

Layer by Layer Validation of Geometrical Accuracy in Additive Manufacturing processes

Kai Zeng^a, Nachiket Patil^a, Hengfeng Gu^a, Haijun Gong^a, Deepankar Pal^a,
Thomas Starr^b, Brent Stucker^a

^a Department of Industrial Engineering, University of Louisville

^b Department of Chemical Engineering, University of Louisville

1 Abstract

Geometrical inaccuracy from shrinkage and residual stress-induced deformations are key sources of defects in Additive Manufacturing (AM). In most AM processes the CAD model is represented by an STL file which is sliced into layers. The Common Layer Interface (CLI) and StereoLithography Interface (SLI) files are two layer file formats which store this slice information and can be data mined to analyze sources of error. By comparing the original STL file against a 'stacked' model based upon SLI and CLI files, a "theoretical" part accuracy can be created. In addition, these SLI/CLI files can be combined with detailed process parameter knowledge such as scan pattern, laser power, scan speed, etc., to enable a numerical prediction of part accuracy and deformations on a layer-by-layer basis. This paper introduces a research project which is developing software and hardware tools to enable prediction and measurement of part dimensions on a layer-by-layer basis with the goal of real-time part quality validation and closed loop control.

2 CLI and SLI comparison

CLI and SLI file formats are commonly used to represent the layer information for sliced STL/CAD models. Sliced files have the contour information of each layer which includes points' coordinates, orientation information to distinguish the inner and exterior contours, layer thickness and number, and hatches filling the contour. Different machine manufactures use different slicing strategies and file formats which increase difficulties for file exchange. Even though each file format has unique ways to represent and store the model information, the data is supposed to be the same regardless of file format. Take EOS DMLS machines, for example; the SLI file format is the default input file format. Although the SLI file format has no public, standard description of its format and structure, a CLI file has an open format and standards, with makes it easier for researchers to use. CLI permits access to the detailed layer information and it was posited that CLI file information could help us understand the proprietary SLI file. The main question to answer thus became whether the CLI and SLI files have the same information for the same model. In order to test this, the template matching method was adopted to answer this question.

2.1 Template matching method

Since there is no way for a standard user to directly access an SLI file, a comparison of a snapshot of the layers produced by EOS software with CLI layers enables one to understand if they are comparable. The steps used in this comparison are as follows: input and display the same layer of the SLI and CLI file in Netfabb software; take screenshots of the images of the layer; make one of the pictures the template and calculate the normalized correlation coefficient of the other picture compared to the template.

Template matching is used to match the two images by calculating the normalized correlation coefficient. The correlation coefficient is described in equation (1).

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X\sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X\sigma_Y} \quad (1)$$

where $\mu_X, \mu_Y, \sigma_X, \sigma_Y$ are the mean and variance of X and Y respectively

The normalized correlation coefficient is shown in equation (2)

$$\gamma(u, v) = \frac{\sum_{x,y}[f(x, y) - \bar{f}_{u,v}][t(x - u, y - v) - \bar{t}]}{\left\{\sum_{x,y}[f(x, y) - \bar{f}_{u,v}]^2 \sum_{x,y}[t(x - u, y - v) - \bar{t}]^2\right\}^{0.5}} \quad (2)$$

where:

- f is the image
- \bar{t} is the mean of the template
- $\bar{f}_{u,v}$ is the mean of $f(x,y)$ in the region under the template

2.2 Example

Figure 1(left) shows a CAD model of a hemisphere which has been manufactured in an EOS M270 machine. One of its cross-sections is shown in Figure 1(right). Cross-sections of the 200th layer from CLI and SLI files are shown. The normalized correlation coefficient of these two pictures is shown in Figure 2. The maximum of the correlation coefficient is 1 which means these two pictures are identical.

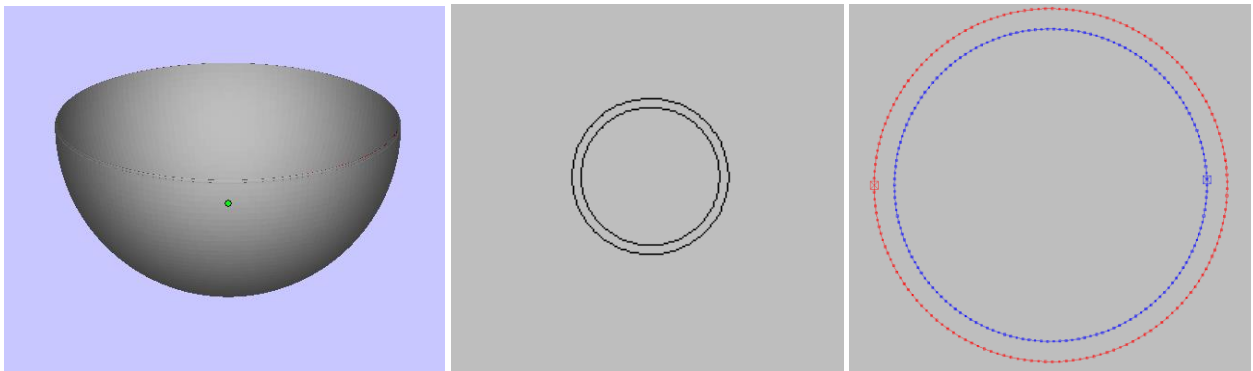


Figure 1 (Left): Hemisphere CAD model; (Middle): cross section of 200th layer from SLI screen shot
(Right): points in cross section of 200th layer from CLI file

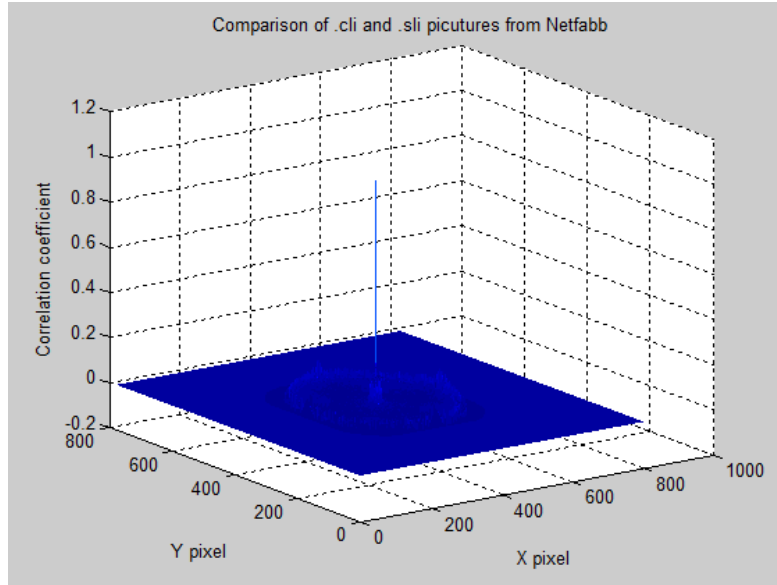


Figure 2 Correlation between two cross-sections corresponding to CLI and SLI

3. Stacking

Geometrical accuracy and deformation are two main defects in Additive Manufacturing (AM). When an AM machine reads the cross section information from CLI/SLI files and builds and stacks these layers to create the 3D geometry (Figure 3), geometrical errors due to the CAD model can be found by comparing the STL against the 'stacked' model.

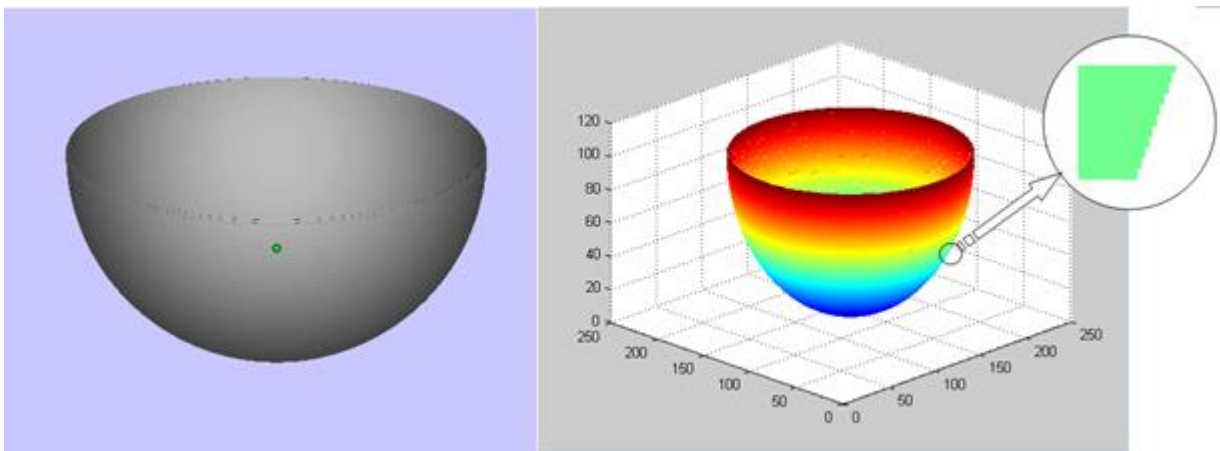


Figure 3 Comparison of original CAD model of a semi sphere against the stacked model

Left: Solid CAD model

Right: Stacked model from layers of cross section with stair errors

3.1 STL file

Taking the example from figure 3, the cross section in the vertical direction can be compared against the STL model. Figure 4 shows an STL model of the Sphere from figure 3 [1].

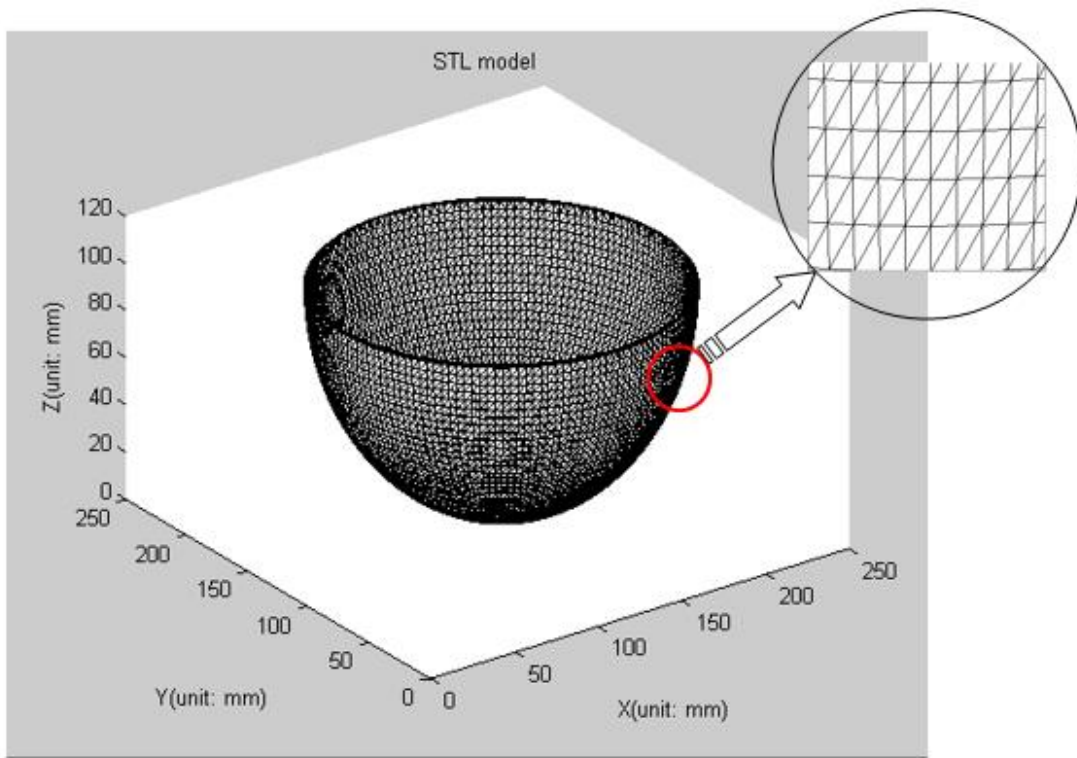


Figure 4 STL model of Sphere showing in figure 3

The contours of vertical cross sections are used in this study. Figure 5 (left) shows triangles which cross the center plane in the X direction and the right picture shows the shadow view of these triangles in the YZ plane. There are total of 352 triangles crossing this cross sections' exterior and interior surface.

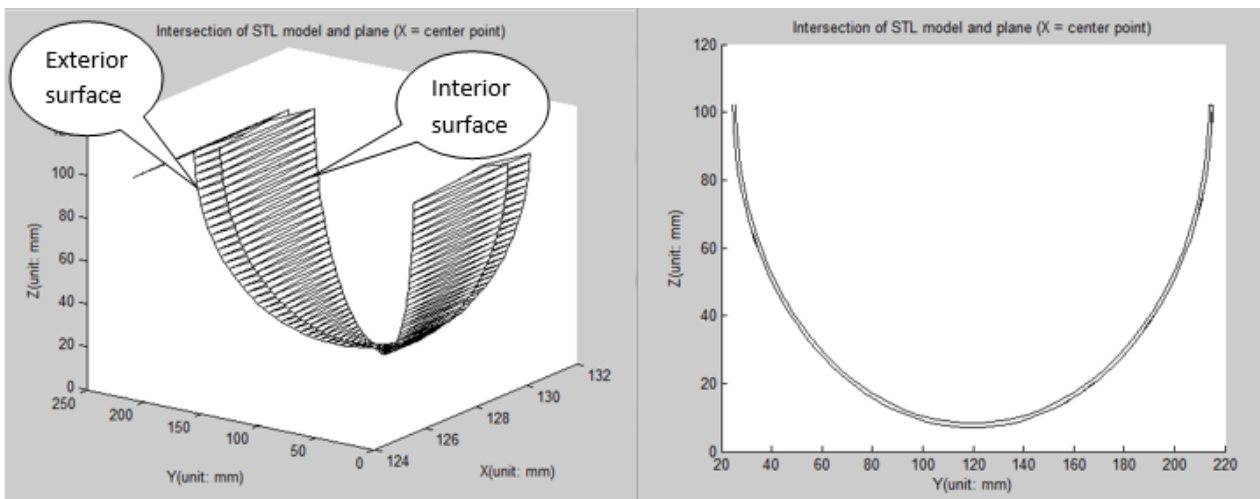


Figure 5 (left)Triangles in STL file which intersect plane $X = 129.4962$, (right) STL model seen from YZ plane

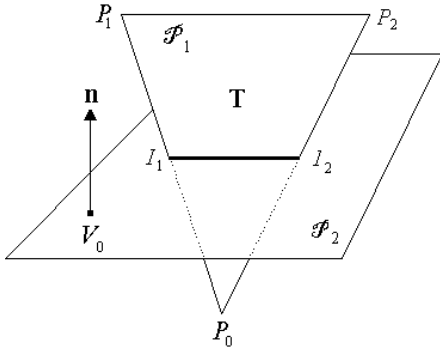


Figure 6 intersection of triangles and plane
<http://geomalgorithms.com/a06-Intersect-2.html>

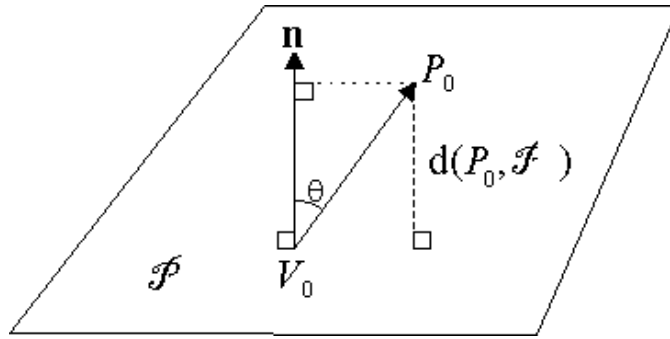


Figure 7 Distance of point to plane
<http://geomalgorithms.com/a04-planes.html#Distance-Point-to-Plane>

The contour in the right picture is the projection profile of the STL sphere in that plane. The specific coordinates of the contour can be captured by calculating the intersection of the triangles and the cutting plane, which is shown in figure 8. The intersection of a triangle and a plane is easily calculated using geometry and is widely used in computer graphics and collision detection methods.

3.2 Intersection of Triangle and plane

The intersection of a triangle and a plane is determined by first deciding the relative position of the triangle's three vertices against the plane. Figure 6 shows the triangle T intersecting with the plane V_0 in line I_1I_2 . Line P_1P_0 and P_2P_0 clearly intersect the plane. However, the distance of a point to a plane must be used to determine that P_1 and P_2 are on one side of V_0 and P_0 is on the other side. As shown in figure 7, $P_0 = (x_0, y_0, z_0)$ is a random point, plane φ is described by $a*x + b*y + c*z + d = 0$. The distance from point P_0 to plane φ is calculated using equation (3).

$$d(P_0, \varphi) = |P_0 - V_0| \cos \theta = \frac{ax + by + cz + d}{\sqrt{a^2 + b^2 + c^2}} \quad (3)$$

The distance $d(P_0, \varphi)$ is a signed value. For the three vertices in a triangle, positive multiplying of either one of the two distances means those two vertices are on the same side of a plane.

3.3 CLI file

There are two types of CLI files: Binary and ASCII. Both of their formats are well documented [2]. The CLI file is a 2½ D representation of a model. Each contour consists of coordinates of vertex points which are connected continuously as a polyline. In order to distinguish contours for exterior surfaces and interior surfaces, clockwise and counter clockwise point orders are used to describe internal and external contours respectively. As a result an original model's characteristics in the XY plane for each layer is kept; however, the Z dimension only has layer thickness information, which is why CLI is a 2½ D representation of a model.

As shown in figure 1, points and contours in a layer can be read from a CLI file. This contour is the exact shape that the AM machine should build for each layer. Assuming the final shape of each layer is exactly represented by the CLI file, the whole AM building process can be simulated by stacking layers

regardless of the scan pattern in each layer. We can see a representative mismatch between STL and CLI boundaries in figure 8. The STL boundary is cut by the vertical plane, however, the CLI boundary is first sliced by the horizontal slicing plane and then projected to the vertical plane. The spots 2 and 3 are locations which have high mismatch between the STL and CLI boundaries. The average mismatch would be less than what is shown. These errors are known as the stair-step effect. In this case, the building of the i^{th} block is based on the i^{th} layer contour [3].

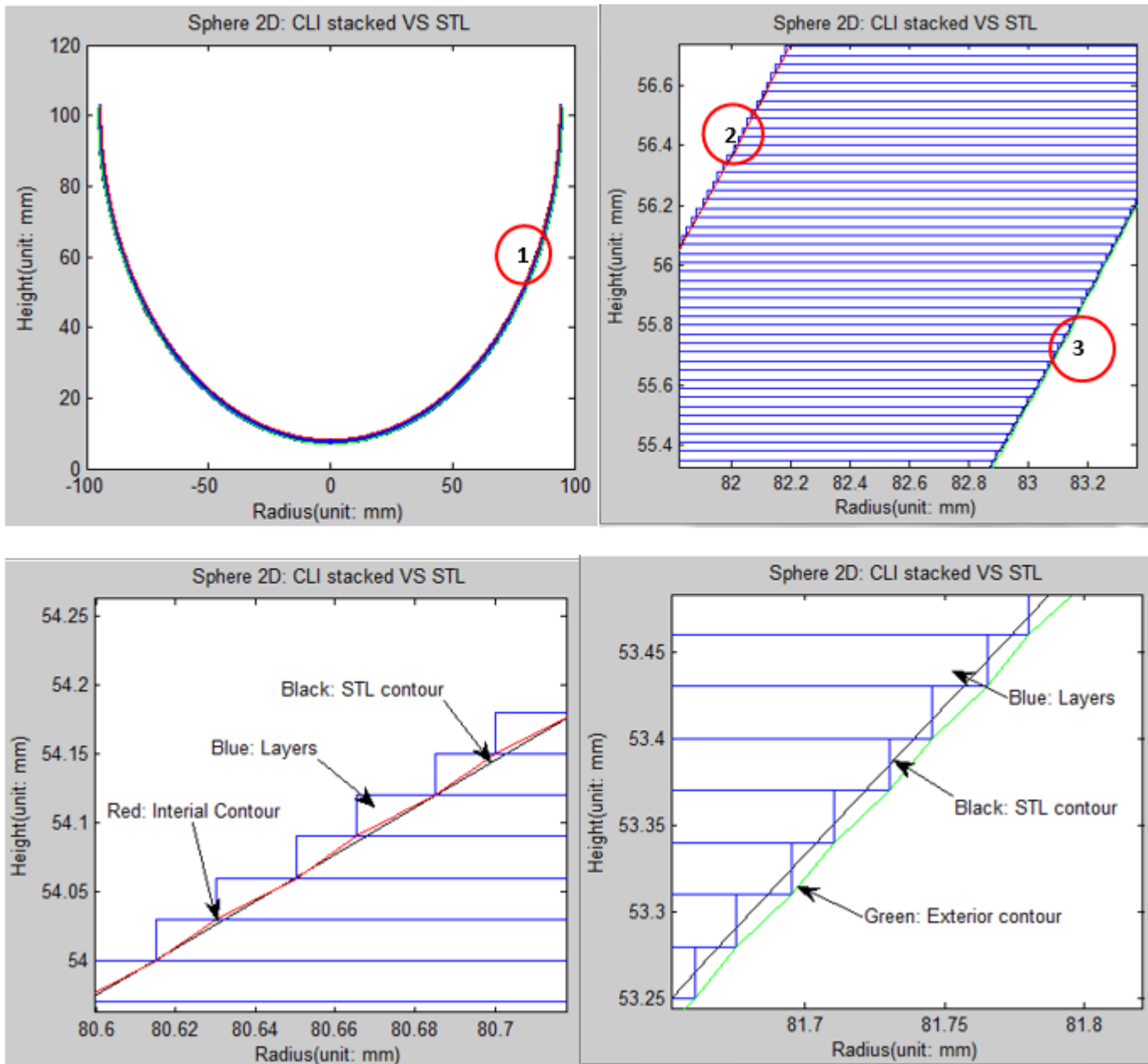


Figure 8 comparison of the CLI contour against the STL contour for the cross section in figure 5
 (Upper left): CLI VS STL of the sphere in 2D; (Upper right): The magnified picture of 1 in upper left
 (Lower left): The magnified picture of 2 in upper right; (Lower right): The magnified picture of 3 in upper right

In figure 8, the exterior stair-step is inside the CLI boundary, unlike the interior stairs. The stair-step error can be calculated quantitatively as cusp height, as shown in figure 9 [3]. From the histogram of the cusp height, the mean value of the cusp height is half the layer thickness which is 30 microns, that means if there is post processing to smooth the surface, at least 15 microns of the surface will be undersized.

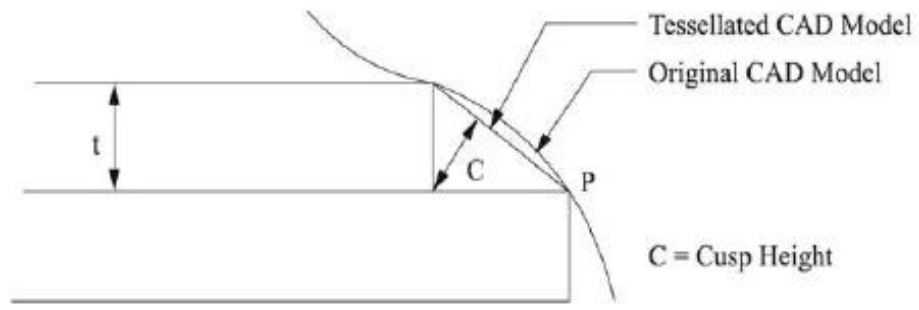


Figure 9 cusp height

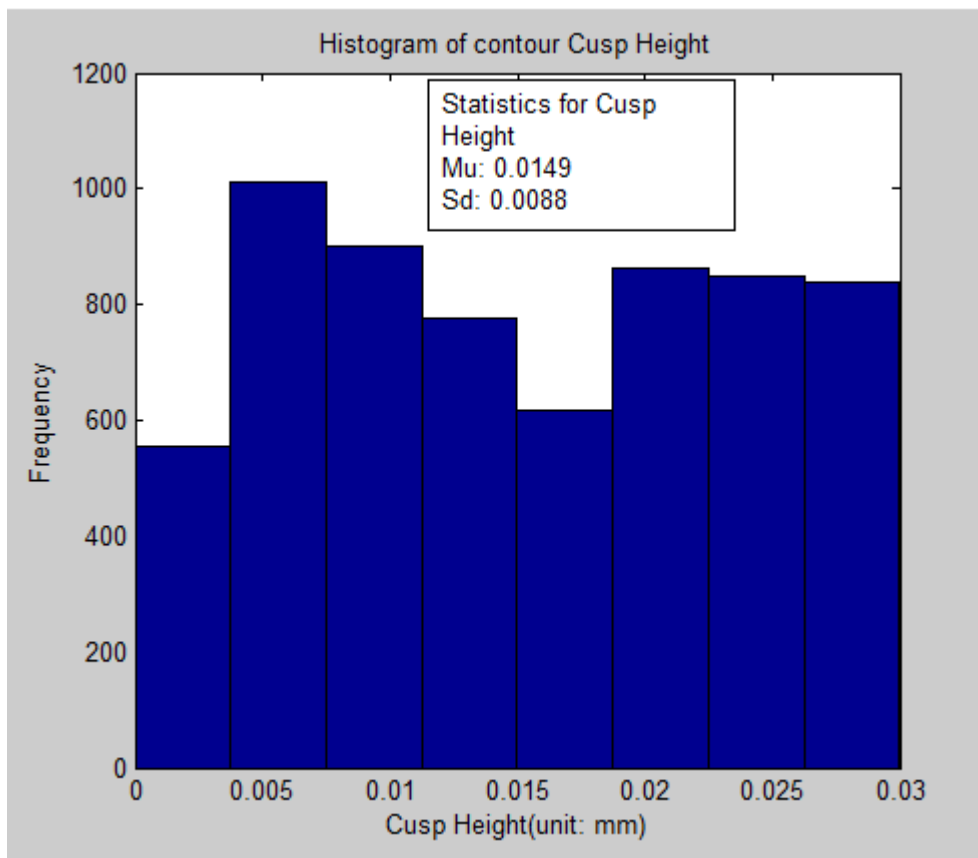


Figure 10 Histogram of contour Cusp Height

4 Scan Pattern

Scan pattern and process parameters are key factors for understanding quality control in AM processes. Specific details including scanning orientation, hatch spacing, and beam offset are able to be extracted from the manufacturing process rather than assuming a certain pattern. Scan pattern plays an important role in thermal evolution, and thus deformation. By using the actual scan pattern rather than an assumed pattern, the thermal model can produce more reliable simulation results.

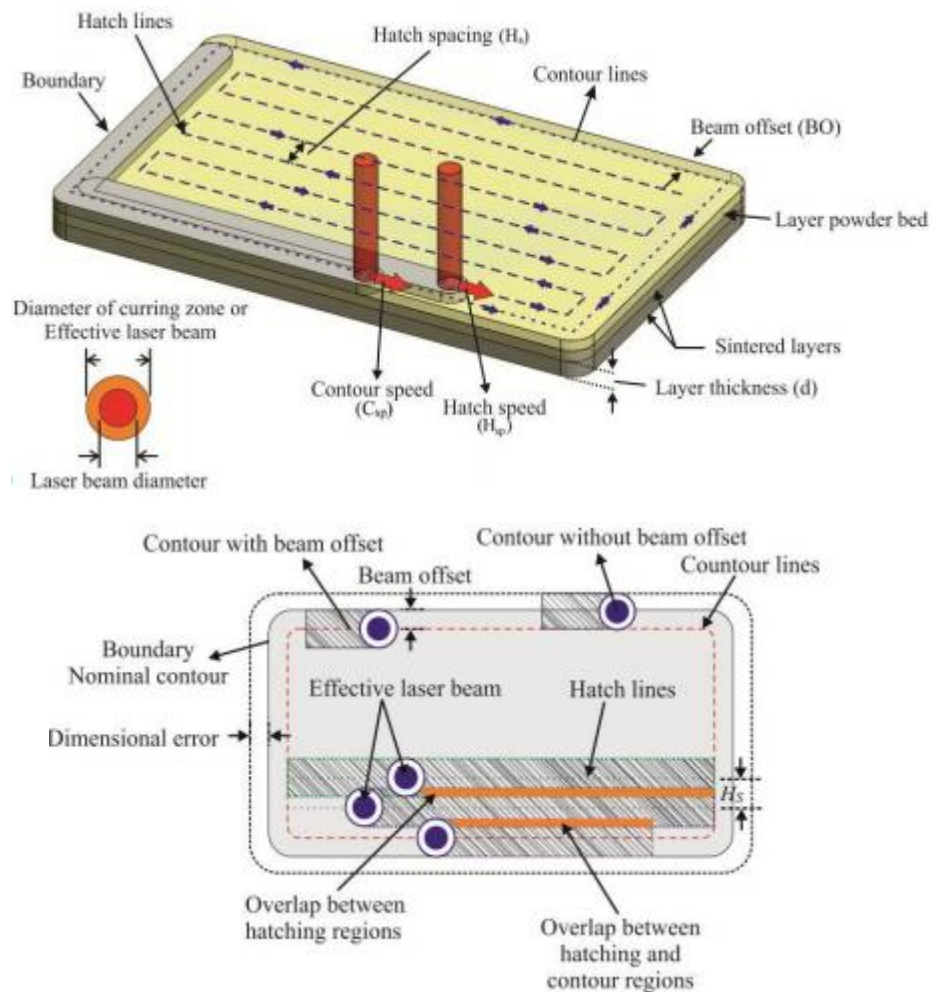


Figure 11 Exposure strategy and process parameters [4]

A laser beam has an energy profile, such as a Gaussian distribution. As shown in figure 11, the laser beam moves in a pattern which has a hatch spacing and velocity. In order to reduce the dimensional error, a beam offset is used to account for the effective diameter of the laser beam. When the laser scans the boundary, it will increase the dimension of the part by half of the beam's diameter unless the boundary contour is shifted towards the inside of the part by the radius of the beam to compensate for dimensional error.

The scan pattern shown in figure 11 is the traditional S pattern with scan lines parallel to the boundary. However, commercial AM machines usually scan parts in patches. Figure 12 is a real scan pattern picture taken from an EOS M270 machine. Red lines in the picture are the boundary contours. Green lines represent scan lines. From the picture, the scan lines are not parallel to either of the boundaries. And the discontinuous 'bold lines' are actually junctions of the scan lines in adjacent patches. The 'bold lines' are illusion of those junctions and 'lines' do not actually exist. The upper picture in figure 12 indicates there are four strips which are scanned independently. The lower picture is a magnification of the corner of the upper picture. Hatch spacing, beam offset and scan lines are shown in the picture.

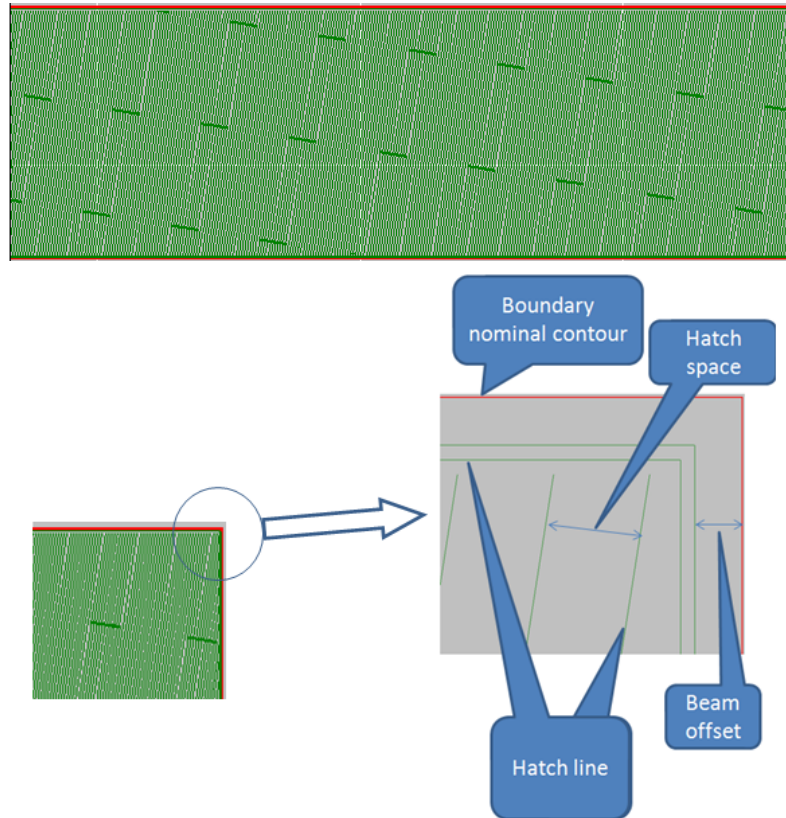


Figure 12 Real scan pattern from EOS M270 machine

4.1 Line detection: Hough transform

In order to extract the quantitative scan pattern from the pictures taken from an AM machine, the Hough transform line detection method is used. It is well known that a straight line can be described using equation 4 in Cartesian coordinates.

$$y = ax + b \quad (4)$$

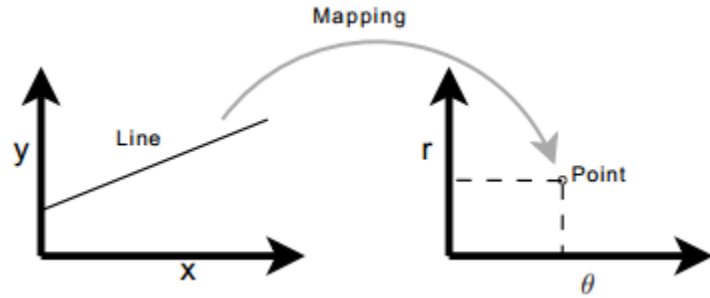


Figure 13 Line representation in Cartesian and Polar Coordinate [5]

The slope 'a' and intercept 'b' characterize the equation. However, when the line is parallel to the Y axis, the slope b approaches infinity. A line in polar coordinates can avoid this problem which is shown in equation 5. The line we see in Cartesian coordinates is a point in the polar coordinate. What is more, one point (x, y) in the Cartesian coordinate gives a sinusoid in Polar Coordinates. M collinear points lying on the line will give m sinusoidal curves which intersect at the same point in Polar coordinates. The idea of the Hough transform is to discretize the (ρ, θ) domain and find the local point which has the maximal intersects.

$$\rho = x \cos \theta + y \sin \theta \quad (5)$$

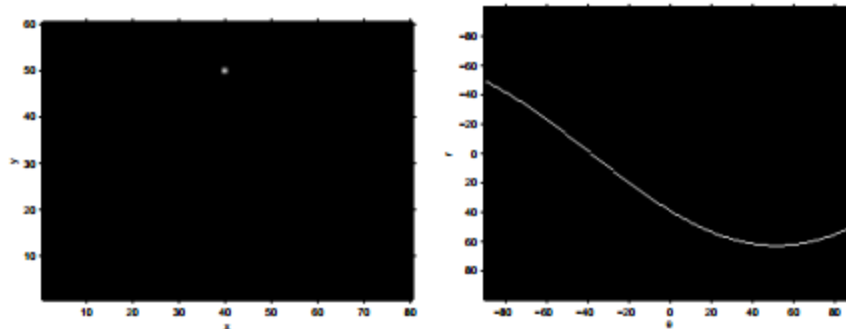


Figure 14 point in Cartesian and Polar Coordinate [5]

According to the Hough transform line detection method, the scan pattern from one of the pictures, as shown in figure 15, can be decomposed into process parameters. The beam offset is 50 microns, the hatch spacing is 30 microns, and the patch distance is 5 millimeters. As shown in figure 15, there are four scan patches, and the laser scan through these four patches from 1 to 4. The laser beam will scan the boundary contour before and the inner scanning at the end of a layer. According to observations of a video of the process, there will be a 12-14 seconds pause to allow the machine to bring down the platform and spread new powder. Assuming the melt pool shape is an ellipse with ratio of long and short radius of 2 [6], laser scanning in a layer can be captured, as shown in figure 16. Simulation of laser scanning shows that there are some voids because of improper process parameters.

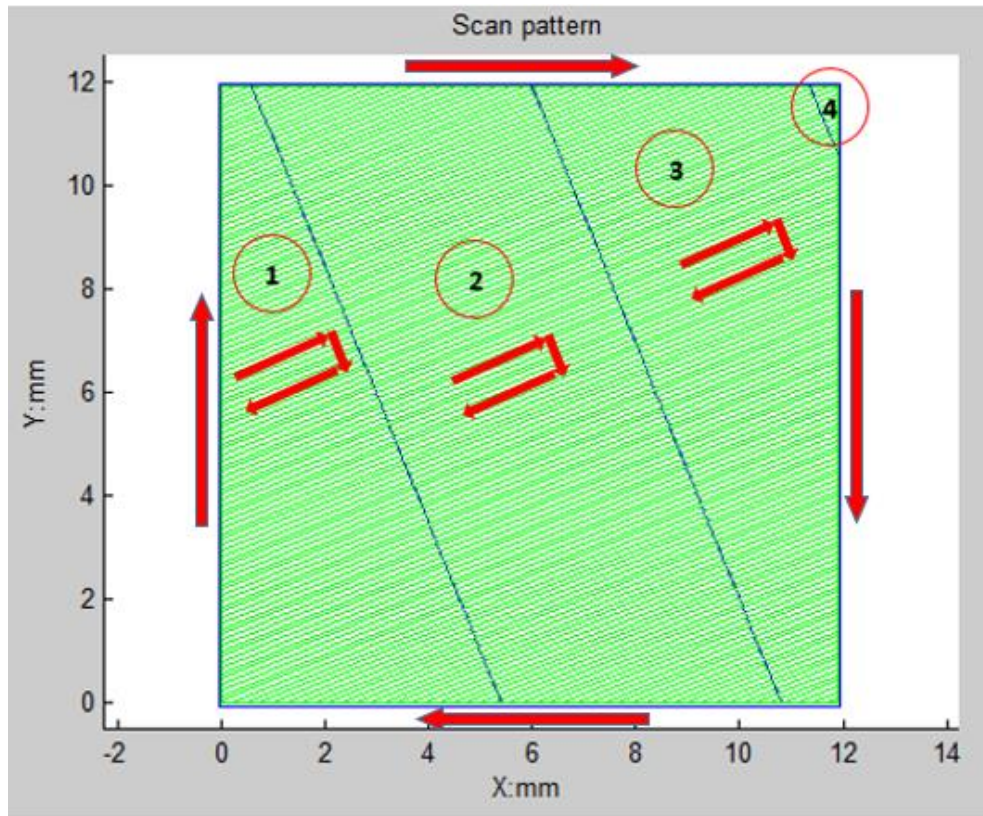


Figure 15 Scan pattern extracted from picture

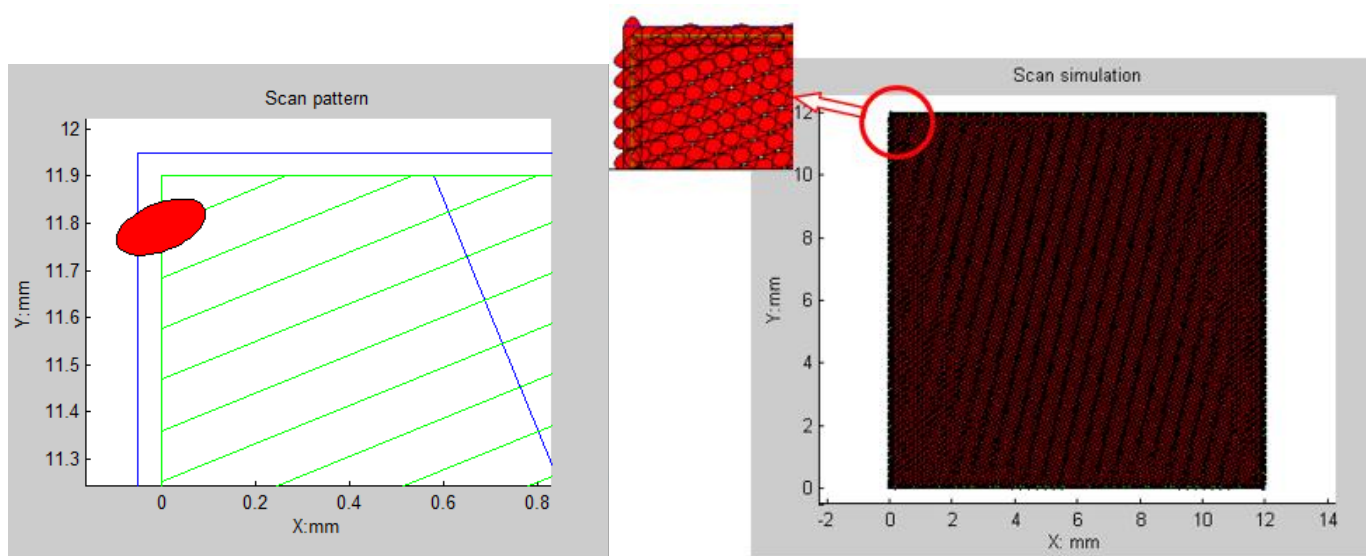


Figure 16 Scan simulation with assumed melt pool shape

5 Future work

This paper covers the current abilities of software tools being developed to enable automated process parameter and geometry capture for prediction of part dimensions on a layer-by-layer basis with the goal of real-time part quality validation and closed loop control. The geometry in this case study was a hemisphere, but more features and geometries need to be tested to ensure universal applicability. In addition, although the scan pattern can be extracted from the AM process and fed directly into the model, there are some other factors like scanner acceleration/deceleration and other beam compensation techniques that might not be captured. These factors may also be very important for predicting dimensional errors and quality control. These and other aspects will be considered in the future.

References

[1] Ian Gibson, David W. Rosen, Brent Stucker. Additive Manufacturing Technologies: Rapid Prototyping to Direct Digital Manufacturing. Springer New York 2009

[2] http://www.forwiss.uni-passau.de/~welisch/papers/cli_format.html

[3] Pulak M. P, N. Venkata R and Sanjay G. Dhande. Slicing Procedure In Layered Manufacturing: A Review, Rapid Prototyping Journal, V9 N5 2003 pp. 274-288

[4] Aulus Roberto Romão Bineli, Ana Paula Gimenez Peres, André Luiz Jardini, Rubens Maciel Filho. Direct metal laser sintering (DMLS): Technology for design and construction of microreactors. 6^o CONGRESSO BRASILEIRO DE ENGENHARIA DE FABRICAÇÃO 11 a 15 de Abril de 2011 . Caxias do Sul . Associação Brasileira de Engenharia e Ciências Mecânicas 2011

[5] http://www.cvmt.dk/education/teaching/f09/VGIS8/AIP/hough_09gr820.pdf

[6] A. V. Gusarov, I. Yadroitsev, Ph. Bertrand, I. Smurov. Model of Radiation and Heat Transfer in Laser-Powder Interaction Zone at Selective Laser Melting. Journal of Heat Transfer, July 2009, Vol. 131