The Pennsylvania State University

The Graduate School

College of Engineering

DATA FUSION FOR ADDITIVE MANUFACTURING PROCESS INSPECTION

A Thesis in

Electrical Engineering

by

Jacob P. Morgan

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2019

The thesis of Jacob P. Morgan was reviewed and approved* by the following:

Richard L. Tutwiler
Associate Professor of Acoustics
Thesis Advisor

William E. Higgins
Distinguished Professor of Electrical Engineering

Edward W. Reutzel
Associate Professor of Engineering Science and Mechanics

Kultegin Aydin
Professor of Electrical Engineering
Head of the Department of Electrical Engineering

*Signatures are on file in the Graduate School.

Abstract

In-situ monitoring of the powder bed fusion additive manufacturing (PBFAM) process is a rapidly expanding area of interest because it offers insight into process physics and is potentially a lower cost alternative to current post-build nondestructive inspection. Ultimately, sensor data may be used as feedback for a real-time fault remediation system. However, it is unclear what defects look like in the sensor data and multiple modalities cannot be used together because they are in arbitrary frames of reference. The goal of this thesis is to present a framework for automatically registering in-situ sensor data to post-build inspection data. This will enable defects found in the post-build inspection to be mapped to the sensor data to serve as a ground truth for developing automatic defect recognition (ADR) algorithms.

In this work, high resolution images and multispectral point sensor data collected during the build are registered to a post-build computed tomography (CT). These sensing modalities can be thought of as 2D raster data, 2D point cloud data, and 3D raster data respectively. A unique optimization approach for registering each modality to a common frame of reference is presented. The process is automated so that large datasets may be generated for use in developing future ADR algorithms. Voids that are clearly visible in the CT can be mapped into the in-situ sensing modalities.

Contents

List of Figures

List of Tables

List of Symbols

β	Vector of parameters being optimized
δ	Levenberg-Marquardt update to β
ε	Residual
λ	Levenberg-Marquardt damping factor
μ	Vector of mixture means
σ	Vector of mixture variances
ϕ	Mixture weight
Ψ	Signed distance function
ω	Vector of Euler angles
Ω	Set that defines model
a	Vector of polynomial coefficients for the x coordinate
b	Vector of polynomial coefficients for the y coordinate
D_{far}	Depth of farthest object in focus
D_{near}	Depth of nearest object in focus
E	Sum of squared residuals
f	Focal length
g	Polynomial transformation
$\tilde{h}$	Vector of homography components
H	Homography
H_{ε}	Hessian of ε with respect to β
H_E	Hessian of E with respect to β
$\widetilde{H}_E$	Gauss-Newton approximation of H_E
H_f	Hyperfocal distance
I_B	Filtered build image
I_S	Synthetic image
J	Jacobian matrix of first derivatives
J_{ε}	Jacobian of ε with respect to β
M	Lens magnification
R	Rotation matrix
R_x	Rotation about the x-axis
R_y	Rotation about the y-axis
R_z	Rotation about the z-axis
t	Translation vector
T	Transformation matrix
V	CT volume
V_{mask}	Binary mask of the CT

Chapter 1

Introduction

1.1 Problem Statement

This work aims to fuse multiple data modalities derived from different sensors to support process inspection for powder bed fusion additive manufacturing (PBFAM). In the PBFAM process, a high-powered laser fuses metal powder on a build plate to form a single layer of the build. The build plate is lowered, new powder is spread across the build plate, and the process is repeated for each new layer forming a 3D part. There is growing interest in using sensors to observe the build as it takes place, but the data from each is in a different frame of reference. Thus, in order to make use of all sensing modalities simultaneously a registration technique is required.

The movement of the laser is known and stored as a set of vectors called the laser scan vectors. All data modalities are registered to the laser scan vectors and therefore will be implicitly aligned to each other. Registration algorithms are developed for high resolution images, multispectral point sensors, and post-build computed tomography (CT). These sensing modalities can be thought of as a 2D raster, point cloud, and 3D raster respectively and thus a different registration is required for each.

The use of post-build CT is important for developing future process inspection algorithms. One defect of interest in the additive manufacturing community is interior voids. These voids are clearly visible in the CT scans, but it is not obvious what they look like in other sensing modalities. Once registered, voids that are visible in the post-build CT can be mapped into different sensing modalities to serve as ground truth data for automatic void detection algorithms.

1.2 Related Work

Process qualification has remained a constant challenge as discontinuities such as lack of fusion, porosity, and cracks may arise during the PBFAM process. The authors of [1], [2], and [3] have shown that these discontinuities correlate to negative mechanical properties. Therefore, it is desirable to detect these discontinuities as they arise during the build process using in-situ sensors.

A common approach to process monitoring is melt pool sensing. The authors of [4] showed that an infrared camera can be used to monitor the thermal behavior of the molten metal. Melt pool sensors have been incorporated into feedback control systems by [5] and [6]. However, analysis of melt pool sensing systems is complicated by the emissivity of the melt pool [7].

An alternative to melt pool sensors is high resolution electro-optical imagery. The authors of [8] presented a layerwise monitoring system using a CCD camera which was able to capture process irregularities. The authors of [9] utilized a tilt-shift lens to capture images of the build surface at a resolution of 32.3 microns per pixel. They were able to correct for the perspective distortion by using calibration markers fused on the build surface [10].

Another promising sensing modality is optical emission spectroscopy (OES). The author of [11] showed that spectral emissions vary with processing parameters like laser power or scan speed. This work was expanded upon in [12] to show that lack of fusion defects could be detected using the line-to-continuum ratio about certain spectral bands.

Despite the work being done on in-situ monitoring, post-build analysis remains the main method of certifying parts. The authors of [13] give an overview of non-destructive post-build analysis techniques for PBFAM parts, including Archimedes methods, metallographic imaging, and CT. Post-build CT is the preferred analysis technique because it enables the quantification of the size and shape of interior voids.

1.3 Outline

Chapter 2 presents the necessary background for developing the results in Chapter 3. Each registration is formulated as a non-linear least squares optimization, and so Chapter 2 begins by introducing the general solution. Next, an overview of signed distance functions is presented, which are a useful model representation that is used for registering both the multispectral point sensors and post-build CT scan.

Chapter 3 discusses the specific techniques for registering high resolution images, multispectral point sensors, and post-build CT to the laser scan vectors. For each modality, the necessary background information, pre-processing steps, and specific optimization are presented.

Chapter 4 gives a brief overview of the experimental setup and summarizes the results. The registration results for post-build CT, high resolution images, and multispectral point sensors are presented and voids from the CT are mapped into both the images and multispectral data.

Chapter 5 offers some conclusions and directions for future work. Appendix A contains additional figures for the CT registration and Appendix B contains additional figures for the multispectral registration.

Chapter 2

Background

This chapter will discuss the relevant background material needed to develop the results in Chapter 3. Section 2.1 discusses how to solve non-linear least squares problems using the Levenberg-Marquardt algorithm. This is important because all of the registrations are posed as non-linear optimizations. Section 2.2 discusses signed distance functions and how they can be generated from the laser scan vectors. Signed distance functions are used in the registration of the multispectral point sensors and post-build CT scans.

2.1 Non-linear least squares

Data modalities are registered using non-linear least squares optimization. The cost function and parameters being optimized differ for each modality, but when posed as a non-linear least squares optimization, they are solved in the same manner. This section discusses the general solution to non-linear least squares problems using the Levenberg-Marquardt algorithm [14].

In non-linear least squares problems, the objective is to minimize a sum of squared residuals. Let $\beta \in \mathbb{R}^m$ be the parameter vector being optimized and $\varepsilon(\beta) \in \mathbb{R}^n$ be the residual vector. Then the non-linear least squares problem is formulated as

$$\min_{\mathrm{\beta}} \left\{ E(\mathrm{\beta}) = \frac{1}{2} \varepsilon(\beta)^T \varepsilon(\beta) \right\} \tag{2.1}$$

The solution to (2.1) is obtained by setting the derivative of $E(\beta)$ to zero. Let the Jacobian matrix of first derivatives of $\varepsilon(\beta)$ be given by $J_\varepsilon = d\varepsilon/d\beta \in \mathbb{R}^m$. Then

$$\frac{dE}{d\beta} = g = J_\varepsilon^T \varepsilon(\beta) = 0 \tag{2.2}$$

Equation (2.2) is a non-linear system of equations that can be solved using Newton's method. Let $\widetilde{H}_E = J_\varepsilon^T J_\varepsilon \in \mathbb{R}^{m \times m}$ be the Gauss-Newton approximation of the true Hessian matrix of second order derivatives of $E(\beta)$. Then given an initial set of parameters β^0 and damping coefficient λ the update is given by

$$\begin{gathered} \left(\widetilde{H}_E + \lambda \mathrm{diag}(\widetilde{H}_E)\right)\delta = -g \\ \beta^{(k+1)} = \beta^{(k)} + \delta \end{gathered} \tag{2.3}$$

Updates are only kept if they improve the error and the damping coefficient is varied heuristically for each iteration. After each update, if the error improves λ is decreased and if the error gets worse λ is increased. For small values of λ Levenberg-Marquardt tends towards the Gauss-Newton algorithm and for large values of λ Levenberg-Marquardt tends towards the method of steepest descent. This is ideal because Gauss-Newton converges faster near the minimum and steepest descent is more robust far from the minimum.

2.2 Signed distance functions

For certain data modalities, it is desirable to minimize the average distance from a point cloud to the laser scan vectors. However, calculating this distance for each point at every iteration is slow and intractable for large point clouds and models. To avoid these problems, the distance to the model can be precomputed at every point in space. This representation is known as a signed distance function (SDF). The laser scan vectors can be transformed into an SDF representation by rasterizing them into a binary image and then using the Euclidean distance transform.

SDFs are continuous functions that define the distance to the nearest surface of a given model. Thus, the SDF evaluated at the point cloud points can be directly used as a residual for non-linear least squares. This section will formally define SDFs, present useful properties of SDFs, and discuss how to generate SDFs from the laser scan vectors.

Let $\Psi(x)\colon \mathbb{R}^n \to \mathbb{R}$ represent the SDF that maps an n-dimensional point to a distance to the model $\Omega \subset \mathbb{R}^n$. The SDF takes on negative values for points located inside of Ω and positive values for points located outside of Ω. Now let $\partial\Omega \in \mathbb{R}^n$ be the boundary of Ω. Then

$$\Psi(x) = \begin{cases} -\mathrm{d}(x, \partial\Omega) & x \in \Omega \\ \mathrm{d}(x, \partial\Omega) & x \in \Omega^C \end{cases} \tag{2.4}$$

where $\mathrm{d}(x, \partial\Omega)$ is the Euclidean distance from point x to the nearest surface $\partial\Omega$.

SDFs defined in this manner have several properties that are useful for gradient descent optimization:

1. If Ω has a piecewise smooth boundary, then $\Psi(x)$ is differentiable almost everywhere.
2. The gradient of $\Psi(x)$ satisfies the eikonal equation $|\nabla\Psi(x)| = 1$ for all x.
3. The gradient of $\Psi(x)$ is always perpendicular to the nearest surface of Ω.

The first property is important because the Jacobian matrix in equation (2.2) is obtained by taking the derivative of the residual, so the residual must be differentiable. The second two properties are useful because the gradient vector for a given point will be in the direction of the nearest surface and with proper sign.

In this work, SDFs are utilized in the registration of the multi-spectral point data and the post-build CT. Each modality is registered to the laser scan vectors, so as an intermediate step the scan vectors are converted to an SDF. The method used here is to first convert the laser scan vectors into a binary image and then use the Euclidean distance transform [15] to obtain the SDF of the binary image. The accuracy of this method depends on the chosen pixel/voxel resolution. A pixel/voxel is defined as an element of the uniform grid of values that is a discretization of the continuous function in 2D or 3D respectively. Higher resolution will improve the accuracy at the cost of additional memory.

First, consider the 2D single layer case. The laser scan vectors are composed of two parts: the exterior contour and the interior hatching used to fill the contour. A binary raster is obtained by overlaying the exterior contours on a grid and filling in each pixel that is contained within the polygons formed by the contours. Edge cases are determined by the percentage of the pixel contained within the polygon.

The Euclidean distance transform computes the distance to the nearest non-zero pixel for all pixels in the binary image. The SDF can be obtained by subtracting the distance transform of the binary image from the distance transform of its complement, which yields negative distance inside the mask and positive distance outside the mask. Let I_M be the binary image and $\mathrm{D}(I_M)$ be the Euclidean distance transform of the image. Then the signed distance function $\Psi(x)$ is given by $\Psi(x) = D(I_M^C) - D(I_M)$. Figure 2.1 shows an example of a set of external contour vectors, the corresponding binary raster, and distance function obtained from the Euclidean distance transform.

Now instead of a single layer, consider every layer of the build for the 3D case. It is not sufficient to simply stack the 2D SDFs from before, because this would not account for the distance in the z-direction. Instead, the binary images of each single layer are stacked into a 3D binary volume where the voxel size in the z-direction is determined by the physical layer thickness of the PBFAM process. Then the Euclidean distance transform is used in the same manner as before to get the full 3D SDF.

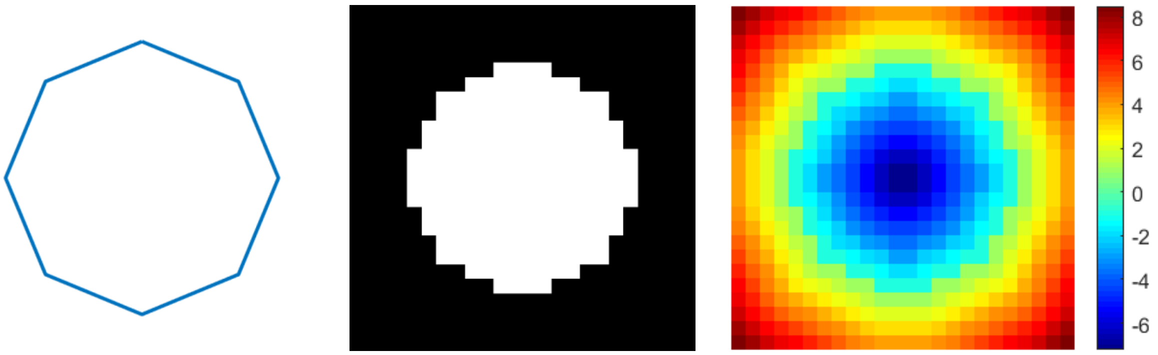


Figure 2.1: Signed distance function from binary raster. Explicitly defined polygon (left), binary raster (middle), and SDF obtained from the Euclidean distance transform (right).

Chapter 3

Data Registration

This chapter will discuss the registration techniques for each data modality. Section 3.1 will discuss the high-resolution images, section 3.2 will discuss the multispectral point sensors, and section 3.3 will discuss the post-build CT. Each section will first discuss the transformation that is being optimized. Next, the relevant pre-processing steps are outlined. Section 3.1 contains an additional discussion on the camera integration and design considerations. Finally, the optimization is presented.

3.1 High Resolution Images

A high-resolution camera is used to capture images of the build as each layer is fused. Due to the location of the camera, the image plane of the camera is not parallel to the build plate. This introduces a projective distortion because the build plate is planar, which can be modeled by a planar homography. Typically, a homography would be calculated from feature correspondences in the images with features in the laser scan vectors. However, there are no well-defined edges or corners in the imagery to choose. A better solution is to directly use the imagery to optimize the parameters of the homography.

3.1.1 Camera design considerations

It is desirable to keep the camera outside of the harsh processing environment of the system, but this introduces some minimum standoff distance from the build plate. Additionally, the viewing angle is off-normal because the laser optics occupy the center of the chamber directly above the build plate. With these imposed constraints, the goal is to image the build surface at the highest possible resolution.

Resolution is defined as the physical size of a single pixel, which at the center of the image is equal to the size of the detector on the camera sensor divided by the magnification. Let f be the focal length of the camera and s be the distance to the object. Then the magnification can be approximated using the Gaussian lens equation [16]

$$M = \frac{f}{s - f} \tag{3.1}$$

Because the viewing angle is off-normal, the depth of field of the camera must also be taken into consideration. The depth of field is defined by the nearest and furthest object distance given by

D_{near} and D_{far} respectively that appears sharp in the image. An object that is outside of the depth of focus will appear blurry in the image. D_{near} and D_{far} are given in terms of the hyperfocal distance H_f

$$D_{near} = s\left(\frac{H_f - f}{H_f + s - 2f}\right)$$
$$D_{far} = s\left(\frac{H_f - f}{H_f - s}\right) \tag{3.2}$$

where D is the aperture diameter and d is the physical size of the detector and H_f is given by

$$H_f = \frac{Df}{d} + f \tag{3.3}$$

Depth of field is maximized when D_{near} is small and D_{far} is large. Analysis of equations (3.2) and (3.3) gives three ways to improve the depth of field: decrease the aperture diameter, decrease the focal length, or increase the detector size. Decreasing the aperture means less light enters the camera, which means the build surface will need to be brighter. Decreasing the focal length will decrease the magnification given by equation (3.1). Increasing the detector size is not possible without getting a different camera. A better solution to increasing the depth of field is to simply tilt the plane of focus, which can be achieved using a tilt-shift lens.

The build area is planar, so the entire surface can ideally be aligned with the plane of sharp focus. However, this process is complicated because adjusting the focus of the lens changes the plane of focus and vice versa. Therefore, an iterative process of adjusting the focus and then the tilt of the lens is used until the image is in focus.

Figure 3.1 shows a representative depth of field for the camera and lens with a tilt of 0°. The lens center is placed in its approximate location in the build chamber, the optical axis is 30° off-normal of the build plate, and the aperture is set to $f/32$. The colored line at the bottom of the build chamber is the 275mm build plate. Green means the build plate is in focus and red means it is not in focus.

Figure 3.2 shows the same scenario as Figure 3.1, but the image plane has been tilted at an angle α with respect to the lens plane such that the plane of focus coincides with the build plate. The Scheimpflug principle [17] states that the image plane, lens plane, and plane of focus intersect at a line, which is denoted by S in Figure 3.2 (shown as a point because the line is perpendicular to the image). The depth of field is now a wedge and D_{near} and D_{far} intersect at the hinge line P. Because the build surface is a plane, the depth of field no longer matters so the aperture diameter may be set to a larger value. This increases the amount of light entering the camera which improves the signal to noise ratio of the image.

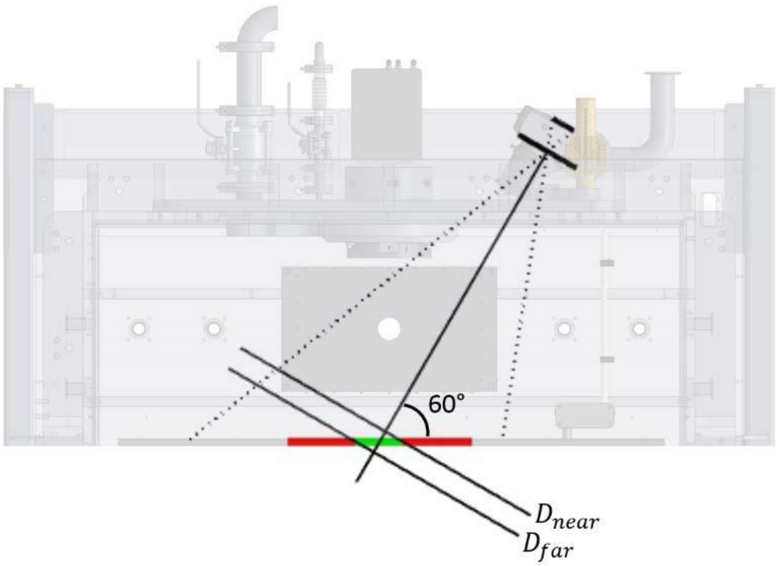


Figure 3.1: Depth of field with 0° tilt (i.e. conventional lens).

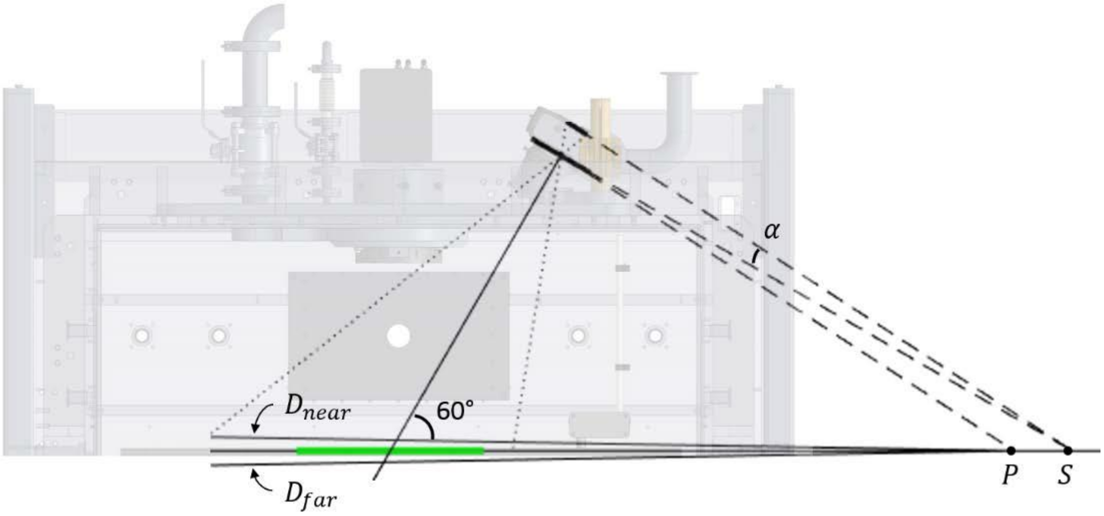


Figure 3.2: Depth of field with tilt α. The plane of focus coincides with the build plate.

3.3.2 Projective Geometry of 2D

Projective distortion arises when a plane is imaged by a perspective camera. Figure 3.3 shows the geometry of a camera looking at a planar surface. Note that points on a line in the planar surface remain on a line in the image plane, but parallel lines on the surface do not necessarily remain parallel in the image. The mapping of the image plane to the planar surface is represented by a projective transformation.

Because the build plate is planar, the transformation from the high-resolution images to the laser scan vectors is described by 2D projective geometry. The 2D projective geometry is defined by a

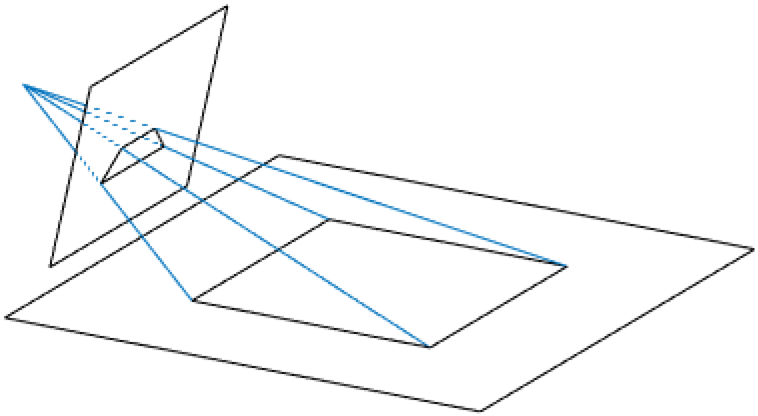


Figure 3.3: Projective distortion introduced by non-parallel image plane.

homography matrix and applied to homogeneous coordinates. Given a set of corresponding points, a simple method of estimating the homography is presented.

Homogeneous coordinates enable the non-linear projective transformation to be represented as a linear set of equations. Let $\bar{x} \in \mathbb{R}^3$ denote the homogeneous coordinate of $x = [u \quad v]^T$. Then the homogeneous projection of $\bar{x}$ is defined as

$$\begin{gathered} \bar{x} = [wu \quad wv \quad w]^T \\ x = \text{proj}(\bar{x}) = [u \quad v]^T \end{gathered} \tag{3.4}$$

A projective transformation is defined as the linear transformation of $\bar{x} \in \mathbb{R}^3$ by the non-singular homography $H \in \mathbb{R}^{3\times3}$ such that

$$\begin{gathered} H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \\ \bar{x}' = H\bar{x} \end{gathered} \tag{3.5}$$

Given a set of $n \geq 4$ point correspondences $\bar{x}_i \leftrightarrow \bar{x}_i{}'$, the homography that maps points x_i to $x_i{}'$ can be approximated by an overdetermined set of linear equations. Let $h_{33} = 1$, which is valid because the homogeneous coordinate is scale invariant i.e. a scale factor does not change u and v in equation (3.4). Then the following set of equations holds

$$\begin{bmatrix} 0 & 0 & 0 & -u_i w_i{}' & -v_i w_i{}' & -w_i w_i{}' & u_i v_i{}' & u_i v_i{}' \\ u_i w_i{}' & v_i w_i{}' & w_i w_i{}' & 0 & 0 & 0 & -u_i u_i{}' & -v_i u_i{}' \end{bmatrix} \tilde{h} = \begin{bmatrix} -w_i v_i{}' \\ w_i u_i{}' \end{bmatrix} \tag{3.6}$$

where $\tilde{h} = [h_{11} \quad h_{21} \quad h_{31} \quad h_{12} \quad h_{22} \quad h_{32} \quad h_{13} \quad h_{23}]$. The matrices for each point are stacked into the full set of equations $A\tilde{h} = b$ where $A \in \mathbb{R}^{2n\times8}$ and $b \in \mathbb{R}^{2n\times1}$. Such an equation can then be solved using the standard least-squares solution $\tilde{h} = (A^T A)^{-1} A^T b$.

3.1.3 Image Construction

There are no well-defined edges or corners in the build imagery to use for point correspondences, so a direct image alignment approach is used instead. The direct alignment is done using a filtered build image and a synthetic image derived from the laser scan vectors. These images are constructed such that they have high intensities where there is fused material and low intensities where there is unfused powder. The images must look as similar as possible so that they can be aligned by minimizing the difference

The build images are filtered using a local standard deviation filter. A standard deviation filter is chosen because the variance of the fused metal is higher than that of the unfused powder. Furthermore, the intensity of the unfused powder is normally distributed and has constant variance, which can be seen in Figure 3.4. The histogram has a normal shape, which is important because when the image is filtered with a standard deviation filter the powder will have a constant value that can be subtracted out. The fused material is more specular which leads to bright reflections and high variance.

The laser scan vectors are not an image, so a synthetic image must be constructed. Further, the synthetic image and filtered build image should look as similar as possible. To construct the synthetic image, the laser scan vectors are first rasterized into a binary image using the same method discussed in section 2.2. Though the binary image could be used directly, this would not account for the filter artifacts induced by the standard deviation filter. Next, the binary 1s in the rasterized image are replaced with white noise and the new image is filtered using the same filter as the build images. White noise must be added so that the filter response is non-zero. Figure 3.5 shows examples of the original build image and filtered build image, noisy binary image, and filtered exemplar.

The local standard deviation filter has some window size associated with it that must be large enough to obtain a good approximation of the powder variance. However, the size of the window also determines how much the edge of the part is dilated. This dilation is implicitly accounted for

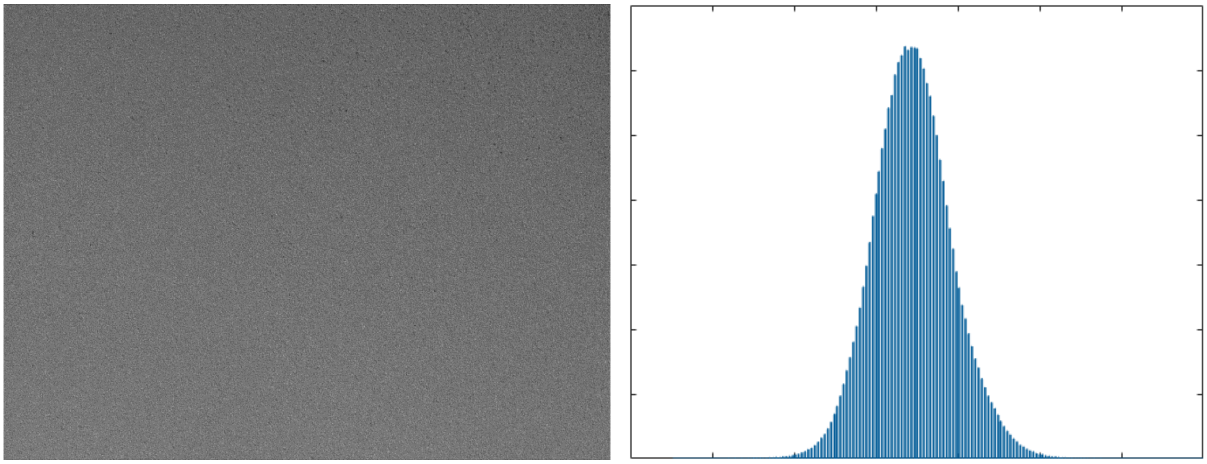


Figure 3.4: Histogram of unfused powder. Image of unfused powder (left) and corresponding histogram (right).

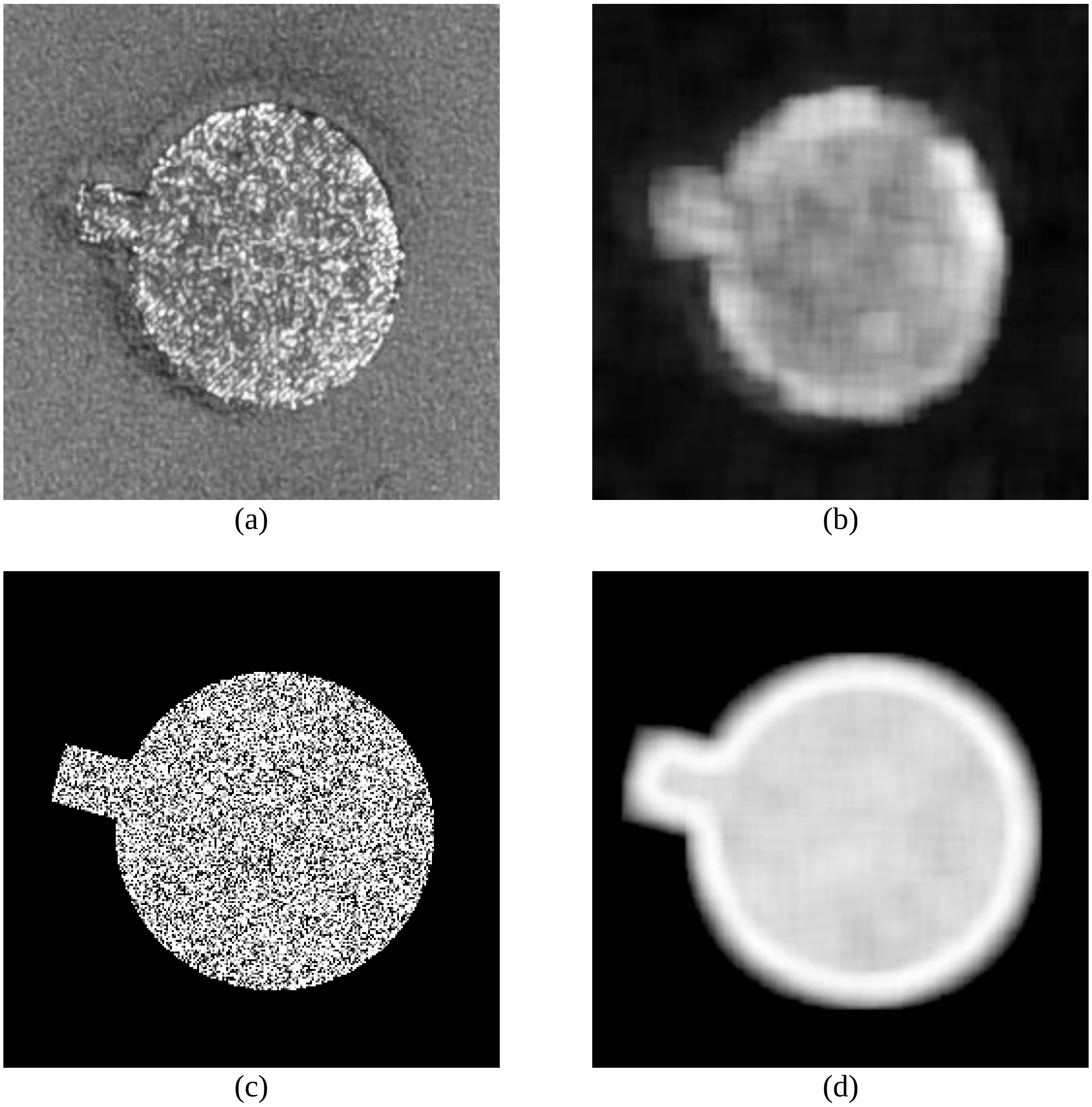


Figure 3.5: Image construction for registration. Unfiltered build image (a), build image filtered by standard deviation filter (b), synthetic binary image with added noise (c), synthetic image filtered by standard deviation filter (d). The filtered image (b) and synthetic image (d) are used for direct image alignment.

because the same filter is applied to both images. For the images in this work, a window size of 11 was empirically chosen to maximize the filter response within the part whilst minimizing the edge effects caused by the window size.

3.1.4 Parameter Optimization

The direct image alignment is formulated as a non-linear least squares optimization where the difference between the filtered build image and synthetic image is minimized. This section presents the error and update equations for the direct image alignment and the construction of the Jacobian matrix used in the Levenberg-Marquardt algorithm.

Let I_B and I_S be the filtered build image and synthetic image derived from the laser scan vectors respectively. Also, let H be the planar homography that maps points from I_B to I_S. The coefficients of H are optimized by minimizing the sum of squared differences between I_B and I_S. Then for iteration k, the optimization with respect to ΔH is given by

$$e^{(k+1)} = \min_{\Delta H} \frac{1}{2} \sum_i \left(I_S\big(\mathrm{proj}(\Delta H \bar{x}_i)\big) - I_B\left(\mathrm{proj}\big(H^{(k)} \bar{x}_i\big)\right)\right)^2$$
$$H^{(k+1)} = H^{(k)} (\Delta H)^{-1} \tag{3.7}$$

Note that updates are calculated for the laser scan vector coordinates and then the inverse is applied to the build image coordinates, which is referred to as inverse composition. This improves performance by enabling the Jacobian to be pre-computed, as will be seen later.

Because updates are iteratively computed, an initial homography $H^{(0)}$ must be given. This initial alignment may be obtained by having a user roughly hand select point correspondences and then applying equation (3.6). Alternatively, a prior calibration can be used if it is available.

To minimize the error $e^{(k+1)}$, the Levenberg-Marquardt algorithm is used. Equation (2.2) is derived by applying the chain rule as follows

$$\left.\frac{de_i^{(k+1)}}{d\Delta H}\right|_{\Delta H = I} = \left(I_S(x_i) - I_B\left(\mathrm{proj}\big(H^{(k)} \bar{x}_i\big)\right)\right) \nabla I_S(x_i)^T \left(\left.\frac{d\mathrm{proj}\big(H^{(k)} \bar{x}_i\big)}{d\Delta H}\right|_{\Delta H = I} \right) \tag{3.8}$$

$$\left.\frac{d\mathrm{proj}\big(H^{(k)} x_i\big)}{d\Delta H}\right|_{\Delta H = I} = \begin{bmatrix} u_i & v_i & 1 & 0 & 0 & 0 & -u_i^2 & -u_i v_i \\ 0 & 0 & 0 & u_i & v_i & 1 & -u_i v_i & -v_i^2 \end{bmatrix} \tag{3.9}$$

where u_i and v_i are the column and row of the exemplar image coordinates. The last two terms of (3.8) form the Jacobian $J_i \in \mathbb{R}^{1\times 8}$ of a single coordinate. The full Jacobian matrix $J \in \mathbb{R}^{n\times 8}$ is obtained by stacking all individual Jacobians. Because the inverse composition is used, J_i does not depend on $H^{(k)}$ so it does not change between iterations which means it can be pre-computed.

A Laplacian pyramid [18] is used to account for large differences between the filtered image and the synthetic image. For each level in the pyramid, the images are down-sampled by a factor of two. The homography is optimized for each level starting with the lowest resolution and then successively optimized at each layer starting at the lowest resolution layer. The result of the previous level is used to initialize the next. This method enables a coarse to fine registration where the differences are refined at each level of the pyramid.

3.2 Multispectral Point Sensors

Two photodiodes monitor the process and are sampled in time as the laser fuses a layer. The photodiodes are filtered to measure light emissions at two separate wavelengths. The idea is that when the laser does not properly fuse the powder, the energy will be diverted somewhere else which can be measured by some ratio of the emissions.

The approximate position of the laser at each time instance is known; however, there is some non-linear transformation that has not been accounted for between the position that is read and the actual position. This section will discuss how the transformation can be modeled using a polynomial transform and how to directly optimize its parameters using a signed distance function.

Multispectral data is collected for every layer of a build and various signals are read for each time instance. Table 3.1 details the properties that are stored for each point. Control signals are measured directly and there is some propagation delay that must be accounted for.

Table 3.1: Fields for each multispectral point.

Property	Description
Time	Time of sample instance.
Position	Laser position on the build plane.
Laser Off/On	Laser control signal (binary).
Photodiode 1	Spectral intensity of photodiode 1.
Photodiode 2	Spectral intensity of photodiode 2.
Power 1	Input control signal for laser power.
Power 2	Measured laser power output.

3.2.1 Polynomial Transformation

The position of each point in the sensor data should correspond to a location on the laser scan vectors. However, due to some measurement error and model approximations, the position of each point has some unknown non-linear distortion. Because there is no model for how the positions are calculated, the distortion is approximated by a general polynomial transformation.

Let $x_i \in \mathbb{R}$ and $y_i \in \mathbb{R}$ be the 2D position of the i-th point. Then the polynomial transformation is given by $g(x_i, y_i; a, b) = \ [x_i' \quad y_i']^T$ where

$$x_i'(x_i, y_i; a) = \sum_{j=0}^{M} \sum_{k=0}^{M-j} a_{jk} x_i^j y_i^k \qquad y_i'(x_i, y_i; b) = \sum_{j=0}^{M} \sum_{k=0}^{M-j} b_{jk} x_i^j y_i^k \tag{3.10}$$

M is the polynomial order and $\{a\}$ and $\{b\}$ are the polynomial coefficients that will be optimized.

A single transformation could be applied to all of the points; however, the polynomial order must be large to get a good fit and the number of parameters scales quadratically with the polynomial order. This can be circumvented by applying the transformation in a two-stage piecewise fashion. The first stage applies a global transformation to all points and the second stage applies a separate transformation locally for each individual part. This reduces the number of free parameters and improves the overall accuracy. The optimization process is the same for each stage, but a smaller subset of points are used for optimizing the second stage.

3.2.2 Data Preprocessing

The parameters of the polynomial transformation are optimized by aligning the contours of the multispectral points to the contours of the laser scan vectors. However, the multispectral points do not differentiate hatching from contours in their properties. Therefore, the contours must be algorithmically extracted in a preprocessing step before they can be used in optimization. Preprocessing consists of two steps: removing points where the laser is off and extracting points on the outer contour.

When the laser is off, points that lie on a hatch should be contained within the external contour. The propagation delay is then chosen so that all points outside the external contour are labeled as off. These points are then removed.

Contour points are obtained using an alpha shape [19], which creates a bounding area around a set of 2D points. The α parameter is the maximum distance between two points such that the points are connected and can be varied to tighten or loosen the fit around the points. Here, α is chosen to be the distance between hatches so that there are no gaps between the hatches and each part is a solid shape. Once an alpha shape is obtained, the perimeter points are extracted and used for optimization. Figure 3.6 shows a subset of multispectral points for a given part before preprocessing, after removing points where the laser is off, and after extracting the contour points using an alpha shape.

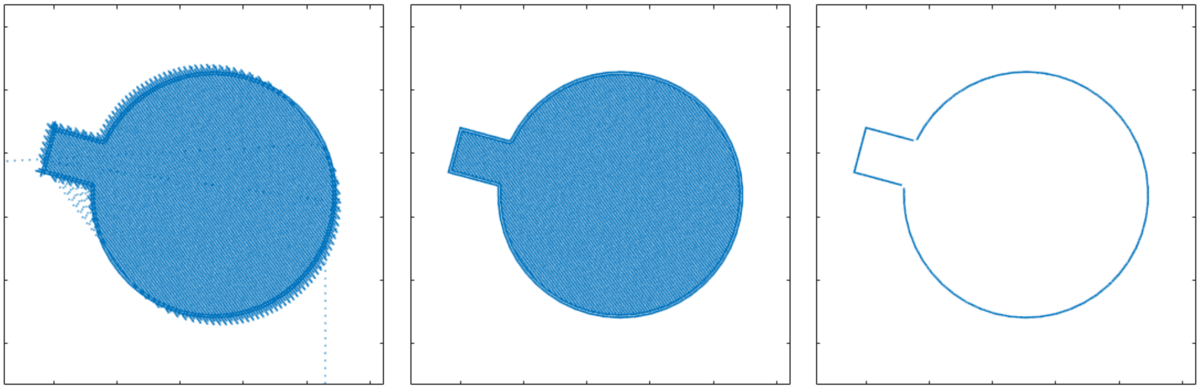


Figure 3.6: Multispectral point cloud preprocessing. Raw unprocessed points (left), points where the laser is on (middle), and extracted contour points (right).

3.2.3 Parameter Optimization

The coefficients of the polynomial transformation are optimized by minimizing the distance of the multispectral points to the laser contours. This distance is efficiently computed using the signed distance function discussed in Chapter 2. This section presents the error and update equations for the optimization and the construction of the Jacobian matrix used in the Levenberg-Marquardt algorithm.

The points are initially not close to the laser scan vectors, so they are scaled and translated such that the bounding box of the points is aligned with the bounding box of the vectors. This gives a close enough fit to initialize the optimization.

Let $\Psi(x)$ be the signed distance function that represents the Euclidean distance to the contours of the laser scan vectors. Then for iteration k, the optimization with respect to Δa and Δb is given by

$$\begin{gathered} e^{(k+1)} = \min_{\Delta a, \Delta b} \frac{1}{2} \sum_i \Psi\left(g\left(x_i, y_i; a^{(k)} + \Delta a, b^{(k)} + \Delta b\right)\right)^2 \\ a^{(k+1)} = a^{(k)} + \Delta a \\ b^{(k+1)} = b^{(k)} + \Delta b \end{gathered} \tag{3.11}$$

This optimization cannot be formulated as an inverse composition because the polynomial transformation is not invertible. $a^{(0)}$ and $b^{(0)}$ are initialized to the identity $g\left(x_i, y_i; a^{(0)}, b^{(0)}\right) = [x_i \quad y_i]^T$,

$$a_{ij}^{(0)} = \begin{cases} 1 & (i,j) = (1,0) \\ 0 & else \end{cases} \qquad\qquad b_{ij}^{(0)} = \begin{cases} 1 & (i,j) = (0,1) \\ 0 & else \end{cases} \tag{3.12}$$

As before, the Levenberg-Marquardt algorithm is used to minimize the error $e^{(k+1)}$. Let $p_i = [x_i \quad y_i]^T$ be the 2D position of multispectral point i. Equation (2.2) is derived by applying the chain rule as follows

$$\left.\frac{de_i^{(k+1)}}{d\Delta a, \Delta b}\right|_{\Delta a, \Delta b = 0} = \Psi\left(p_i^{(k)}\right) \Delta\Psi\left(p_i^{(k)}\right)^T \left(\left.\frac{dg\left(x_i, y_i; a^{(k)} + \Delta a, b^{(k)} + \Delta b\right)}{d\Delta a, \Delta b}\right|_{\Delta a, \Delta b = 0}\right) \tag{3.13}$$

$$\left.\frac{dg}{d\Delta a, \Delta b}\right|_{\Delta a, \Delta b = 0} = \begin{bmatrix} \frac{\partial g}{\partial a_0} & \frac{\partial g}{\partial a_1} & \dots & \frac{\partial g}{\partial a_N} & \frac{\partial g}{\partial b_0} & \frac{\partial g}{\partial b_1} & \dots & \frac{\partial g}{\partial b_N} \end{bmatrix} \tag{3.14}$$

where $N = M(M-1)/2$ and the partial derivatives are given by

$$\frac{\partial g}{\partial a_l} = [x^j y^k \quad 0]^T \qquad\qquad \frac{\partial g}{\partial b_l} = [0 \quad x^j y^k]^T \tag{3.15}$$

The exponents j and k are given by the linear mapping from a_l and b_l, $l = j + \sum_{n=1}^{k}(M - n)$, $0 \leq j \leq M$ and $0 \leq k \leq j$. The last two terms of (3.13) form the Jacobian $J_i \in \mathbb{R}^{1\times 8}$ of a single coordinate. The full Jacobian matrix $J \in \mathbb{R}^{n\times 8}$ is obtained by stacking all individual Jacobians.

3.3 Post-Build Computed Tomography

After a build is completed, parts are scanned in a CT machine. CT penetrates the interior of the part and enables embedded defects to be found without destroying the part. It is important to register the post-build CT so that these embedded defects can be mapped back into the other sensing domains and serve as a ground truth for automatic detection algorithm development. This section will discuss how the post-build CT is registered to the laser scan vectors.

3.3.1 Projective Geometry of 3D

The CT scan is initially in some arbitrary frame of reference and must be rotated and translated to match the laser scan vectors. This transformation is described by the 3D rigid transformation which is a special case of the 3D projective transformation. This section presents the 3D rigid transformation and the minimal representation used for optimization.

Similarly to the 2D case, 3D projections can be represented as linear transformations using homogeneous coordinates. Let $\bar{p} \in \mathbb{R}^4$ denote the homogeneous coordinate of the 3D point $p = [x \quad y \quad z]^T$. Then the homogeneous projection of $\bar{p}$ is defined as

$$\begin{gathered} \bar{p} = [wx \quad wy \quad wz \quad w]^T \\ p = \mathrm{proj}(\bar{p}) = [x \quad y \quad z]^T \end{gathered} \tag{3.16}$$

The results developed here utilize a special case of the 3D projective transform: the rigid transformation. A rigid transformation is composed of a rotation matrix $R \in \mathrm{SO}(3)$ and translation vector $t \in \mathbb{R}^{3\times 1}$ where $SO(3) = \{R \in \mathbb{R}^{3\times 3} \mid RR^T = I, \det(R) = 1\}$ is the special orthogonal group. The rigid transformation matrix $T \in \mathbb{R}^{4\times 4}$ is given by

$$\begin{gathered} T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \\ \bar{p}' = T\bar{p} \end{gathered} \tag{3.17}$$

For optimization, a minimal representation of T is required. The rotation matrix R can be minimally represented by three Euler angles $\omega \in \mathbb{R}^3$. Here the rotation matrix is given by $R(\omega) = R_x(\omega_1)R_y(\omega_2)R_z(\omega_3)$ where

$$R_x(\omega_1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega_1 & -\sin\omega_1 \\ 0 & \sin\omega_1 & \cos\omega_1 \end{bmatrix}$$

$$R_y(\omega_2) = \begin{bmatrix} \cos\omega_2 & 0 & \sin\omega_2 \\ 0 & 1 & 0 \\ -\sin\omega_2 & 0 & \cos\omega_2 \end{bmatrix} \tag{3.18}$$

$$R_z(\omega_3) = \begin{bmatrix} \cos\omega_3 & -\sin\omega_3 & 0 \\ \sin\omega_3 & \cos\omega_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The translation vector is already a minimal representation. Therefore, T is minimally represented by six parameters $(\omega \in \mathbb{R}^3, t \in \mathbb{R}^3)$. During optimization, updates are calculated with respect to each parameter, but are then composed and accumulated as a full transformation matrix

3.3.2 Point Cloud Extraction

The parameters of the rigid transformation are optimized by minimizing the distance between the surface of the part in the CT and the contours of the laser scan vectors. However, the CT is a 3D raster, so first the surface of the part in the CT volume is extracted as a 3D point cloud to be used for optimization. Because the density of the part is higher than the density of the background, a mask of the part is obtained by thresholding the CT volume. The surface points of this mask are then extracted.

The threshold is automatically chosen and will differ depending on the type of material used. The distribution of intensity values in the CT can be modeled by a two-component Gaussian mixture model. The intensity threshold is implicitly defined where the probability of part intensities is equal to the probability of non-part intensities. The probability density function $f(x)$ is given by the weighted sum of two normal distributions. Let $\phi \in [0,1]$ be the mixing coefficient and μ_i and σ_i be the mean and standard deviation of each normal. Then

$$f(x; \phi, \mu_1, \sigma_1, \mu_2, \sigma_2) = \phi N(x; \mu_1, \sigma_1^2) + (1 - \phi)N(x; \mu_2, \sigma_2^2) \tag{3.19}$$

The parameters of the pdf can be estimated using the expectation maximization algorithm outlined in [20]. Once the parameters are obtained, the CT is thresholded to a binary volume as follows. Let V be the CT volume. Then the binary mask of the part V_{mask} is given by

$$V_{mask}(x) = \begin{cases} 1 & (1 - \phi)N(V(x); \mu_2, \sigma_2^2) > \phi N(V(x); \mu_1, \sigma_1^2) \\ 0 & else \end{cases} \tag{3.20}$$

The surface points are given by part voxels x in the mask $V_{mask}(x)$ that are next to non-part voxels. These points are collected into a point cloud and used for optimization.

Figure 3.7 shows the histogram of intensities for a single part. The distribution of intensities is roughly two Gaussians and the mixture model is a good fit. This method allows a threshold to be automatically determined regardless of part material.

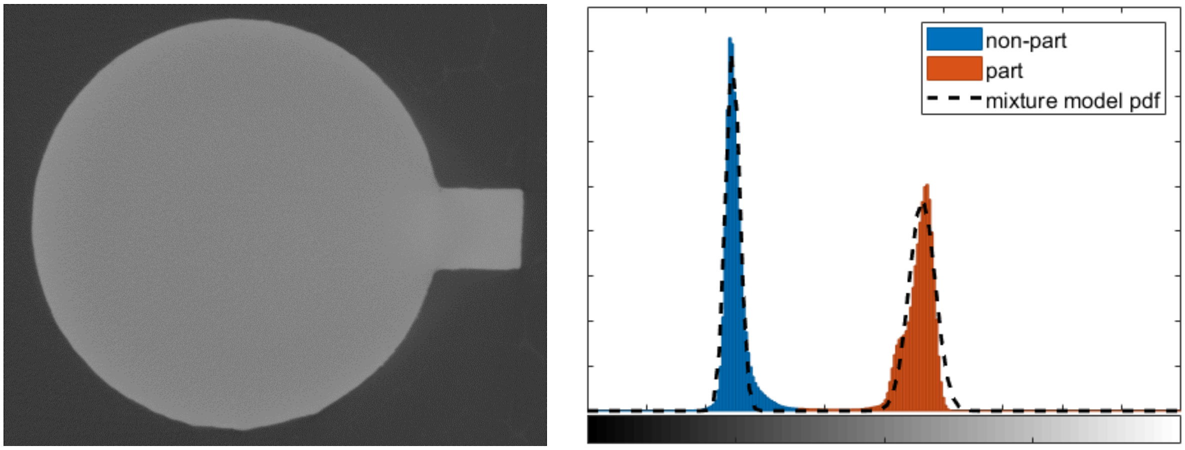


Figure 3.7: Histogram of CT intensities and mixture model pdf. A slice taken from the CT (left) and corresponding histogram of intensities (right). The mixture model is plotted as a dashed black line and the histogram is thresholded into two regions: part and non-part.

3.3.3 Parameter optimization

The parameters of the rigid transformation are optimized by minimizing the distance of the CT point cloud to the laser contours. This distance is efficiently computed using the signed distance function discussed in Chapter 2. This section presents the error and update equations for the optimization and the construction of the Jacobian matrix used in the Levenberg-Marquardt algorithm.

Let T be the rigid transformation that maps the CT point cloud to the laser scan vectors given in equation (3.17). As discussed in section 3.3.1, T can be minimally represented by six parameters $(\omega \in \mathbb{R}^3, t \in \mathbb{R}^3)$. Let $\Psi(x)$ be the signed distance function that represents the distance to the laser scan vectors. When the CT point cloud is aligned to the scan vectors, the value of $\Psi(x)$ will be small because the distance to the scan vectors is small. Therefore, the registration is formulated as a non-linear least squares optimization where the sum of squared distances is minimized with respect to an update $(\Delta\omega, \Delta t)$ to the rigid transformation

$$e^{(k+1)} = \min_{\Delta\omega,\Delta t} \frac{1}{2} \sum_i \Psi\left(\operatorname{proj}\left(\begin{bmatrix} R(\Delta\omega) & \Delta t \\ 0 & 1 \end{bmatrix} T^{(k)} x_i\right)\right)^2$$

$$T^{(k+1)} = \begin{bmatrix} R(\Delta\omega) & \Delta t \\ 0 & 1 \end{bmatrix} T^{(k)} \tag{3.21}$$

where k is the iteration number. Note that updates are calculated with respect to the minimal representation but then accumulated in the full transformation matrix T.

As before, some initial $T^{(0)}$ must be given to initialize the algorithm. When the CT operator exports the volume, they can choose the orientation of the part. Here, the operator roughly orients the part to match the orientation of the laser scan vectors in which case $R^{(0)} = I$ and $t^{(0)}$ may be obtained by aligning the centroid of the CT surface to the centroid of the laser scan vectors.

The Levenberg-Marquardt algorithm is used to minimize $e^{(k+1)}$. Let p_i be the i-th point in the CT point cloud and $p_i^{(k)} = T^{(k)} p_i = \begin{bmatrix} x_i^{(k)} & y_i^{(k)} & z_i^{(k)} \end{bmatrix}^T$. Equation (2.2) is derived by applying the chain rule as follows

$$\left.\frac{de_i^{(k+1)}}{d\Delta\omega, \Delta t}\right|_{\Delta\omega,\Delta t=0} = \Psi\left(p_i^{(k)}\right) \Delta\Psi\left(p_i^{(k)}\right)^T \left(\left.\frac{d\operatorname{proj}\left(\begin{bmatrix} R(\Delta\omega) & \Delta t \\ 0 & 1 \end{bmatrix} p_i^{(k)}\right)}{d\Delta\omega, \Delta t}\right|_{\Delta\omega,\Delta t=0} \right) \tag{3.22}$$

$$\left.\frac{d\operatorname{proj}\left(\begin{bmatrix} R(\Delta\omega) & \Delta t \\ 0 & 1 \end{bmatrix} p_i^{(k)}\right)}{d\Delta\omega, \Delta t}\right|_{\Delta\omega,\Delta t=0} = \begin{bmatrix} 0 & z_i^{(k)} & -y_i^{(k)} & 1 & 0 & 0 \\ -z_i^{(k)} & 0 & x_i^{(k)} & 0 & 1 & 0 \\ y_i^{(k)} & -x_i^{(k)} & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.23}$$

The last two terms of (3.22) form the Jacobian $J_i \in \mathbb{R}^{1\times 6}$ of a single point. The full Jacobian matrix of all terms $J \in \mathbb{R}^{n\times 6}$ is obtained by stacking all individual Jacobians. The Jacobian depends on the current transformation $T^{(k)}$, so it must be recomputed at every iteration.

Chapter 4

Results

This chapter presents the registration results for a single build. Section 4.1 gives an overview of the experimental setup and build layout. Next, section 4.2 presents a quantitative analysis of the CT to laser scan vector registration. Sections 4.3 and 4.4 present registration results for the high resolution images and multispectral point sensors respectively. Voids found in the post-build CT are then mapped into these sensing modalities.

4.1 Experimental Setup

For this work, all builds were completed on a 3D Systems Prox320 powder bed fusion additive manufacturing system. To collect images, the system is outfitted with a Nikon D810 camera with a PC-E Micro NIKKOR 45mm f/2.8D ED tilt-shift lens. With this setup the build plate is imaged at a resolution of roughly 60 µm per pixel. Figure 4.1 shows a cross sectional view of the build chamber and the mounted camera. To collect multispectral data, two avalanche photodiodes are mounted outside the build chamber. Post-build analysis was completed with a GE phoenix v|tome|x m CT scanner.

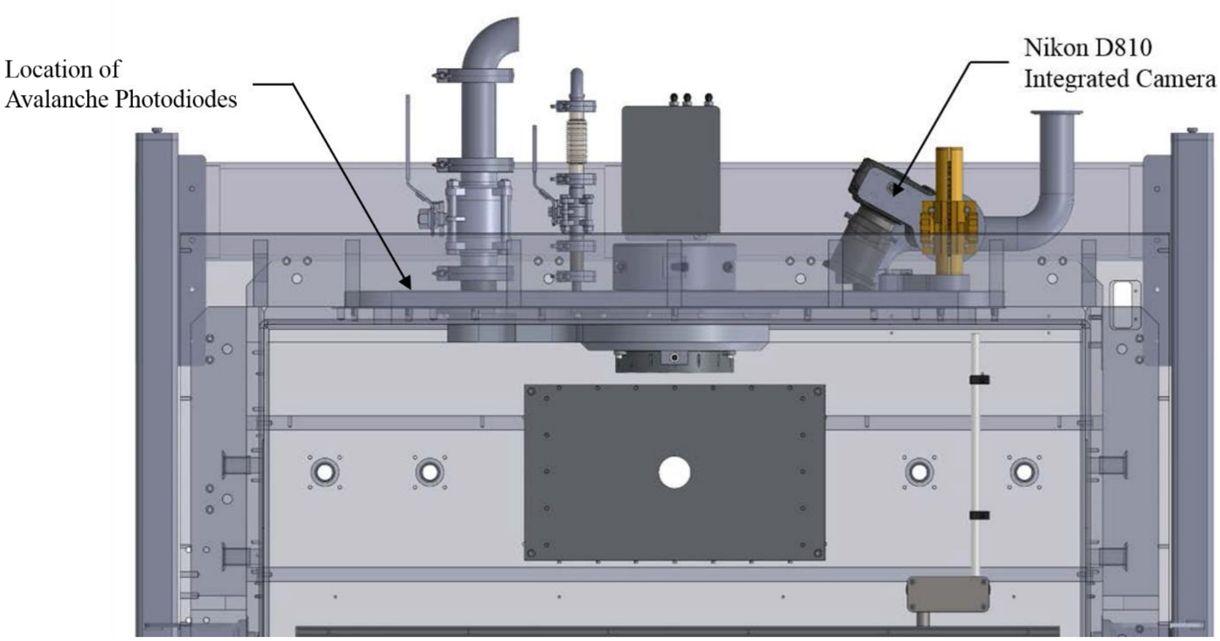


Figure 4.1: 3D Systems ProX320 build chamber and integrated camera.

A special cylinder was constructed for testing the registration algorithms and developing automatic defect detection algorithms. Figure 4.2 shows the design for one such cylinder. A cylinder was chosen to minimize reflections in the CT caused by sharp edges and corners. To eliminate rotational ambiguity, a small tab was added to the bottom of the cylinder. Additionally, each cylinder has a unique number that is printed at the top.

A total of twenty-three cylinders were printed for the test build. These cylinders were printed under one of four scenarios: nominal conditions (control), varied laser scan direction relative to gas flow, varied hatch-contour overlap distance, or varied machine parameters (laser power, laser speed, hatch spacing). Figure 4.3 shows the location of each cylinder on the build plate.

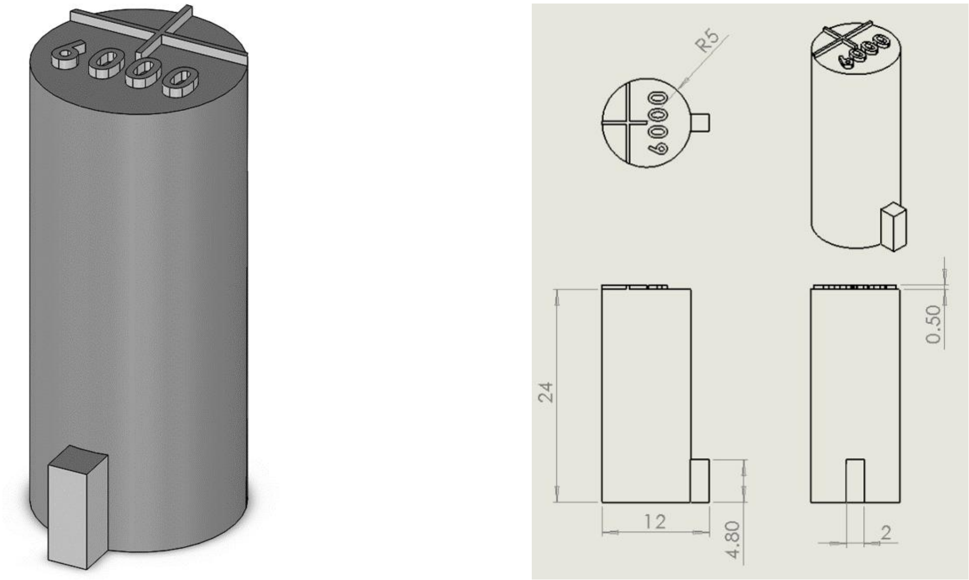


Figure 4.2: Cylinder design used for testing.

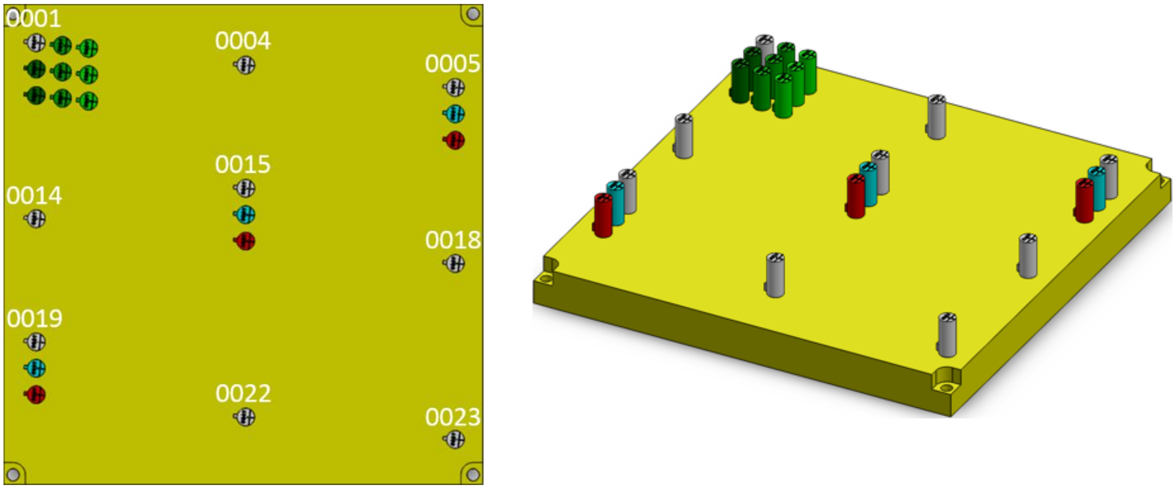


Figure 4.3: Build layout. The gray cylinders are printed under nominal conditions, red with varied scan direction, blue with varied hatch-contour overlap, and green with varied machine parameters.

400 layers were supposed to be printed, but due to a technical malfunction the camera stopped recording images after layer 72. The build itself was then halted prematurely after layer 172. Therefore, the cylinders are roughly half the height they are supposed to be. To account for this error in the analysis, the laser scan vectors beyond layer 172 are ignored for registration.

4.2 Post-Build Computed Tomography

For the test build, only the nine nominal cylinders were CT scanned. Each of these cylinders was registered to the laser scan vectors using the method discussed in section 3.3. Recall that the end goal is to map voids found in the CT to the other sensing modalities. Thus, any error in the CT registration affects the mapping of voids to all sensing modalities, so it is even more important that this registration is as accurate as possible. This section presents the registration of the CT and discusses the results.

Figure 4.4 shows a registered point cloud for cylinder 0001. The remaining cylinders are shown in Appendix A. The point cloud is colored according to the SDF distance evaluated at each point. Note that the point cloud SDF distances are clustered around 100 μm. This is because in reality, the printed vectors have some non-zero width, so the actual part that is printed has a slightly dilated external contour. This width is not accounted for in the SDF so the optimal SDF distance is achieved by centering the vectors in the CT surface, which causes the distances of points on the side of the cylinder to be some number greater than zero. This phenomenon does not occur in the z-direction, so the SDF distances at the top of the cylinder are close to zero.

Additional errors may be caused by either post-build warping or artifacts in the CT scan. Post-build warping introduces some non-linear transformation of the part that cannot be accurately modeled by a rigid transformation. Additionally, artifacts may arise in the CT scan when there are sharp corners and edges which cause the CT to be dilated.

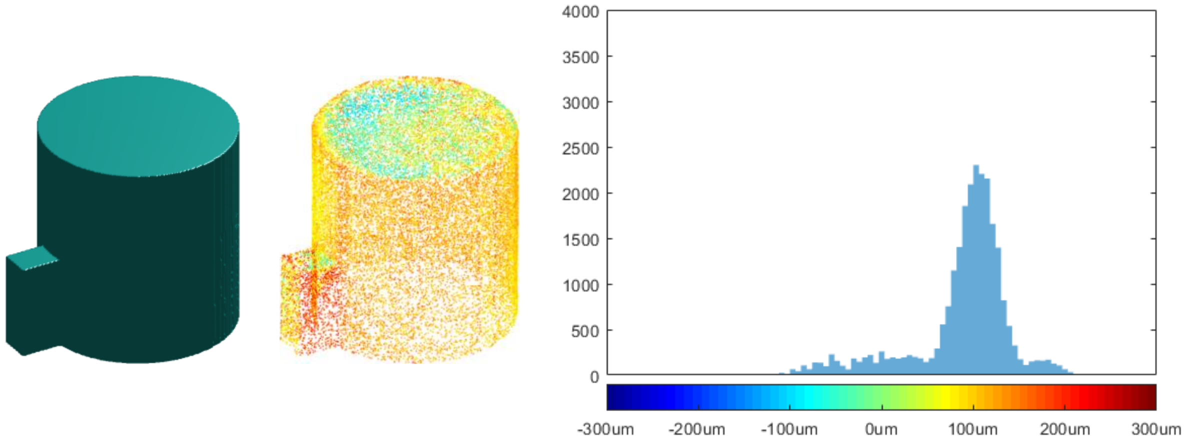


Figure 4.4: Registered CT point cloud for cylinder 0001. SDF isosurface (left), registered point cloud (middle), and histogram of point cloud SDF values.

The root mean squared error for each cylinder is reported in Table 4.1. The average error was 107 µm which is expected due to the laser width. Recall that the resolution of the camera is roughly 60 µm per pixel. This means that voids mapped from the CT to the high-resolution images will be ±1.5 pixels from their true position on average. The laser width could be accounted for by dilating the external contours by half the laser width (or eroding the CT mask by half the laser width). This is difficult because the exact laser width is unknown.

Table 4.1: CT point cloud registration error.

Part Name	RMSE (µm)
0001 - Nominal	103.4210
0004 - Nominal	109.1089
0005 - Nominal	106.5773
0014 - Nominal	109.2762
0015 - Nominal	108.1504
0018 - Nominal	99.9308
0019 - Nominal	95.7102
0022 - Nominal	107.4913
0023 - Nominal	122.7757

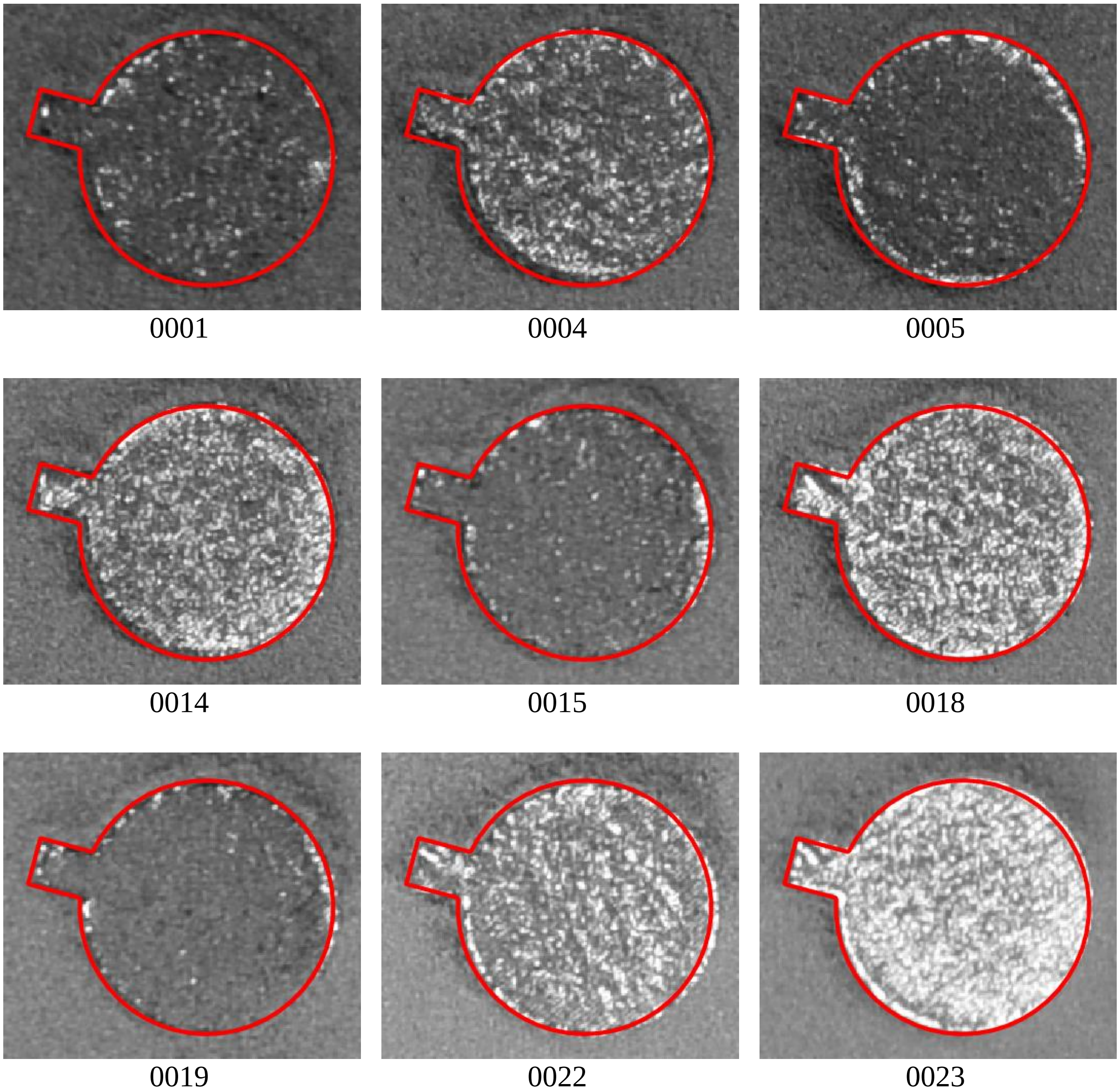
4.3 High Resolution Images

The camera captures an image of the entire build plate so data for all twenty-three cylinders is available. However, since the end goal is to map voids from the CT to the images, the registration results are only presented for the nine nominal cylinders. Recall that the registration error being minimized is a difference between image intensities. This has no physical meaning, and the exact edge in the image cannot be located with a known level of accuracy, so only a qualitative analysis is presented.

Figure 4.4 shows images of each of the nominal cylinders for a layer 10 with the laser scan vectors overlaid on top. The registration is good if the scan vector contours follow the edge of the part in the images. Most of the cylinders align well; however, cylinders 0004 and 0014 are slightly misaligned on the bottom edge. This is likely caused by the dark shadow located in this position. This shadow will stand out in the standard deviation filter the same way the part would which causes the skew. Qualitatively, this misalignment is bounded by four pixels in width and given the pixel size is around 60 µm, this means the error is bounded by 240 µm.

Cylinder 0001 illustrates the challenge of the parts not having well defined corners or edges. The interior of the part looks very similar to the powder aside from a small number of reflections. Despite this, the direct image alignment approach gives good registration results for this part.

Figure 4.6 shows two examples of voids found in the post-build CT that are mapped into the images. Patches around void locations in the image domain can now be studied further to see if


0001

0004

0005

0014

0015

0018

0019

0022

0023

Figure 4.5: Registered images with laser scan vectors overlaid in red.

there is any distinguishing features that can be used for automatic detection algorithms. These patches could also be used to train a supervised machine learning algorithm. To ensure that a defect is actually contained within the patch, the radius of the patch should be equal to the largest possible error. Compounding the CT registration error and image registration error, a patch must be at least 340 µm wide (six pixels) to guarantee that the void will be contained within it.

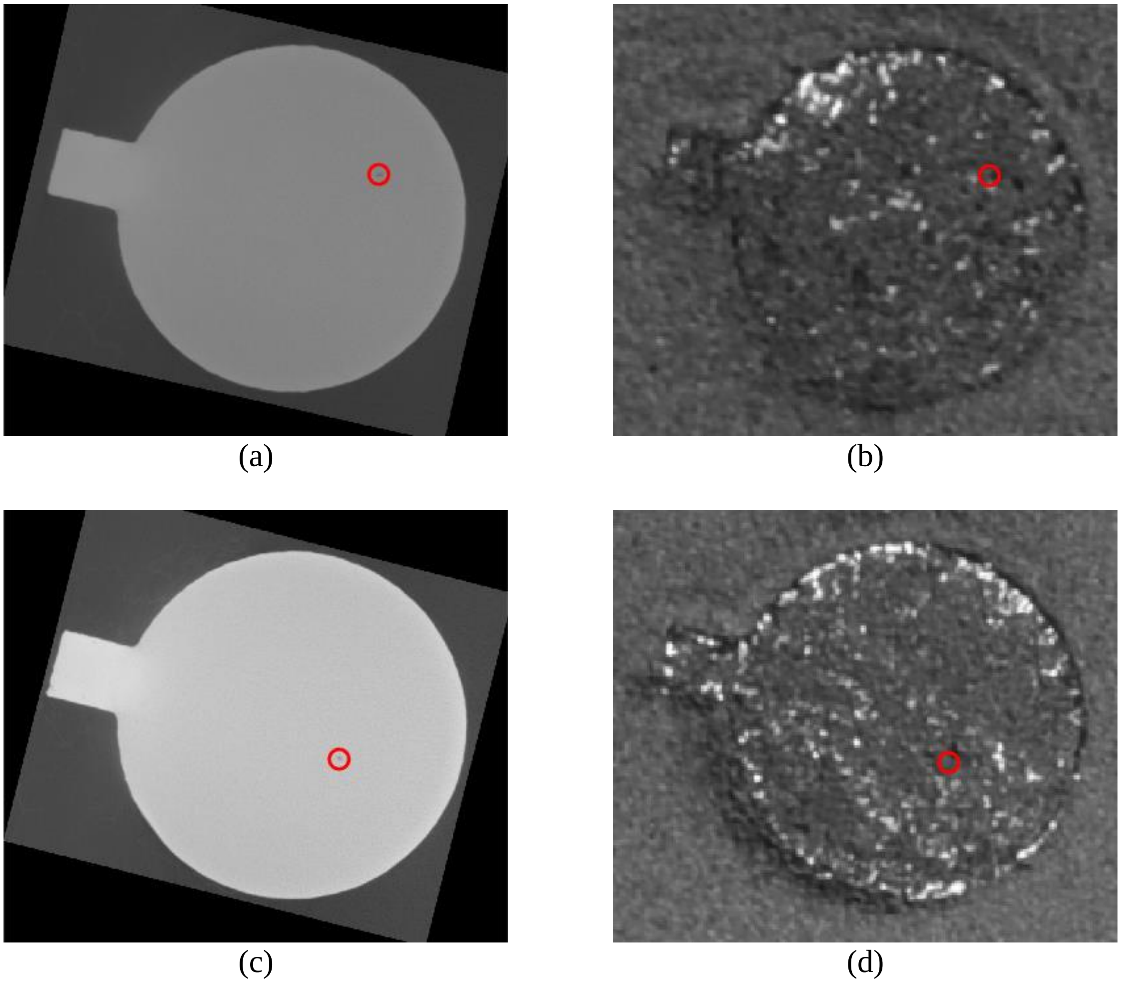


Figure 4.6: Voids found in CT mapped to images. (a) and (b) are cylinder 0001 and (c) and (d) are cylinder 0004.

4.3 Multispectral Point Sensors

This section presents the registration results for the multispectral point sensors. As discussed in section 3.2, the multispectral data is stored as a point cloud. This work is focused on the registration of the 2D position of the points, which is independent of the spectral intensities. Because the points are aligned to an SDF, the error that is minimized has a physical meaning. Therefore, a quantitative analysis is presented in addition to the qualitative analysis.

Figure 4.7 shows the registered multispectral points for cylinder 0001. The remaining eight nominal cylinders are shown in Appendix B. Only the outer contour is shown, because those are the points that are used in the optimization. Points are colored according to their SDF distance. The registration accuracy is very good with most of the points falling within ± 10 μm of the laser scan vectors and the error is contained mostly within the tab of the cylinder.

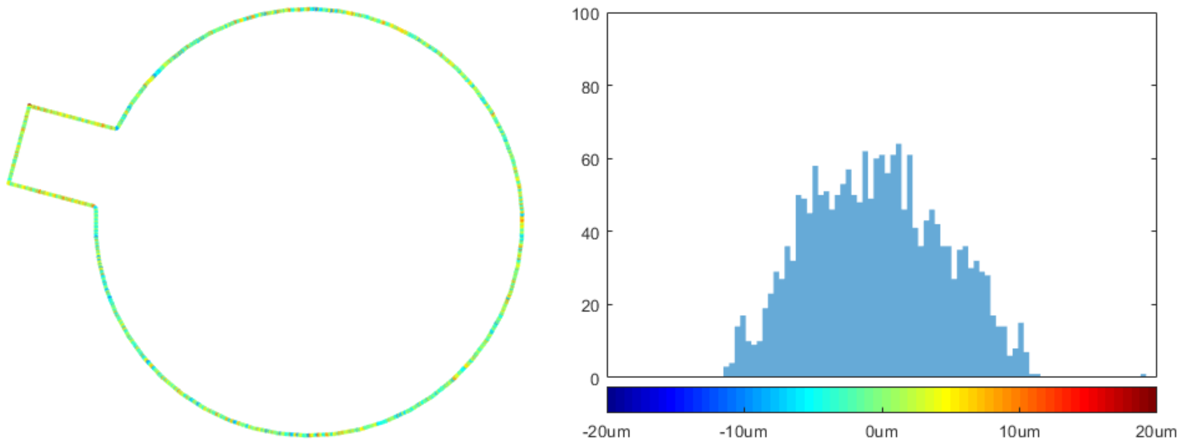


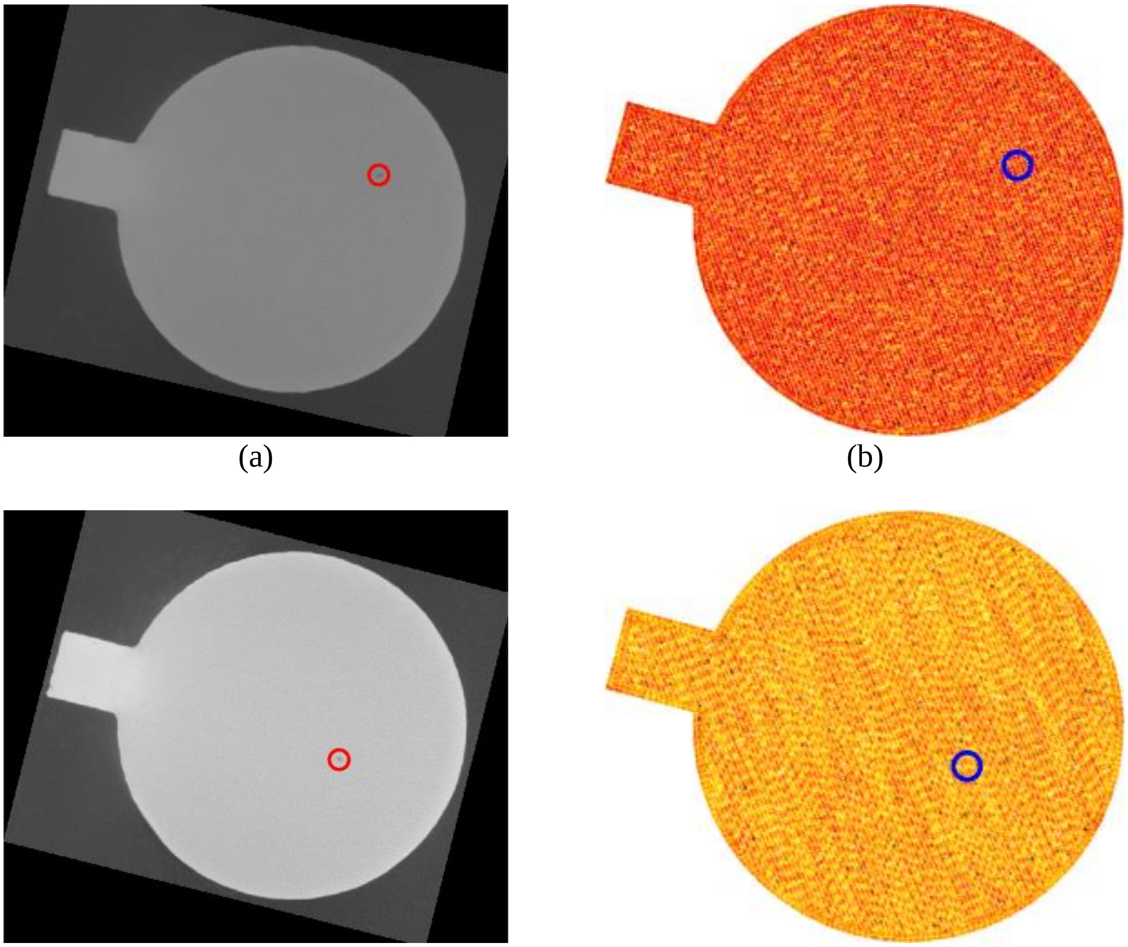
Figure 4.7: Registered multispectral point cloud for cylinder 0001. Point cloud colored according to SDF distance (left) and histogram of SDF distances (right).

Table 4.2 presents the root mean squared error for each of the nominal cylinders. The average across all cylinders was 5.6954 µm. This is significantly less than the registration error of the CT meaning that when voids are mapped into the multispectral domain, the error is almost entirely due to the CT registration.

Table 4.2: Multispectral point cloud registration error.

Part Name	RMSE (µm)
0001 - Nominal	4.9066
0004 - Nominal	5.2680
0005 - Nominal	6.7046
0014 - Nominal	5.2806
0015 - Nominal	6.2074
0018 - Nominal	5.6023
0019 - Nominal	5.5841
0022 - Nominal	5.4295
0023 - Nominal	6.2756

Figure 4.8 shows two examples of voids found in the post-build CT that are mapped into the multispectral point clouds. The point clouds are colored according to the ratio of the intensities of the multispectral sensors.



(a) (b) (c) (d)

Figure 4.8: Voids found in CT mapped to multispectral. (a) and (b) are cylinder 0001 and (c) and (d) are cylinder 0004. The multispectral point clouds are colored according to the spectral intensity at each point.

Chapter 5

Conclusion

This thesis has introduced three applications of non-linear least squares optimization to register multi-modal data sources to laser scan vectors. High resolution images and multi-spectral point sensors monitor the powder bed fusion additive manufacturing process in-situ and post-build inspection is done using a computed tomography. Each data modality is registered to the laser scan vector frame of reference and thus all modalities are implicitly registered to each other. Voids that are larger than 200 µm in diameter are the most concerning in terms of the structural integrity of the part. The registration error from CT to images is on the order of 340 µm and the registration error from CT to multispectral is on the order of 105 µm. The CT registration could be further improved by accounting for the width of the laser in the distance function.

Future work will use registered data sets to develop in-situ automatic defect detection algorithms. Voids that are found via post-build inspection can be mapped to the other sensing modalities to serve as a ground truth for development. Patches around the location of voids in the high-resolution images and in the multispectral point cloud may be used as labeled training data for a machine learning algorithm. Eventually automatic detection algorithms may be used in conjunction with feedback control for fault remediation as the build is processed. In conclusion, the registration techniques presented in this thesis provide an automated method of aligning multi-modal data sources and a framework extendible to other sensors.

Appendix A

Registered CT Point Clouds

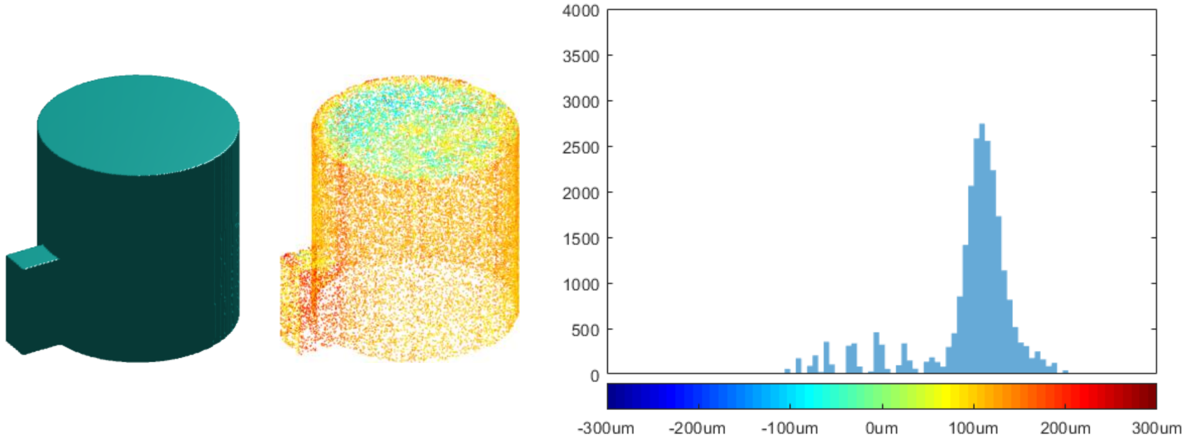


Figure A.1: Registered CT point cloud for cylinder 0004. SDF isosurface (left), registered point cloud (middle), and histogram of point cloud SDF values.

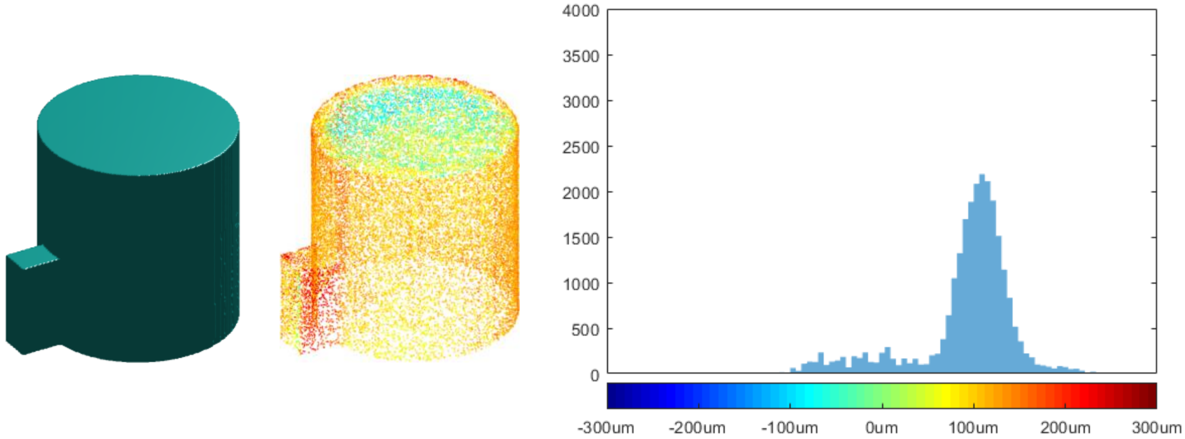


Figure A.2: Registered CT point cloud for cylinder 0005. SDF isosurface (left), registered point cloud (middle), and histogram of point cloud SDF values.

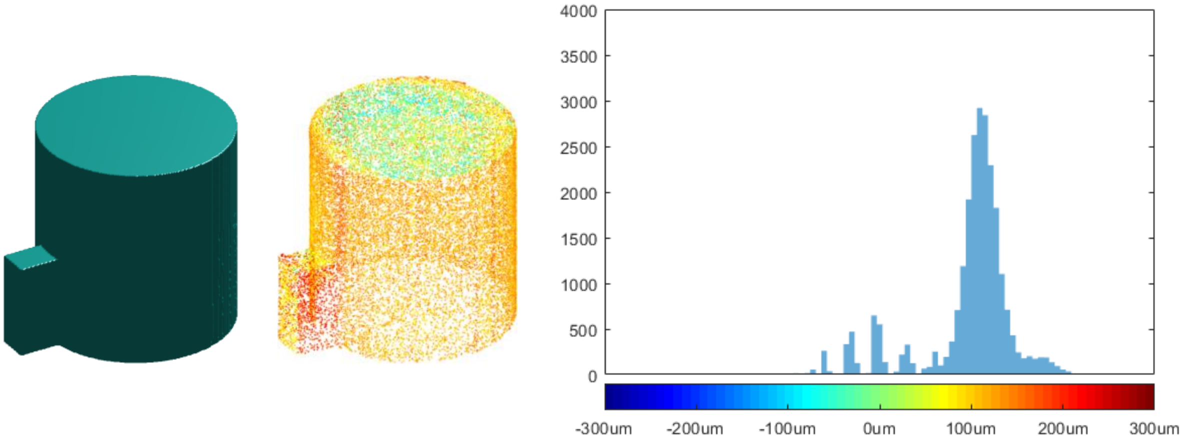


Figure A.3: Registered CT point cloud for cylinder 0014. SDF isosurface (left), registered point cloud (middle), and histogram of point cloud SDF values.

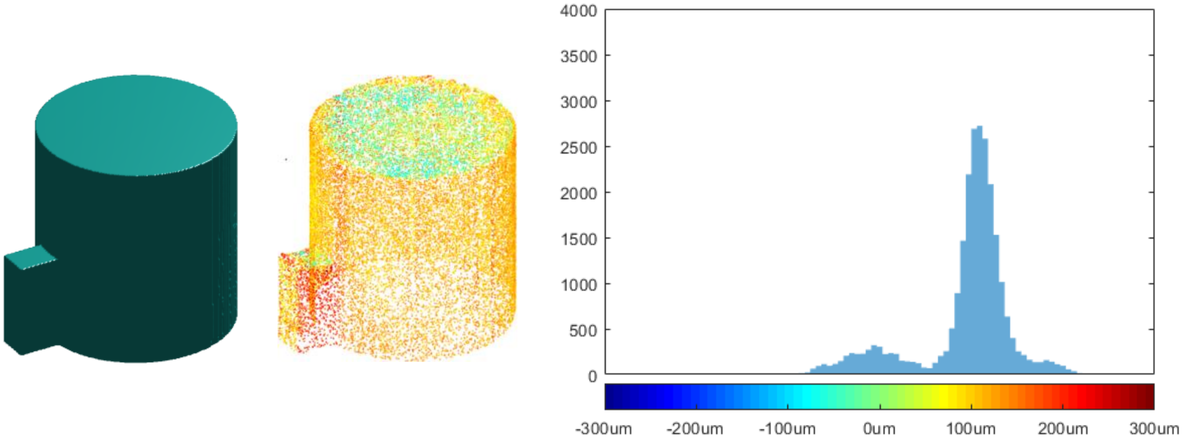


Figure A.4: Registered CT point cloud for cylinder 0015. SDF isosurface (left), registered point cloud (middle), and histogram of point cloud SDF values.

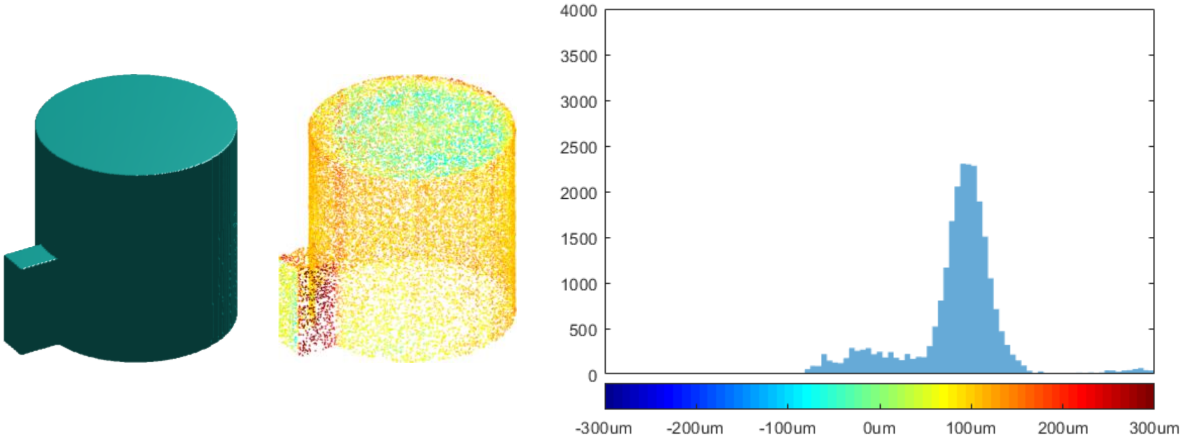


Figure A.5: Registered CT point cloud for cylinder 0018. SDF isosurface (left), registered point cloud (middle), and histogram of point cloud SDF values.

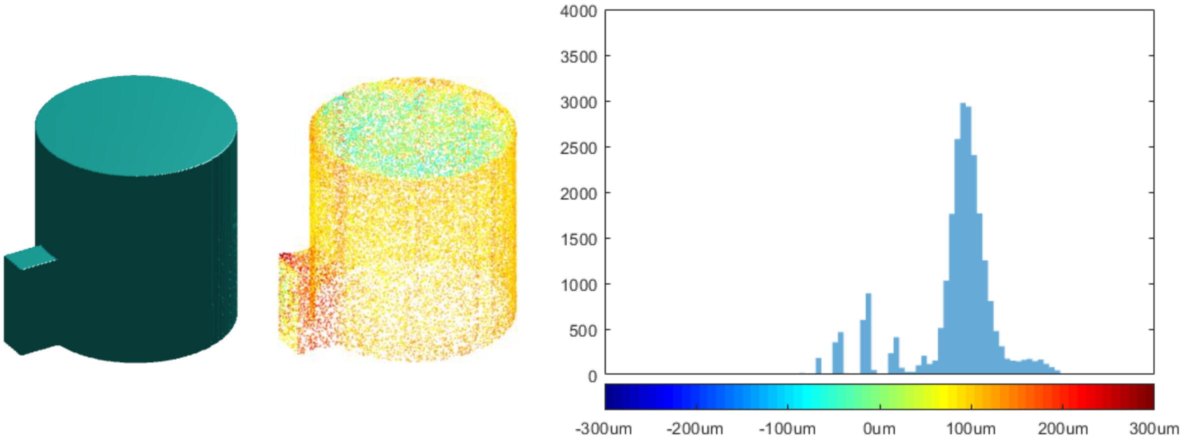


Figure A.6: Registered CT point cloud for cylinder 0019. SDF isosurface (left), registered point cloud (middle), and histogram of point cloud SDF values.

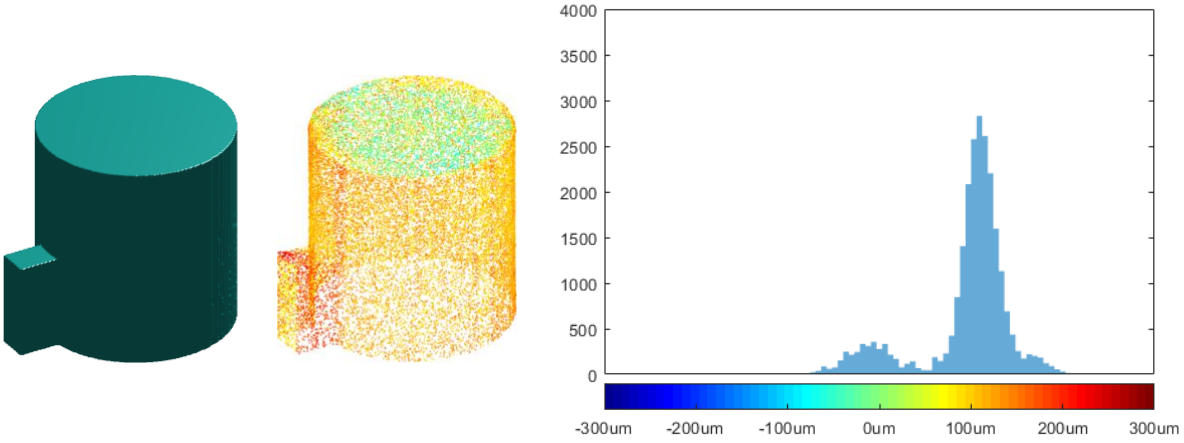


Figure A.7: Registered CT point cloud for cylinder 0022. SDF isosurface (left), registered point cloud (middle), and histogram of point cloud SDF values.

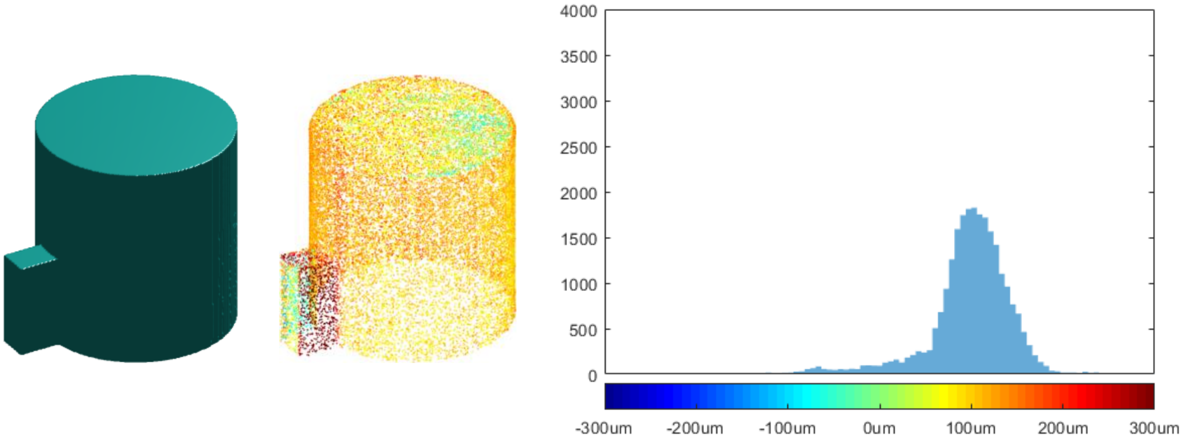


Figure A.8: Registered CT point cloud for cylinder 0023. SDF isosurface (left), registered point cloud (middle), and histogram of point cloud SDF values.

Appendix B

Registered Multispectral Point Clouds

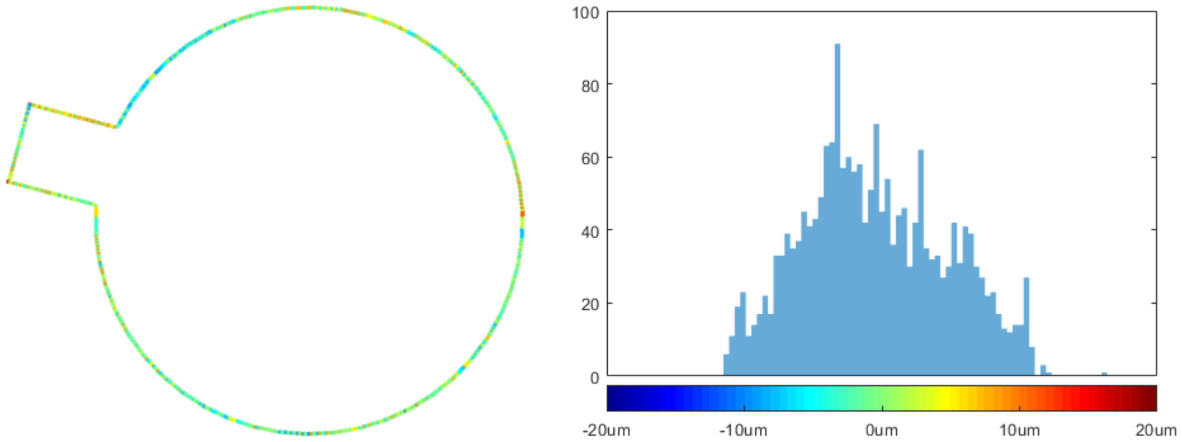


Figure A.9: Registered multispectral point cloud for cylinder 0004. Point cloud colored according to SDF distance (left) and histogram of SDF distances (right).

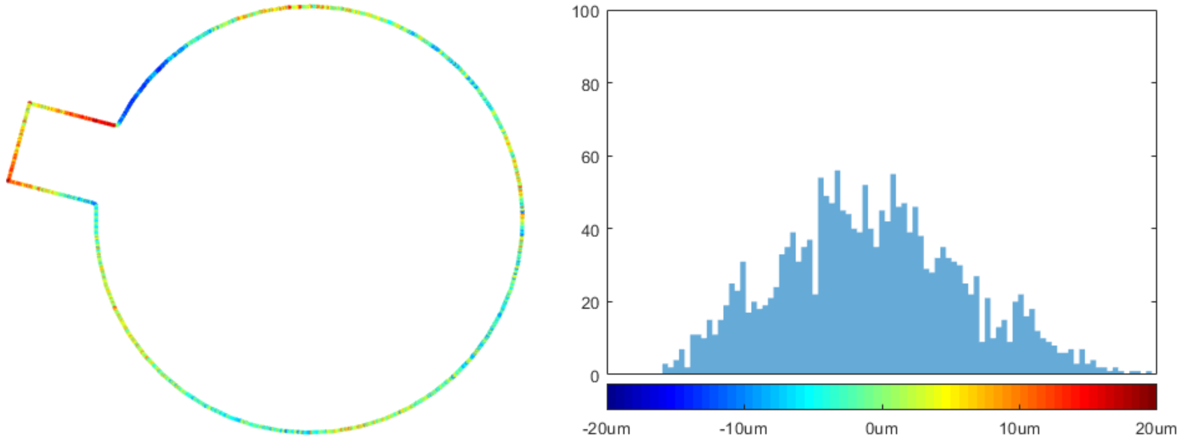


Figure A.10: Registered multispectral point cloud for cylinder 0005. Point cloud colored according to SDF distance (left) and histogram of SDF distances (right).

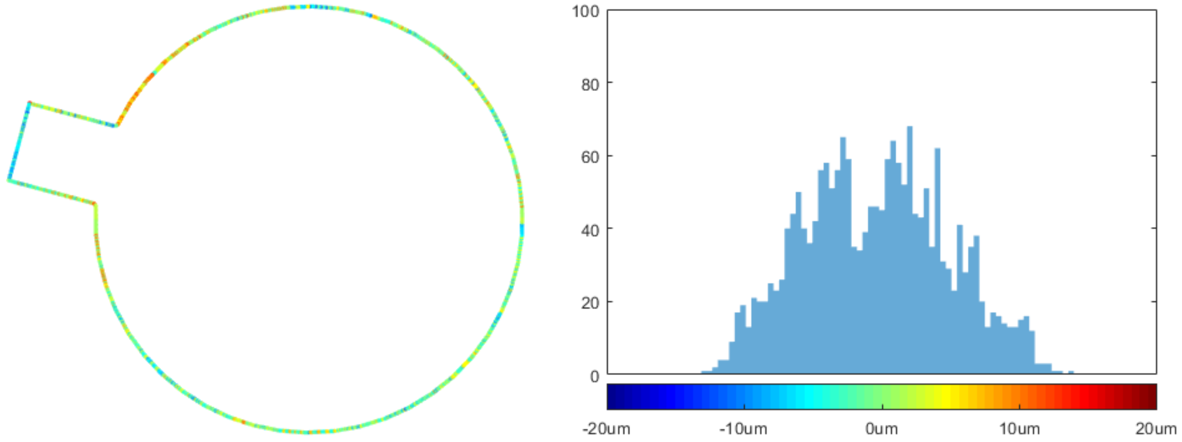


Figure A.11: Registered multispectral point cloud for cylinder 0014. Point cloud colored according to SDF distance (left) and histogram of SDF distances (right).

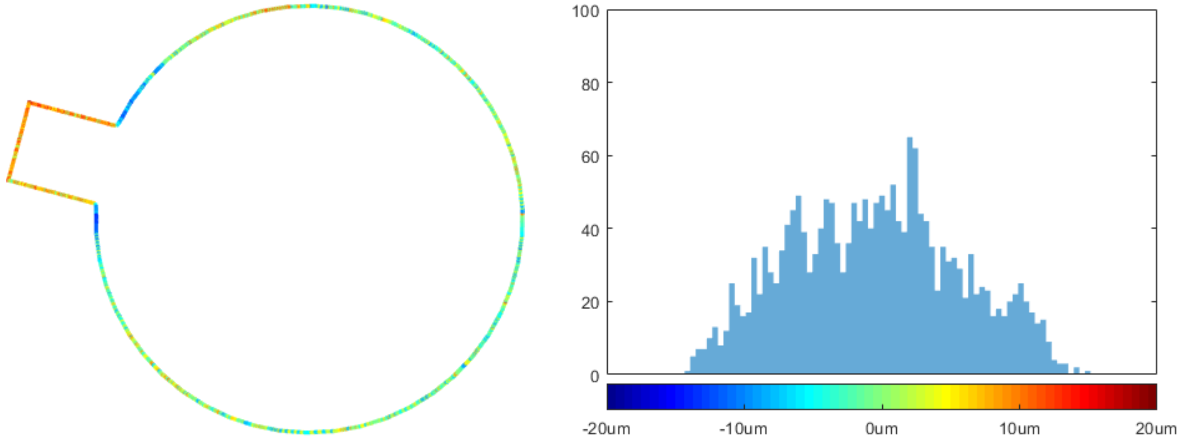


Figure A.12: Registered multispectral point cloud for cylinder 0015. Point cloud colored according to SDF distance (left) and histogram of SDF distances (right).

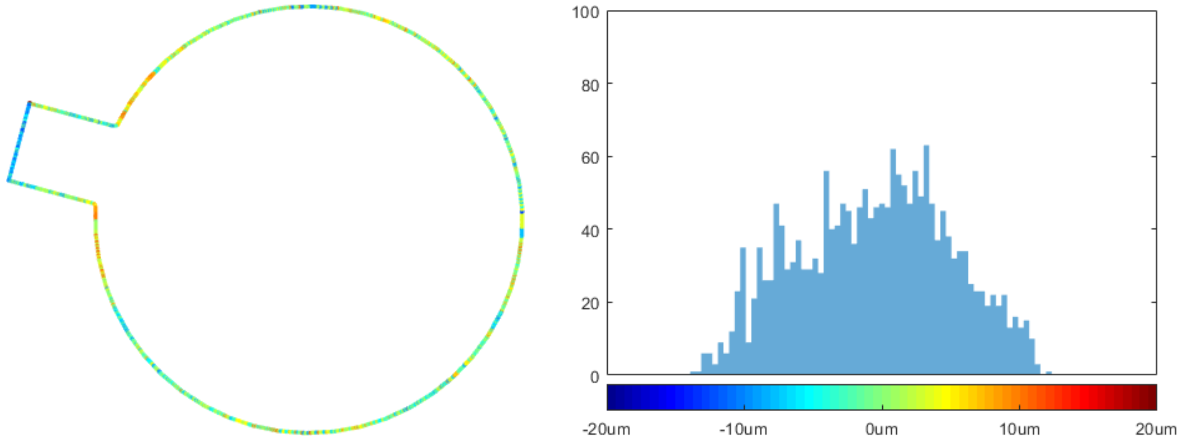


Figure A.13: Registered multispectral point cloud for cylinder 0018. Point cloud colored according to SDF distance (left) and histogram of SDF distances (right).

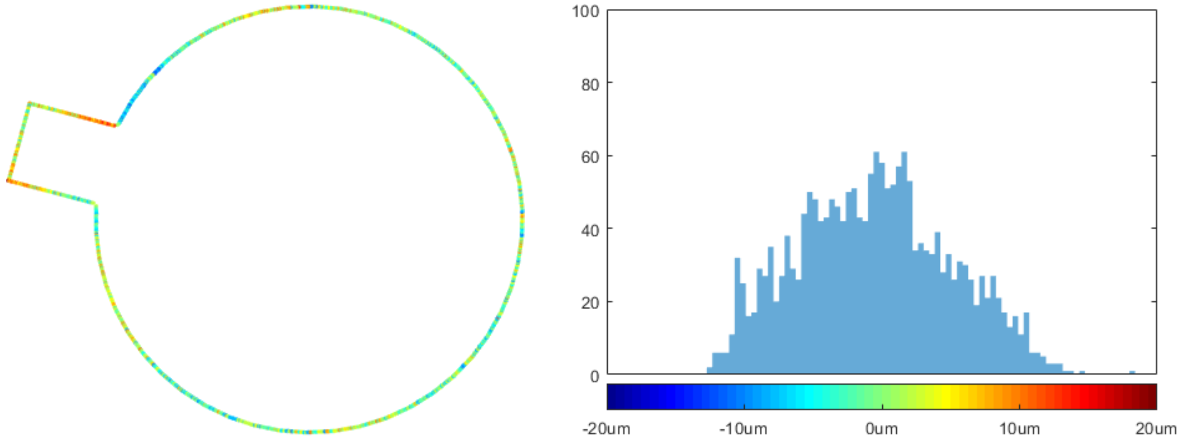


Figure A.14: Registered multispectral point cloud for cylinder 0019. Point cloud colored according to SDF distance (left) and histogram of SDF distances (right).

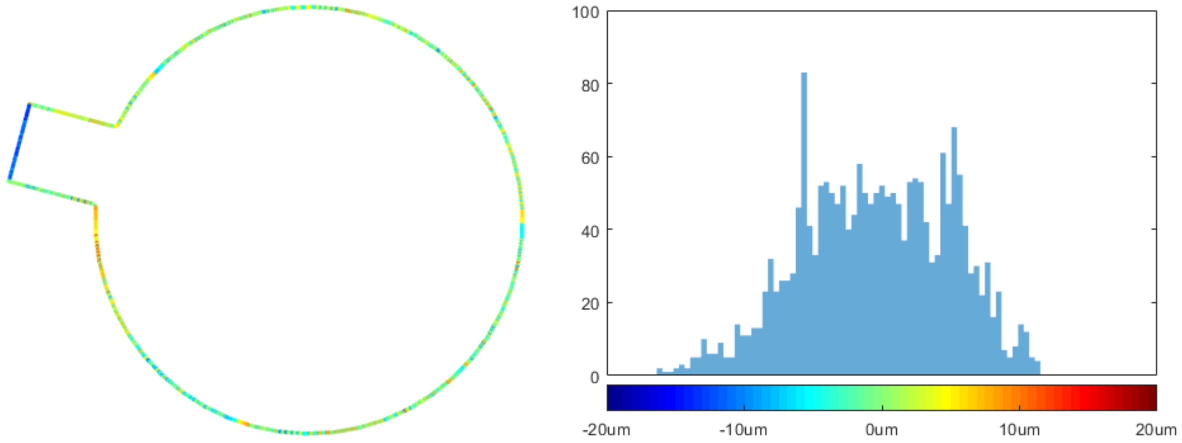


Figure A.15: Registered multispectral point cloud for cylinder 0022. Point cloud colored according to SDF distance (left) and histogram of SDF distances (right).

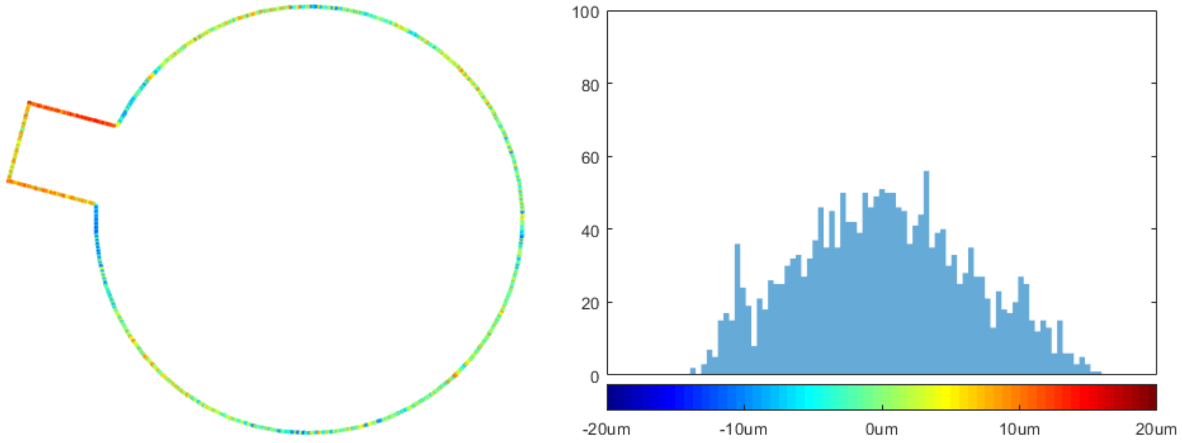


Figure A.16: Registered multispectral point cloud for cylinder 0023. Point cloud colored according to SDF distance (left) and histogram of SDF distances (right).

References

[1] H. Gong, K. Rafi, H. Gu, G. J. Ram, T. Starr and B. Stucker, "Influence of defects on mechanical properties of Ti–6Al–4 V components produced by selective laser melting and electron beam melting," *Materials & Design,* vol. 86, pp. 545-554, 2015.

[2] S. Yavari, S. Ahmadi, R. Wauthle, B. Pouran, J. Schrooten, H. Weinans and A. Zadpoor, "Relationship between unit cell type and porosity and the fatigue behavior of selective laser melted meta-biomaterials," *Journal of the mechanical behavior of biomedical materials,* vol. 43, pp. 91-100, 2015.

[3] Q. Liu, J. Elambasseril, S. Sun, M. Leary, M. Brandt and P. Sharp, "The effect of manufacturing defects on the fatigue behavior of Ti-6Al-4V specimens fabricated using selective laser melting," *Advanced Materials Research*, vol. 891, pp. 1519-1524, 2014.

[4] D. Hu and R. Kovacevic, "Modelling and measuring the thermal behaviour of the molten pool in closed-loop controlled laser-based additive manufacturing," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture,* vol. 217, no. 4, pp. 441-452, 2003.

[5] T. Craeghs, F. Bechmann, S. Berumen and J.-P. Kruth, "Feedback control of Layerwise Laser Melting using optical sensors," *Physics Procedia,* vol. 5, pp. 505-514, 2010.

[6] D. Hu and R. Kovacevic, "Sensing, modeling and control for laser-based additive manufacturing," *International Journal of Machine Tools and Manufacture,* vol. 43, no. 1, pp. 51-60, 2003.

[7] G. Tapia and A. Elwany, "A review on process monitoring and control in metal-based additive manufacturing," *Journal of Manufacturing Science and Engineering,* vol. 136, no. 6, 2014.

[8] S. Kleszczynski, J. z. Jacobsmühlen, J. Sehrt and G. Witt, "Error detection in laser beam melting systems by high resolution imaging," in *Proceedings of the Solid Freeform Fabrication Symposium*, 2012.

[9] J. z. Jacobsmühlen, S. Kleszczynski, D. Schneider and G. Witt, "High resolution imaging for inspection of laser beam melting systems," in *IEEE International Instrumentation and Measurement Technology Conference*, 2013.

[10] J. zur Jacobsmühlen, J. Achterhold, S. Kleszczynski, G. Witt and D. Merhof, "Robust calibration marker detection in powder bed images from laser beam melting processes," in *IEEE International Conference on Industrial Technology*, 2016.

[11] K. Bartkowiak, "Direct laser deposition process within spectrographic analysis in situ," *Physics Procedia,* vol. 5, pp. 623-629, 2010.

[12] A. R. Nassar, T. J. Spurgeon and E. W. Reutzel, "Sensing defects during directed-energy additive manufacturing of metal parts using optical emissions spectroscopy," in *Solid Freeform Fabrication Symposium,* 2014.

[13] A. Spierings, M. Schneider and R. Eggenberger, "Comparison of density measurement techniques for additive manufactured metallic parts," *Rapid Prototyping Journal,* vol. 17, no. 5, pp. 380-386, 2011.

[14] J. J. Moré, "The Levenberg-Marquardt algorithm: implementation and theory," *Numerical analysis,* pp. 105-116, 1978.

[15] C. R. Maurer, Rensheng Qi and Vijay Raghaven, "A linear time algorithm for computing exact Euclidean distance transforms of binary images in arbitrary dimensions," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 25, no. 2, pp. 265-270, 2003.

[16] E. Hecht, Optics, 4th, San Francisco: Addison-Wesley, 2002.

[17] H. M. Merklinger, Focusing the view camera, Seaboard Printing Limited 5, 1996.

[18] P. Burt and E. Adelson, "The Laplacian Pyramid as a Compact Image Code," *IEEE Transactions on Communications,* vol. 31, no. 4, pp. 532-540, 1983.

[19] H. Edelsbrunner, D. Kirkpatrick and R. Seidel, "On the shape of a set of points in the plane," *IEEE Transactions on Information Theory,* vol. 29, no. 4, pp. 551-559, 1983.

[20] J. A. Bilmes, "A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," *International Computer Science Institute,* vol. 4, no. 510, p. 126, 1998.