular motion of particles among them. The direction of circular motion and its angular velocity were constant for each pair of images. Similar to the linear motion, 16 images were simulated to represent a circular motion, as depicted in Figure 22. The particles moved in a counterclockwise circular motion with an angular velocity of 0.2 radians/frame.



**Figure 21. A pair of binary images that were simulated to represent a circular motion**

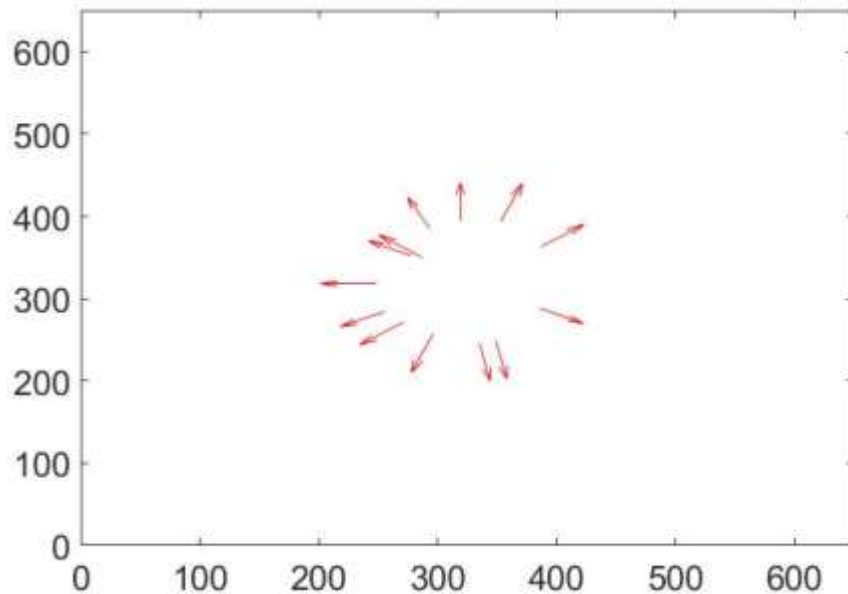
## 4.2. Particle tracking

The simulated images were analyzed using a particle tracking algorithm created as a MATLAB script. The particle tracking algorithm detects both the magnitude and direction of the displacements of circular objects in a pair of images. The magnitude and direction of the displacements of each circular object were then illustrated by using a vector plot. The calculated displacements were compared to the model displacements used in the binary image generation to calibrate the particle tracking code.

### 4.2.1. Vector plots

Vector plots were used to depict the relative magnitude and direction of displacements of particles in a pair of binary images analyzed by using the particle tracking MATLAB script. The

quiver function in MATLAB was used to create vector plots of the calculated displacements. The linear radial outward motion in Figure 20 was tracked and displayed as vector plots in Figure 23, which revealed that the particles had a constant

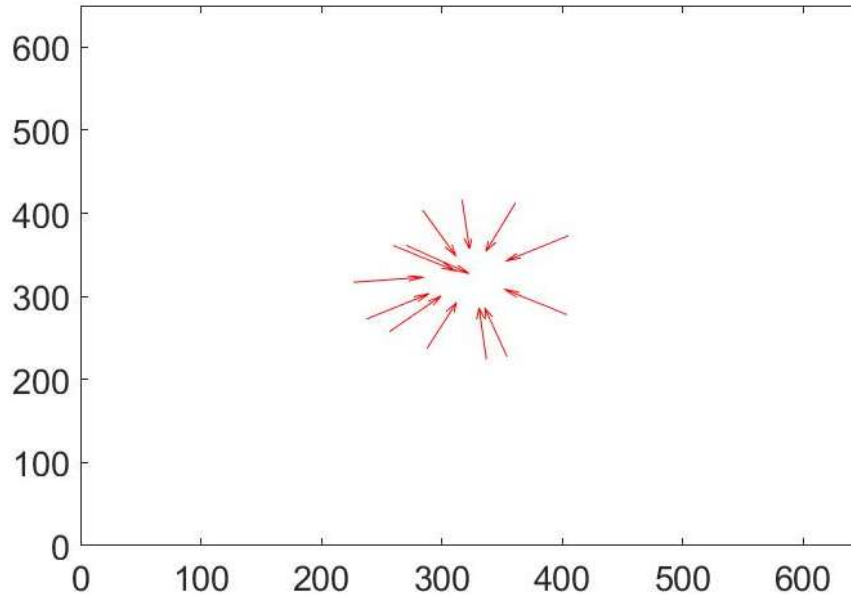


**Figure 23. Result from the particle tracking algorithm: radial outward motion displayed as a vector plot. The particles had a constant linear displacement in a radial outward direction.**

magnitude of displacements in a radial outward direction (away from the center of melt pool).

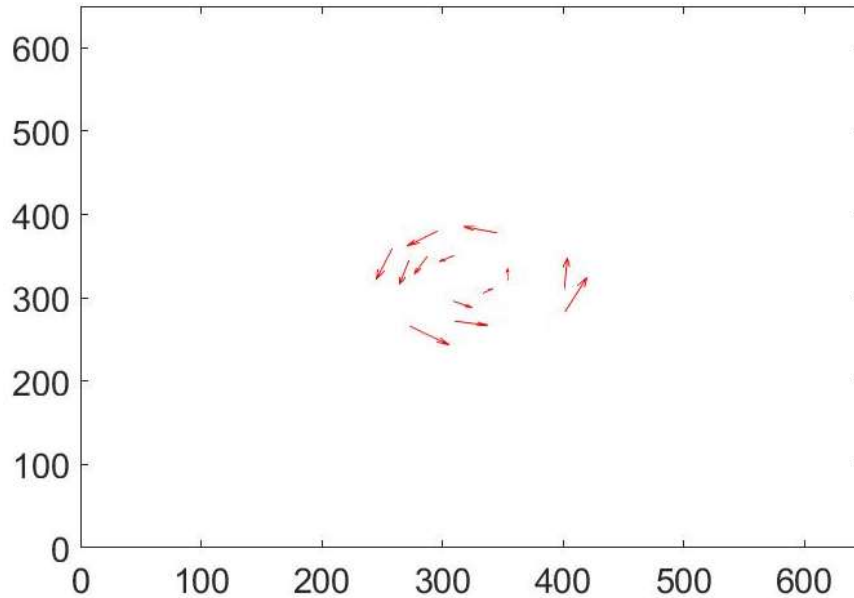
The linear motion of the particles in the melt pool was also tracked during the radial inward motion. Figure 24 displays a vector plot generated by tracking the pair of binary images

in Figure 21 demonstrating a radial inward motion. The vector plot revealed that the particles had a constant magnitude of radial inward displacements moving towards the center of the melt pool.



**Figure 24. Result from the particle tracking algorithm: radial inward motion displayed as a vector plot. The particles had a constant linear displacement in a radial inward direction.**

Circular motion of the particles was tracked using the particle tracking algorithm. The circular motion represented by a pair of binary images in Figure 22 was tracked and displayed as a vector plot in Figure 25. In a circular motion, the linear displacement of the particles increases with an increase in the distance of the particle from the center of rotation. The vector plot revealed large magnitude of displacements for particles that are away from the center and small magnitude of displacements for particles that are near the center of rotation.



**Figure 25. Result from the particle tracking algorithm: circular motion displayed as a vector plot. The particles had a varying linear displacement in a counterclockwise direction. The linear displacements of particles were larger when the particles were away from the center of rotation.**

#### **4.2.2. Displacement measurement**

The displacement of each surface oxide particle was calculated by identifying the pixel location with the highest of cross-correlation coefficient. The linear displacement in pixels during the linear radial motion was calculated using equations explained in detail in the Appendix A.

Angular displacements were calculated for the circular motion because the particles have a varying linear displacement during circular motion. The linear displacement of each particle during circular motion was calculated similar to that during a linear motion and then converted to angular displacements in radians. The calculated linear displacements were used to determine the new cartesian coordinates of the centers of the surface oxides after they moved in a circular motion in between the image pair. Then, the angles subtended by the initial location of surface particles and the final location of surface particles at the center of the melt pool on the horizontal



axis passing through the center were calculated; the difference between the two angles is the angular displacement of the particles.

The binary images were simulated based on the experiments designed using the Design Expert 12. A total of 52 experimental runs were completed with different levels of each factor. The melt pool sizes and flow paths were considered as categorical factors whereas the image pixel size and the number of surface oxides were considered numeric factors in the CCD design of experiments. The experimental design for the linear flow was divided into radial outward motion and radial inward motion because of the difference in the displacements of particles during two different motions. An error threshold of 20% was considered to be acceptable for the displacement measurement in both the linear and circular motion.

#### **4.2.2.1. 2,000 $\mu\text{m}$ melt pool**

The experiments were first designed for the melt pool size of 2,000  $\mu\text{m}$  with two numeric factors i.e. the image pixel size and the number of surface oxides. The experiments were repeated for radial outward motion, radial inward motion, and circular motion because flow type was a categorical factor in the experimental design.

The result of the displacement measurement for a melt pool size of 2,000  $\mu\text{m}$  is presented in Table IV. The probability that the error of displacement measurement was greater than or equal 20% was zero for all the runs during the linear radial outward motion. The probability of error was mostly zero during the linear radial inward motion but the run 9 with highest number of particles had a non-zero probability of error, which might have occurred because of clustering of particles. However, during a circular motion, the probability that the error of measurement was greater than or equal 20% was non-zero, but was randomly distributed and less than 10% in all the cases.

**Table IV. Probability that the error of measurement is greater than or equal to 20% for a 2,000  $\mu\text{m}$  melt pool.**

Run	Factor 1 A: Image pixel size (pixels)	Factor 2 B: Number of surface oxides	Linear motion		Circular motion
			Radial outward motion	Radial inward motion	
1	400 x 400	18	0.00%	0.00%	7.05%
2	650 x 650	13	0.00%	0.00%	6.15%
3	300 x 300	13	0.00%	0.00%	5.13%
4	650 x 650	13	0.00%	0.00%	6.15%
5	650 x 650	5	0.00%	0.00%	5.33%
6	900 x 900	18	0.00%	0.00%	5.53%
7	400 x 400	8	0.00%	0.00%	5.83%
8	1000 x 1000	13	0.00%	0.00%	5.21%
9	650 x 650	20	0.00%	2.70%	4.92%
10	900 x 900	8	0.00%	0.00%	5.83%
11	650 x 650	13	0.00%	0.00%	6.15%
12	650 x 650	13	0.00%	0.00%	4.62%
13	650 x 650	13	0.00%	0.00%	5.64%

#### 4.2.2.2. 200 $\mu\text{m}$ melt pool

The levels of the numeric factors varied with the melt pool size. Therefore, new experiments were designed for the melt pool size of 200  $\mu\text{m}$  and used for both the linear and circular motion. The probability that the error of measurement was greater than or equal to 20 % was higher for a melt pool size of 200  $\mu\text{m}$  than for the 2,000  $\mu\text{m}$  melt pool, Table V. The probability was significantly higher during the radial outward motion for a 200  $\mu\text{m}$  melt pool with a displacement of 2 pixels in between each successive binary image. However, the probability that the error of measurement was greater than or equal 20 % was zero or low for the radial inward motion that had a displacement of 8 pixels in between each successive binary image. The probability that the error of displacement measurement was greater than or equal to 20% during a linear radial inward motion was highest for run 11. The simulated linear displacements of particles initiate as a radial outward motion from near the center toward the edge of the melt pool and then transition to a radial inward motion when the particles are near the melt pool edge. The image pixels size for run 11 was extremely small and therefore the size of interrogation windows had to be decreased to not allow the particles to cross the image boundary. The surface particles were extremely close to image boundary in the final few binary images representing a radial outward motion, and initial binary image representing a radial inward motion because of the small image pixel size. The probability of error increased when a smaller interrogation window was used. For the circular motion, the probability that the error of measurement was greater than or equal to 20% was distributed in a similar way to that of a 2,000  $\mu\text{m}$  melt pool, but was slightly higher.

**Table V. Probability that the error of measurement is greater than or equal to 20% for a 200  $\mu\text{m}$  melt pool.**

Run	Factor 1 A: Image pixel size (pixels)	Factor 2 B: Number of surface oxides	Linear motion		Circular motion
			Radial outward motion	Radial inward motion	
1	440 x 440	18	26.97%	5.26%	6.84%
2	300 x 300	13	14.12%	0.00%	9.38%
3	300 x 300	20	6.33%	11.11%	10.22%
4	300 x 300	13	25.81%	0.00%	9.74%
5	440 x 440	8	9.09%	0.00%	6.78%
6	300 x 300	13	12.00%	0.00%	12.50%
7	160 x 160	18	12.79%	0.00%	7.30%
8	160 x 160	8	6.94%	0.00%	5.13%
9	300 x 300	5	24.14%	0.00%	5.33%
10	300 x 300	13	11.76%	5.56%	6.04%
11	100 x 100	13	28.95%	5.88%	7.20%
12	500 x 500	13	6.25%	0.00%	13.07%
13	300 x 300	13	11.76%	5.56%	5.33%

#### **4.2.3. Effects of various factors on the displacement measured by the particle tracking algorithm**

A test plan was developed to analyze the effects of various factors (both categoric and numeric) on the displacement measured by using the particle tracking algorithm. Experiments were designed using a CCD in Design Expert 12. A quadratic model analysis was performed

individually for all six models that included models for radial outward and inward motion, and circular motion for the two melt pool sizes. The results from analysis of variance (ANOVA) depicted that the models were not significant for most of the cases with a high p-value.

Models can be designed for data that already exists. Consequently, a historical analysis of the models was performed using all six models simultaneously. Three numeric factors were considered for the historical model: melt pool size, image pixel size, and number of surface oxides. Flow type was the only categoric factor in the historical model with three levels for three different flow types. There were 13 runs in each of the initially designed models and hence the historical model comprised of 78 observations from 6 different models.

The historical analysis revealed that the historical model was significant with a p-value  $<0.0001$ , and melt pool size and flow type were significant factors. Lack of fit of the residual was not significant but had a high F-value and low p-value indicating that the data might have been under fit. The coefficient of variance (CV) was 70.68%, which is a high value even though some inherent errors were expected during the execution of the code. The predicted value for  $R^2$ , 50.5%, was low but close to the adjusted  $R^2$  of 61.78%, which shows reasonable agreement. The box-cox plot for power transform in the model diagnostics recommended a log transform with a  $k$  value of 0.00027.

A natural log transform was used in the historical model with the constant  $k=0.001$  (smallest number that could be input). The model then had three significant factors: melt pool size, flow type, and number of surface oxides. The lack of fit of residuals was not significant with a high p-value of 0.66. The CV was reduced to 20.18% and the predicted  $R^2$  was increased to 81.40%, a value that was very close to the adjusted  $R^2$  of 83.83%. The model was thought to be more efficient to predict the behavior of error of displacement measurement at

various levels of each of the significant factors. The efficiency of the model was evaluated during validation of the model and explained in a later section. The ANOVA result of the historical model after natural log transform are presented in Table VI and Table VII.

**Table VI. ANOVA for reduced 2FI model (historical model).**

Source	Sum of Squares	df	Mean Square	F-value	p-value
<b>Model</b>	319.25	8	39.91	50.9	< 0.0001
A-Melt pool size	91.74	1	91.74	117.01	< 0.0001
C-Number of surface oxides	5.45	1	5.45	6.95	0.0104
D-Flow type	139.44	2	69.72	88.92	< 0.0001
AD	73.62	2	46.95	46.95	< 0.0001
CD	9	2	5.74	5.74	0.0049
Residual	54.1	69	0.7841		
Lack of Fit	33.56	45	0.7458	0.8714	0.6632

**Table VII. ANOVA fit statistics for the historical model.**

Statistic	Value
Std. Dev.	0.8855
Mean	-4.39
C.V. %	20.18
R <sup>2</sup>	0.8551
Adjusted R <sup>2</sup>	0.8383
Predicted R <sup>2</sup>	0.814
Adeq Precision	20.2002

The effects of the melt pool size, image pixel size, number of surface oxide, and flow type on the error of displacement measurement using the particle tracking algorithm are discussed in the sections to follow. The above-mentioned discussions will be followed by model validation that recommended independent studies for further analysis. The effects of size of surface oxides and velocity of particles in between an image pair on the error of measurement using the particle tracking algorithm was quantified from two independent studies that used the optimized parameter from the historical model.

#### 4.2.3.1. Effect of the size of melt pool

The size of the melt pool was found to be a significant factor in the historical analysis of models with a p-value of  $<0.0001$ . The melt pool size also had a significant impact on the error of displacement measured during a linear radial outward flow as displayed in Figure 26. The probability that error of displacement measurement was greater than or equal to 20% was high when the melt pool size was low i.e. 200  $\mu\text{m}$ , and had a broad 95% confidence interval (CI). The

probability of error

was almost zero with a

narrow 95% CI when

the melt pool size was

increased to 2,000  $\mu\text{m}$ .

The probability of

error decayed

exponentially with the

increase in the size of

melt pool. The effect

of the melt pool size

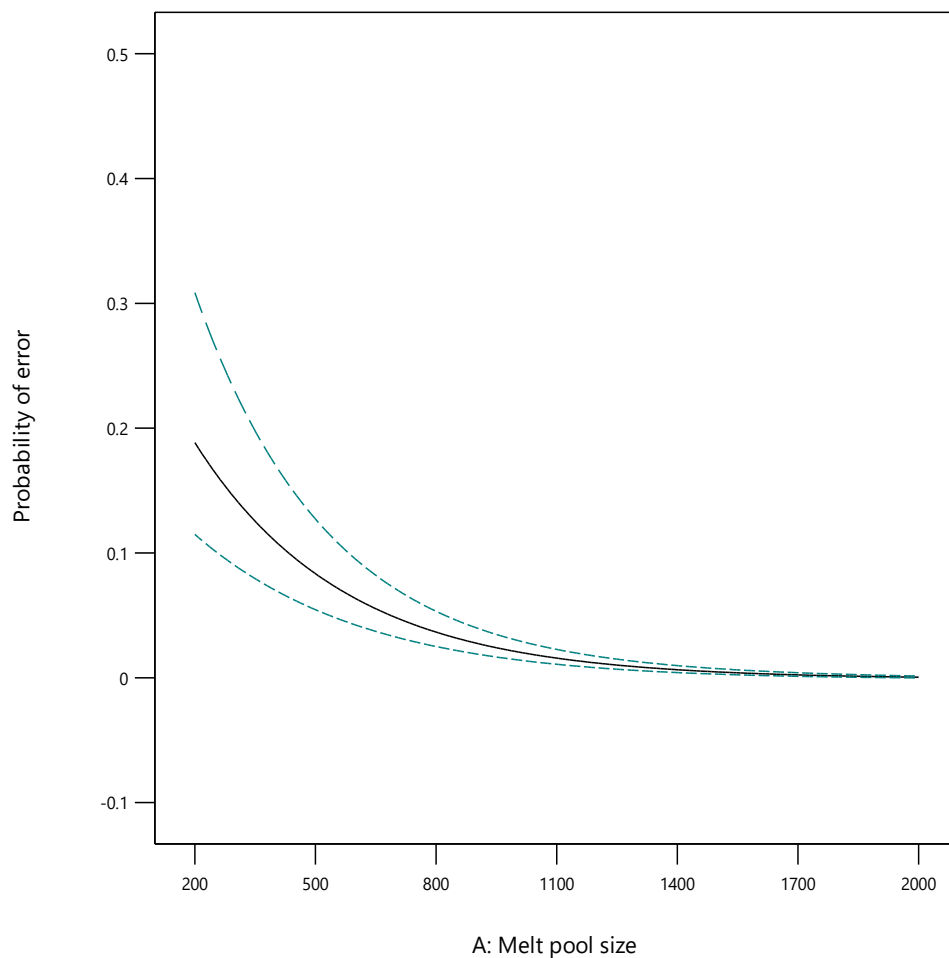
on the error of

displacement

measurement during a

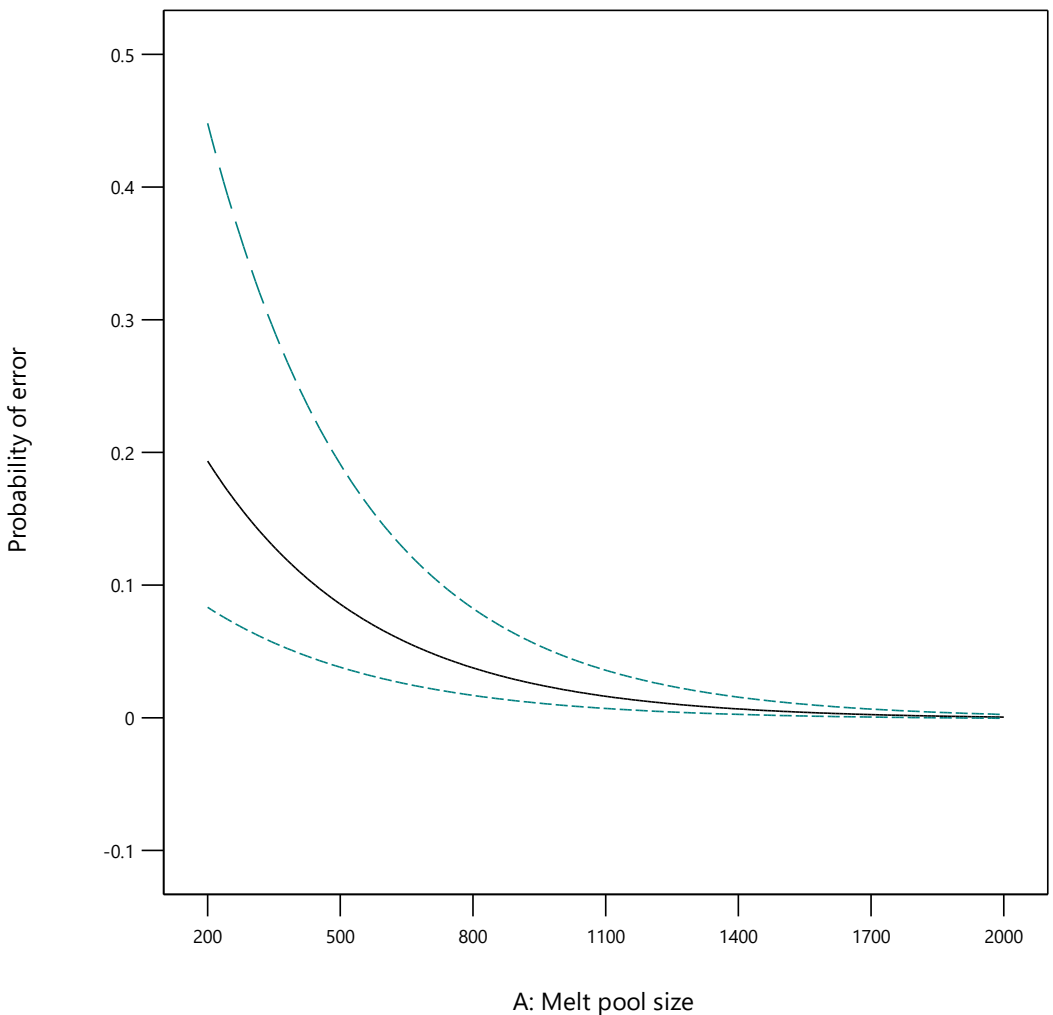
linear radial outward

flow with 13 particles (a mid-range value) is displayed in Figure 26.



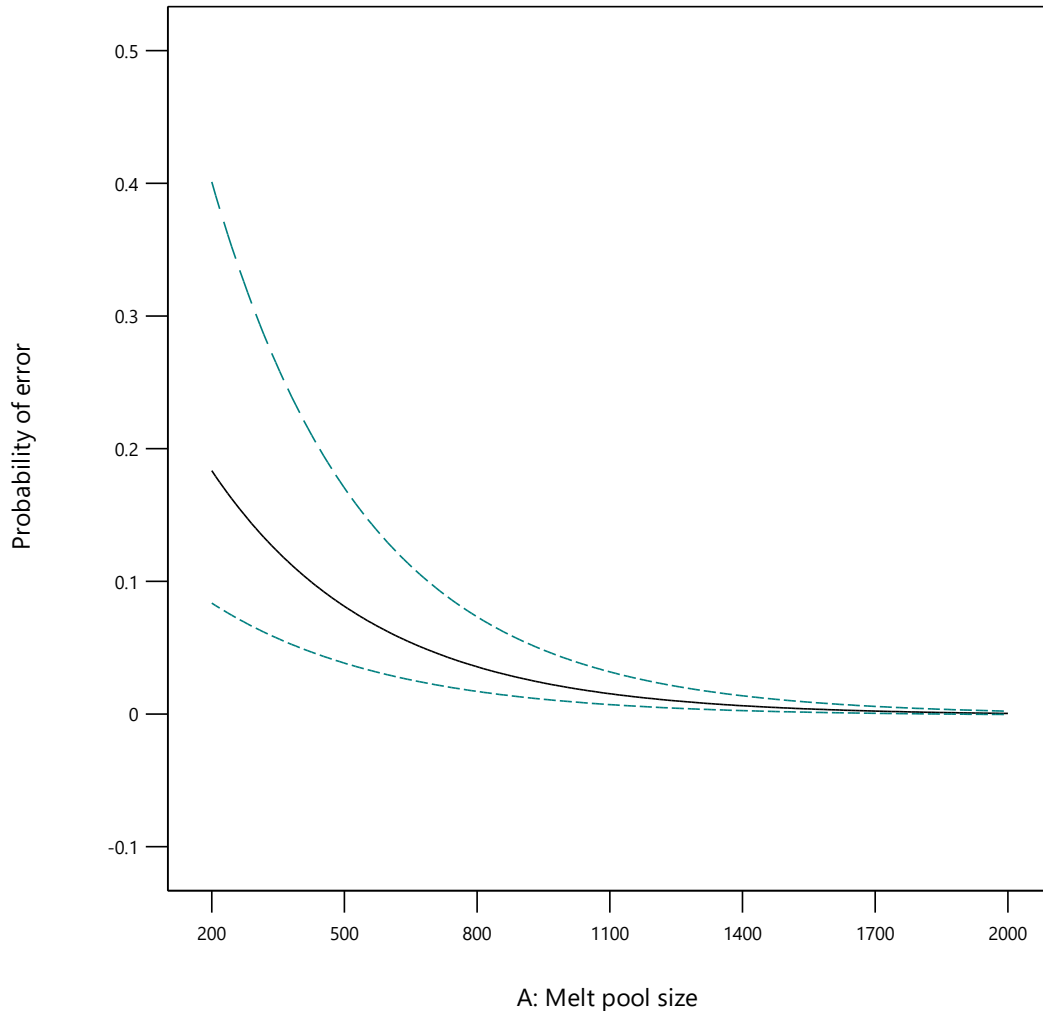
**Figure 26. Effect of melt pool size on the error of measurement of a linear radial outward motion when the number of particles = 13. The probability of error decreased with an increase in the melt pool size.**

As the number of surface oxides was in the extremes of the varied range (5 or 20), the 95% CI of the probability of error increased during linear radial outward motion, as displayed by Figure 27 and Figure 28, with 5 and 20 particles respectively. When the number of particles approached the mid-range of values, the probability of error had a low CI as depicted in Figure 26.



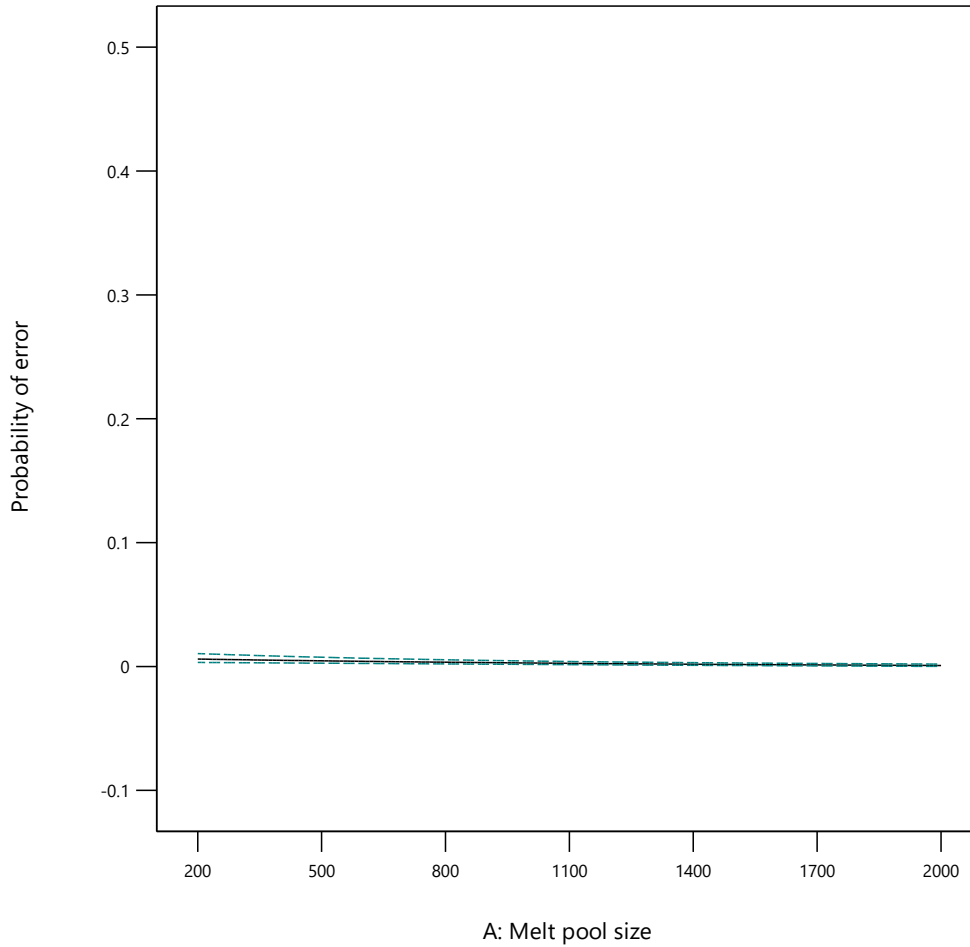
**Figure 27. Effect of melt pool size on the error of measurement of a linear radial outward motion when the number of particles = 5. The 95% CI was higher for smaller melt pool size when the number of surface oxides was decreased to 5.**





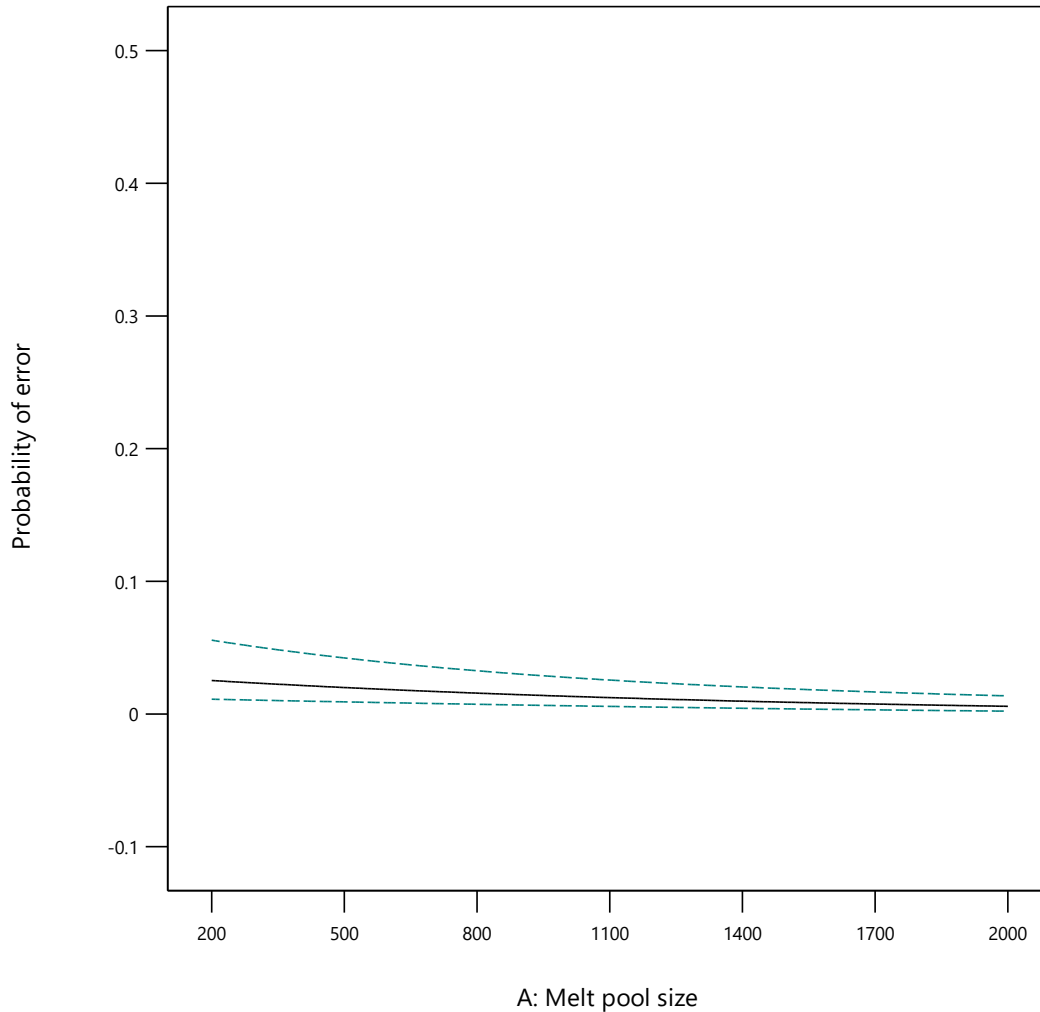
**Figure 28. Effect of melt pool size on the error of measurement of a linear radial outward motion when the number of particles = 20. The 95% CI was higher for smaller melt pool size when the number of surface oxides was increased to 20.**

The error of measurement during a linear radial inward flow was not much affected by the melt pool size as displayed in Figure 29. The probability that the error of displacement measured during a linear radial inward flow was greater than or equal 20% remained low and close to zero with a narrow 95% CI for the melt pool size range used in the study.



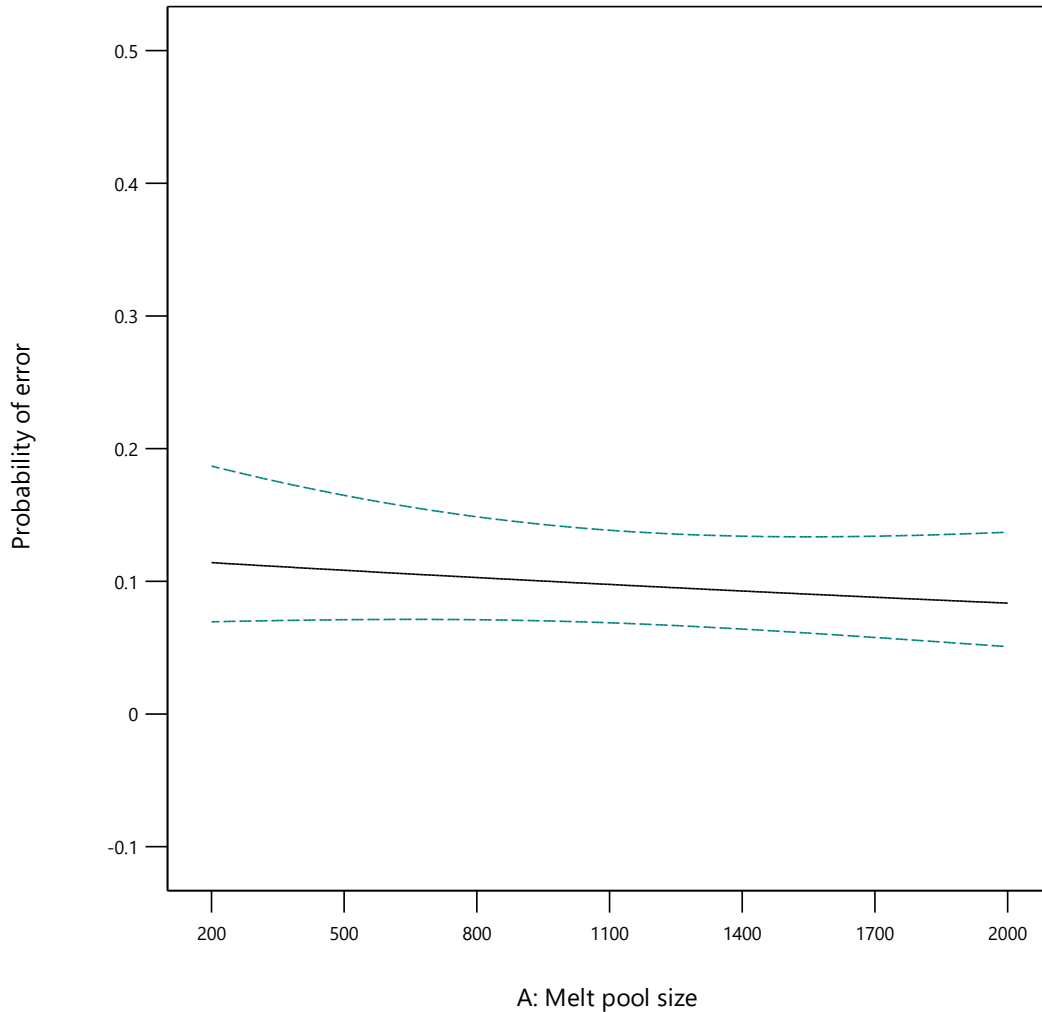
**Figure 29. Effect of melt pool size on the error of measurement of a linear radial inward motion when the number of particles = 13. The probability of error was not affected by the changes in the melt pool size.**

When the number of surface particles increased, the 95% CI of the probability of error became wider with a slight increase in mean probability of error. No significant change was observed in the probability of error of displacement measured at lower number of particles. The effect of the melt pool size on the error of displacement measurement during a linear radial inward flow with 20 particles is displaced in Figure 30.



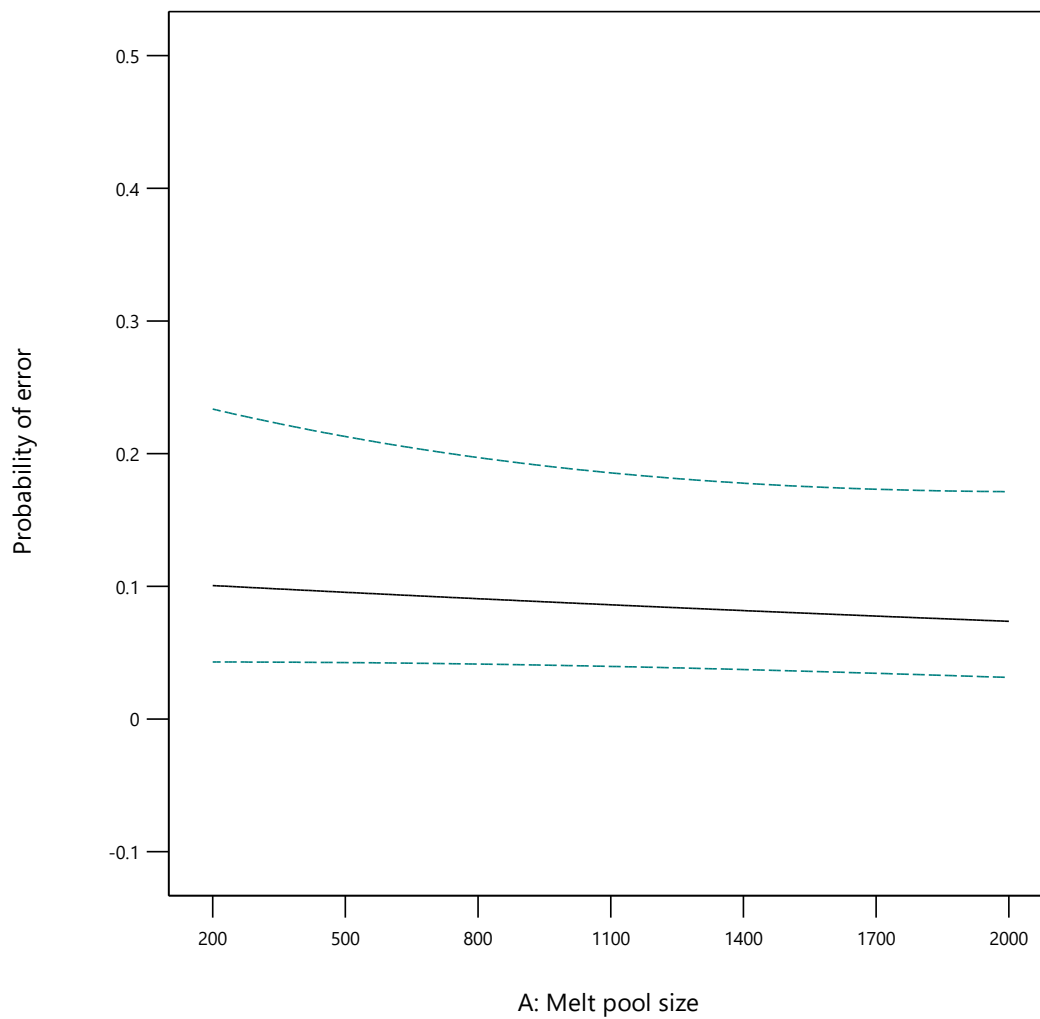
**Figure 30. Effect of melt pool size on the error of measurement of a linear radial inward motion when the number of particles = 20. The mean probability of error increased slightly with a broader 95% CI as the number of particles was increased to 20.**

Circular flow had an almost constant probability of error of displacement measurement for the range of melt pool sizes selected for this study. However, the 95% CI of the probability of error was wider in the extremes of the melt pool size range and narrower in the mid ranges of melt pool size. The effect of melt pool size on the error of displacement measured using the particle tracking algorithm during a circular flow with 13 particles is displayed in Figure 31.

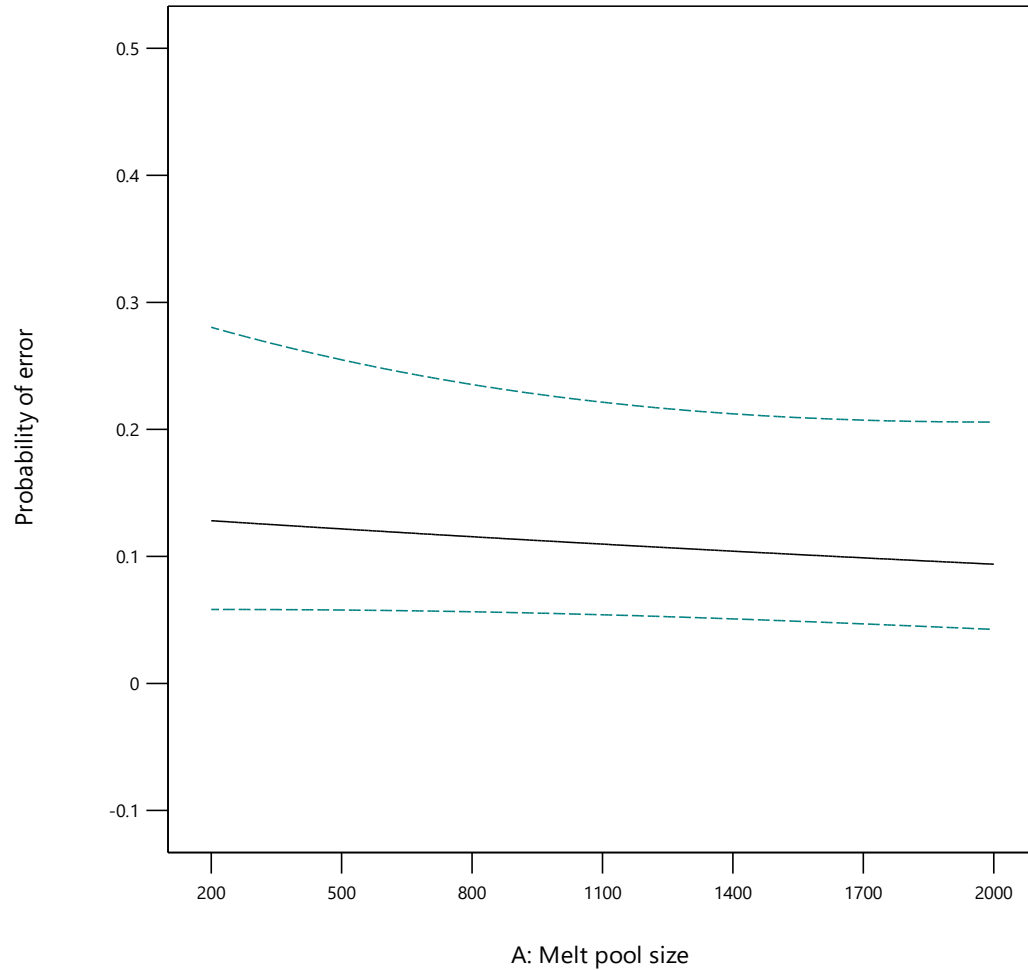


**Figure 31. Effect of melt pool size on the error of measurement of a circular motion when the number of particles = 13. The probability of error was unaffected by the changes in melt pool sizes but the 95% CI was broader when the melt pool size was in the extremes of the range.**

The 95% CI of the probability of error increased whenever the number of particles were near the extremities of the range. The probability that the error of displacement measured using the particle tracking algorithm was greater than or equal to 20% increased slightly with the increase in the number of particles during a circular motion. Figure 32 and Figure 33 demonstrate the effect of melt pool sizes on the error of displacement measurement during a circular motion when the number of particles were 5 and 20, respectively.



**Figure 32. Effect of melt pool size on the error of measurement of a circular motion when the number of particles = 5. The 95% CI became broader as the number of particles was reduced to 5.**



**Figure 33. Effect of melt pool size on the error of measurement of a circular motion when the number of particles = 20. The mean probability of error increased slightly and the 95% CI became broader as the number of particles was increased to 20.**

#### **4.2.3.2. Effect of the image pixel size**

The effect of the image pixel size on the error of displacement measured using the particle tracking algorithm was analyzed to determine the optimum image size needed to accurately measure the displacement of surface flow during SLM. The image pixel size had no apparent effect on the error of displacement measurement using the particle tracking algorithm. The ANOVA of the historical model revealed that the image pixel size was not a significant factor and had a high p-value of 0.85 in the historical model analysis. Therefore, the image pixel size was not included in the historical analysis of the model.

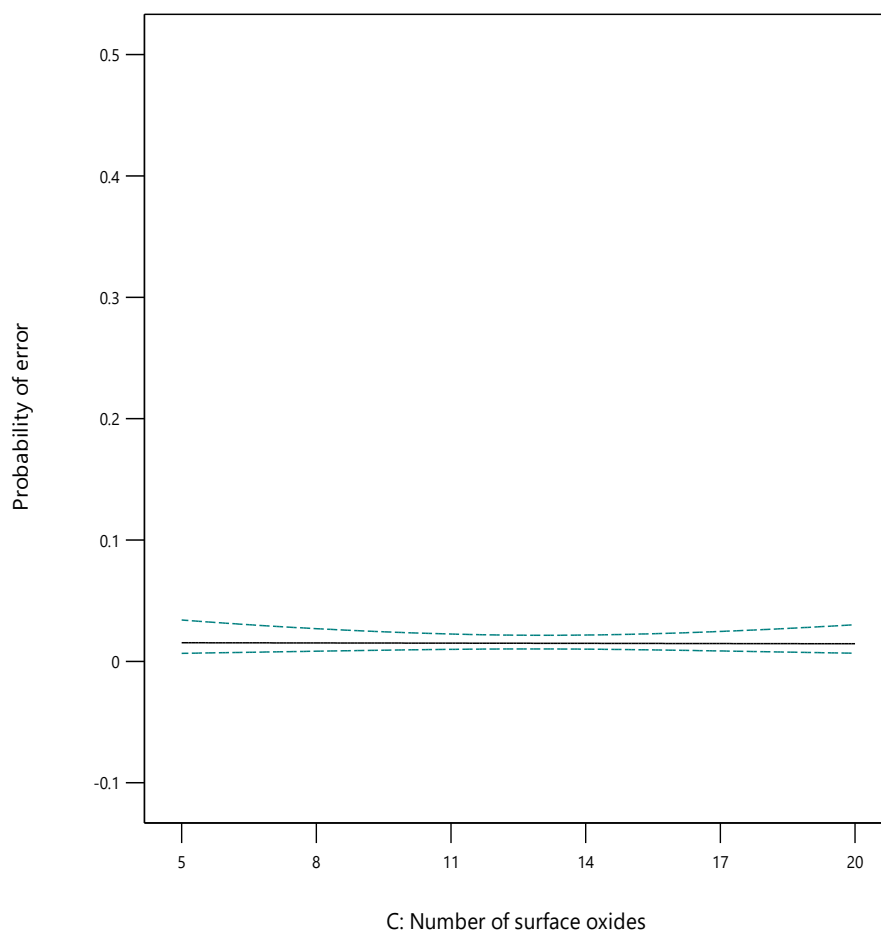
#### 4.2.3.3. Effect of the number of surface oxides

The number of surface oxides in a melt pool primarily depends on the amount of oxygen in the building atmosphere. The effect of the number of surface oxides on the error of displacement measurement using the particle tracking algorithm was studied to determine the optimum number of surface oxides needed to track the melt pool flow more accurately. The number of surface oxides was a significant factor in the historical analysis of the models with a p-value of 0.01. The probability that the error of displacement measured using the particle tracking algorithm was greater than or equal to 20% was almost constant with the changes in the

number of surface particles during a linear radial outward motion. However, the 95% CI of the probability of error was broader when the number of particles was in the extremes of the range.

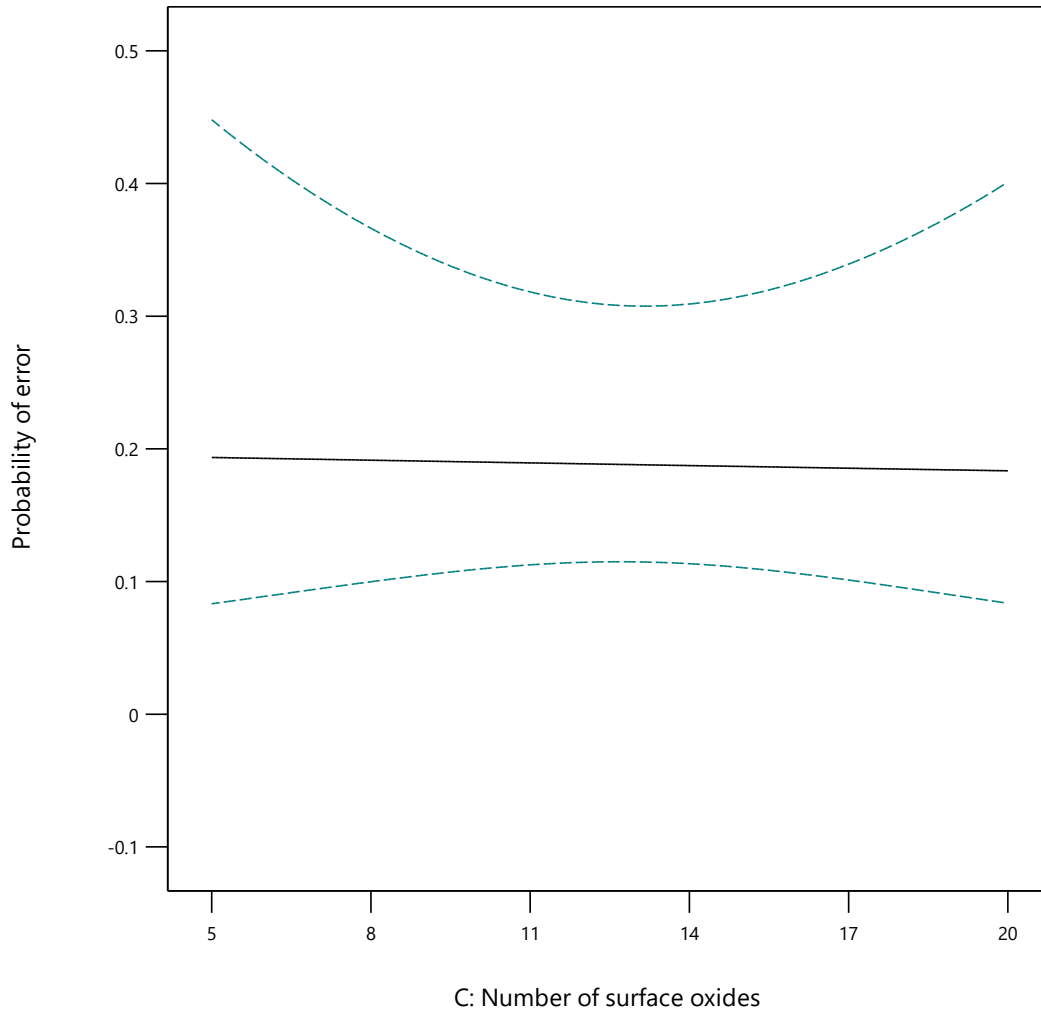
Figure 34 displays the effect of the number of surface oxides on the error of displacement

measurement during a linear radial outward motion when the melt pool size was about 1,100  $\mu\text{m}$  (mid-range).



**Figure 34. Effect of the number of surface oxides on the error of measurement of a linear radial outward motion when the melt pool size = 1, 100  $\mu\text{m}$ . Mean probability of error was unaffected by the changes in the number of surface oxides but the 95% CI became broader when the number of particles approached the extremes of the range.**

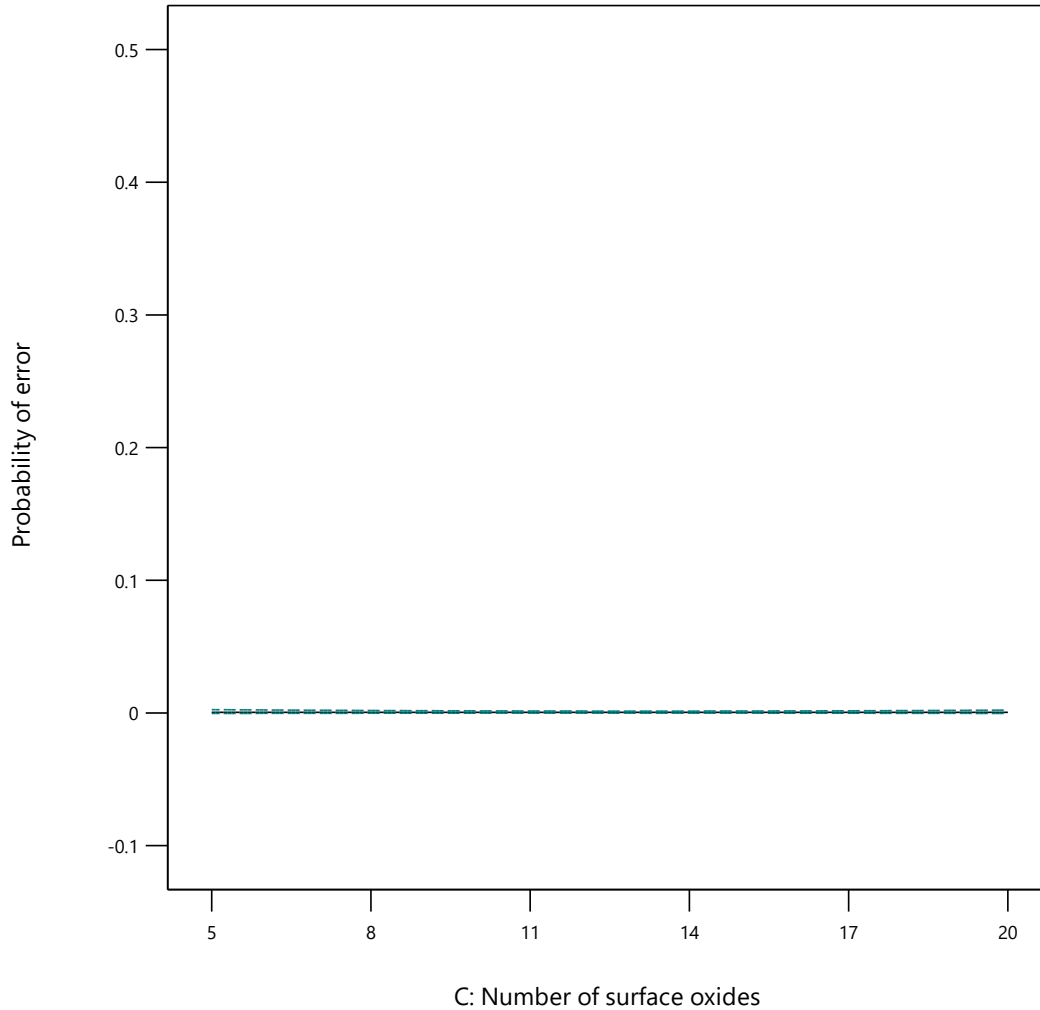
When the melt pool size was decreased from 1,100  $\mu\text{m}$ , the probability that the error of displacement measurement was greater than or equal to 20% increased significantly and had a broader 95% CI, as displayed in Figure 35 for a 200  $\mu\text{m}$  melt pool size.



**Figure 35. Effect of the number of surface oxides on the error of measurement of a linear radial outward motion when the melt pool size = 200  $\mu\text{m}$ . The probability of error increased significantly with a broader 95% CI when the melt pool size was decreased.**

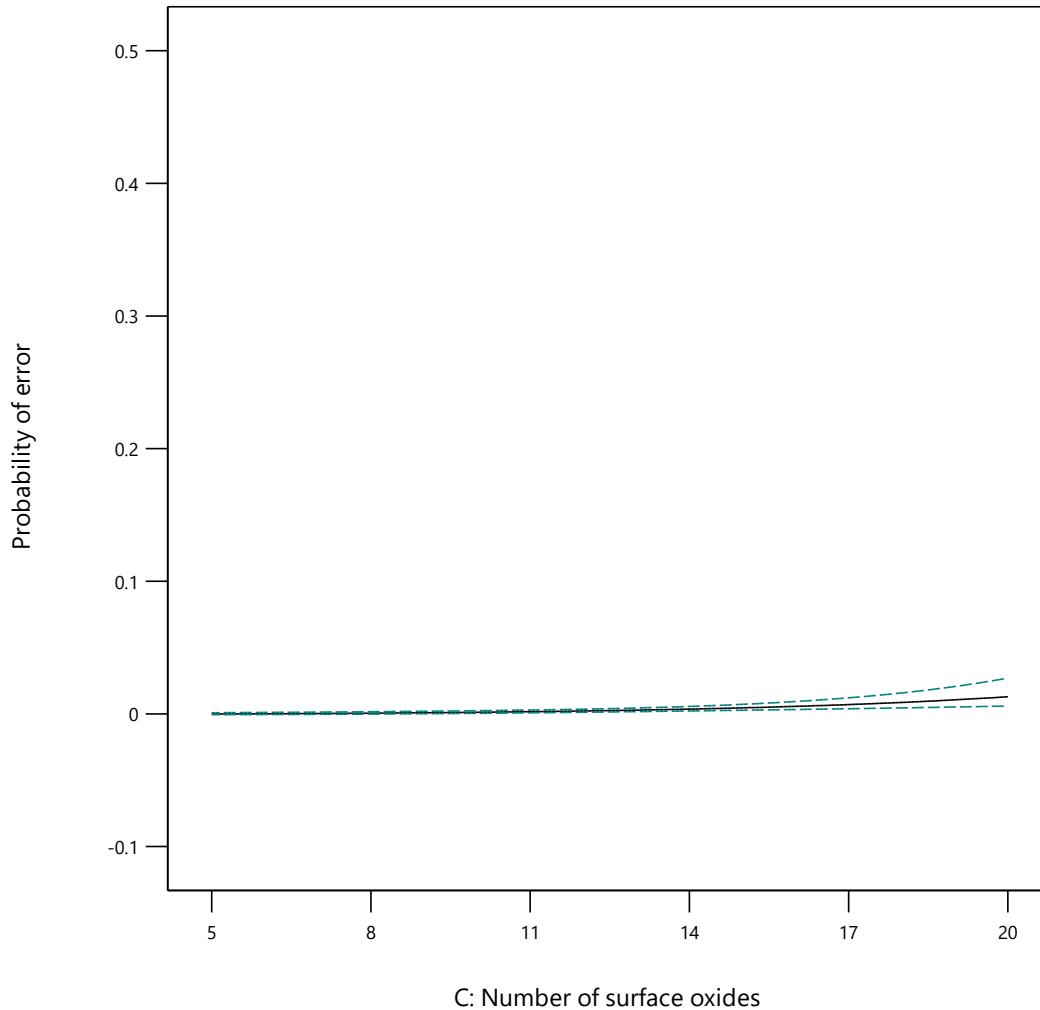
When the melt pool size was increased, the probability of error decreased significantly and the 95% CI became narrower. The effect of the number of surface particles on the error of displacement measurement during a linear radial outward motion at 2,000  $\mu\text{m}$  melt pool size is displayed in Figure 36.





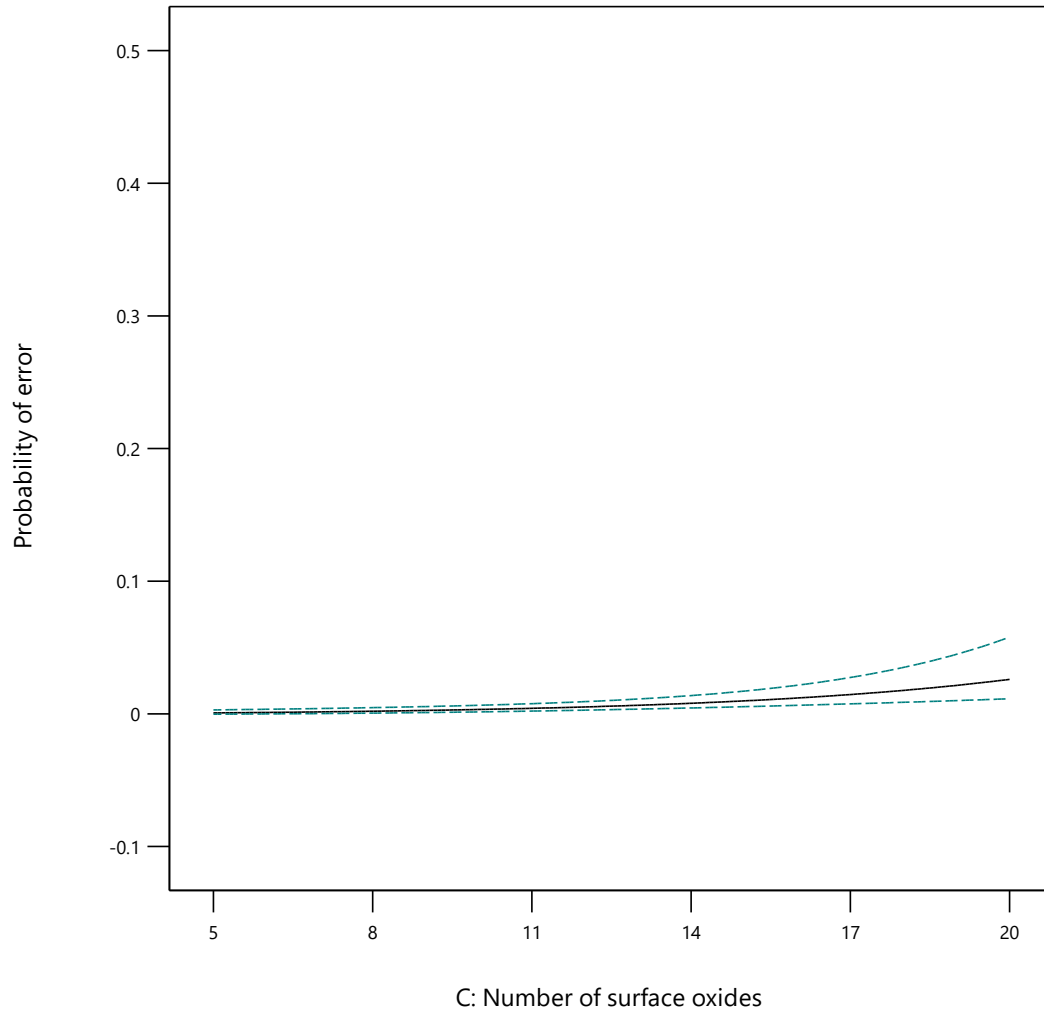
**Figure 36. Effect of the number of surface oxides on the error of measurement of a linear radial outward motion when the melt pool size = 2,000  $\mu\text{m}$ . The probability of error diminished to about zero with a narrower 95% CI when the melt pool size was increased.**

The probability that the error of displacement measurement was greater than or equal to 20% during a linear radial inward motion remained constant with the change in the number of surface oxides but had a broader 95% CI when the number of particles was high. Figure 37 displays the effect of the number of surface oxides on the error of measurement during a radial inward motion at a melt pool size of 1,100  $\mu\text{m}$  and depicts that the confident interval got broader as the number of particles approached 20.



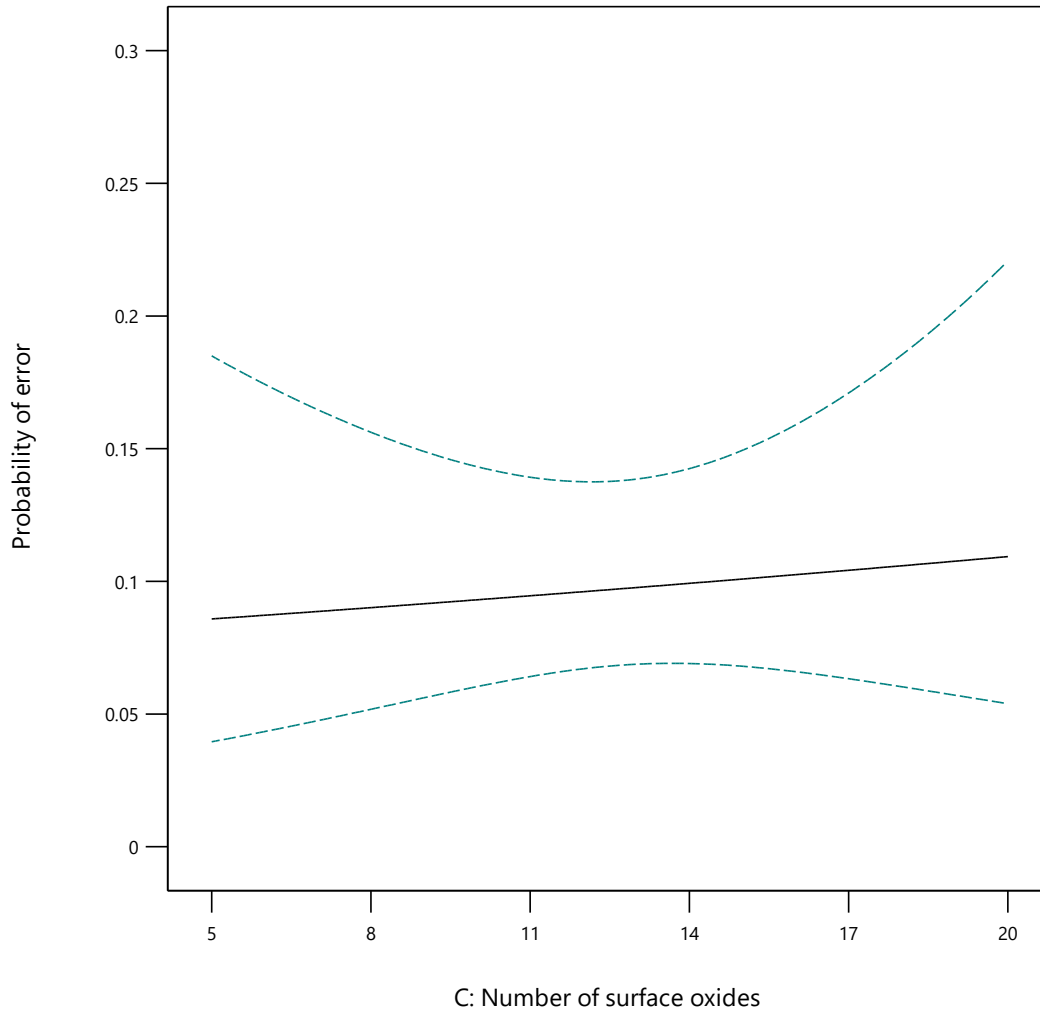
**Figure 37. Effect of the number of surface oxides on the error of measurement of a linear radial inward motion when the melt pool size = 1,100  $\mu\text{m}$ . Mean probability of error was not affected by the changes in the number of particles but the 95% CI became broader as the number of particles approached 20.**

For the melt pool size of 200  $\mu\text{m}$ , the probability that the error of displacement measurement during linear radial inward motion was greater than or equal to 20% remained low when the number of particles was low but increased slightly when the number of particles was high. The 95% confidence became broader with higher number of particles when the melt pool size was 200  $\mu\text{m}$ , as displayed in Figure 38. The probability of error remained approximately the same when the melt pool size was increased from 1,100  $\mu\text{m}$  to 2,000  $\mu\text{m}$  during linear radial outward motion.



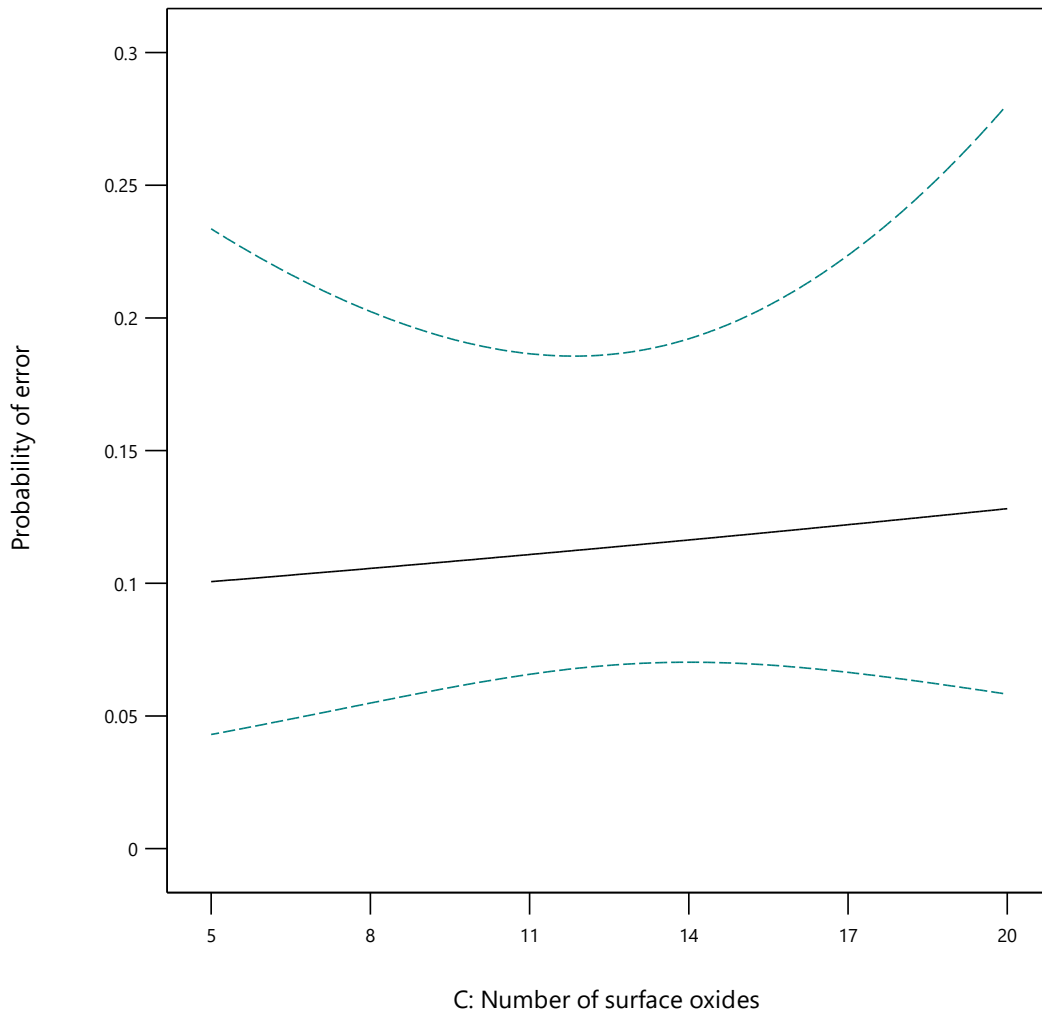
**Figure 38. Effect of the number of surface oxides on the error of measurement of a linear radial inward motion when the melt pool size = 200  $\mu\text{m}$ . Probability of error was not much affected when the number of particles were low. However, the probability of error slightly increased with a broader 95% CI as there were higher number of particles.**

For a circular motion, the probability that the error of displacement measured using the particle tracking algorithm was greater than or equal to 20% increased slightly with the increase in number of surface oxides. The 95% CI was broader in the extremes of the range of the number of surface oxides. The effect of the number of surface oxides on the probability of error of displacement measurement during a circular motion at a melt pool size of about 1,100  $\mu\text{m}$  is displayed in Figure 39.



**Figure 39. Effect of the number of surface oxides on the error of measurement of circular motion when the melt pool size = 1,100  $\mu\text{m}$ . Probability of error increased slightly with the increased in the number of particles and the 95% CI became broader as the number of surface oxides approached the extremes of the range.**

The probability of error decreased slightly but the 95% CI remained unchanged when the size of melt pool was increased from 1,100  $\mu\text{m}$  to 2,000  $\mu\text{m}$ . However, when the size of melt pool was decreased to 200  $\mu\text{m}$ , there was a significant increase in the probability of error and the width of its 95% CI, as displayed in Figure 40.

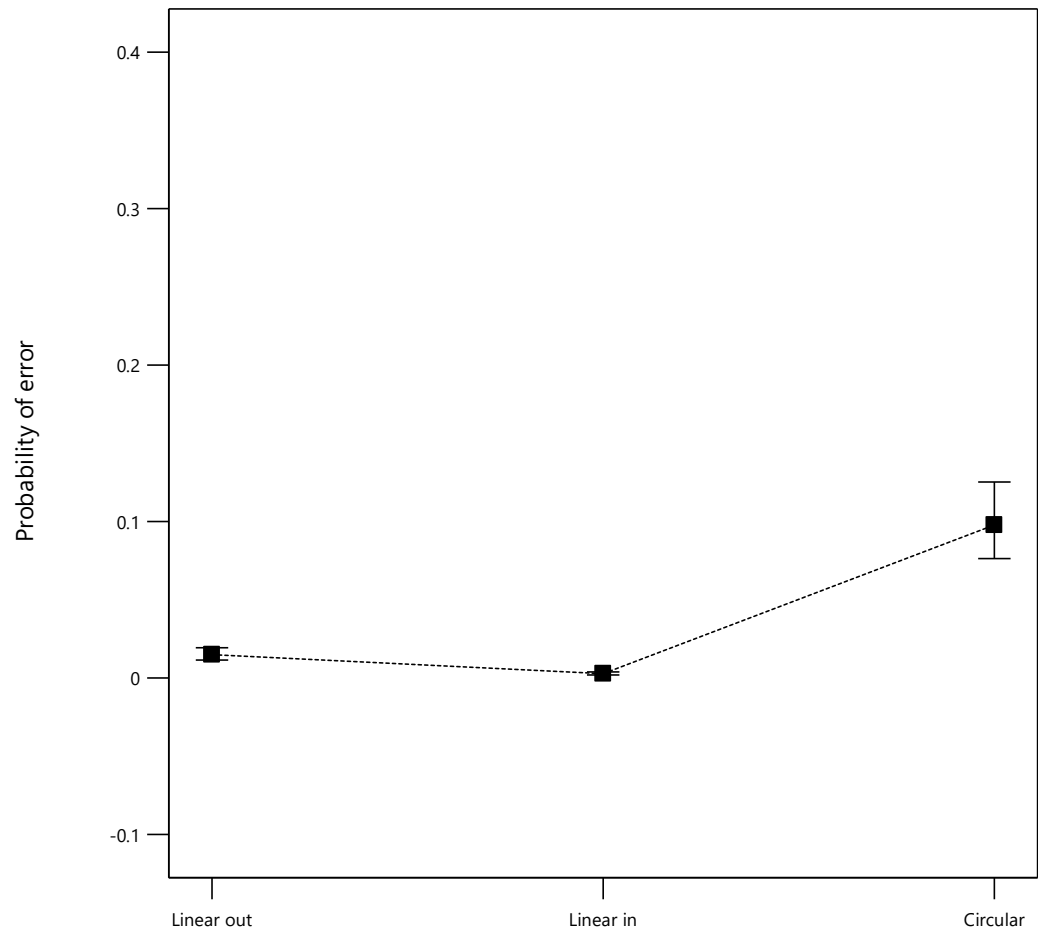


**Figure 40.** Effect of the number of surface oxides on the error of measurement of circular motion when the melt pool size = 200  $\mu\text{m}$ . Both the probability of error and the width of its 95% CI increased as the melt pool size decreased.

#### 4.2.3.4. Effect of the flow path

Effect of the flow path on the measured displacement was analyzed to determine if the particle tracking code would be able to track surface flow in a melt pool during different flow patterns with equal accuracy. The probability that the error of the displacement measurement was greater than or equal to 20% during a linear radial outward motion was low for a larger melt pool and high for a smaller melt pool. Probability that the error of displacement measurement was greater than or equal to 20% was usually low during a linear radial inward flow but depended on

the melt pool size and the number of surface oxides. However, the probability of error of measurement for a circular flow changed slightly but its 95% CI changed significantly as the melt pool size and the number of surface oxides changed. Figure 41 displays the probability of error as a function of flow type when the melt pool size is about 1,100  $\mu\text{m}$  and the number of surface oxides is 13 (mid values of the range).

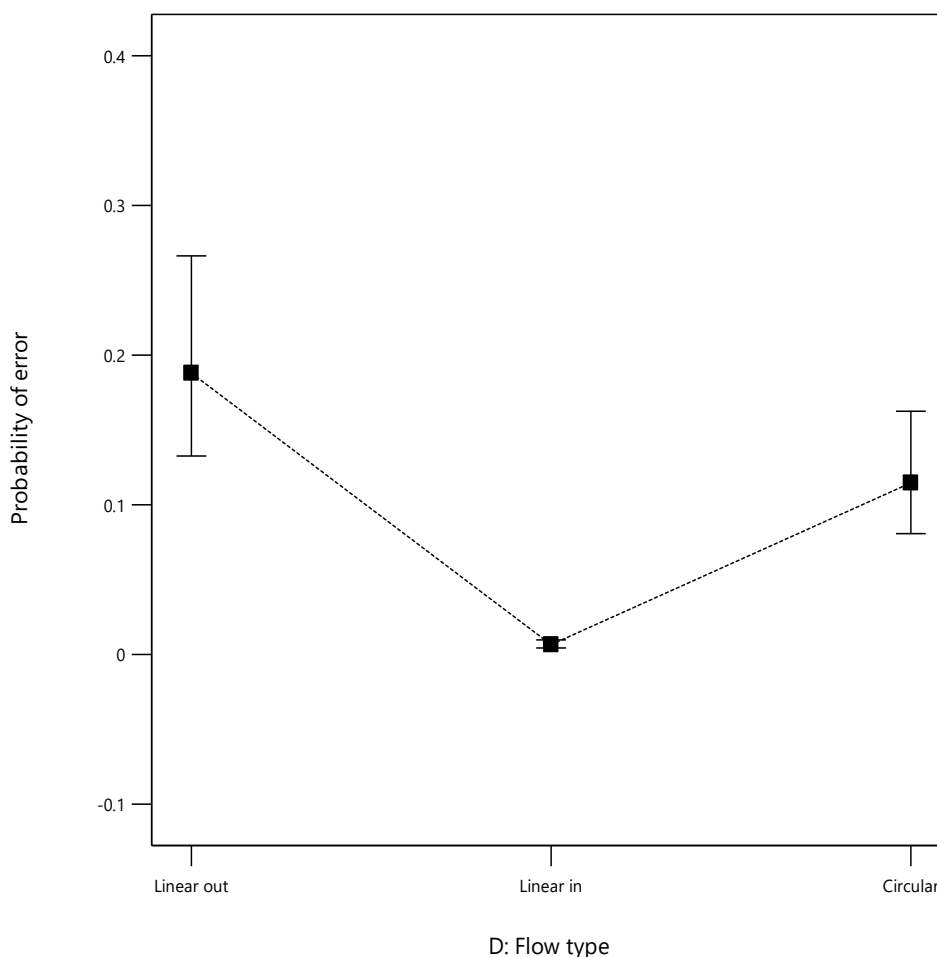


D: Flow type

**Figure 41. Effect of the flow paths on the error of measurement when the melt pool size = 1,100  $\mu\text{m}$  and number of surface oxides = 13. Probability of error is low during linear motion and high during circular motion.**

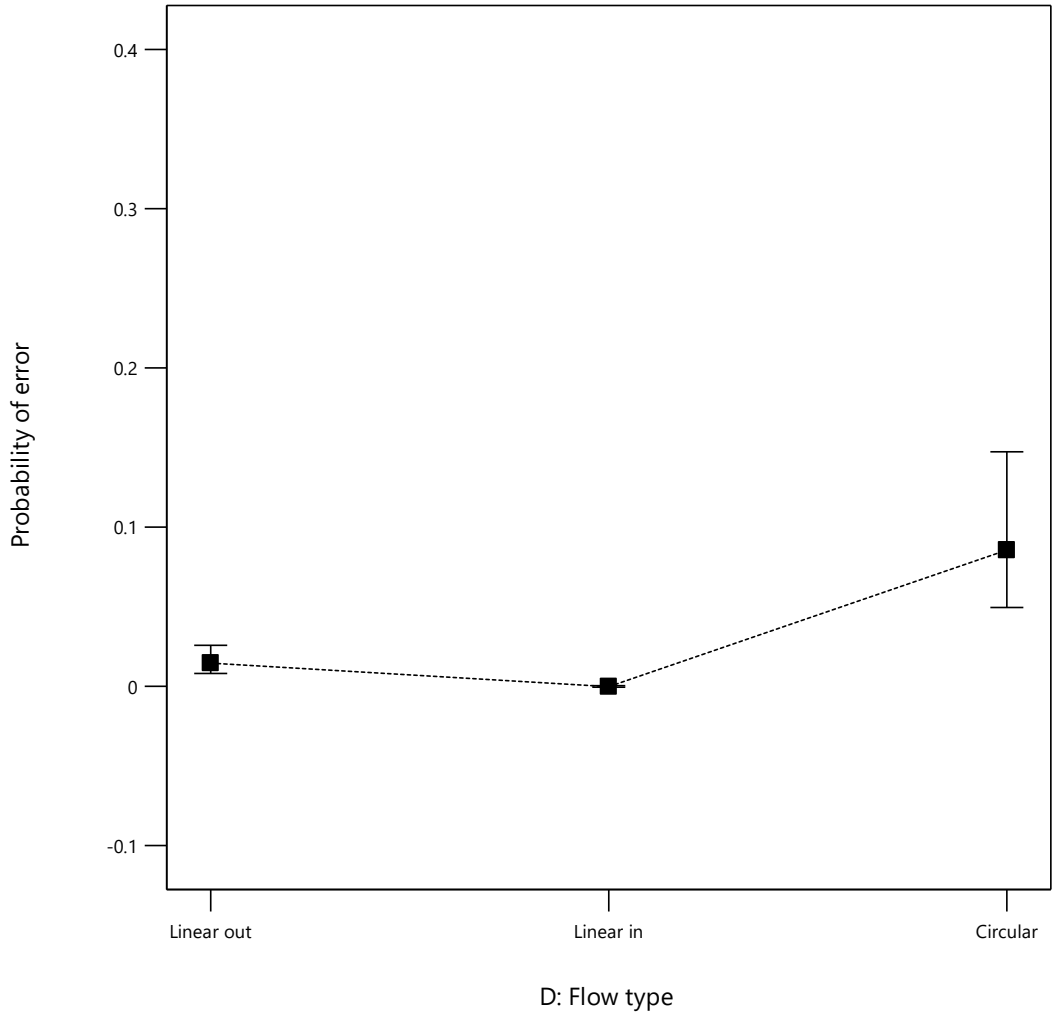
When the size of the melt pool was decreased and the number of surface oxides was kept constant (13), the probability that the error of measured displacement was greater than or equal to 20% increased drastically and the 95% CI became broader during a linear radial outward flow. However, the probability of error of displacement measurement for a linear radial inward flow was almost unaffected. During a circular flow, the probability of error increased slightly and the 95% CI became significantly broader with the decrease in melt pool size. Figure 42 displays the probability of error of the measured displacement as a function of flow type when the melt pool size is 200  $\mu\text{m}$  and

the number of particles kept constant at 13. The probability of error of displacement measurement was not significantly affected as the melt pool size was increased from 1,100  $\mu\text{m}$  to 2,000  $\mu\text{m}$  with a constant number of particles.



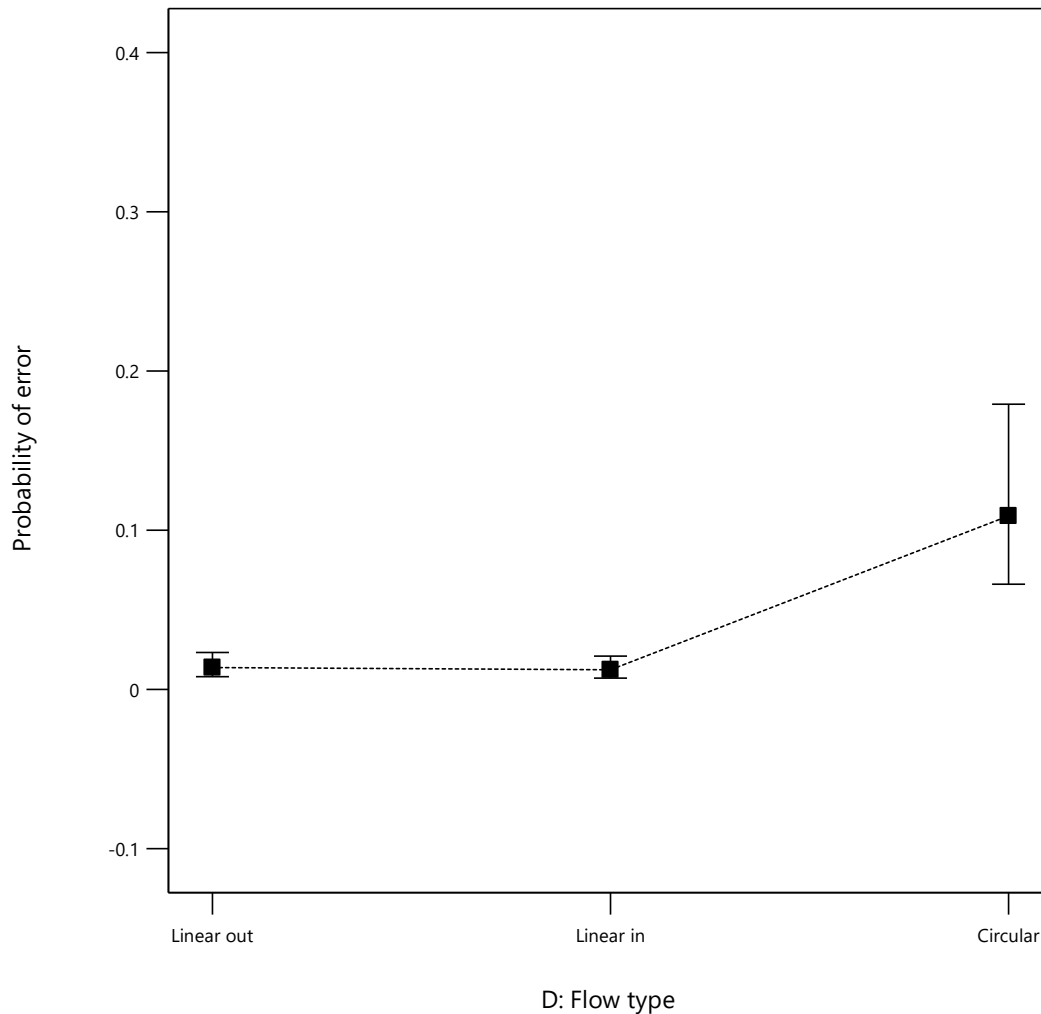
**Figure 42. Effect of the flow paths on the error of measurement when the melt pool size = 200  $\mu\text{m}$  and number of surface oxides = 13. Probabilities of error during a linear radial outward motion and circular motion was significantly increased with a broader 95% CI whereas the probability of error during a linear radial inward motion was unaffected.**

The changes in the number of surface oxides didn't show a significant influence as the changes in melt pool size had on the error of the measured displacement. When the melt pool size was kept constant at 1,100  $\mu\text{m}$ , the probability that the error of displacement measurement was greater than or equal to 20% remained almost constant but the 95% CI became broader for each flow type as the number of surface oxides were in the extremes of the range. Figure 43 and Figure 44 display the effect of the flow path on the error of the displacement measurement when the number of surface oxides are 5 and 20, respectively at a melt pool size of 1,100  $\mu\text{m}$ .



**Figure 43. Effect of the flow paths on the error of measurement when the melt pool size = 1,100  $\mu\text{m}$  and number of surface oxides = 5. Mean probability of error was unaffected but the 95% CI became broader as the number of particles approached the lower-limit of the range.**





**Figure 44. Effect of the flow paths on the error of measurement when the melt pool size = 1,100  $\mu\text{m}$  and number of surface oxides = 20. Mean probability of error was unaffected but the 95% CI became broader as the number of particles approached the upper-limit of the range.**

#### 4.2.3.5. Model validation

The result of the historical model was verified by running validation tests and exporting the results to the ‘Confirmation’ section of the ‘Post Analysis’ tab in the Design Experts 12. The validation tests were performed for three flow types with various levels of each factor that were different from the levels used in the historical model, as presented in Table VIII. The size range of the oxide particles was chosen according to the size of the melt pool so that the particles don’t

form a cluster. The levels of each factor were chosen randomly and each test was performed at least twice to make sure they represent the true value.

The sizes of the melt pool were chosen randomly between 200  $\mu\text{m}$  and 2,000  $\mu\text{m}$ . The melt pool of 2,000  $\mu\text{m}$  had a scaling factor of 8.9  $\mu\text{m}/\text{pixel}$  and a size of 225 pixels in the simulated binary images. Whereas the melt pool of 200  $\mu\text{m}$  had a scaling factor of 2.2  $\mu\text{m}/\text{pixel}$  and a size of 91 pixels in the binary images. The sizes of the melt pool (between 91 pixels and 225 pixels) chosen for the validation tests were interpolated between 200  $\mu\text{m}$  and 2,000  $\mu\text{m}$  to determine their sizes in  $\mu\text{m}$ . Since the size of the first melt pool (120 pixels) in the validation test plan was small, the radius range of surface particles used in the binary image simulation was similar to that for a 200  $\mu\text{m}$  melt pool i.e. 6 – 9 pixels. The radius range of surface particles in the binary images of other validation tests was similar to the radius range used for a 2,000  $\mu\text{m}$  melt pool i.e. 6 – 12 pixels. It was observed during particle tracking of linear motion that the probability of error was high when particle velocity was low, but the particle velocity was not one of the factors in the model. For the validation tests, the particle velocities during linear motion were kept low on purpose even for large melt pool size to check if they affected the particle tracking algorithm. A constant angular velocity of 0.2 radians/frame was used for circular motion. The simulated binary images portraying linear motion for validation tests had either radial outward or radial inward motion only in between image pairs and not a combination of both outward and inward motion as in the original tests of the historical model.

**Table VIII. Validation test plan.**

<b>Melt pool sizes (pixels)</b>	<b>Melt pool sizes (<math>\mu\text{m}</math>)</b>	<b>Number of surface oxides</b>	<b>Flow type</b>	<b>Radius range (pixels)</b>	<b>Velocity</b>
120	590	7	Linear in	6 – 9	2 pixels/frame
150	990	10	Circular	6 – 12	0.2 radians/frame
180	1,400	16	Linear out	6 – 12	3 pixels/frame

Table IX illustrates the results from the validation tests. The probability that the error of displacement measured using the particle tracking algorithm was greater than or equal to 20% for three flow types is also presented in Table IX. Three trials were performed for the linear flows because of significant discrepancies between the first two trials. The probability that the error of displacement measured using the particle tracking algorithm was greater than or equal to 20% during linear motion decreased significantly on the second trial and therefore a third trial was necessary. On contrary, the first two trials for the circular flow generated the same results and therefore, a third trial was not necessary.

**Table IX. Result of the validation tests. Probability that the error of displacement measurement was greater than or equal to 20% during three different flow types.**

<b>Trials</b>	<b>Linear radial outward motion</b>	<b>Linear radial inward motion</b>	<b>Circular motion</b>
1	9.71%	22.45%	5.33%
2	1.94%	6.32%	5.33%
3	6.67%	7.14%	-

#### **4.2.3.5.1. Linear radial outward motion model**

Three different trials of the experiment were performed for the validation of the historical model for a linear radial outward motion. The probability that the error of displacement measured using the particle tracking algorithm was greater than or equal to 20% was calculated from the various trials. The response variable from each trial was exported to the 'Confirmation' section of Design Expert 12. The historical model predicted a low mean probability of error (0.0063) and a 95% CI of (-0.000195 – 0.0298) for this set of factors. The response variable from the second trial was close to the predicted mean value and within the 95% CI. However, even the average of the last two trials (with lower response values) was outside the 95% CI. This model failed to predict the probability of error of displacement measured using the particle tracking algorithm during a radial outward motion for the given set of factors.

#### **4.2.3.5.2. Linear radial inward motion model**

The historical model was also validated for the linear inward motion by running three trials of an experiment. The probability that the error of displacement measured using the particle tracking algorithm was greater than or equal 20% during a linear radial inward motion was calculated from each trial and used as a response variable that was next exported to the 'Confirmation' section of Design Expert 12. The response variables from the second and third trials were used in the model validation as their values were close to each other and seemed to represent the true value. The average of the response variables from the two trials was 0.067, a value significantly higher than the predicted mean value of 0.00085 and out of the 95% CI of (-0.00096 – 0.004). This model also failed to predict the probability of error of displacement measured using the particle tracking algorithm during a radial inward motion for the given set of factors.

#### **4.2.3.5.3. Circular motion model**

The validation test of the historical model for circular motion required only two trials as both results had the same response variable. The response variable was the probability that the error of angular displacement measured using the particle tracking algorithm was greater than or equal to 20% during a circular motion. The average response variable from the two trials (0.053) was slightly lower than the predicted mean value of 0.095 but it was within the 95% CI of (0.016 – 0.242). This model was successful to predict the probability of error of angular displacement measured using the particle tracking algorithm during a circular motion for the given set of factors.

#### **4.2.3.5.4. Conclusions from model validation**

The historical model was successfully validated for circular motion, but the model validation was unsuccessful for linear radial motions. The historical model was validated for circular motion from the validation tests that used the same angular velocity (i.e. 0.2 radians/frame) of particles in between image pair as the actual tests in the model. On contrary, the velocities of particles in the simulated binary images used to validate the historical model for linear motions were different from the particle velocities used in the actual tests. Velocities that were different from the original tests were chosen on purpose to understand the effect of velocity and displacements between the particles on the displacement measurement. The factor “velocity” was not initially considered in the experimental design, but it was noticed that higher displacement measurements yielded lower error. Therefore, velocity was added as the other factor on the experiments meant to be designed to analyze the effect of the size of surface oxide on the displacement measured using the particle tracking algorithm.

#### 4.2.3.6. Independent studies

Two independent studies were conducted to analyze the effects of the particle velocity, the size of the surface particles, and the number of surface particles on the error of displacement measured using the particle tracking algorithm after the model validation failed for linear motion. The levels of the other factors such as the melt pool size, the image pixel size, and the flow type were chosen from the historical analysis of the model that yielded the least amount of error. The melt pool size of 2,000  $\mu\text{m}$  (225 pixels) was used for the independent studies as its probability of error for all the flow types was low. Since the image pixel size was an insignificant factor in the historical analysis, a random image size of 500 x 500 pixels was chosen for this study. The linear radial flow (both outward and inward) had a low probability of error in a melt pool of 2,000  $\mu\text{m}$  and hence linear radial outward motion was chosen for the independent studies.

The first independent study was conducted to analyze the effects of the velocity of surface particles and the size of surface particles on the displacement measured using the particle tracking algorithm. In the historical analysis the probability of error was lower when the number of surface oxides was in the mid-range values, and therefore the number of surface oxides chosen for the first independent study was 12. Table X presents the probability that the error of displacement measured using the particle tracking algorithm was greater than or equal to 20% for various levels of the velocity and the size of the surface particles. The probability that the error of displacement measurement was greater than or equal to 20% was zero when the particle velocities were 6 or 10 pixels/frame for any level of the size of surface oxides. The probability was not zero when the velocity of particles was 2 pixels/frame.

**Table X. Results to quantify the effects of the velocity of surface particles and the size of surface oxides on the displacement measured using the particle tracking algorithm.**

<b>Run</b>	<b>Factor 1 A: Velocity of surface oxides (pixels/frame)</b>	<b>Factor 2 B: Size of surface oxides</b>	<b>Probability that error <math>\geq</math> 20%</b>
1	10	Mid-range	0
2	6	Mid-range	0
3	10	High-range	0
4	6	Low-range	0
5	6	Mid-range	0
6	6	Mid-range	0
7	6	High-range	0
8	2	Low-range	5.30%
9	10	Low-range	0
10	2	Mid-range	9.90%
11	6	Mid-range	0
12	6	Mid-range	0
13	2	High-range	18.90%

The probability that the error of displacement measured using the particle tracking algorithm was greater than or equal to 20% was zero for most runs during the first independent study and a model could not be established. The probability was non-zero only when the particle velocity was 2 pixels/frame, which can be useful in developing a model. Therefore, experiments were designed for a second independent study at a constant linear velocity of 2 pixels/frame to

quantify the effects of the size of surface oxides and the number of surface particles on the error of displacement measurement. The probability that the error of displacement measurement was greater than or equal to 20% was non-zero for all 13 runs when the velocity of particles was 2 pixels/frame, as demonstrated in Table XI.

**Table XI. Results to quantify the effects of the size of surface oxides and the number of surface oxides on the displacement measured using the particle tracking algorithm.**

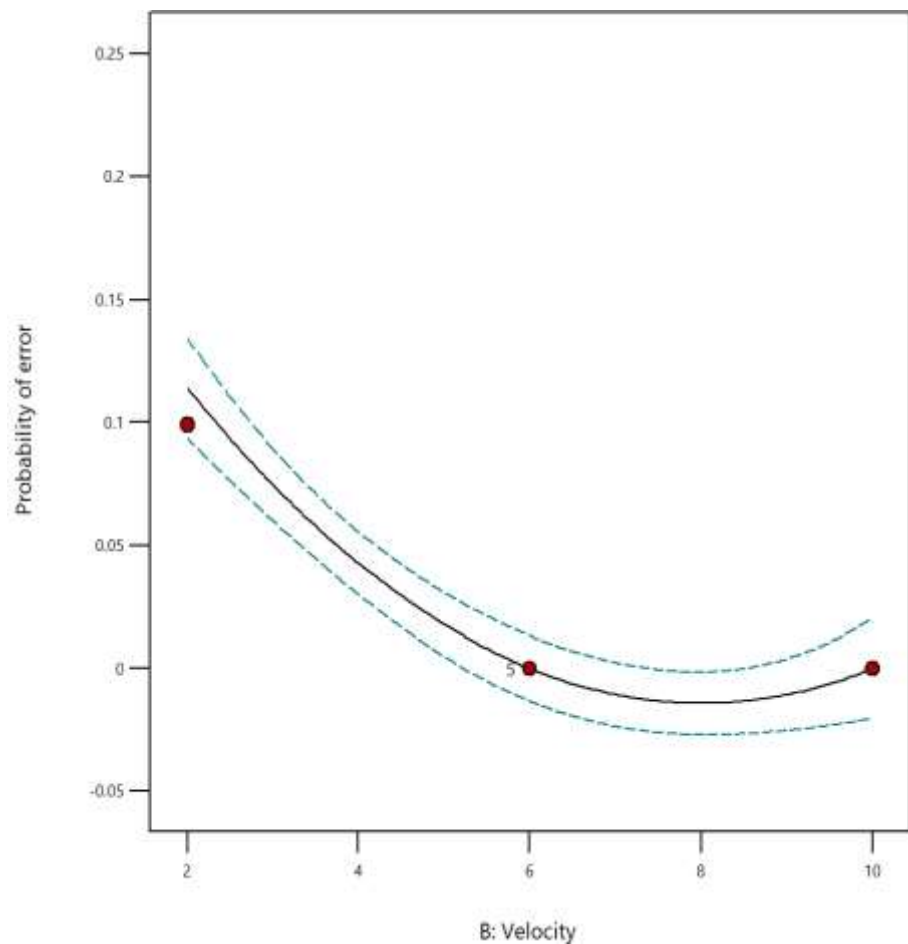
<b>Run</b>	<b>Factor 1</b> <b>A: Size of surface oxides</b>	<b>Factor 2</b> <b>B: Number of surface oxides</b>	<b>Probability that</b> <b>error <math>\geq</math> 20%</b>
1	Mid-range	13	10.40%
2	Low-range	5	13.30%
3	Mid-range	13	11.80%
4	Mid-range	13	7.80%
5	Mid-range	5	25.30%
6	High-range	5	5.30%
7	Mid-range	13	6.30%
8	Mid-range	13	2.90%
9	Low-range	20	13.40%
10	Low-range	13	14.50%
11	High-range	13	2.52%
12	Mid-range	20	12.80%
13	High-range	20	7.80%



#### 4.2.3.6.1. Effect of the velocity of particles

The historical analysis of the models and its validation tests suggested that the displacements of the particles in between the image pair might affect the error of displacement measured using the particle tracking algorithm. The probability of error was low when large displacements (i.e. 5, 8, and 20 pixels) were measured and high when small displacements (i.e. 2 pixels) were measured. Therefore, experiments were designed to quantify the effects of the velocity of the particles on the error of displacement measurement. Similar to the historical model, the probability that the error of displacement measurement was greater than or equal to 20% was only greater than zero when the velocity of particles was 2 pixels/frame. ANOVA

results depicted that velocity was a significant factor in the model with a p-value of  $<0.0001$ . The probability of error was high when the particle velocity was 2 pixels/frame but quickly diminished to zero as the velocity approached 6 pixels/frame as displayed in Figure 45.



**Figure 45. Effect of the velocity of particles on the error of measurement. Probability of error decreased with the increase in the particle velocity.**

The probability that the error of displacement measurement was greater than or equal to 20% was also zero in all the runs when a displacement of 5 pixels was measured during the initial tests for a melt pool of 2,000  $\mu\text{m}$ . The results of the ANOVA of the model for the first independent study are presented in Table XII and Table XIII.

**Table XII. ANOVA for reduced quadratic model of the first independent study.**

Source	Sum of Squares	df	Mean Square	F-value	p-value
<b>Model</b>	0.0376	4	0.0094	40.41	< 0.0001
A-Size of surface oxides	0.0031	1	0.0031	13.14	0.0067
B-Velocity	0.0195	1	0.0195	83.72	< 0.0001
AB	0.0046	1	0.0046	19.71	0.0022
B <sup>2</sup>	0.0105	1	0.0105	45.08	0.0002
Residual	0.0019	8	0.0002		
Lack of Fit	0.0019	4	0.0005		

**Table XIII. ANOVA fit statistics for the model of the first independent study.**

Statistic	Value
Std. Dev.	0.0152
Mean	0.0263
C.V. %	58
R <sup>2</sup>	0.9528
Adjusted R <sup>2</sup>	0.9293
Predicted R <sup>2</sup>	0.6817
Adeq Precision	20.3983

#### **4.2.3.6.2. Effect of the size and the number of surface oxides**

The size of the surface oxides also emerged as a significant factor in the first independent study quantifying the effects of the velocity and the size of surface oxides on the error of displacement measurement. However, most probabilities of error were zero and therefore enough evidence was not present to conclude that the size of surface oxides had a significant effect on the error of displacement measurement. Consequently, experiments were designed for a second

independent study at a velocity that yields non-zero response variables. The second independent study analyzed the effects of the size and the number of surface oxides on the error of displacement measured using the particle tracking algorithm at a constant linear particle velocity of 2 pixels/frame. The ANOVA result for the model of the second independent study depicted that model was not significant with a high p-value of 0.85. The size and the number of surface oxides were also insignificant factors with high p-values of 0.87 and 0.59, respectively in that model. Furthermore, the lack of fit of residuals was significant with a low p-value of 0.04. Therefore, both the size and the number of surface oxides had no significant effect on the error of displacement measured using the particle tracking algorithm. The results of the ANOVA of the model for the second independent study are presented in Table XIV and Table XV.

**Table XIV. ANOVA for linear model of the second independent study.**

<b>Source</b>	<b>Sum of Squares</b>	<b>df</b>	<b>Mean Square</b>	<b>F-value</b>	<b>p-value</b>
<b>Model</b>	0.0017	2	0.0009	0.161	0.8535
A-Size of surface oxides	0.0001	1	0.0001	0.0248	0.8780
B-Number of particles	0.0016	1	0.0016	0.2971	0.5976
Residual	0.0542	10	0.0054		
Lack of Fit	0.0493	6	0.0082	6.77	0.0427

**Table XV. ANOVA fit statistics for the model of the second independent study**

<b>Statistic</b>	<b>Value</b>
Std. Dev.	0.0736
Mean	0.1211
C.V. %	60.80
R <sup>2</sup>	0.0312
Adjusted R <sup>2</sup>	-0.1626
Predicted R <sup>2</sup>	-0.7023
Adeq Precision	1.1942

#### 4.2.4. Error analysis

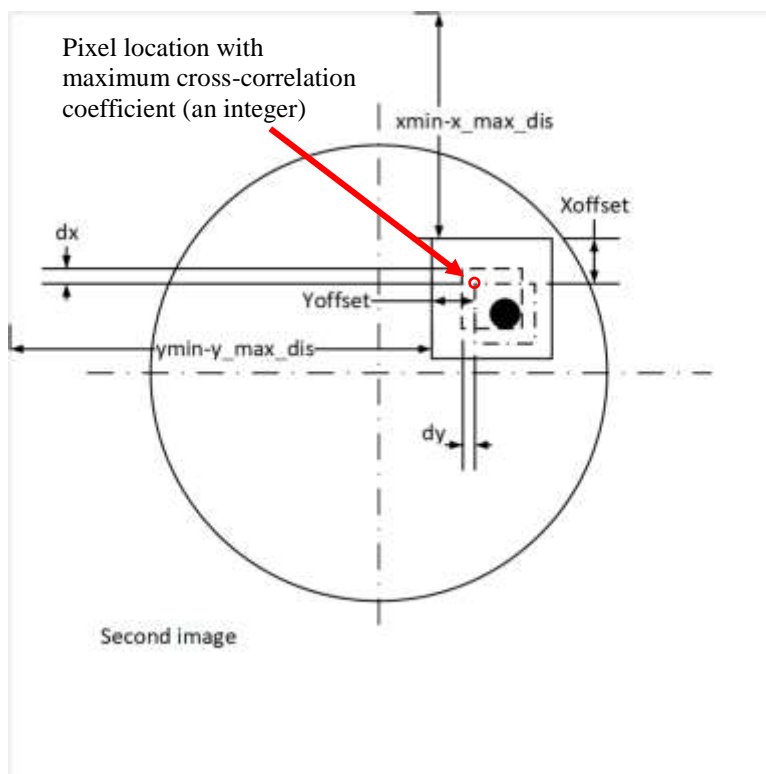
The possible causes of error of displacement measured using the particle tracking algorithm during two different flow types were studied. A test plan was designed to determine the factors that had the most influence on the error of displacement measured by using the particle tracking code. ANOVA results concluded that some factors that were initially thought to be significant did not provide enough evidence to prove their significance in the model while other factor (velocity of particles) emerged to be significant in the model. An error of 20% was considered to be the threshold to define the efficacy of the code in measuring the displacement of particles in between a pair of images. Two different motion types i.e. the linear motion and the circular motion were tracked using the particle tracking algorithm and the probability that the error of the displacement measurement was greater than or equal to 20% was monitored.

The particle tracking algorithm calculates the displacements of the particles in between two successive images by cross-correlating the interrogation window of one image to that of the other image of the image pair. Circular objects were detected in the first image of the image pair and the cartesian coordinates of their centers were rounded off to a nearest integer values. Interrogation windows were created in both the images that were centered about the detected centers of each circles in the first image of the image pair. The detected radius of each circle was also rounded off to a nearest integer that is greater or equal to the radius of the detected circle to define the size of the window in the first image of the image pair. Size of the interrogation window in the first image depends on the size of detected circle whereas the window size in the second image depends on the value of  $r$ . The  $r$  value is a numeric variable that was used to define the size of window in the second image of the image pair with respect to its size in the first image. In the current code, the default value of  $r$  was set to 0.5 but was changed based on the displacement being measured. When the value of  $r$  was 0.5, the interrogation window in the first

image was increased by half its size on both sides in each axis to create the interrogation window in the second image of image pair. Therefore, the size of the interrogation window in the second image was twice the size of the window in the first image at  $r = 0.5$ . When higher displacements of particles in between the image pair were measured, the  $r$  value was increased to 1 or 2.

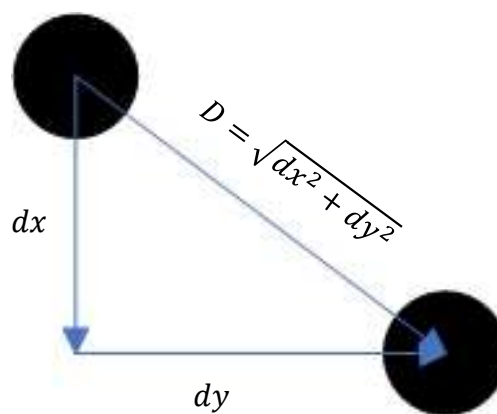
#### 4.2.4.1. Linear motion

During linear motion, high errors were observed when low displacements of particles in between the image pair were measured. The size of the interrogation window created in the first image was an integer value because it was equal to twice the radius of detected circle that is rounded off to the nearest integer greater than or equal to the radius. The value of  $r$  used in the code for tracking linear motion was either 0.5, 1, or 2, all of which result in an integer sized interrogation window in the second image of the image pair. The `normxcorr2` function calculates the normalized 2-D cross-correlation between the two windows extracted from an image pair as discussed in the “Particle tracking algorithm” sub-section of the “Methods” section. The peak of the cross-correlation lies at the pixel location that is an integer value, where the window of the first image (i.e. template) has the maximum cross-correlation coefficient with respect to the window of the second image (i.e. image) of the image pair, as displayed in Figure 46. The displacements calculated using the equations in the “Particle tracking algorithm” section of “Appendix A” resulted in integer displacement values. If  $dx$  is the displacement in x-direction and  $dy$  is the displacement in y-direction then,  $dx$  and  $dy$  values are always integers based on the current algorithm.



**Figure 46. The detection of a new location of the particle using the normxcorr2 function in MATLAB. The peak of the cross-correlation lies at the integer pixel location where there is maximum correlation coefficient between the matrix template and the image. The peak of the cross-correlation is used to calculate the displacement of particle in both the direction.**

The 2-D linear displacement of the particle in between the image pair was calculated using the Pythagoras theorem as demonstrated in Figure 47. The 2-D displacement of the particle is expressed as  $D$  and calculated by using the displacements of the particle in both the x and y-directions. When a particle has a certain displacement in any direction with an integer magnitude, it does not necessarily mean that the magnitudes of displacements of the particle in both the directions are also integers. However,



**Figure 47. Pythagoras theorem used to calculate the displacement of particles in between a pair of images.**

the displacements detected by the current code in each direction are always integers. Therefore, the calculated displacements possess some error that occur because of the rounding off and the integer sizes of windows. For example, when a displacement of 2 pixels was detected, the particles can only have integer displacement of 0 or 2 in either direction when they were travelling along the axial direction. The displacement of all the other particles were not integers in either direction. The error is small in magnitude and only significant when small displacements are measured, while the error seems to vanish when larger displacements are measured using the particle tracking algorithm.

The `normxcorr2` function sometimes detected multiple peaks during cross-correlation between two windows. Both windows were observed when such an error occurred, which revealed that most of the encountered multiple peaks were one pixel apart from each other. Therefore, the first peaks were selected as the detected location for those cases. Such practice could lead to an error, but the error was about a pixel in value because the detected multiple peaks were adjacent to each other in most cases. Again, such small errors are significant when measuring small displacements of 2 or 3 pixels.

#### **4.2.4.2. Circular motion**

The possible causes of error of the displacement measured using the particle tracking algorithm during a circular motion were also analyzed. A comprehensive error analysis study was conducted to analyze the relation between the error of the measured displacement during a circular motion and the distance of the particles from the center of rotation. The tests designed for the 2,000  $\mu\text{m}$  melt pool portraying circular motion were re-run and the distance of each particle from the center of the melt pool (or the binary image) was observed. The goal was to determine whether the error of displacement measurement during a circular motion depended

upon the distance of the particles from the center of the rotation or not. A detailed study report of the error analysis of circular motion is presented in the Appendix A, and has the following conclusions:

- No concrete evidence exists to indicate whether the error is high when the particles are near to or far from the center of rotation.
- The particles that are in the mid-range, though have a low probability of error, sometimes generate high error of measurement.
- The error of displacement measurement during circular flow is almost constant and does not follow a pattern.

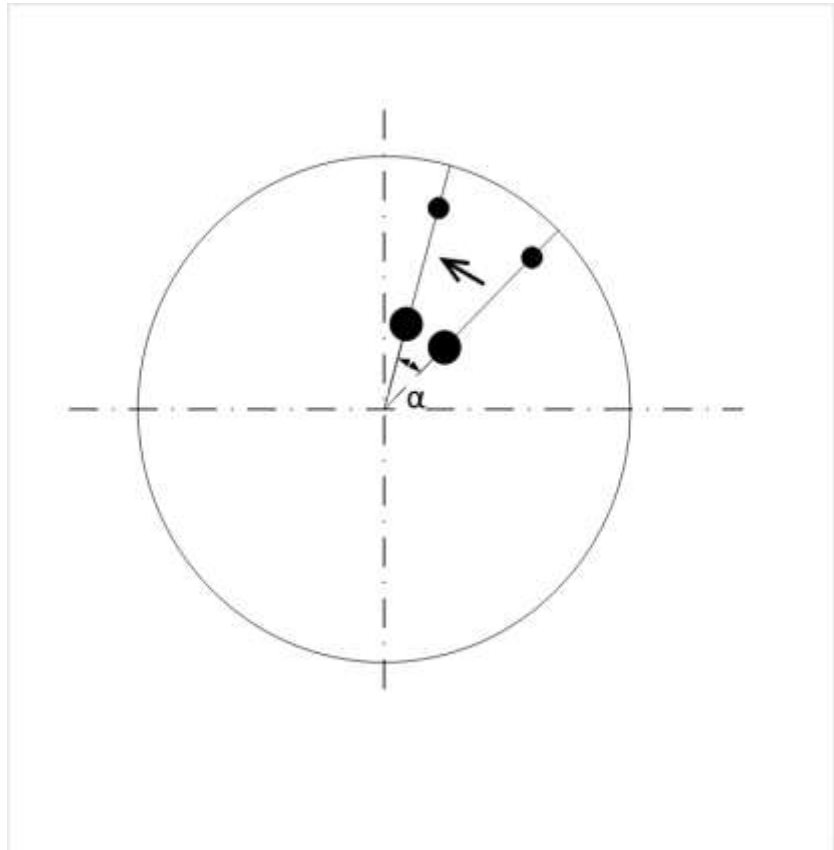
The linear displacements of the particles in a circular motion are not equal and depend upon the distance of particles from the center of rotation. However, the size of the interrogation window for each circular object in the current code is only dependent on its detected radius. The interrogation windows in the second image are centered about the detected center of the circular objects in the first image of the image pair. Therefore, the interrogation window in the second image of the image pair must be large enough to accommodate the displaced object. The size of the interrogation window in the second image depends only on the size of the window in the first image and the  $r$  value in the current algorithm. The  $r$  value is the same for all the particles in the binary image pair. Therefore, having the same window sizes in the second image, even though the particles have different linear displacements based on their distances from the center of rotation, can result in erroneous detection. The window in the second image might not be large enough to accommodate the movement of a particle that is small but away from the center of rotation during a circular motion. In addition, the cross-correlation can detect other particles as



the tracked particle if similar sized particles appear in the interrogation window of the second image of the image pair.

Figure 48 considers an example where two particles are in a circular motion with equal angular displacement of  $\alpha$ . The bigger particle is closer to the center whereas the smaller particle is away from the center.

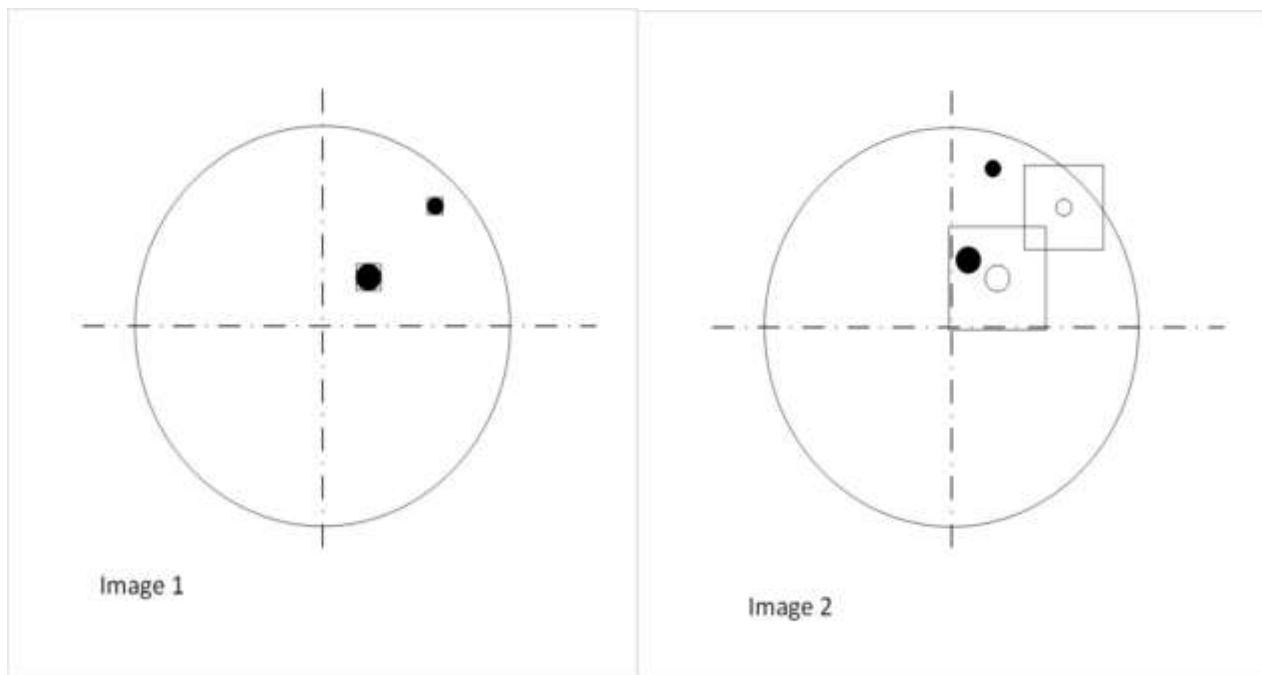
Suppose there are two images representing the motion in Figure 48 that are named image 1 and image 2. Now, interrogation windows are created in both the images by the particle tracking algorithm



**Figure 48. An example of circular motion of two particles with same angular displacement of  $\alpha$ . The bigger particle is near the center of rotation while the smaller particle is away from the center.**

around the centers of the detected circular particles in image 1. The size of the interrogation window in image 1 is equal to the diameter of the detected circle, as displayed in Figure 49. The size of interrogation window in the image 1 is increased by a common factor  $r$  to determine the size of interrogation window in image 2. The value of  $r$  in this example is equal to 2 and results in interrogation windows in image 2 that have 5 times the size of the windows in image 1. The interrogation windows in images 2 are also centered about the centers of detected circles in image 1 as displayed in Figure 49. The bigger circular particle is inside the interrogation window

whereas the smaller one is outside the interrogation window in image 2. In such situation, if there is another particle of similar size to the smaller circular particle that appears in the interrogation window of that particle in image 2, the code will detect that particle as the tracked particle and the peak of correlation will be at that location. Such error can be encountered when tracking circular motion in a binary image pair by using the current algorithm.



**Figure 49.** Creation of interrogation windows in an image pair by the particle tracking algorithm to track a circular motion of particles. The interrogation windows in both the images are created around the center of detected circles in in image 1. The size of the interrogation window in image 1 is equal to the diameter of detected circle whereas the size of the interrogation window in image 2 depend upon the value or  $r$  and the size of the window in image 1. The value of  $r$  in this example is 2, which creates interrogation window in image 2 whose size is five time the size of the window in image 1.

## 5. Summary and conclusions

### 5.1. Executive summary

Existing work on SLM, Marangoni convection, and the use of high-speed visualization of a melt/weld pool were reviewed. The needs were identified as the basic understanding of 1) the relation between processing parameters and melt pool flow and 2) the relation between melt pool flow and various SLM defects [2]. In order to understand the melt pool flow of a SLM, the melt pool must be visualized by using a high-speed camera system, and the recorded high-speed images should be analyzed by tracking the movement of the surface particles using a particle tracking algorithm.

The high-speed images of a melt pool of SLM were simulated using a MATLAB script. A particle tracking algorithm was developed using various functions from the MATLAB Image Processing Toolbox (MIPT), an add-on for MATLAB. The size of the melt pool, the image pixel size, the number of surface oxides, the flow path, and the size of surface oxides were the various factors assumed to affect the particle tracking algorithm. A test plan was designed and optimized using Design Expert 12 to analyze the effect of the relevant factors on the error of displacement measured by using the particle tracking algorithm.

Binary images were simulated with different levels of each factor and analyzed using the particle tracking algorithm. The linear displacements were measured in pixels during linear motions whereas angular displacements were measured in radians during circular motion. The linear motion was divided into radial outward motion and radial inward motion because of the difference in the magnitude and direction of the displacements of each. ANOVA was performed to determine if any factor had a significant impact on the error of displacement measured using the particle tracking algorithm.

## 5.2. Conclusions

The experimental results, ANOVA, and the historical model validation distinguished the factors that had a significant influence on the error of displacement measured by using the particle tracking algorithm from the insignificant factors. The cause of error of displacement measurement for the two different types of motion in a melt pool i.e. linear and circular motion were analyzed. The important conclusions from the current research project are as follows:

- The velocity of the particles in between the image pairs and the flow type were the two significant factors that affected the particle tracking algorithm. Other factors considered in the study were not significant.
- The melt pool size appeared as a significant factor in the historical analysis of the models, however the two melt pool sizes in the initially designed experiments had different particle velocities. The first independent study used different velocities of particles for a constant melt pool size and the result proved that the error of displacement measured using the particle tracking algorithm is dependent on the velocity and not the melt pool size.
- ANOVA identified the number and the size of surface oxides as significant factors during their initial studies. However, a second independent study was performed at a constant velocity. The results were used to conclude that both these factors were insignificant in the model.
- The percentage error of displacement measurement was high when measuring small linear displacements because displacements in pixel integer values were reported by the code in both the x and y directions. The actual displacement calculated using Pythagoras theorem had a significant error while measuring

small displacements, but the error was not significant when large displacements were measured.

- The error of displacement measurement during circular flow was nearly constant and did not follow a particular pattern. The error of displacement measurement during a circular flow might be present due to the use of similar sized interrogation windows for particles with different linear displacements.
- Detection of multiple peaks by the cross-correlation led to minor error that became substantial when measuring small displacements.
- The measurement error of the particle tracking algorithm has been quantified and validated against simulated data, therefore it can be used to analyze the actual melt pool data with known confidence.

### 5.3. Recommendations for future work

The particle tracking algorithm was accurate in tracking particles with large displacements during a linear flow. However, the accuracy of the code in tracking particles with small linear displacement and during circular motion can be improved. Some of the recommendations for future research to improve the accuracy of the particle tracking algorithm and increase its application scope are as follows:

1. Linear motion:

Work on improving the accuracy of the particle tracking algorithm when measuring small displacements. Perform the cross-correlation without using the rounding functions (round and ceil).

2. Circular motion:

Analyze of the effect of angular velocity on the error of displacement measured using the particle tracking algorithm. This study will give evidence of whether the flow type is a significant factor or not. Furthermore, the size of the interrogation window in the second image can be made a function of both the size of detected circle and its distance from the center to possibly reduce the error of angular displacement measurement.

3. Imfindcircles function:

Change the input arguments such as sensitivity and edge threshold to detect more circular particles than from default settings and analyze their effect on the accuracy of displacement measurement.

4. Noisy images:

Use of the particle tracking algorithm to analyze images that have noise. The noise might distort the shape of circular particles.

## 6. References

- [1] T. Wohlers and T. Gornet, "Wholers Report," Wohlers Associates Inc., Fort Collins, CO, USA, 2014. Accessed: Dec. 11, 2020. [Online]. Available: <http://www.wohlersassociates.com/history2014.pdf>
- [2] P. Gautam, T. O. Winsor, I. Kulseng-Hansen, and P. Lucon, "Marangoni Convection in Selective Laser Melting (SLM) of 316L Stainless Steel," in *Proc. of the 2019 Additive Manuf. with Powder Metallurgy Conf.*, Phoenix, AZ, Jun. 24, 2019, pp. 32–50.
- [3] H. K. D. H. Bhadeshia, "Additive manufacturing," *Mater. Sci. and Technol.*, vol. 32, no. 7, pp. 615–616, 2016, doi: 10.1080/02670836.2016.1197523.
- [4] S. H. Huang, P. Liu, A. Mokasdar, and L. Hou, "Additive manufacturing and its societal impact: A literature review," *Int. J. Adv. Manuf. Technol.*, vol. 67, no. 5–8, pp. 1191–1203, 2013, doi: 10.1007/s00170-012-4558-5.
- [5] F. Bechmann, "Changing the future of additive manufacturing," *Met. Powder Rep.*, vol. 69, no. 3, pp. 37–40, 2014, doi: 10.1016/S0026-0657(14)70135-3.
- [6] T. D. Ngo, A. Kashani, G. Imbalzano, K. T. Q. Nguyen, and D. Hui, "Additive manufacturing (3D printing): A review of materials, methods, applications and challenges," *Compos. Part B Eng.*, vol. 143, no. February, pp. 172–196, 2018, doi: 10.1016/j.compositesb.2018.02.012.
- [7] F. H. Kim and S. P. Moylan, "Literature Review of Metal Additive Manufacturing Defects," *NIST Adv. Manuf. Ser. 100–16*, 2018, doi: 10.6028/NIST.AMS.100-16.
- [8] C. R. Heiple, J. R. Roper, R. T. Stagner, and R. J. Aden, "Surface Active Element Effects on the Shape of GTA, Laser, and Electron Beam Welds," *Weld. J.*, vol. 62, no. 3, pp. S72–S77, 1983.
- [9] C. Limmaneevichitr and S. Kou, "Visualization of Marangoni convection in simulated weld pools containing a surface-active agent," *Weld. J.*, vol. 79, no. 11, pp. 324S-330S, 2000.
- [10] M. Rombouts, J. P. Kruth, L. Froyen, and P. Mercelis, "Fundamentals of selective laser melting of alloyed steel powders," *CIRP Ann. - Manuf. Technol.*, vol. 55, no. 1, pp. 187-192, 2006, doi: 10.1016/S0007-8506(07)60395-3.
- [11] X. Zhou, X. Liu, D. Zhang, Z. Shen, and W. Liu, "Balling phenomena in selective laser melted tungsten," *J. Mater. Process. Technol.*, vol. 222, pp. 33-42, 2015, doi: 10.1016/j.jmatprotec.2015.02.032.
- [12] T. Campbell, C. Williams, O. Ivanova, and B. Garrett, "Could 3D Printing Change the World? Technologies, Potential, and Implications of Additive Manufacturing," Atlantic Council, Washington, DC, USA, Strategic Foresight Report, 2011. [Online]. Available: [https://www.atlanticcouncil.org/wp-content/uploads/2011/10/101711\\_ACUS\\_3DPrinting.PDF](https://www.atlanticcouncil.org/wp-content/uploads/2011/10/101711_ACUS_3DPrinting.PDF)
- [13] ISO/ASTM52900-15, "Standard Terminology for Additive Manufacturing - General Principles - Terminology," ASTM International, West Conshohocken, PA, USA, 2015. Accessed: Dec. 11, 2020. [Online]. Available: <http://www.astm.org/cgi-bin/resolver.cgi?ISOASTM52900-15>
- [14] W. J. Sames, F. A. List, S. Pannala, R. R. Dehoff, and S. S. Babu, "The metallurgy and processing science of metal additive manufacturing," *Int. Mater. Rev.*, vol. 61, no. 5, pp. 315–360, 2016, doi: 10.1080/09506608.2015.1116649.
- [15] J. P. Kruth, L. Froyen, J. Van Vaerenbergh, P. Mercelis, M. Rombouts, and B. Lauwers,

- “Selective laser melting of iron-based powder,” vol. 149, no. 1-3, pp. 616-622, 2004, doi: 10.1016/j.jmatprotec.2003.11.051.
- [16] H. Lee, C. H. J. Lim, M. J. Low, N. Tham, V. M. Murukeshan, and Y. J. Kim, “Lasers in additive manufacturing: A review,” *Int. J. of Precis. Eng. and Manuf. - Green Tech.* 4, vol. 4, no. 3, pp. 307-322, 2017, doi: 10.1007/s40684-017-0037-7.
- [17] M. Yakout, A. Cadamuro, M. A. Elbestawi, and S. C. Veldhuis, “The selection of process parameters in additive manufacturing for aerospace alloys,” *Int. J. Adv. Manuf. Technol.*, vol. 92 (5-8), pp. 2081-2098, 2017, doi: 10.1007/s00170-017-0280-7.
- [18] B. Zhang, L. Dembinski, and C. Coddet, “The study of the laser parameters and environment variables effect on mechanical properties of high compact parts elaborated by selective laser melting 316L powder,” *Mater. Sci. Eng. A*, vol. 584, pp. 21-31, 2013, doi: 10.1016/j.msea.2013.06.055.
- [19] E. Liverani, S. Toschi, L. Ceschini, and A. Fortunato, “Effect of selective laser melting (SLM) process parameters on microstructure and mechanical properties of 316L austenitic stainless steel,” *J. Mater. Process. Technol.*, vol. 249, pp. 255-263, 2017, doi: 10.1016/j.jmatprotec.2017.05.042.
- [20] Z. Sun, X. Tan, and S. B. Tor, “Effects of chamber oxygen concentration on microstructure and mechanical properties of stainless steel 316L parts by selective laser melting,” in *Proc. 3rd Int. Conf. Progress Addit. Manuf.*, C. K. Chua, W. Y. Young, M. J. Tan, E. Lui, and S. B. Tor, 2018, pp. 470-475, doi: 10.25341/D4CS38.
- [21] U. Scipioni Bertoli, A. J. Wolfer, M. J. Matthews, J. P. R. Delplanque, and J. M. Schoenung, “On the limitations of Volumetric Energy Density as a design parameter for Selective Laser Melting,” *Mater. Des.*, vol. 113, pp. 331-340, 2017, doi: 10.1016/j.matdes.2016.10.037.
- [22] E. O. Olakanmi, “Selective laser sintering/melting (SLS/SLM) of pure Al, Al-Mg, and Al-Si powders: Effect of processing conditions and powder properties,” *J. Mater. Process. Technol.*, vol. 213, no. 8, pp. 1387-1405, 2013, doi: 10.1016/j.jmatprotec.2013.03.009.
- [23] T. Kurzynowski, E. Chlebus, B. Kuźnicka, and J. Reiner, “Parameters in selective laser melting for processing metallic powders,” in *Proc. SPIE 8239, High Power Laser Materials Processing: Lasers, Beam Delivery, Diagnostics, and Applications, 823914*, , San Francisco, CA, USA, 2012, doi: 10.1117/12.907292.
- [24] S. L. Anna. (2009). Interfaces and Multiphase Flows in Microfluidics [Presentation]. Available: [https://www.ima.umn.edu/materials/2009-2010/T12.5-6.09/8894/SLA\\_IMA\\_Tutorial\\_Interfaces\\_Drops\\_Multiphase.pdf](https://www.ima.umn.edu/materials/2009-2010/T12.5-6.09/8894/SLA_IMA_Tutorial_Interfaces_Drops_Multiphase.pdf)
- [25] P. Sahoo, T. Debroy, and M. J. McNallan, “Surface tension of binary metal-surface active solute systems under conditions relevant to welding metallurgy,” *Metall. Trans. B*, vol. 19 (3), pp. 483-491, 1988, doi: 10.1007/BF02657748.
- [26] C. Limmaneevichitr and S. Kou, “Visualization of Marangoni convection in simulated weld pools,” *Weld. J. (Miami, Fla)*, vol. 79, no. 5, pp. 126S-135S, 2000.
- [27] C. Y. Yap *et al.*, “Review of selective laser melting: Materials and applications,” *Applied Physics Reviews.*, vol. 2. no. 4 2015, doi: 10.1063/1.4935926.
- [28] B. Zhang, Y. Li, and Q. Bai, “Defect Formation Mechanisms in Selective Laser Melting: A Review,” *Chinese Journal of Mechanical Engineering (English Edition).*, vol. 30 (3), pp. 515-527, 2017, doi: 10.1007/s10033-017-0121-5.
- [29] M. Wilhelm, K. D. Wissenbach, and A. D. Gasser, “Shaped body especially prototype or replacement part production,” Germany, Patent DE19649865C1, Feb. 12 1998.



- [30] C. X. Zhao *et al.*, “The effect of oxygen on transitional Marangoni flow in laser spot welding,” *Acta Mater.*, vol. 58, no. 19, pp. 6345-6357, 2010, doi: 10.1016/j.actamat.2010.07.056.
- [31] M. M. Pariona, A. F. Taques, and L. A. Woiciechowski, “The Marangoni effect on microstructure properties and morphology of laser-treated Al-Fe alloy with single track by FEM: Varying the laser beam velocity,” *Int. J. Heat Mass Transf.*, vol. 119, pp. 10-19, 2018, doi: 10.1016/j.ijheatmasstransfer.2017.11.097.
- [32] M. Mizutani, S. Katayama, and A. Matsunawa, “Observation of molten metal behavior during laser irradiation - Basic experiment to understand laser welding phenomena,” in *Proc. SPIE 4831, First Int. Symp. High-Power Laser Macroprocessing*, Osaka, Japan, 2003, doi: 10.1117/12.497909.
- [33] P. Henrikson, “Visualization of weld pool surface flow during TIG welding on stainless steel 316L plates,” *Math. Model. Weld Phenom.* 7, pp. 125–147, 2005.
- [34] C. X. Zhao, V. Van Steijn, I. M. Richardson, Z. Saldi, and C. R. Kleijn, “Experimental Characterization of GTA Weld Pool Surface Flow Using PIV,” in *Proc. 8th Int. Conf. Trends Weld. Res.*, Pine Mountain, GA, USA, pp. 201-210, 2009, doi: 10.1361/cp2008twr201.
- [35] C. Zhao and I. M. Richardson, “Complex flow motions during laser welding,” in *40th AIAA Plasmadynamics and Lasers Conference*, San Antonio, TX, USA, 2009, doi: 10.2514/6.2009-3739.
- [36] C. Zhao, “Measurements of fluid flow in weld pools,” Ph.D. dissertation, Dept. Mat. Sci. and Eng., Delft University of Technology, Delft, The Netherlands, 2011.
- [37] V. Gunenthiram *et al.*, “Experimental analysis of spatter generation and melt-pool behavior during the powder bed laser beam melting process,” *J. Mater. Process. Technol.*, vol. 251, pp. 375-386, 2018, doi: 10.1016/j.jmatprotec.2017.08.012.
- [38] N. Otsu, “A Threshold Selection Method from Gray-Level Histograms,” *IEEE Trans. Syst. Man. Cybern.*, vol. 9, no. 1, pp. 62–66, 1979, doi: 10.1109/TSMC.1979.4310076.
- [39] MathWorks®, “Image Processing Toolbox™: User’s Guide (R2020b),” 2020. Accessed: Dec. 11, 2020. [Online]. Available: [https://www.mathworks.com/help/pdf\\_doc/images/images\\_ug.pdf](https://www.mathworks.com/help/pdf_doc/images/images_ug.pdf).
- [40] Arthur Coste, “Image Processing Project 4 Hough Transform,” The University of Utah, Salt Lake City, UT, USA, CS6640, Nov. 2012. Accessed: Dec. 11, 2020. [Online]. Available: [http://www.sci.utah.edu/~acoste/uou/Image/project4/ArthurCOSTE\\_Project4.pdf](http://www.sci.utah.edu/~acoste/uou/Image/project4/ArthurCOSTE_Project4.pdf)
- [41] V. K. Yadav, S. Batham, A. K. Acharya, and R. Paul, “Approach to accurate circle detection: Circular Hough Transform and Local Maxima concept,” in *2014 Int. Conf. Electron. Commun. Syst. ICECS*, Coimbatore, Tamil Nadu, India, Feb. 2014, doi: 10.1109/ECS.2014.6892577.
- [42] J. Westerweel, “Digital Particle Image Velocimetry - Theory and Application,” Ph.D. dissertation, Dept. Mech. Maritime and Mat. Eng., Delft University of Technology, Delft, The Netherlands, 1993.
- [43] L. Sarno, A. Carravetta, Y. C. Tai, R. Martino, M. N. Papa, and C. Y. Kuo, “Measuring the velocity fields of granular flows – Employment of a multi-pass two-dimensional particle image velocimetry (2D-PIV) approach,” *Adv. Powder Technol.*, vol. 29, no. 12, pp. 3107-3123, 2018, doi: 10.1016/j.appt.2018.08.014.
- [44] D. Lin, J. Grundmann, and A. Eltner, “Evaluating Image Tracking Approaches for Surface

- Velocimetry With Thermal Tracers,” *Water Resour. Res.*, vol. 55, no. 4, pp. 3122-3136, 2019, doi: 10.1029/2018WR024507.
- [45] A. K. Prasad, “Particle image velocimetry,” *Curr. Sci.*, vol. 79, no. 1, pp. 51-60, 2000.
- [46] P. Koutalakis, O. Tzoraki, and G. Zaimes, “UAVs for Hydrologic Scopes: Application of a Low-Cost UAV to Estimate Surface Water Velocity by Using Three Different Image-Based Methods,” *Drones*, vol. 3 (1), pp. 14-28, 2019, doi: 10.3390/drones3010014.
- [47] D. Dabiri, “Cross Correlation Digital Particle Image Vecolimetry - A Review,” University of Washington, Seattle, WA, USA, 2006. Accessed: Dec. 11, 2020. [Online]. Available: <https://www.aa.washington.edu/sites/aa/files/faculty/dabiri/pubs/piV.Review.Paper.final.pdf>.

## 7. Appendix A

### 7.1. Particle tracking algorithm

In MATLAB, images are stored as matrices and each pixel is specified by its row number followed by column number. The measure of row number starts from the top left corner and increases in a vertically downward direction, which is taken as x direction. The y direction is taken as the horizontal direction from left to right representing the column count for that pixel. The displacements of circular object in x and y directions are  $dx$  and  $dy$  respectively.

From Figure 12,

x-coordinates of the top edge of window in image 1 =  $x_{min}$

x-coordinates of the bottom edge of window in image 1 =  $x_{max}$

y-coordinates of the left edge of window in image 1 =  $y_{min}$

y-coordinates of the right edge of window in image 1 =  $y_{max}$

x-coordinates of the center of the detected circle in image 1 =  $k_1$

y-coordinates of the center of the detected circle in image 1 =  $h_1$

width of the window in image 1 =  $w\_width = size(template, 1)$

height of the window in image 1 =  $w\_height = size(template, 2)$

$x\_max\_dis$  and  $y\_max\_dis$  are the distances in x and y directions by which the size of interrogation window in the first image is increased in both the directions to form a window in the second image. Therefore,  $x\_max\_dis$  is the distance between the top edges of the windows in images 1 and 2 and  $y\_max\_dis$  is the distance between the left edges of the windows in images 1 and 2. The displacements in the x and y directions are calculated using Equations 10 and 11.

$$dx = x_{min} - x_{max\_dis} + x_{offset} - x_{min} \quad \text{Equation 10}$$

$$dy = y_{min} - y_{max\_dis} + y_{offset} - y_{min} \quad \text{Equation 11}$$

The Equations 8 and 9 are then combined with Equations 10 and 11 to generate Equations 12 and 13 that will be used in the script to calculate the displacements of particles in the x and y direction.

$$dx = x_{peak} - x_{max\_dis} - w\_width \quad \text{Equation 12}$$

$$dy = y_{peak} - y_{max\_dis} - w\_height \quad \text{Equation 13}$$

## 7.2. Error analysis of circular motion

Run	Nearest circle	Farthest circle	In-between
1	13 pixels: Has a higher error in most cases but not always greater than 20% 30 pixels: error is sometimes high and sometimes low but rarely >20%	109: sometimes high and rarely >20% (once) 99: sometimes high and rarely >20% (once)	Mostly very less error but every once in a while, shows high error >20% to very high error
2	21 pixels: Sometimes high error and rarely higher than 20%: once 29, 30,33 pixels: Sometimes high error and rarely higher than 20%: once for 29 and 30	87 pixels: Generally low error but once very high >20% 77,78 pixels: Usually very low to low error but high error >20% once at 78 (measured 0 radians) 74: Mostly low error but once high error >20%	Mostly low error but some have high error of >20%

3	<p>28,29 pixels: Slight high error but only once &gt;20% at 29.</p> <p>41 pixels: Some high error with once &gt;20%</p>	<p>95 pixels: Low error with once high error &gt;20%</p> <p>88,86,85 pixels: Few high errors and only once &gt;20% at 85</p>	<p>Mostly low error but some have high error of &gt;20%</p>
4	<p>29 pixels: Error is sometimes high and only once &gt;20%</p> <p>38,38,39 pixels: Sometimes high and few &gt;20%: twice at 39 pixels and once at 38</p>	<p>94,93 pixels: Mostly low twice greater than 20%, once each at 94 and 93.</p> <p>83,80: Mostly low twice greater than 20%, once each at 83 and 80</p>	<p>Mostly low error and rarely have high error of &gt;20%: Twice</p>
5	<p>58 pixels: rarely high error never &gt;20%</p> <p>61 pixels: rarely high error and once &gt;20%</p>	<p>98 pixels: rarely high error and once &gt;20%</p>	<p>Mostly low but twice &gt;20%</p>
6	<p>35 pixels: Sometimes high error and once &gt;20%</p>	<p>95 pixels: Mostly low and once &gt;20%</p> <p>91 pixels: Mostly low and once &gt;20%</p>	<p>Mostly low error but occasionally high error of &gt;20%</p>

	<p>39 pixels: low error and once &gt;20%</p> <p>46,48 pixels: mostly low error but twice &gt;20%: once each at 46 and 48.</p>	<p>88 pixels: Mostly low and once &gt;20%</p>	
7	<p>24 pixels: many high errors but only once &gt;20%</p> <p>44,45 pixels: mostly low and Twice &gt;20: once each at 44 and 45</p>	<p>77 pixels: Mostly low but once &gt;20%</p> <p>69 pixels: Mostly low but once &gt;20%</p>	<p>Mostly low but twice &gt;20%</p>
8	<p>22 pixels: High error but only once &gt;20%</p> <p>39 pixels: Sometimes high error but only once &gt;20%</p> <p>44,45, 46 pixels: Mostly low. But twice</p>	<p>84 pixels: Mostly low but once &gt;20%</p> <p>75 pixels: Mostly low but once &gt;20%</p> <p>73 pixels: Mostly low and never &gt;20%</p>	<p>Mostly low but thrice &gt;20%</p>

	>20 at 46 and once at 44		
9	25 pixels: Sometimes high error but never >20% 33 pixels: Many high errors but only once >20% 38 pixels: Sometimes high error but only once >20%	107 pixels: sometimes high error but never >20% 92 pixels: Mostly low and never >20% 78 pixels: Mostly low but once >20%	Mostly very less error but every once in a while, shows high error >20% to very high error
10	36 pixels: Sometimes high error but only once >20% 58,59 pixels: Mostly low but twice >20%: one each at 58 and 89 pixels	104 pixels: Rarely high but none >20% 78,76 pixels: Mostly low but twice >20%: one each at 78 and 76 pixels	Mostly low but twice >20%
11	28 pixels: Sometimes high error but only once >20%	86,85 pixels: Mostly low but once >20% at 86	Mostly low but every once in a while, shows high



	<p>32 pixels: High errors but only once &gt;20%</p> <p>38 pixels: Sometimes high error but only once &gt;20%</p>	<p>77 pixels: Mostly low and never &gt;20%</p>	<p>error &gt;20% to very high error</p>
12	<p>28 pixels: high errors but only once &gt;20%</p> <p>35 pixels: Sometimes high error but only once &gt;20%</p> <p>37,38 pixels: Mostly low. But once &gt;20 at 37 and once at 38</p>	<p>85 pixels: Mostly low and never &gt;20%</p> <p>74 pixels: Low and never &gt;20%</p> <p>71,70 pixels: Mostly low twice greater than 20%, once each at 71 and 70</p>	<p>Mostly low but thrice &gt;20%</p>
13	<p>17,18 pixels: High errors but only twice &gt;20%: once each at 17 and 18 pixels</p> <p>38,38 pixels: sometimes high but twice &gt;20% once each</p>	<p>93 pixels: Mostly low and never &gt;20%</p> <p>85 pixels: Mostly low but once &gt;20%</p> <p>82 pixels: Mostly low but once &gt;20%</p>	<p>Mostly low but every once in a while, shows high error &gt;20% to very high error</p>

## 8. Appendix B

### 8.1. Particle tracking MATLAB script

#### 8.1.1. Linear motion

```

%%-RTR Research Group Template-----
-----%
% Created By:    Prakash Gautam
% Program Name:  PIV_test.m
% Created On:    2020-05-07
% Modified By:Prakash Rev:2 Modified    Name:PIV    Modified Date:7/14/2020
Notes:-----%
% Peter A. Lucon____000_____Matlab_Template_r00_____2020-05-
04___: Made the Template.
% Prakash Gautam____001_____Matlab_Template_r00_____2020-05-
08___:
% Added script
% Prakash Gautam____002_____Matlab_Template_r00_____2020-07-
14___:
% Used imfindcircles function to detect the circles first and then built
% window around the centers of circles
% Prakash Gautam____003_____Matlab_Template_r00_____2020-07-
14___:
% Used imfindcircles function to detect the circles first and then built
% window around the centers of circles
% Prakash Gautam____004_____Matlab_Template_r00_____2020-07-
14___:
% Used imfindcircles function to detect the circles first and then built
% window around the centers of circles
% Prakash Gautam____005_____Matlab_Template_r00_____2020-07-
23___:
% used the coordinates of center of each detected circle to plot the vector
% plot using quiver

%% Program Description-----
-----%
% This program first defines all the variables that are used to compute
% cross-correlation of two binary images
disp('Program Running')
%% Conversion Variables
ft_m = 0.3048;                %'m/ft'    % Converts ft to meters
%% Initialize Program-----
-----%

clear all;
clc;
close all;
%Reading images and specifying origin
image1 = imread('meltpool15.tif');
image2 = imread('meltpool16.tif');
%% Variables-----
-----%

[Nx,Ny] = size(image1);

```

```

%detection of circles using imfindcircles function that output the
%coordinates of center and the radius of detected circles
[centers1,radii1,metric1] = imfindcircles(image1,[6
20], 'ObjectPolarity', 'bright', 'Sensitivity',0.85, 'EdgeThreshold',0.3);
[centers2,radii2,metric2] = imfindcircles(image2,[6
20], 'ObjectPolarity', 'bright', 'Sensitivity',0.85, 'EdgeThreshold',0.3);

n=numel(radii1);

%Test matrices are created in both the template(smaller image) and the larger
image based on the size of window
%and the search zone. The point of maximum correlation specifies the
%location in larger images where the template would match and has the highest
correlation.
%the final average displacement can be calculated based on the peaks.
%Initially, the test images and the displacements are initialized to zero.

test1= 0;
test2 = 0;
dx(n) = 0; %displacement
dy(n) = 0;
xpeak =0;
ypeak = 0;

%Co-ordinates of the center of detected circles in both the images
h1=round(centers1(:,1));
k1=round(centers1(:,2));
h2=round(centers2(:,1));
k2=round(centers2(:,2));
k=Nx-k1;
h3=transpose(h1);
k3=transpose(k);

%% Solution-----%
-----%
%c0 to count the windows
%test matrices extracted from both the images and cross correlated
for c0=1:(n)
    max_correlation = 0;
    w_width=2*ceil(radii1(c0));
    w_height=2*ceil(radii1(c0));

    %Range of search window in image 2
    r=0.5; %search zone on either side of the window
    x_max_dis = r*w_width;
    y_max_dis = r*w_height;

    %specifying the location of interrogation windows
    xmin = k1(c0)-(w_width/2);
    xmax = k1(c0)+(w_width/2-1);
    ymin = h1(c0)-(w_height/2);
    ymax = h1(c0)+(w_height/2-1);

    %creation of interrogation windows in both the images

```

```

    test1 = image1(xmin:xmax,ymin:ymax);
    test2 = image2((xmin-x_max_dis):(xmax+x_max_dis),(ymin-
y_max_dis):(ymax+y_max_dis));
    correlation = normxcorr2(test1,test2);

    [xpeak,ypeak] = find(correlation==max(correlation(:)));

    %calculation of displacements in x and y directions

    dx(c0) = xpeak-w_width-x_max_dis;
    dy(c0) = ypeak-w_height-y_max_dis;
end
d=sqrt(dx.^2+dy.^2); %displacement using Pythagoras theorem

%vector plot
h=quiver(h3,k3,dy,-dx);
%h.ShowArrowHead='off'; %arrow heads can be turned off

%specifying x and y limits of the plot
xlim([0 Nx])
ylim([0 Ny])

%h.AutoScale='off';;%auto scaling of the magnitude of vectors can be turned
off

%color of arrows can be specified
h.Color='red';
%% Closing Program-----%
-----%
disp('Program Finished')
```

## 8.1.2. Circular motion

```

%%-RTR Research Group Template-----
-----%
% Created By:   Prakash Gautam
% Program Name: PIV_test.m
% Created On:   2020-05-07
% Modified By:Prakash Rev:2 Modified   Name:PIV   Modified Date:7/14/2020
Notes:-----%
% Peter A. Lucon_____000_____Matlab_Template_r00_____2020-05-
04___: Made the Template.
% Prakash Gautam_____001_____Matlab_Template_r00_____2020-05-
08___:
% Added script
% Prakash Gautam_____002_____Matlab_Template_r00_____2020-07-
14___:
% Used imfindcircles function to detect the circles first and then built
% window around the centers of circles
% Prakash Gautam_____003_____Matlab_Template_r00_____2020-07-
14___:
% Used imfindcircles function to detect the circles first and then built
% window around the centers of circles
% Prakash Gautam_____004_____Matlab_Template_r00_____2020-07-
14___:
% Used imfindcircles function to detect the circles first and then built
% window around the centers of circles
% Prakash Gautam_____005_____Matlab_Template_r00_____2020-07-
23___:
% used the coordinates of center of each detected circle to plot the vector
% plot using quiver
% Prakash Gautam_____006_____Matlab_Template_r00_____2020-08-
04___:
% calculation of angular displacement between images by using the linear
% displacement

%% Program Description-----
-----%
% This program first defines all the variables that are used to compute
% cross-correlation of two binary images
disp('Program Running')
%% Conversion Variables
ft_m = 0.3048;                %'m/ft'      % Converts ft to meters
%% Initialize Program-----
-----%
clear all;
clc;
close all;
%Reading images and specifying origin
image1 = imread('meltpool1.tif');
image2 = imread('meltpool2.tif');
%% Variables-----
-----%
[Nx,Ny] = size(image1);

```

```

%detection of circles using imfindcircles function that output the
%coordinates of center and the radius of detected circles
[centers1,radii1,metric1] = imfindcircles(image1,[6
20], 'ObjectPolarity','bright','Sensitivity',0.85,'EdgeThreshold',0.3);
[centers2,radii2,metric2] = imfindcircles(image2,[6
20], 'ObjectPolarity','bright','Sensitivity',0.85,'EdgeThreshold',0.3);

n=numel(radii1);

%Test matrices are created in both the template(smaller image) and the larger
image based on the size of window
%and the search zone. The point of maximum correlation specifies the
%location in larger images where the template would match and has the highest
correlation.
%the final average displacement can be calculated based on the peaks.
%Initially, the test images and the displacements are initialized to zero.

test1= 0;
test2 = 0;
dx(n) = 0; %displacement
dy(n) = 0;
xpeak = 0;
ypeak = 0;

%Co-ordinates of the center of detected circles in both the images
h1=round(centers1(:,1));
k1=round(centers1(:,2));
rho=sqrt(((Ny/2)-h1).^2+((Nx/2)-k1).^2);

k=Nx-k1;
h3=transpose(h1);
k3=transpose(k);

%% Solution-----
-----%
%c0 to count the windows
%test matrices extracted from both the images and cross correlated
for c0=1:(n)
    max_correlation = 0;
    w_width=2*ceil(radii1(c0));
    w_height=2*ceil(radii1(c0));

    %Range of search window in image 2
    r=1; %search zone on either side of the window
    x_max_dis = r*w_width;
    y_max_dis = r*w_height;

    %specifying the location of interrogation windows
    xmin = k1(c0)-w_width/2;
    xmax = k1(c0)+(w_width/2-1);
    ymin = h1(c0)-w_height/2;
    ymax = h1(c0)+(w_height/2-1);

```

```

    %creation of interrogation windows in both the images
    test1 = image1(xmin:xmax,ymin:ymax);
    test2 = image2((xmin-x_max_dis):(xmax+x_max_dis),(ymin-
y_max_dis):(ymax+y_max_dis));
    correlation = normxcorr2(test1,test2);

    [xpeak,ypeak] = find(correlation==max(correlation(:)));

    %calculation of displacements in x and y directions

    dx(c0) = xpeak-w_width-x_max_dis;
    dy(c0) = ypeak-w_height-y_max_dis;
    k2(c0)=k1(c0)+dx(c0);
    h2(c0)=h1(c0)+dy(c0);
    theta1(c0)=atan2((Nx/2)-k1(c0),h1(c0)-(Ny/2));
    theta2(c0)=atan2((Nx/2)-k2(c0),h2(c0)-(Ny/2));

end
d=abs(abs(theta1)-abs(theta2)); %angular displacement calculated as the
difference between two angles

%vector plot
h=quiver(h3,k3,dy,-dx);
%h.ShowArrowHead='off'; %arrow heads can be turned off

%specifying x and y limits of the plot
xlim([0 Nx])
ylim([0 Ny])

%h.AutoScale='off';%auto scaling of the magnitude of vectors can be turned
off

%color of arrows can be specified
h.Color='red';
%% Closing Program-----%
-----%
disp('Program Finished')

```

## 9. Appendix C



**METAL POWDER  
INDUSTRIES FEDERATION**

105 COLLEGE ROAD EAST  
PRINCETON, NEW JERSEY 08540-6692  
TEL: (609) 452-7700  
info @ mpif.org • www.mpif.org

### **MEMO**

**DATE:** September 29, 2020

**TO:** Prakash Gautam  
Graduate Teaching Assistant  
Monatan Technological Institute

**FROM:** Turner Abbott  
Publications Manager

**SUBJECT:** Permission to use MPIF material

---

You have permission to use , whole or in parts, the paper “Marangoni Convection in Selective Laser Melting (SLM) of 316L Stainless Steel “*2019 Advances in Additive Manufacturing with Powder Metallurgy*”, MPIF, pp.32-50 (2019).

Please use the credit line stated below.

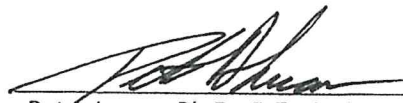
#### **CREDIT LINE:**

Reprinted with permission from “*2019 Advances in Additive Manufacturing with Powder Metallurgy*” Metal Powder Industries Federation, Princeton, New Jersey, USA, 2019.



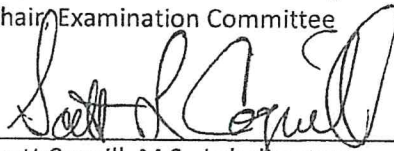
## SIGNATURE PAGE

This is to certify that the thesis prepared by Prakash Gautam entitled "Particle Tracking of a Simulated Melt Pool of Selective Laser Melting" has been examined and approved for acceptance by the Department of Mechanical Engineering, Montana Technological University, on this 1st day of December, 2020.



---

Peter Lucon, Ph.D., P.E., Assistant Professor  
Department of Mechanical Engineering  
Chair, Examination Committee



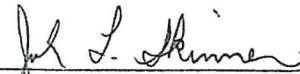
---

Scott Cogull, M.S., Lab director  
Department of Mechanical Engineering  
Member, Examination Committee



---

K. V. Sudhakar, Ph.D., P.E, Professor  
Department of Metallurgical and Materials Engineering  
Graduate School Representative and Member, Examination  
Committee



---

Jack L. Skinner, Ph.D., P.E, Associate Professor and Department Head  
Department of Mechanical Engineering  
Member, Examination Committee