**Print Fidelity Metrics for Additive Manufacturing of Cement-based Materials**

Hajar Taheri Afarani1, Newell H. Moser2, Edward J. Garboczi2,

Ebrahim Nasr Esfahani3 and Joseph J. Biernacki1

1Department of Chemical Engineering, Tennessee Technological University, Cookeville, Tennessee

2National Institute of Science and Technology (NIST), Boulder, Colorado

3Department of Electrical and Computer Engineering, Tennessee Technological University, Cookeville, Tennessee

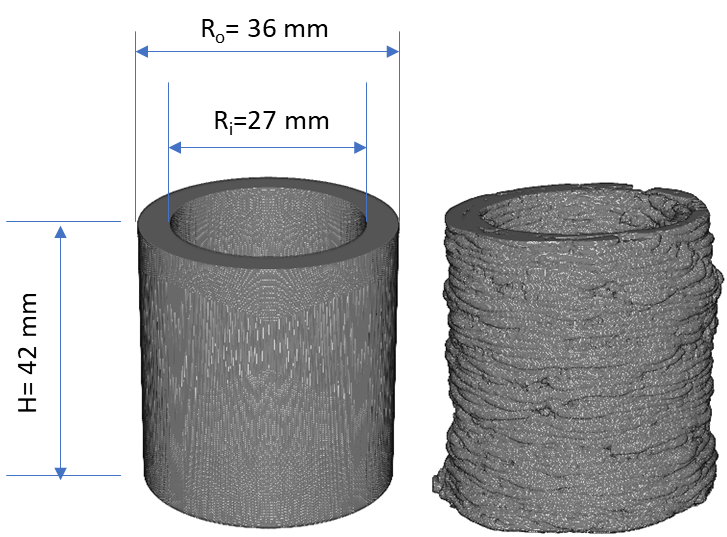
**Abstract**

Additively manufactured cement-based structures for infrastructure applications suffer from in-construction shape deformations, which are a strong function of process conditions and the rheology of the printing material (cement paste, mortar, or concrete). Thus, characterization of the shape of such manufactured objects is critical to establish and ensure fidelity to the original CAD model. In this study, a number of quantitative metrics were used to compare the dimensional and shape accuracy of laboratory-scale printed cement-based objects. A new method is described that uses the axes of minimum moment of inertia and the centroid as the basis for aligning and comparing objects. X-ray computed tomography (XCT) data was used to characterize both internal and external features. Details of the logic and image processing requirements are given and typical sample irregularities that lead to quantification of uncertainty are illustrated. The effect of sampling statistics on metric confidence was studied and guidelines are provided for good sampling protocols. The results show the extent to which different penalty logics provide sensitivity for the detection of specific types of flaws. Furthermore, when the minimum moment of inertia is used as the basis for alignment and comparison, a high correlation is found between boundary-based and volume-based fidelity metrics. Such quantitative printability metrics are necessary to establish a basis for evaluating the repeatable shape fidelity of 3D-printed objects and for quantitatively studying how rheology affects both the manufacturing process and the final built part. The method is illustrated for benchmark printed objects fabricated using three hydrogel forming polymers as printing aids.

**Keywords:** additive manufacturing, 3D printing, fidelity, quantitative, metrics, moment of inertia, flaws, cement, X-ray CT, imaging

1. **Introduction**

Additive manufacturing is the build-up of an objects through the layer-by-layer deposition of a material [1]. Although introduced in 1987, additive manufacturing is currently revolutionizing how a wide range of products is manufactured. Like any manufacturing process, errors affect the quality of 3D printed objects resulting in differences between the 3D printed object and the CAD models they are created from, as seen in Figure 1, which shows the original CAD model and the part printed from this model.

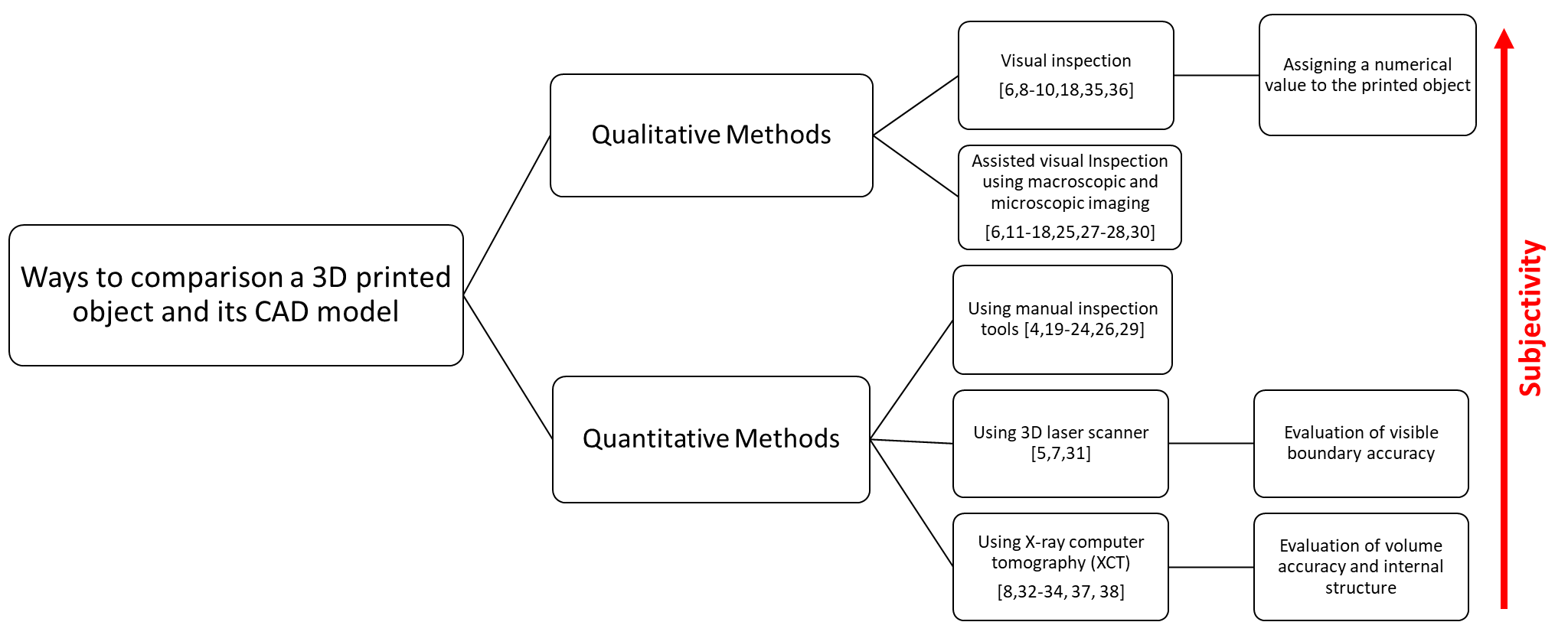


1. (b)

**Figure 1.** Virtual point cloud renderings of: (a) model object and (b) a printed object.

Despite wide-spread application of additive manufacturing in different industries including aerospace, automotive, and medicine, the assessment of dimensional accuracy and printability of manufactured objects has not yet been investigated thoroughly. To quantify the quality (fidelity) of a 3D printed object, it must somehow be compared to the CAD model. Although some additive manufacturing technologies enable the manufacture of objects with high fidelity, variations from the CAD model’s target dimensions are always present [2-4]. Numerous factors contribute to the inaccuracy of 3D printed objects including machine (tool) tolerance, the type of materials being printed, the print (build) logic, and the object size [2-3, 5]. Therefore, developing unbiased methods to assess shape fidelity and dimensional accuracy of printed objects would be beneficial to a broad range of industries.

There are many ways to compare printed objects and the digital counterparts from which they are created. Some of these methods are more subjective than others [6-10]. Figure 2 is a flow chart that summarizes the most commonly used comparison methods including qualitative and quantitative techniques.

****

**Figure 2.** Classification of methods for characterizing the fidelity of 3D printed objects [5-38].

In all cases there is some level of subjectivity which increases as the method becomes more qualitative. Whereas qualitative methods such as visual inspection are subject to the opinion of the observer even quantitative methods that utilize digital scanners or x-ray tomography are subject to the basis upon which the printed object and the model are indexed, i.e., aligned. Qualitative visual inspection has been used for assessing the deviation and shape fidelity of 3D printed objects either by assigning a numerical quality score to the printed object [6, 8-10, 18, 35-36] or by the enhanced visual judgment of 3D printed objects using macroscopic and microscopic imaging [6, 11-18, 25, 27, 28, 30]. For instance, Abdollahi et al. assessed the print quality of 3D printed cylinders of polydimethylsiloxane (PDMS) elastomer resin using visual inspection and by giving numerical scores to three characteristics, i.e. layer fusion, infill, and stringiness [10]. In this case, layer diffusion was referred to as the adhesion between layers during printing; infill was assessed by noting if the inside of the printed cylinders was hollow or not (i.e., filled with printing materials); and the lack of adhesion between the first few layers of the print was defined as the stringiness. A score of zero to 10 was given for each response variable and a total score formed by summation.

Such qualitative methods do not provide accurate measurement of printability since they are subjective and may even overlook some types of defects. To make a more reliable comparison between 3D printed objects and their CAD model, various quantitative methods have been suggested that utilize various hand-made or instrument assisted measurements, e.g., optical or electron microscopy [16-18, 25, 27, 28, 30], 3D laser-scans [5, 7, 31], or X-Ray computer tomographic renderings (XCT) [8, 32-34, 37-38].

For example, Li et al. [16] investigated the quality of direct ink writing (DIW) structures printed using cellulose nanocrystals (CNC) by analyzing the edges of a 1 cm × 1 cm × 1 cm printed cubic. For this purpose, optical images of the cubic shapes were converted into black-and-white and then a deviation for a portion of the perimeter of the projected area was calculated.

Indirect metrics have also been used in an attempt to correlate properties of the printing paste to shape fidelity. In one such study, the shape fidelity of bio-inks was assessed using a filament collapse test, in which the mid-span deflection of a suspended filament of bio-inks was measured [17, 18]. The catenary deflection of a gel filament deposited between two pillars was considered as a quantitative evaluation of shape fidelity. Ribeiro et al. [18] quantified this deflection after deposition of a filament at t=0 s and t=20 s by measurement of the angle of deflection at the edge of the suspended filament using video still images. Habib et al. [17] defined a collapse area factor (Cf) as a metric for determining the shape fidelity of such printed inks, i.e., 100 % times one minus the ratio of the area below the filament catenary and the theoretical area between two adjacent pillars. The value of Cf is between 0 and 100 %, with 100 % resulting for prints that do not retain their shape, and 0 % is for prints that do retain their shape [17].

Manual measurement tools such as a caliper, micrometer, and ruler have also been used for evaluating the accuracy and shape analysis of printed objects with simple geometries through measurement of dimensions such as diameter, height, width, and length [19-24]. Similarly, a printing accuracy metric, i.e., a shape fidelity factor, was developed by determining the total area of a printed object using digital caliper measurements [25-29].

Although the above approaches provide some information about the quality of 3D printed objects, they do not provide precise quantification of printability since they are dependent on how the measurements are done and how many measurements are made. For instance, visual inspection methods depend subjectively on how appealing to the eyes an object is, so that they are only useful for judging objects based on appearance and cannot detect internal defects. Manual measurement-based methods are likewise fraught with procedural errors, e.g., incorrect handling of calipers. Thus, to reduce measurement errors and the subjectivity of such methods, more automated techniques have been developed and reported. The implementation of image analysis, for example, has been demonstrated. Ouyang et al. [30] studied the influence of bio-ink properties and printing parameters on bio-ink printability using two-dimensional (2D) optical microscope images of printed orthogonal screen structures (mesh patterns with a square pitch). In their approach, the printability was evaluated as the ratio of circularity of a perfect square over the actual circularity of the pore shape.

3D laser scanning has also been used for characterization of 3D printed objects [31]. 3D printed nanocellulose cubes were scanned and surface point clouds of high-resolution were obtained. The obtained point clouds were used to extract principal dimensions (X, Y, and Z). Quantification of print quality was performed by comparing the obtained dimensions to the reference geometry from which the object was printed [31]. This concept was further [7] extended and a printability index based on the external dimensional mismatch of the 3D computer-generated model and the 3D printed object was introduced. The printability of silicate-based slurries was assessed in this way using tens of thousands of external surface points [7]. Although this technique was useful, it lacked information about the internal surface geometries and optically hidden defects, i.e. pore structure [7]. Furthermore, this method requires alignment of the scanned point cloud and the surface of the 3D model, a task that can be somewhat subjective and difficult.

Because 3D laser scanners only provide information about the surface of 3D printed objects, X-ray computer tomography (XCT) has more recently been applied since it images internal and external features [32-34]. For instance, the geometric accuracy of a 3D printed aortic valve made from poly-ethylene glycol-diacrylate (PEG-DA) hydrogel supplemented with alginate was assessed using XCT [32]. The XCT scans were reconstructed into stereolithography (STL) geometries [32]. The surface deviations between the reconstructed STL model of the printed geometries and native STL models were quantified through alignment of the scan data and the CAD model and expressed in terms of surface heat maps [32, 37]. The alignment was performed by defining landmark points on the CAD model and reorienting the model and the printed object relative to each other using a fitting procedure. To compensate for the inability of the surface heat map to assess the internal geometric fidelity, a slice-by-slice comparison of each printed layer to the original STL files in the XY-plane was also performed using Boolean operations. For a given slice, regions printed outside the model print area were designated as “overprint” error, and regions that were not printed within the model area were designated “underprint” error [32].

XCT was similarly employed to analyze 3D printed hollow cylinders prepared from two different polymer filaments, polyvinyl alcohol (PVA) and polylactic acid (PLA) [33]. The quality of the printed hollow cylinders was examined based on the co-registration of the CAD model and the XCT data by applying logical operators. Co-registration is the process of transforming different sets of data into one coordinate system by selecting one image as the reference to which all the other images are aligned [38]. Logical operations were performed for the co-registered XCT and CAD model datasets. No additional details were provided regarding how the model and the printed object were aligned. Logical operators were used to identify a sub-volume of the XCT data that did not overlap with the CAD model, which was used to evaluate the printing accuracy.

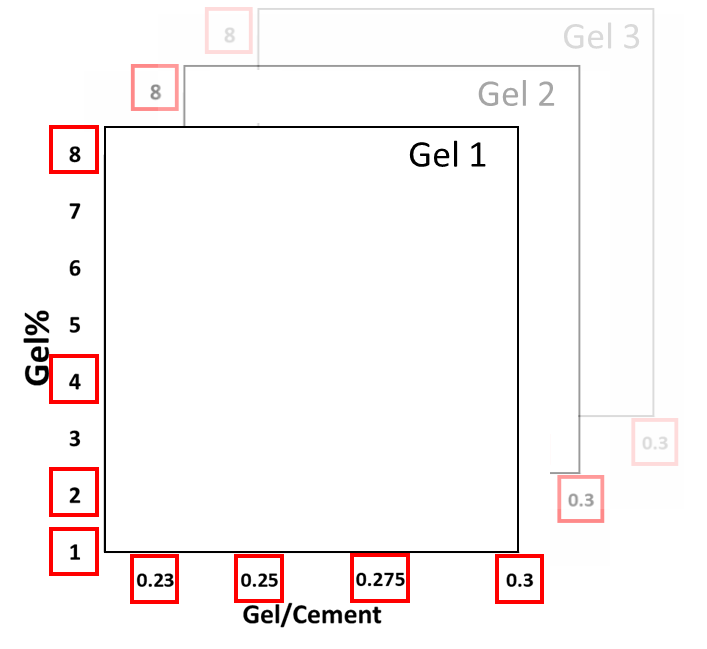
In [34], the so-called residual volume, the volume occupied by one but not both objects i.e. for point sets A and B, was suggested as a metric to quantify differences between 3D images generated from XCT. For this purpose, the XCT image and the CAD model were aligned using a landmark-oriented strategy based on picking at least three points and equating them to points on the CAD model [34]. Details of how the landmarks were aligned is unclear.

Figure 2 summarizes the main strategies that have been used to evaluate print fidelity. The primary gap in the prior work is either the subjectivity of the measurement methods used and/or the subjectivity imposed by the alignment strategy. Furthermore, the need for an unbiased print fidelity metric is apparent in ongoing efforts to model and develop additive manufacturing processes [35-36,39]. The present work is an extension of these techniques; XCT images were used and a new strategy for unbiased alignment was developed that utilizes the center of mass and the axes of minimum moment of inertia as the spatial basis. This approach removes both measurement and alignment subjectivity, does not rely on any form of hand-held instrument or by-eye determinations and utilizes fundamental properties of the object’s geometry, i.e. the axes of minimum moment of inertia and centroid, to provide unbiased alignment. The moment of inertia is related to the distribution of mass in an object with respect to a specified axis, which therefore accounts for porous regions in the build where mass is effectively zero. Purely geometric references, e.g. lining up the basal planes of the real and printed object, cannot account for pores. The porosity of a built part, as well as the local spatial distribution of pores, are important aspects contributing to its overall quality. Moreover, the use of geometric references can result in bias errors when calculating the geometric deviations. Such bias might be minimized if a translation and rotation are applied to re-position the printed object with respect to the model object, and then a minimization procedure used to find the global minimum in the geometric deviation between the printed object and the model object. Not only is a procedure like this nontrivial, but it does not account for pores and other such defects.

The minimum moment of inertia is used here as an alternative alignment strategy. Two quantitative metrics were calculated, one using surface boundaries and the other using volume [8]. The new alignment strategy and two metrics (printability indexes), i.e., boundary-based printability (*PIB*) and volume-based printability (*PIV*), are presented here as alternative quantitative descriptions of geometric fidelity of any printed object. Furthermore, boundary-based (surface-based) comparators do not penalize for internal features when the commonly used landmark-based strategies are used to index, i.e., align, the model and printed object. Thus, by imposing an alignment strategy that is inherently sensitive to the distribution of mass within the object, i.e., the minimum moment of inertia, standard boundary-based comparators will likewise become sensitive to internal flaws. The statistical validity, ability to discern defects of various kinds, high correlation between *PIB* and *PIV* and an application are demonstrated.

1. **Materials and Methodology**

A 4×4×3 full-factorial experimental design was used to print a collection of samples prepared at 48 different conditions using three hydrogel-forming polymers as printing aids at four concentrations and four gel to cement ratios, refer to Figure 3. These samples were discussed in detail in the authors’ prior work [8] and are used here are a basis for comparing the proposed minimum moment of inertia method for characterizing print fidelity. The materials and methods used to print and characterize these samples are presented in the following sections.



**Figure 3**. Matrix illustrating the 4×4×3 full-factorial experimental design using for printing of samples [8].

* 1. **Materials**

Three different hydrogel forming polymers were used: WALOCEL M-20678 hydroxyethyl methyl cellulose (HMEC), METHOCEL™ 240 hydroxypropyl methyl cellulose (HPMC), and DOW Chemical POLYOX WSR 301 polyethylene oxide (PEO). Deionized water (DI) was used for preparation of all the hydrogels. Sakrete Type I/II portland cement from a single sack was used for the preparation of all cement pastes.

* 1. **Printing protocols**

**Mixing** – Hydrogels with four different polymer contents (i.e., 1 %, 2 %, 4 %, and 8 % by mass) were prepared by mixing dry polymer and DI-water until gelation occurred. Gels were aged over night before further use. Cement-based printing pastes were prepared by hand mixing the desired amount of gel and cement and then further shearing by hand using a mortar and pestle until a cohesive paste was obtained, typically 10 min. Prepared pastes were used within 120 min to avoid hydration-induced stiffening. All characterized samples were printed immediately after mixing, generally within the first 30 min after mixing. Additional information on the preparation of gels and cement-based pastes can be found elsewhere [8].

**Printing** – A Hydra 16A (Hyrel **3D,** Norcross, GA) printer[[1]](#footnote-2) was used. A 25 cm3 paste canister (reservoir) and a 0.15 cm diameter nozzle was used. Objects were printed at a print speed of 0.10 cm/s with a layer height of 0.1 cm. Hollow cylinders were printed with an inner and outer diameter and height of 2.7 cm, 3.6 cm, and 4.2 cm, respectively. More details on the printing procedures can be found elsewhere [8].

* 1. **X-ray computed tomography (XCT) procedures**

3D printed hollow cylinders were scanned using a Zeiss Versa XRM500 CT scanner and XCT datasets were obtained [8]. The scanner had a voltage range of 30 kV to 160 kV, with maximum power of 10 W. The resolution of the instrument was controlled using lenses (0.4×, 4×, 20×, 40×) covered with scintillation material, with the 0.4× lens used for all scans. A beam voltage of 80 kV, a power of 10 W, exposure times between 0.8 s to 1.90 s, and full 360° rotation were used. All of the 3D printed hollow cylinders were scanned with a resolution of 1004 pixels × 1024 pixels in each plane and voxel sizes ranging from 42 μm to 47 μm, which were determined by fitting the width of each specimen into the 1024-pixel x 1024-pixel field of view (FOV). Scanning with the abovementioned specifications produced 16-bit/pixel grayscale images corresponding to 216 shades of gray. Complete XCT scans typically produced about 750 grayscales reconstructed XCT images, which were 2D slices across the sample. Vertically stacking these images gave a 3D image of each sample.

* 1. **Processing of the grayscale images** 
     1. **Pre-processing: grayscale normalization, thresholding and binarization**

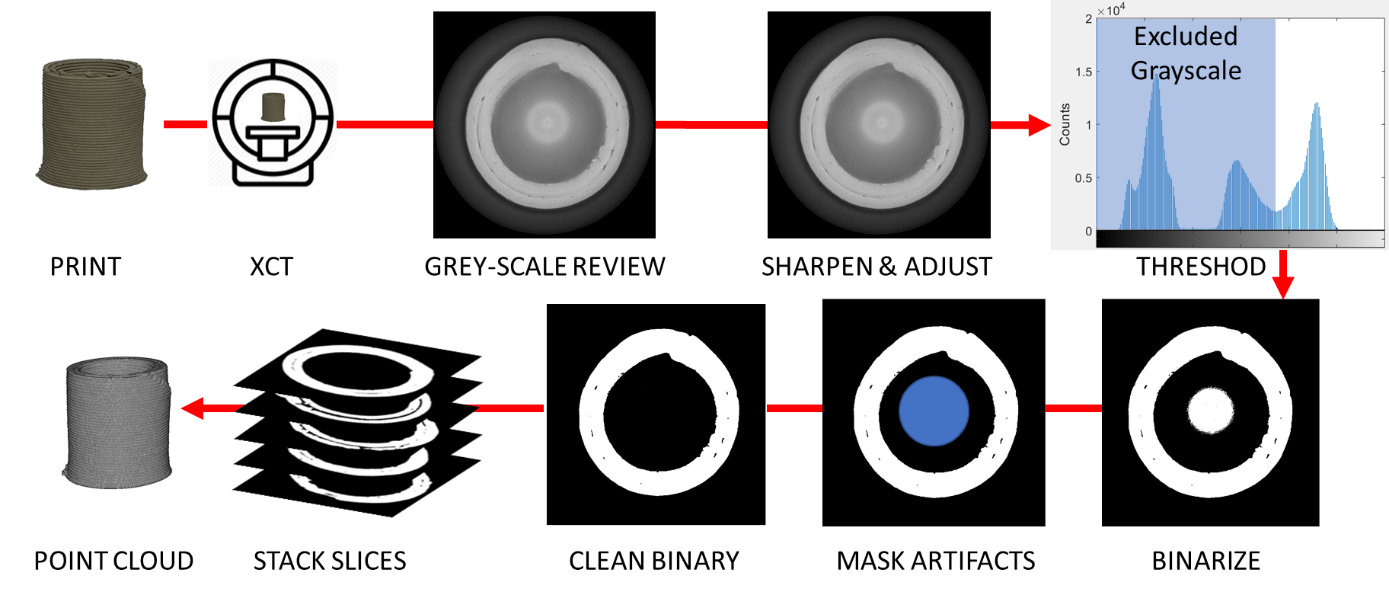
A series of image conditioning treatments were applied to the grayscale XCT images to normalize the grayscale contrast, improve edge identification, remove any reconstruction artifacts, and segment into binary images. To help with feature identification, all 2D images were first sharpened to improve edge discrimination. Although complete object scans were done using constant instrument settings, since the scans ran for at least one-hour natural fluctuations in the XCT produced images with somewhat different grayscale histograms. This is also an artifact of the reconstruction algorithms, with images near the top and bottom of the FOV being treated a bit differently besides having somewhat different X-ray exposures. Prior to thresholding and segmentation, in some cases it was helpful to establish a grayscale histogram with sufficient contrast to distinguish object features from background [40- 43]. To achieve this, a single 2D slice with native grayscale contrast satisfactory for thresholding and segmentation was identified; this image will be referred to as the “standard image.” All other images were normalized to the standard image grayscale using the MatLab [**mat2gray**](https://www.mathworks.com/help/images/ref/mat2gray.html#d123e212963) function. This process scales the grayscale intensities of a subject image so that the scaled grayscale histogram is best fit to that of the standard image. Table 1 summarizes all of the MatLab functions used and their application. Thus, the object of interest was separated based on the analysis of the image histogram by determining a single grayscale range.

**Table 1.** List ofMatLab functions used and their specific application.

|  |  |
| --- | --- |
| **MatLab Function** | **Use\*** |
| **imread** | reads binarized images |
| **im2double** | converts image to double precision |
| **mat2gray** | normalizes grayscale |
| **bwboundaries** | locates interior and exterior boundaries |
| **fminunc** | finds the minimum moment of inertia |
| **imtranslate** | performs linear translation |
| **vrrotvec** | calculates rotation between two vectors |
| **vrrotvec2mat** | convert rotation from axis-angle to matrix representation |
| **vertcat** | vertically concatenates matrixes |
| **histogram** | generates a histogram |
| **bwconncomp** | locates center of mass of point cloud |
| **Imrotate3** | rotation of 3D image |

\*Some MatLab routines have “options” that control their functionality. In all such cases the default options were used.

Figure 4 is a schematic illustration of the complete process, from obtaining reconstructed grayscale images to the final 3D point cloud representing the scanned sample. The cross-sectional reconstructed images contained a circular artifact near the center of each slice, where no sample material existed, increasing in size and brightness towards the top of the object. With the threshold used, this artifact became white, so was removed using a circular user-defined mask. With the images sharpened, grayscale normalized if necessary and reconstruction artifacts removed, the images were segmented into binary images [44, 45]. Figure 4 includes an illustration of the segmentation thresholds that were used. The segmentation thresholds were set by partitioning the grayscale histogram of the standard image into grayscale ranges that define regions of solid and air. Such “partitioning” generally occurs where there is a clear distinction between grayscale values and is generally located in a trough in the grayscale histogram, as shown in Figure 4. Small variations in the absolute location of the chosen grayscale threshold value thus do not make a significant difference in the dimensions or geometry of the resulting image [46]. The standard image threshold ranges, shown on the grayscale histogram of Figure 4, were then fixed and applied to all images. Notably, the threshold values were selected to eliminate domains that were clearly not features of the printed object and to retain features including large-scale pores. Pores below the scan resolution of nominally 40 m were thus excluded from the final binarized images. While other segmentation strategies might have been used, this simple logic was chosen for demonstrating the proposed alignment strategy.



**Figure 4.** From printed object to point cloud. Printed objects were scanned, grayscale images were reviewed, sharpened, threshold applied, binarized and masked to produce clean binary images for each XCT plane (slice). Planes were stacked and a point cloud generated.

* + 1. **Generation of the model object**

In order to create a digital representation of the model object (i.e., a voxel model of the benchmark geometry at the same resolution as the XCT images), the intersection between a series of planes and a hollow cylinder of the correct dimensions was analytically calculated; the set of plane-intersections is representative of the XCT cross-sectional images of the printed object. These intersection-contours were then transformed into the appropriate coordinate system and binarized so that they could be directly compared to the 3D printed objects.

* + 1. **Generation of point clouds**

A point cloud is a large number of points that form the geometry of a 3D object. The point cloud defines surface (boundary) locations and internal structure, e.g., pore networks, and is usually the set of coordinates defining surface or volume points in 2D or 3D. The digitally generated binary images in this study, both model and printed, were used to produce surface or volume point clouds of integer values where each value of 1 denotes the location of solid matter and is associated with a digital address (i.e., or index) in voxels measured from some origin, usually taken to be at a corner of the image. The integer addresses are easily converted to Cartesian coordinates in physical units, considered to be the center of each voxel or pixel, by multiplying the integer coordinates by the instrument resolution, e.g. in units of length/voxel edge. For the purposes of this paper, further interpolation of the surface at a sub-voxel level was not considered to be necessary.

* + 1. **Calculation of the center of mass (COM) and translation**

The center of mass of both the 3D printed object and the digital model object was determined using the MATLAB functions **regionprops** and **bwconncomp**, which are listed in Table 1. The model object and the actual printed object point clouds were then translated so that their centers of mass coincided with [0, 0, 0]. Translation is a linear operator:

(1)

where is the translated coordinate set, is the original coordinates of the raw point cloud in the coordinates of the XCT, and is the center of mass in the XCT coordinates.

* + 1. **Calculation of centroidal axis of minimum moment of inertia**

Prior to making any quantitative comparison of the similarity between a printed object and a model object, a spatial-basis or datum must be established upon which all comparative measurements could be made. For example, select points on the surface of the model (landmarks) and analogous points on the actual object could be aligned [34, 47], or the axial moments of inertia can be aligned keeping the base planes co-planar [8]. Such methods, however, have limitations and may not penalize for inter structural defects [31, 34, 48]. Alternatively, the coordinates defining the vector producing the minimum moment of inertia (*Imin*) and the center of mass could be used.

For a given rigid-body object, different axes of rotation will have different moments of inertia about those axes. In general, the moments of inertia are not equal unless the object is symmetric about all axes. Assuming that the moments of inertia are not equal, then there exists an axis of rotation corresponding to the minimum moment of inertia. Note that *Imin* will always pass through the center of mass of the rigid object and so is always a centroidal moment of inertia.

In this study, a viscous cement paste mixture was excreted through a nozzle that traversed along predefined toolpaths. A right circular hollow cylinder was chosen as the benchmark geometry. While any shape can be used in theory, shapes that are easily defined using analytical mathematical descriptions are easiest to use. Although a simpler geometry was chosen, the following methodologies can be readily extended to arbitrarily shaped components, as described and demonstrated in Section S.3. of the Supplemental Materials. The dimension of a right circular hollow cylinder is defined by the internal radius, , external radius, , and height, , as shown in Figure 1. The perfect cylinder is oriented such that the height of the cylinder is coaxial with the z-axis of the global coordinate system. Such cylinders either have an *Imin* that is coaxial with the z-axis and perpendicular to the base of the perfect cylinder or lies in the xy-plane which passes through the object’s COM, i.e., the centroidal xy-plan. For the dimensions used in this study, the model cylinder’s *Imin* lies in the centroidal xy-plane. However, the printed cylinders are not perfect due to manufacturing process errors. For example, the real cylinder may contain sizable pores due to lack-of-fusion, and its shape may deviate from that of the benchmark cylinder. Consequently, *Imin* for the real cylinder will not necessarily lie in the centroidal xy-plane of the printed cylinder. In general, each printed cylinder will have a unique *Imin*.

Reconstructed XCT image stacks were used to calculate the axis of rotation corresponding to the minimum moment of inertia for each of the printed cylinders. The experimental parameters used for the XCT measurements were described in more detail in Section 2.3. After post-processing, the cross-sectional images (see Section 2.4.1), the exterior geometry, and the interior defects of each printed cylinder were identified based on pixel values: white pixels corresponded to solid material, and black pixels corresponded to the absence of material. These black and white binary images were then used to calculate *Imin* of the physical cylinders.

The moment of inertia tensor for a discrete particle system is defined by the matrix *T* given by:

(2)

where *mk* is mass density and are solid material (white) positions relative to the center of mass [49,50] and k ranges from 1 to K, where K is the number of solid points defined in the 3D image volume. If only the *Imin* is desired, and because it is assumed that each white pixel represents an equal amount of mass, then each point mass, , can be set equal to one. The direction defined by the unit vector *n* that defines the minimum moment of inertia can then be found by minimizing the following equation:

(3)

given that, (4)

where *In* is the moment of inertia about the centroidal vector *n* and is the minimum moment of inertia.

* + 1. **Rotating the surface of 3D printed object**

For purposes of making surface-based comparisons, only surface point clouds are needed, which reduces the memory burden and number of calculations that must be done. As mentioned before, the direction of the unit vectors that define the minimum moment of inertia for the model object and the 3D printed object are not necessarily the same. Using these unit vectors, a rotation matrix was formed to align them with each other. Though it is arbitrary which unit vector is rotated to align with the other, we chose to rotate the inner and the outer surface point clouds of the actual printed object so that the *Imin* vector was coincident with that of the model object.

The rotation was performed by applying the MatLab functions **vrrotvec** and **vrrotvec2mat** which, when applied in series, produces a rotation matrix equivalent to [51]:

(5)

(6)

(7)

and thus:

(8)

where *R* is the combined rotation matrix and **, ** and ** are angles of rotation about the x-, y- and z-axes respectively. This approach was only used for making surface-based comparisons.

* + 1. **Rotating the volume of the model object**

For purposes of making surface-based comparisons, rotation of either the surface of the printed object’s point cloud into the orientation of the model object’s *Imin* or vice versa is of equal computational effort. However, for making volume-based comparisons, the entire point cloud that identifies the location of all matter within the objects’ volume must be known and stored. Rotation of the printed object’s entire volume point cloud requires numerical interpolation since printed objects by definition will not maintain perfect geometric symmetries. Thus, it is by far computationally more expedient to rotate the model object’s *Imin* into that of the printed objects since the model object, in this case, has a simple analytical description. More detailed information about the creation of the model object and the rotation procedures are provided in the Supplemental Materials.

* + 1. **Addition of blank planes**

To make a correct comparison of the 3D printed object and the model, the height of the two objects, i.e., the number of binarized images or the number of planes, must be the same. But since the printed object may be taller or shorter than the model, blank planes must be added to the top and bottom of the printed object’s image stack. The following equations were used, for example, when the printed object was shorter than the model object, the more common occurrence. Thus, the number of planes and the height of the 3D printed object and the model becomes the same:

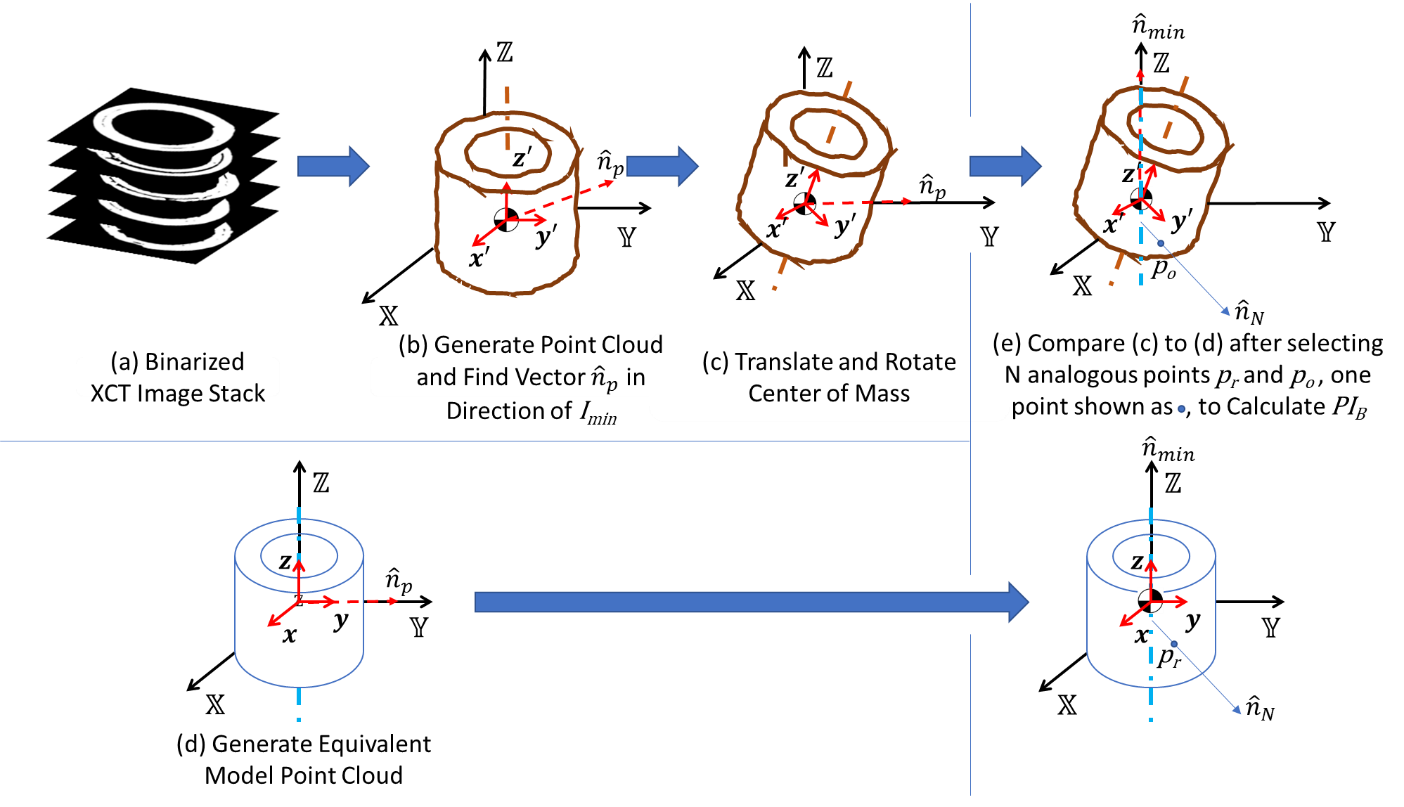
(9)

(10)

where = number of planes in the model object stack, *BP* = the z-index of the bottom plane, generally 1, and = the z-index of the center of mass of the printed object (PO). After these additional top and bottom planes are included, the stack is re-indexed so that the bottom plane is once again number 1.

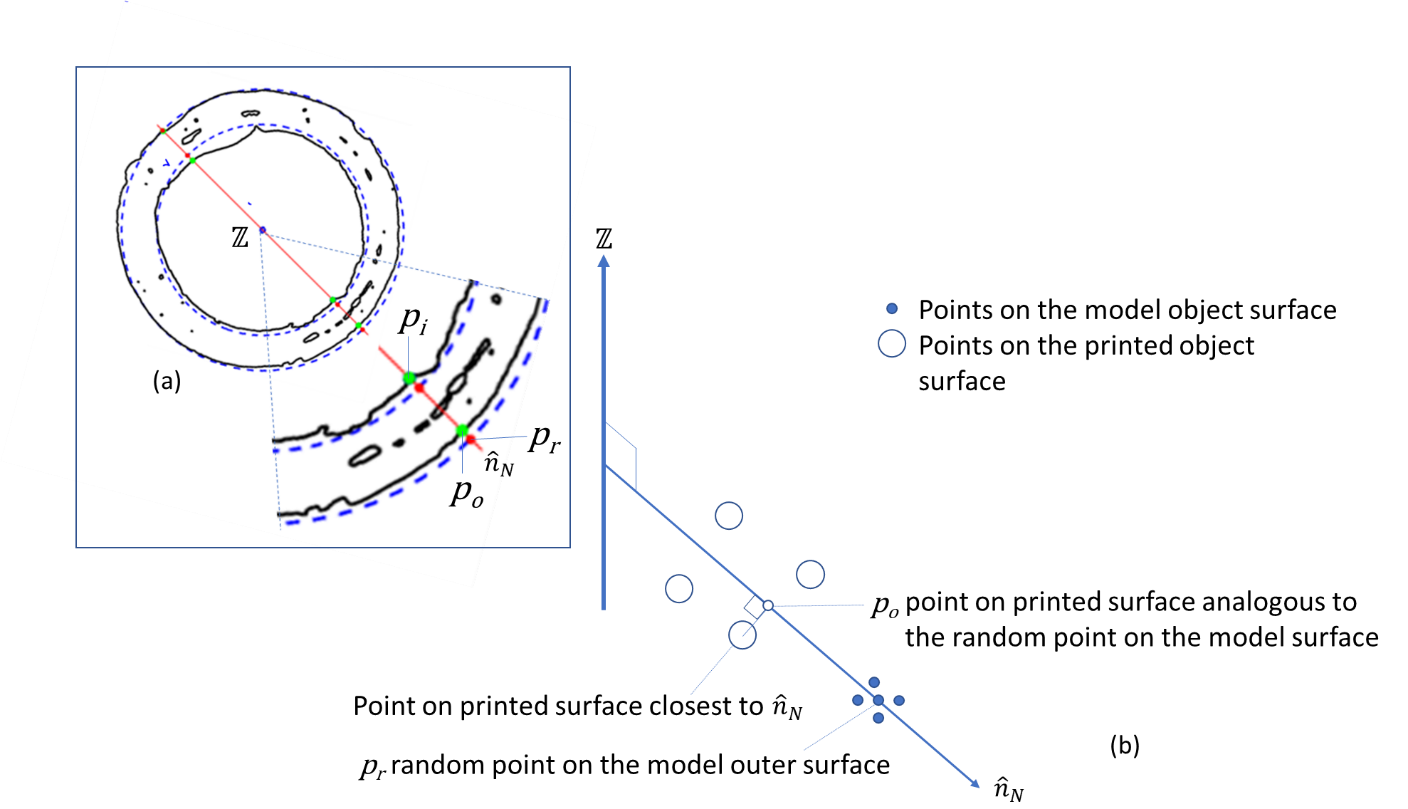
* 1. **Calculation of the boundary-based printability index (*PIB*)**

After obtaining binarized images, the steps presented in Figure 5 were performed to calculate the dimensional accuracy of 3D-printed objects using a boundary-based printability index, i.e., using external features after rotating the printed object surfaces to align the *Imin* with that of the model object.



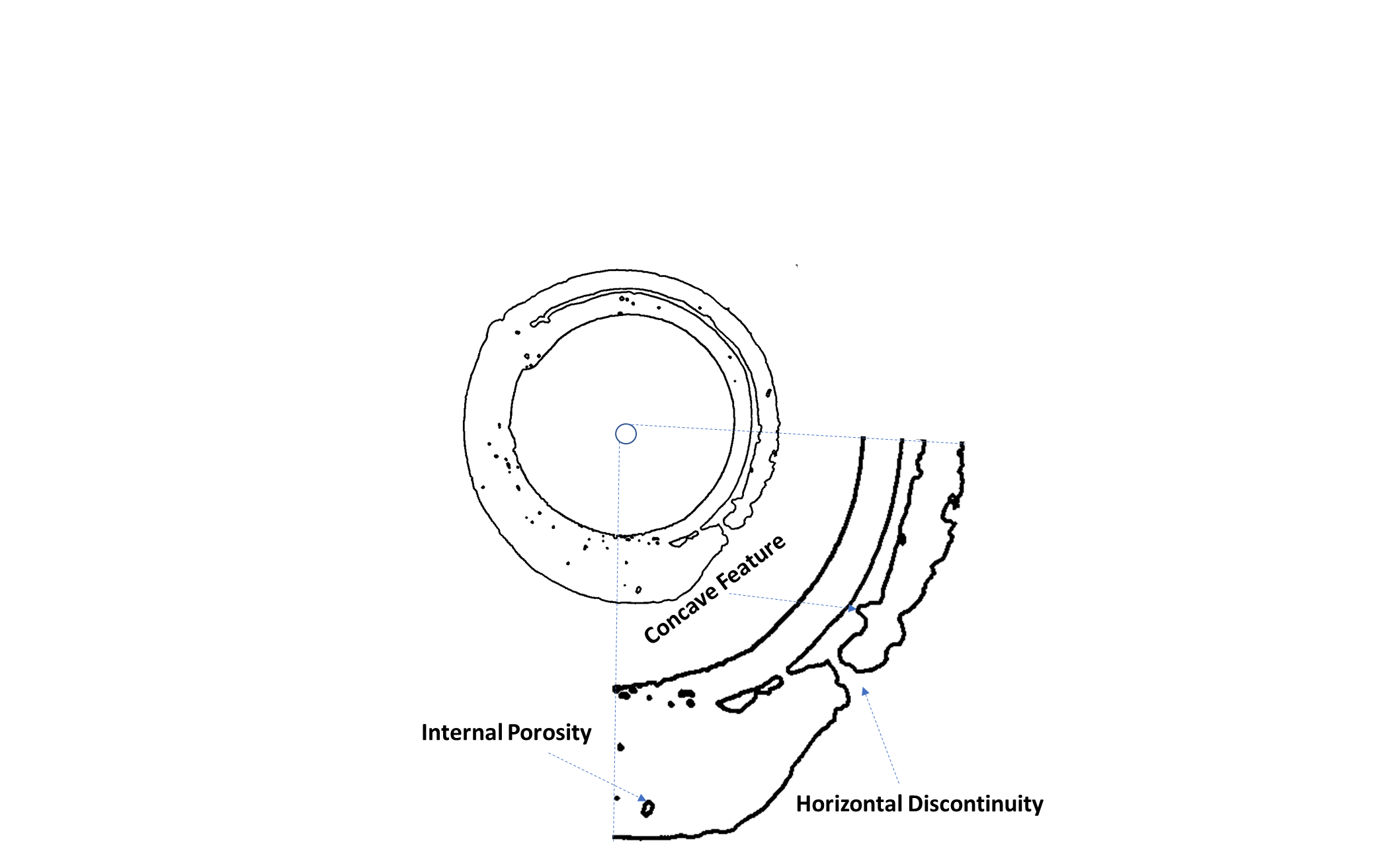
**Figure 5.** Steps in the generation and alignment of point clouds for calculating boundary-based printability metric when the axes of minimum moment of inertia and center of mass are used as the basis for alignment. For purposes of calculating *PIB*, the surface point cloud of the printed object is rotated into the *Imin* of the model object. In panel (e), N analogous points on the surface of the rotated model (c) and printed object (d) are selected, *pr* and *po* respectively, and a point-by-point location comparison made; one such point is shown for illustration. In these images, is the axes of minimum moment of inertia, is one of N lines drawn at random and is the axial direction of the local coordinate system, in this case passes through the COM of both object after alignment and is the axial build direction of the printed object.

The boundary-based printability index (*PIB*) compares the inner-most and outer-most surfaces of a 3D printed object to the model dimensions. Therefore, for any random point on the surface of the benchmark model object, *pr*, the location of a corresponding point on the printed object’s inner- and outer-most surfaces, *pi*and *po*, that lie along the vector passing through the surface point of the model and orthogonal to the axis of *Imin* must be determined. Figure 5e illustrates such points in 3D and Figure 6a illustrates such points in 2D, for simplicity. Figure 6b shows the details of point locating procedure in 3D; both Figures 5 and 6 use a common notation for consistency where is the vector pointing in the direction of axis of *Imin* and is a vector orthogonal to . As shown on Figure 6b, to find such points on the surface of the printed object, the vector orthogonal to the axis of *Imin* passing through a random point on the model surface was found, i.e., , along with the intersection of that vector and the inner and outer surfaces of the printed object, shown as *pi* and *po* respectively on Figures 6a and 6b and as *po* on Figure 5e. A total of N such intersecting points were found on the surface of the actual object, which correspond to 4N points on the surface of the model object since a single line (vector) intersects the printed object in four surface locations as shown on Figure 6a. In practice, such digital surfaces are not continuous but are defined as point clouds, the points of which may not fall on such arbitrarily chosen vectors. Furthermore, rotation to align the centers of mass and the axes of minimum moments of inertia almost guarantees that points on the surface of the actual object will not lie along a vector normal to the axes of minimum moment of inertia and passing through a surface point of the model object’s point cloud. Thus, the intersection points, i.e., *pi* and *po*, on the printed surface boundary were found by calculating the shortest distance between the closest point on the printed surface and the vector . This distance lies along the vector passing through the closest point on the printed surface and normal to the vector *nN* as shown on Figure 6b.



**Figure 6.** In these images, is the axes of minimum moment of inertia, is one of N lines drawn at random generating 4N surface points, two of which are shown, *pi* and *po*, and is the axial direction of the local coordinate system, in this case passes through the COM of both object after alignment and is the axial build direction of the printed object. (a) 2D representation of a single XY-plane through a printed object (the solid boundaries) and model object (the dashed boundaries) after alignment of the COM and after rotating the axis of *Imin* of the print object into that of the model object. The intersecting line, one of N such lines that would be used to compute the boundary-based printability metric, illustrates the points intersecting the inner- and outer-most surface of the printed object. (b) 3D representation showing the location of the interpolated surface point on the printed object *po* corresponding to a point *pr* on the model object.

Moreover, when using a boundary-based method, defining the exterior boundary can be subjective; Figure 7 illustrates a number of characteristic defects encountered when printing including pores, concave features, and discontinuities.



**Figure 7.** This cross-section illustrates various types of defects that challenge the calculation of boundary-based metrics. Shown here are included pores and a type of concave feature that would not be visible by optical methods and a horizontal discontinuity.

Care must be taken to select an outer-most and inner-most surface. The absolute location of the surface is thus subjective i.e. the surface of the outermost boundary or some average of point locations that includes enclosed or concave surfaces could be used. Such included features illustrate the fallibility of this and similar surface-based strategies for characterizing print fidelity. After locating a total of 4N random points on the inner and outer surface of the model and determining 4N analogous points on the inner- and outer-most surfaces of the printed object, a "boundary-based" printability index *PIB* was then calculated that penalizes for deviations of the surface of the actual object from that of the model object. The sum of the squared differences between the position of points on the surface of the printed object and the model object were computed for the 4N points on the inner and outer surface of the model object and the corresponding points on the surfaces of the printed object. The surface deviations at the 4N random points were used to calculate the boundary-based printability metric [8]:

(11)

where: *dA* is the orthogonal distance of a point on the surface of the printed object to the axis of MMI, *dP* is the distance of a point on the surface of the model object to the axis of *Imin*, “*i* ” indicates the inner-most surface, “*o*” indicates the outer-most surface, *Ni*=2N is the number of random points taken on the outer-most surfaces and *No*=2N is the number of points taken on the inner-most surface. It is important also to note that points on the surface of the actual object and the model object, for which *dA* and *dp* are computed, must lie along the same line extending perpendicular to the centroidal axes, i.e., must be co-linear. Thus, a total of N=(*Ni+No*)/4 such lines were drawn that extend through the model and printed object touching the inner- and outer-most surface in two places for a total of four points per line. For a cylindrical object, *dp* is a constant, thus, the co-linearity condition is always met without strict mathematical enforcement [8]. Notably, is a linear indicator of print fidelity, i.e., is based on differences in linear measures and although unitless the unit basis is length/length. This distinction is important when correlating and other printability metrics that have different unit basis as is discussed in Section 2.6.

It should be pointed out that in some cases there are no corresponding printed points, e.g., in cases where the height of the printed object was less than that of the model. In such cases, the squared distance to the axis of *Imin* was added to the summation. Furthermore, in other cases, there were more than two intersection points, i.e., lines might pass through pores. In such situations, only the inner-most and the outer-most points were considered for calculation, i.e., points *pi* and *po* as shown on Figure 6a. Thus, this technique, which uses the inner- and the outer-most surface of the object, clearly neglects interior or otherwise inaccessible concave defects, i.e., pores. Considering only the innermost and the outermost surface points is equivalent to scanning the surface with a laser scanner that ignores internal structures. Therefore, this metric does not account for pore structure and some forms of concave defects such as the large cavity labeled on Figure 7. In order to account for such defects in printed objects and to penalize for them, a more direct measure of internal deviations is needed, which requires direct characterization and comparison of volumes.

* 1. **Calculation of the volume-based printability index (*PIV* )**

The volume-based printability index (*PIV*) enables internal flaws such as pores to be considered in the shape comparison. The calculation of *PIV*, however, is a more complex problem than the boundary-based index. Again, the center of mass of both the 3D-printed object and the model object was calculated and their point clouds translated to [0, 0, 0] and the unit vector of the axis of the minimum moment of inertia of the printed object and the model determined, as described in Sections 2.4.2 to 2.4.5. In this case, rather than rotating the point cloud of the printed object, the point cloud of the CAD model is rotated as described in Section 2.4.7. Since the model has an analytically defined geometry (hollow cylinder), the model object can be rotated analytically and efficiently as shown in Figure 8. However, rotation of the printed object involves moving every point in the point cloud and using interpolation to maintain defined image planes. Figure 8 also shows the steps used to align the model and printed objects for purposes of calculating the volume-based printability index.

The volume-based printability metric used here is defined as [8]:

(12)

where produces a Boolean outcome of 1 for true and 0 for false, thus constraining the value of *PIV*to be 0 ≤ *PIV*  ≤ 1, where

,

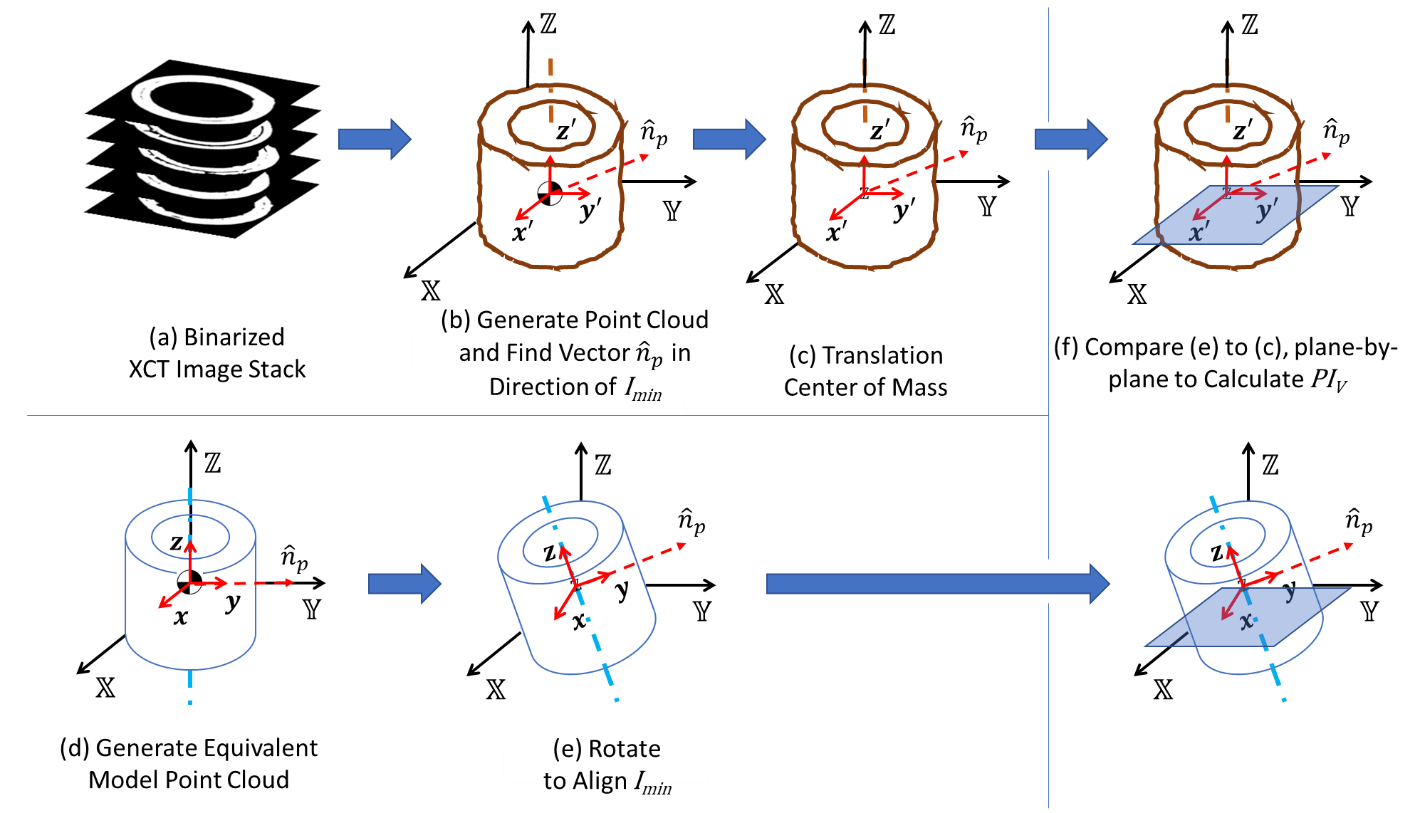
,

, “*p”* here indicates the theoretical “model” object.

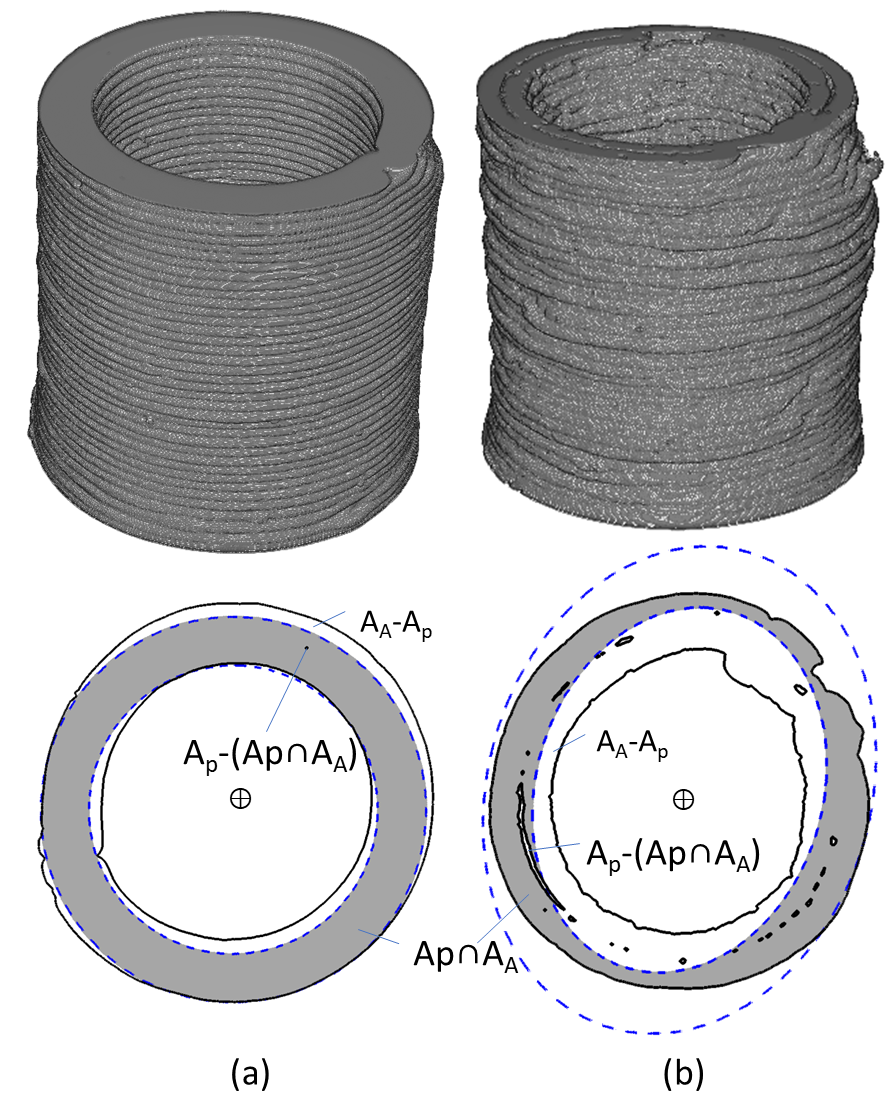
, “*A*” here indicates the “actual” or printed object.

Note also that volumes are calculated by integrating point cloud areas, i.e. where *p*=plane index in point cloud, *P*=number of planes in point cloud and the *Ai* are 2D differential-quantities. Figure 9 illustrates how similarities and differences between a slice of a printed object and a slice of a model object were obtained and how the *PIV* was calculated illustrating the relevant Boolean differential-quantities calculated including , and *AA – Ap*. Thus, the *PIV* index is defined such that *PIV* is equal to 1 for a perfect object, *PIV* is less than one for an imperfect object, and *PIV* is made to be zero for an object that cannot be printed by virtue of the Boolean operator. Since the unconstrained index, i.e., , produces a theoretical value of , the Boolean constraint was applied. This forces grossly deformed objects that lie far from the benchmark model to also have a printability of zero [8].

To perform *PIV* calculations, the internal structure of each plane (slice) of the printed object was compared to the corresponding plane of the rotated model object. is distinctly different from in that has a unit basis of volume/volume, making it a volume-based fidelity metric. For comparative purposes, can either be redefined as or left in the cubed form, as was done here.

****

**Figure 8**. Steps in the generation and alignment of point clouds for calculating volume-based printability metric when the axes of minimum moment of inertia and center of mass are used as the basis for alignment. For purposes of calculating *PIV*, the volume point cloud of the model object is rotated. In these images, is the axes of minimum moment of inertia, is one of N lines drawn at random and is the axial direction of the local coordinate system, in this case passes through the COM of both object after alignment and is the axial direction of the model object.



**Figure 9.** Overlapping planes for printed object and model object after alignment of centers of mass and axes of *Imin* for: (a) well printed object with *Imin* nearly in plane and (b) poorly printed object with *Imin* significantly out of plane. Solid boundaries are the printed object and dash boundaries are the model after rotation to align the COM and *Imin*.

1. **Results and Discussion**

The 4×4×3 experimental design depicted in Figure 3 produces a total of 48 independent variable conditions at which objects were to be printed [8]. From among the 48 conditions, a total of 31 objects were produced with the balance of conditions being unprintable. From among the objects printed, a wide range of print qualities were generated ranging from visibly well-printed objects to partially printed and highly irregular objects. In the authors’ prior work, qualitative and quantitative print quality metrics were correlated to the three-independent variable [8]. In the prior work, quantitative metrics were based on alignment of the basal plane and axial centroidal moments of inertia. Clear domains of printability were identified as a function of gel polymer content, gel-to-cement ratio and polymer type. The three different hydrogel-forming polymers were used as printing aids. Table 2 summarizes the complete printed sample dataset including location of the centers of mass, the direction vector pointing along the axes of minimum moment of inertia, the angle between the printed and model objects’ minimum moment of inertia, and various printability indexes for each of the printed objects. Table 2 also indicates for which printing conditions no object could be printed, in those cases the printability indexes are zero per the definitions of both *PIB* and *PIV*.

**Table 2.** Various object metrics calculated from X-CT data.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Polymer Type | Concentration  (wt %) | Gel/Cement  Ratio | Center of Mass | Direction Vector  (x, y, z) | Angle  (o) | Print Indexes (unitless) | | | |
| *PIB\** | *PIV\** | *PIB* [7] | *PIV* [7] |
| HEMC# | 0.23 | 1 | NA | NA | NA | 0 | 0 | 0 | 0 |
| HEMC | 0.23 | 2 | NA | NA | NA | 0 | 0 | 0 | 0 |
| HEMC | 0.23 | 4 | NA | NA | NA | 0 | 0 | 0 | 0 |
| HEMC | 0.23 | 8 | NA | NA | NA | 0 | 0 | 0 | 0 |
| HEMC | 0.25 | 1 | NA | NA | NA | 0 | 0 | 0 | 0 |
| HEMC | 0.25 | 2 | (544.8, 478.3, 50.4) | (-0.23, 0.97, -0.05) | -2.9 | 0.13 | 0.01 | 0.13 | 0.00 |
| HEMC | 0.25 | 4 | (530.5, 520.8, 374.5) | (0.25, 0.97, -0.07) | -4.1 | 0.95 | 0.44 | 0.99 | 0.73 |
| HEMC | 0.25 | 8 | (521.5, 524.2, 369.1) | (-0.02, 0.99, -0.08) | -4.3 | 0.95 | 0.42 | 0.99 | 0.77 |
| HEMC | 0.275 | 1 | (518.5, 502.4, 314.8) | (0.68, 0.73, -0.00) | -0.4 | 0.83 | 0.34 | 0.85 | 0.45 |
| HEMC | 0.275 | 2 | (521.9, 515.2, 365.4) | (0.56, 0.78, -0.29) | -16.6 | 0.83 | 0.16 | 0.99 | 0.75 |
| HEMC | 0.275 | 4 | (527.7, 499.1, 368.4) | (0.83, -0.03, 0.56) | 34.2 | 0.60 | 0.05 | 0.99 | 0.73 |
| HEMC | 0.275 | 8 | (521.3, 510.4, 371.3) | (0.80, -0.46, 0.37) | 21.9 | 0.75 | 0.15 | 0.99 | 0.77 |
| HEMC | 0.3 | 1 | (527.4, 508.3, 378.2) | (0.44, 0.55, 0.71) | 45.1 | 0.52 | 0.02 | 0.99 | 0.69 |
| HEMC | 0.3 | 2 | (531.3, 527.0, 363.0) | (0.57, 0.80, 0.15) | 8.5 | 0.92 | 0.32 | 0.99 | 0.76 |
| HEMC | 0.3 | 4 | (520.4, 517.3, 362.8) | (0.97, 0.01, 0.24) | 13.8 | 0.87 | 0.26 | 0.99 | 0.71 |
| HEMC | 0.3 | 8 | (516.9, 511.4, 371.2) | (-0.07, 0.96, -0.25) | -14.9 | 0.86 | 0.15 | 0.99 | 0.77 |
| HPMC | 0.23 | 1 | NA | NA | NA | 0 | 0 | 0 | 0 |
| HPMC | 0.23 | 2 | NA | NA | NA | 0 | 0 | 0 | 0 |
| HPMC | 0.23 | 4 | NA | NA | NA | 0 | 0 | 0 | 0 |
| HPMC | 0.23 | 8 | NA | NA | NA | 0 | 0 | 0 | 0 |
| HPMC | 0.25 | 1 | (494.1, 517.0, 323.5) | (0.97, -0.05, 0.24) | 14.1 | 0.58 | 0.00 | 0.73 | 0 |
| HPMC | 0.25 | 2 | (520.2, 511.3, 229.5) | (0.64, 0.76, 0.00) | 0 | 0.51 | 0.20 | 0.79 | 0.26 |
| HPMC | 0.25 | 4 | (513.5, 510.5, 242.1) | (-0.58, 0.82, -0.00) | -0.1 | 0.64 | 0.23 | 0.87 | 0.47 |
| HPMC | 0.25 | 8 | (523.8, 513.0, 371.7) | (0.84, 0.01, 0.53) | 32.3 | 0.64 | 0.07 | 0.99 | 0.73 |
| HPMC | 0.275 | 1 | (520.6, 507.3, 260.7) | (0.21, 0.96, -0.17) | -9.9 | 0.42 | 0.06 | 0.63 | 0.08 |
| HPMC | 0.275 | 2 | (523.7, 523.9, 361.4) | (-0.18, 0.81, -0.56) | -33.8 | 0.61 | 0.02 | 0.99 | 0.63 |
| HPMC | 0.275 | 4 | (520.5, 519.0, 367.8) | (0.43, 0.40, 0.81) | 54.2 | 0.46 | 0.01 | 0.99 | 0.58 |
| HPMC | 0.275 | 8 | (508.8, 514.8, 365.0) | (0.91, 0.16, 0.39) | 22.9 | 0.78 | 0.14 | 0.99 | 0.74 |
| HPMC | 0.3 | 1 | (514.3, 504.9, 369.8) | (0.36, 0.80, 0.47) | 28.4 | 0.73 | 0.07 | 0.99 | 0.66 |
| HPMC | 0.3 | 2 | (537.0, 489.2, 356.1) | (0.75, -0.60, 0.29) | 16.6 | 0.81 | 0.20 | 0.99 | 0.62 |
| HPMC | 0.3 | 4 | (521.7, 513.4, 368.2) | (0.27, 0.41, 0.87) | 60.4 | 0.43 | 0.01 | 0.99 | 0.56 |
| HPMC | 0.3 | 8 | (526.8, 516.9, 370.4) | (0.43, 0.89, 0.17) | 9.6 | 0.91 | 0.27 | 0.99 | 0.72 |
| PEO | 0.23 | 1 | NA | NA | NA | 0 | 0 | 0 | 0 |
| PEO | 0.23 | 2 | NA | NA | NA | 0 | 0 | 0 | 0 |
| PEO | 0.23 | 4 | NA | NA | NA | 0 | 0 | 0 | 0 |
| PEO | 0.23 | 8 | NA | NA | NA | 0 | 0 | 0 | 0 |
| PEO | 0.25 | 1 | NA | NA | NA | 0 | 0 | 0 | 0 |
| PEO | 0.25 | 2 | NA | NA | NA | 0 | 0 | 0 | 0 |
| PEO | 0.25 | 4 | NA | NA | NA | 0 | 0 | 0 | 0 |
| PEO | 0.25 | 8 | NA | NA | NA | 0 | 0 | 0 | 0 |
| PEO | 0.275 | 1 | (538.3, 514.9, 321.4) | (0.70, 0.71, -0.04) | -2.2 | 0.89 | 0.37 | 0.89 | 0.53 |
| PEO | 0.275 | 2 | (522.0, 500.8, 344.6) | (0.42, 0.81, -0.41) | -24.0 | 0.59 | 0.03 | 0.85 | 0.26 |
| PEO | 0.275 | 4 | (517.5, 525.8, 374.5) | (-0.40, 0.84, 0.36) | 21.0 | 0.76 | 0.04 | 0.99 | 0.71 |
| PEO | 0.275 | 8 | (514.2, 509.5, 363.8) | (0.15, 0.68, 0.72) | 46.2 | 0.52 | 0.02 | 0.98 | 0.52 |
| PEO | 0.3 | 1 | (527.5, 515.3, 368.3) | (0.86, 0.18, 0.48) | 28.6 | 0.70 | 0.07 | 0.99 | 0.70 |
| PEO | 0.3 | 2 | (552.9, 497.6, 367.8) | (-0.58, 0.81, 0.13) | 7.4 | 0.93 | 0.24 | 0.99 | 0.80 |
| PEO | 0.3 | 4 | (527.8, 523.2, 366.4) | (0.64, 0.48, 0.61) | 37.4 | 0.58 | 0.04 | 0.99 | 0.66 |
| PEO | 0.3 | 8 | (525.2,518.2, 370.6) | (-0.20, 0.63, 0.74) | 47.9 | 0.49 | 0.01 | 0.99 | 0.65 |

#Conditions in shaded cells produced no printed object, i.e., were not printable.

\*Based on sampling of 1000 points. One-thousand points was determined to produce a consistent standard deviation, population-to-population when N random points were sampled m times, where m×N a large number, refer to Section 3.2 for a more detailed explanation.

* 1. **Effect of rotation on the results of *PIB***

In their previous work, the authors assessed both boundary-based and volume-based printability indices [8]. However, the centroidal axial moment of inertia was used as the alignment basis and the centroids were not translated to a common zero. Instead, the object bases were kept co-planar. In the present work, the axes of minimum moment of inertia were used as an alternative alignment bases. The use of surface landmarks, for example, raises questions of how to choose analogous landmarks on the model and printed object and on what basis in 3D space are the analogous points to be placed, i.e., aligned. The use of some common and calculable geometrically-based property of the object which globally represents the distribution of matter within the object, e.g. *Imin*, removes such subjective choices. To illustrate the effect of using the minimum moment of inertia strategy, a comparison between values of *PIB* based on the alignment of the centroidal axial moment of inertia [8] and the minimum moment of inertia along with the COM is shown in Figure 10. The value of *PIB* is plotted as a function of the angle between the unit vectors along the minimum moment of inertia of the model and printed object, which is referred to in the figure as the “deviation angle.” When outliers representing objects that did not print completely are removed, there is a strong correlation between the *PIB* and deviation angle when the minimum moment of inertia is used as the alignment basis. As expected, however, there is no correlation when the centroidal axial moments are aligned. Partially printed objects generally occurred for print formulations that were too stiff to flow through the print nozzle. This generally results in a short and in some cases very short object. Such objects can have axes of minimum moment of inertia that are close to being in the xy-plane since low aspect ratio (height/diameter) cylinders have minimum moment of inertia in the xy-plane by definition. In such cases, the deviation angle will be low or even near zero and the *PIB* will also be low. Such points cannot be compared to the progression generated by completely printed objects. By comparison, when the centroidal axial moment of inertia is used as the alignment basis, there is a small, but in most cases insignificant penalty levied for poor distribution of mass, but not enough to provide correlation with the deviation angle. This is generally true because the distance between the axial centroids of the printed objects and the model is not well correlated with the deviation angle. In fact, the dataset used in this illustration exhibits high boundary-based printability indexes when the centroidal axial moment is used as the alignment basis but when the axes of minimum moment of inertia is the alignment basis, printability becomes a strong function of the deviation angle. This implies that fidelity metrics that do not adequately penalize for internal structure can artificially reward for appearance only while ignoring the soundness of internal structure. Furthermore, Figure 10 illustrates that when objects are aligned using the axes of minimum moment of inertia, even a boundary-based index correctly penalizes for internal structure defects, as indicated by the strong correlation between deviation angle and *PIB*.

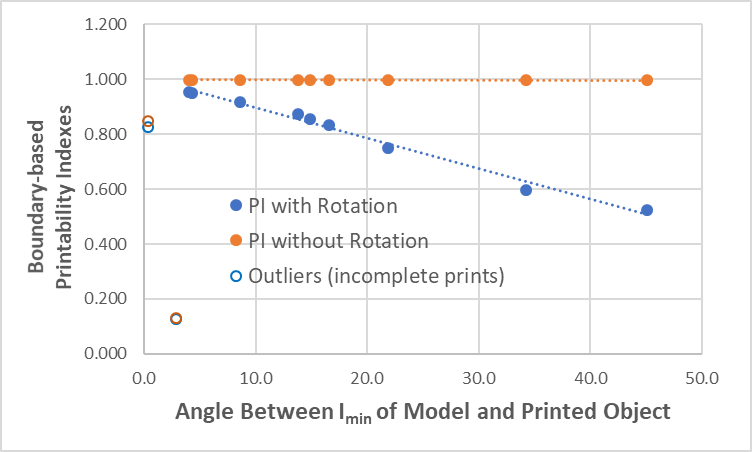
* 1. **Establishing good sampling statistics for using the *PIB***

To establish a statistically representative sample size, N was chosen to be the number of sampling points needed to statistically capture deviations on the surface of the printed object, *PIB* was calculated for N between 5 and 5120 performing m replicates such that the total number of sampling points was equal to 5120 e.g. when N=5, m=5120/5=1054 replicates. Figures 11(I a and b) illustrate how *PIB* changes with N for two different objects. As expected, for low sample sizes, i.e. low values of N, the standard deviation in the observed mean is large but asymptotically approaches a small value as N becomes large. This analysis shows that a surprisingly small number of sampling points can be used to establish a high degree of confidence for this printability index, about 200 points minimum with about 1000 points preferred, with higher numbers of points needed for objects with lower print fidelity. Objects with lower surface (boundary) fidelity exhibit higher standard deviation, as expected and as shown by comparing Figure 11(a, I and b, I). The lower surface fidelity object, i.e., Object 11(a), has much larger surface location variation and requires a larger number of surface points to be sampled to establish an apparent stable level of deviation. This is important since even relatively low-resolution XCT point clouds can contain millions of individual surface points, most of which provide no additional information regarding shape fidelity when measured using *PIB*. *PIB* becomes approximately invariant with N when the number of sampling points is at least 1000 though objects with higher print fidelity show less variability and converge to an asymptotic value of *PIB* at much lower values of N (e.g. 300).

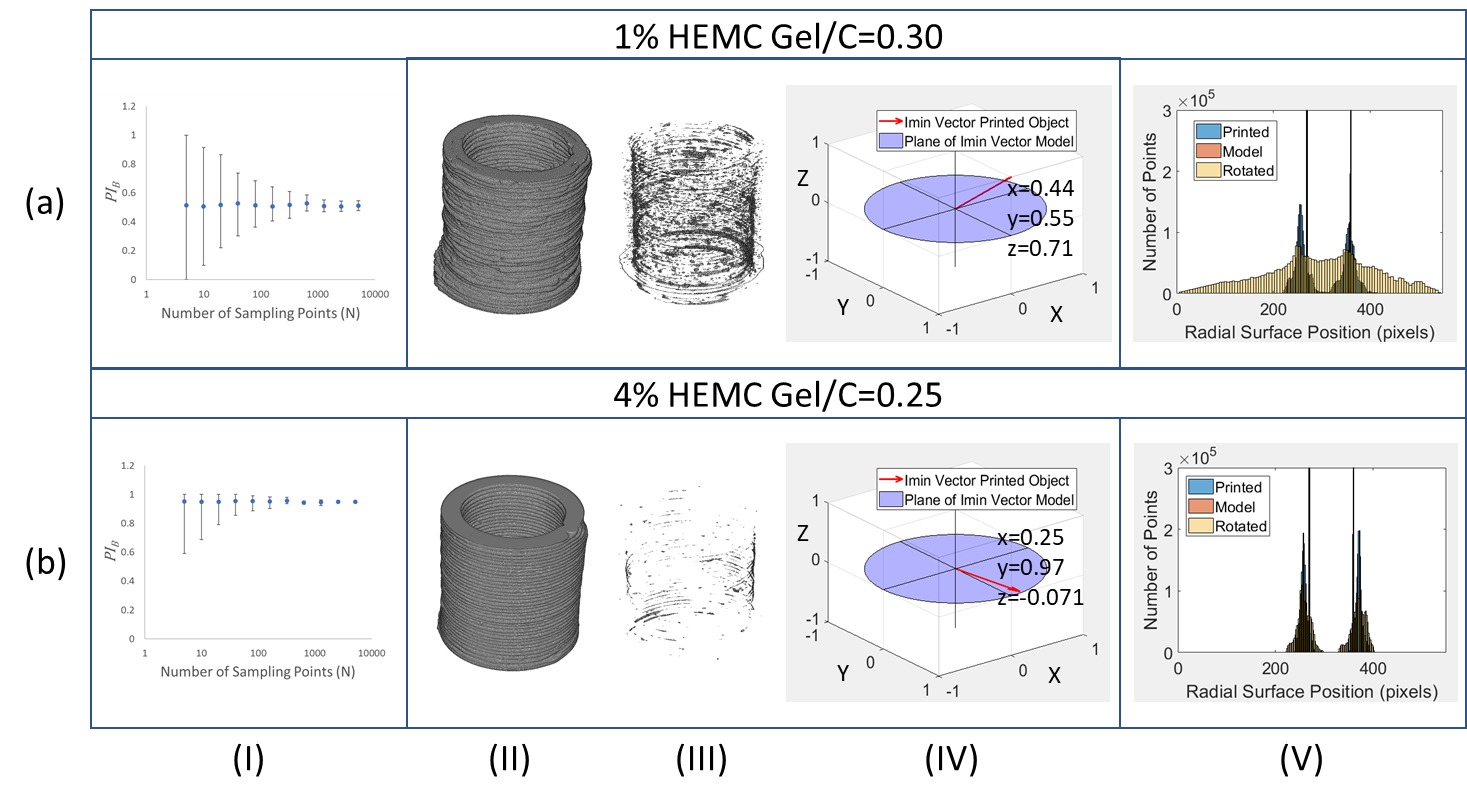
Figure 11 also shows the internal structures of the two printed objects, 11(III), i.e., the pores inside of the printed objects, the direction of the unit vector of minimum moment of inertia, 11(IV), and a histogram of the surface points of the printed object before and after rotation compared to that of the model object, 11(V). Comparing Figures 11(a) and 11(b) shows that for these two objects the visibly more perfect object, compare 11(b, II) to 11(a, II), had fewer internal flaws, compare 11(b, III) to 11(a, III). The visibly more perfect object also has a higher quantitative measure of shape fidelity, i.e., the *PIB* for the visibly more appealing object is >0.9 whereas for the less appealing object it is ~0.5. Such relationships, however, are not necessary universally true. Furthermore, the object with greater internal defects, and in this case the less visibly appealing object, has an *Imin* vector that is significantly out of plane, 11(a, IV). Conversely, the visibly more appealing object has an *Imin* vector that is nearly in plane, 11(b, IV). These differences in orientation of the vector of *Imin* lead to vastly different surface location histograms, 11(a, V) and 11(b, V). Such surface location histograms might be used as a quick semi-quantitative visual indicator of print fidelity. When surface location histograms are compared for objects aligned with and without the use of the axes of minimum moment of inertia, the impact of internal defects is quickly realized. The surface location histogram in Figure 11 for the model object consists of a bimodal distribution of two Kronecker delta impulses, one impulse at the inner and one at the outer radius of the model. As expected, printed objects have a distribution of surface locations falling around the Kronecker impulses of the model object, the breadth of which is a graphical indicator of print surface fidelity. Two comparisons are made on Figures 11(a, V) and 11(b, V): (1) the surface locations for the unrotated printed objects and (2) the surface locations for the rotated printed objects. A sharp distinction is noted when the rotated and unrotated histograms are compared. Objects that have poorly aligned *Imin* exhibit broadening of the surface location histogram as illustrated by Figure 11(a, V) whereas objects with well aligned *Imin* exhibit minimal broadening. This simple form of graphical analysis can be used to quickly screen print fidelity.

* 1. **Example application – using *Imin* -based printability indexes to select printing aids.**

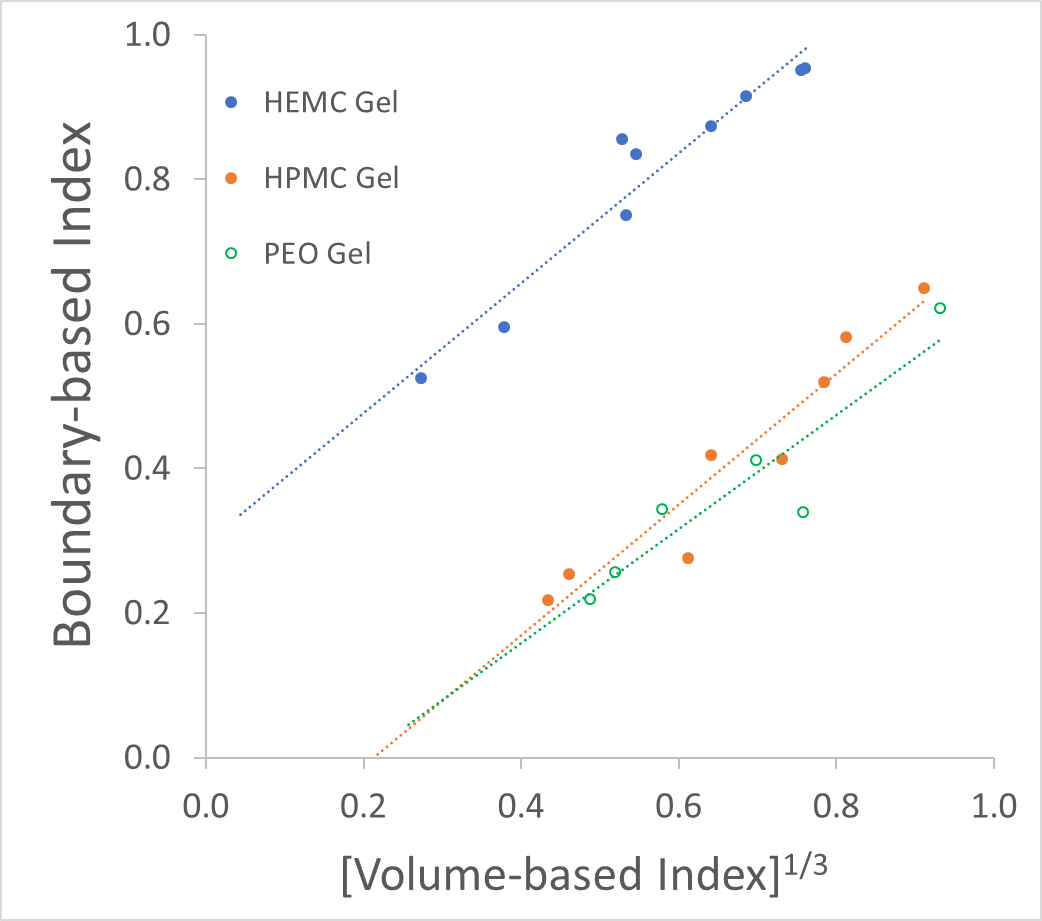
The collection of printed objects based on the 4×4×3 full-factorial experimental design is used here as an illustrative example of how the proposed *Imin* -based printability indexes could be applied to discriminate between different print process treatments. Figure 12 illustrates the relationship between *PIB* and *PIV*. As expected, *PIB* and are highly correlated since *PIB* is effectively a linear measure of object fidelity and a cubic measure of object fidelity. Again, this correlation implies that when surface deviations are computed using the axes of minimum moment of inertia as the basis for alignment, changes in external features caused by internal deviations are the only way that internal deviations can be accounted for. The empirical correlations shown in Figure 12 illustrates a strong correlation between *PIB* and *PIV*, as expected. Figure 12 also illustrates a convenient way to compare print paste performance in terms of the relationship between surface- and volume-based printability measures. A clear distinction between the internal and external fidelity is seen for the three printing aids where HEMC clearly produces higher surface fidelity at same volume fidelity when compared to HMPC and PEO. Such plots can enable distinctions that might otherwise be difficult to discern by eye.



**Figure 10.** Correlation between the angle between the axes of minimum moment of inertia of the printed and model objects, i.e. the deviation angle, and boundary-based printability index with and without rotation. Outliers were not included in the correlation lines and represent incompletely printed objects.

****

**Figure 11**. Comparison of sampling statistics for two printed samples: (a) poor print with *Imin* well out of plane and (b) good print with *Imin* in plane. Panel (I): Variation of printability index *PIB* as a function of number of points sampled (N). The error bars are mean standard deviations in observed *PIB* when N points are sampled. Panel (II): Virtual point cloud renderings from XCT data. Panel (III): Internal defect structure from XCT data. Panel (IV): Direction of vector of minimum moment of inertia. Panel (V): Surface location histograms shown for the printed object, model object and rotated-printed object.



**Figure 12**. Correlation between *PIB* and *PIV* illustrating that the boundary-based index penalizes for internal defects when the axes of *Imin* is used as the alignment basis. Shown here also are datasets for three different printing aids quantitatively confirming differences in performance.

1. **Summary and Conclusions**

Assessing the fidelity of 3D printed objects is a complex challenge. Most techniques utilize some form of subjective strategy such as alignment of surface landmarks. Furthermore, surface-only methods reward “beauty” at the expense of structural fidelity. To address these deficiencies and to remove the subjectivity associated with the use of surface landmarks as alignment references; this study instead used the axes of minimum moment of inertia and the centers of mass (COM) as the alignment basis. The axis of minimum moment of inertia is a geometric property of the object that embodies the distribution of matter within the object volume and thus it provides an inherently unbiased spatial reference. This strategy imposes a printability penalty for printing objects with internal defects, despite surface compliance. When applied, boundary-based, i.e., external shape-based, metrics such as the *PIB* described above, suitably penalize for internal flaws. Furthermore, when comparing the shape of objects, included voids and concave features make the location of surface boundaries ambiguous. To avoid this ambiguity, this study utilized the outer-most and inner-most surface points. The moment-based technique, however, requires rotation to align objects. When alignment requires rotation of either the model or printed object point cloud, it may be computationally more expedient to rotate the model object if the model object can be analytically defined. When objects are aligned using the axes of minimum moment of inertia, even a boundary-based index correctly penalizes for internal structural defects. A small number of sampling points can be used to establish a high degree of confidence in such metrics when comparing shape fidelity though objects with lower surface, i.e., boundary, fidelity exhibit higher standard deviation, thus a larger number of surface points are required to establish an apparent stable level of variation. By correlating quantitative measures of shape and volume fidelity, those with higher surface fidelity were found to have quantitatively fewer internal flaws, i.e. higher volume fidelity, though this is not expected to be a generality. Surface location histograms can be used as a quick semi-quantitative visual indicator of print fidelity. A sharp distinction between such histograms for a rotated and not rotated object are seen for objects with high deviation angle, i.e. when the angle between the printed object and the model object axis of minimum moment of inertia is large. A potential drawback of using any moment-based alignment technique is the need to have internal structural information. When used, however, a volume-based metric, *PIV*, will have strong correlation with a boundary-based metric, *PIB*, and mappings of *PIB* verses *PIV* provides clear distinctions between the fidelity of different printed objects, e.g., when using different printing aids.

Existing techniques that utilize surface landmarks, basal plane alignment or similar methods, can misrepresent the distribution of matter even when volume-based approaches are utilized. Surface landmarks, basal plane and similar features are not properties of the object but rather subjective choices for convenient ways to align objects for comparison. The new print fidelity technique, which utilizes XCT data to render a virtual image and to make measurements, combines alignment using the axes of minimum moment of inertia and the centroid to fill the gaps in existing techniques, as illustrated in Figure 2. Since the moment of inertia and centroid are inherently dependent on the distribution of matter, objects with perfect surface fidelity are penalized for internal defects without the need for utilizing more than one metric, i.e. a surface-based metric and a volume-based metric.

**Acknowledgements**

The authors acknowledge financial support from the National Science Foundation (NSF) though grant number CMMI-1563173 and from the Tennessee Technological University (TTU) Center for Manufacturing Research (CMR).The authors also acknowledge the contribution of printed rocker arm objects from the laboratory of Dr. Ismail Fidan at TTU as reported on in the Supplemental Materials.

1. **References**
2. Duda T, Raghavan L V, 3D Metal Printing Technology, IFAC-PapersOnLine 2016; 49:103-110, <https://doi.org/10.1016/j.ifacol.2016.11.111>.
3. Alexandru T, Andrei J.Voxel-Based Assessment of Printability of 3D Shapes in Mathematical Morphology and Its Applications to Image and Signal Processing - 10th International Symposium, ISMM 2011, Verbania-Intra, Italy, 2011.
4. Nutma M. A quick guide to dimensional accuracy for 3D printing. 2019. <https://www.3dprintingmedia.network/quick-guide-dimensional-accuracy-3d-printing/>.
5. Melenka GW, Schofield JS, Dawson MR, et al. Evaluation of dimensional accuracy and material properties of the MakerBot 3D desktop printer. Rapid Prototyp J 2015;21:618 –627. https://doi:[10.1108/RPJ-09-2013-0093](https://doi.org/10.1108/RPJ-09-2013-0093).
6. Tiwari K, Kumar S. Analysis of the factors affecting the dimensional accuracy of 3D printed products. Mater Today: Proceedings 2018;5: 18674-18680. <https://doi.org/10.1016/j.matpr.2018.06.213>.
7. Huang CY. Extrusion-based 3D Printing and Characterization of Edible Materials. Waterloo, Ontario, Canada, 2018.
8. Ketel S, Falzone G, Wang B, et al. A printability index for linking slurry rheology to the geometrical attributes of 3D-printed components. Cement Concrete Comp 2019;101:32-43. <https://doi.org/10.1016/j.cemconcomp.2018.03.022>.
9. Taheri Afarani H, Carroll W, Garboczi E J, et al. Designing 3D printable cementitious materials with gel-forming polymers. Constr Build Mater 2021;268:121709. <https://doi.org/10.1016/j.conbuildmat.2020.121709>.
10. Lille M, Nurmela A, Nordlund E, et al. Applicability of protein and fiber-rich food materials in extrusion-based 3D printing. J Food Eng 2018;220:20-27. <https://doi.org/10.1016/j.jfoodeng.2017.04.034>.
11. Abdollahi S, Davis A, Miller JH, et al. Expert-guided optimization for 3D printing of soft and liquid materials. PLoS ONE 2018;13:e0194890. <https://doi.org/10.1371/journal.pone.0194890>.
12. Skardal A, Devarasetty M, Kang HW, et al. A hydrogel bioink toolkit for mimicking native tissue biochemical and mechanical properties in bioprinted tissue constructs, Acta Biomater 2015;25:24-34. https://doi: 10.1016/j.actbio.2015.07.030.
13. Stichler S, Jungst T, Schamel M, et al. Thiol-ene Clickable Poly(glycidol) Hydrogels for Biofabrication. Ann Biomed Eng 2017;45:73–285. <https://doi.org/10.1007/s10439-016-1633-3>.
14. Kim YB, Lee H, Yang GH, et al. Mechanically reinforced cell-laden scaffolds formed using alginate-based bioink printed onto the surface of a PCL/alginate mesh structure for regeneration of hard tissue. J Colloid Interface Sci 2016;461:359-368. <https://doi.org/10.1016/j.jcis.2015.09.044>.
15. Lee HJ, Kim YB, Ahn SH,et al. A New Approach for Fabricating Collagen/ECM-Based Bioinks Using Preosteoblasts and Human Adipose Stem Cells. Adv Healthcare Mater 2015;4:1359–1368. https://doi: 10.1002/adhm.201500193.
16. Zhao Y, Li Y, Mao S, et al. The influence of printing parameters on cell survival rate and printability in microextrusion-based 3D cell printing technology. Biofabrication 2015;7:045002. https://doi: 10.1088/1758-5090/7/4/045002.
17. Li VCF, Dunn CK, Zhang Z, et al. Direct Ink Write (DIW) 3D Printed Cellulose Nanocrystal Aerogel Structures. Sci Rep 2017;7:8018.<https://doi.org/10.1038/s41598-017-07771-y>.
18. Habib MA, Khoda B. Development of clay based novel hybrid bio-ink for 3D bio-printing process. J Manuf Process 2019;38:76–87. <https://doi.org/10.1016/j.jmapro.2018.12.034>.
19. Ribeiro A, Blokzijl MM, Levato R, et al. Assessing bioink shape fidelity to aid material development in 3D bioprinting. Biofabrication 2017;10:014102. https://doi: 10.1088/1758-5090/aa90e2.
20. Hao L, Mellor S, Seaman O, et al. Material characterisation and process development for chocolate additive layer manufacturing. Virtual Phys Prototyp 2010;5:57-64. <https://doi.org/10.1080/17452751003753212>.
21. Huan S, Ajdary R, Bai L, et al. Low solids emulsion gels based on nanocellulose for 3D-printing. Biomacromolecules 2019;20:635-644.

<https://doi.org/10.1021/acs.biomac.8b01224>

1. Xin S, Chimene D, Garza JE, et al. Clickable PEG Hydrogel Microspheres as Building Blocks for 3D Bioprinting. Biomater Sci 2019;7:1179-1187.<https://doi.org/10.1039/C8BM01286E>
2. Kitsakis K, Moza Z, Iakovakis V, et al. An investigation of dimensional accuracy of Multi-Jet Modeling parts, in Proceedings of the International Conference Applied Mathematics, Computational Science & Engineering (AMCSE 2015). Agios Nikolaos, Crete, Greece, 2015.
3. Alsoufi MS, Elsayed AE. Surface Roughness Quality and Dimensional Accuracy—A Comprehensive Analysis of 100% Infill Printed Parts Fabricated by a Personal/Desktop Cost-Effective FDM 3D Printer. Mater Sci Appl 2018;9:11-40. https://doi:10.4236/msa.2018.91002.
4. Asadi-Eydivand M, Solati-Hashjin M, Farzad A, et al. Effect of technical parameters on porous structure and strength of 3D printed calcium sulfate prototypes. Robot Comput Integr Manuf 2016;37:57-67. <https://doi.org/10.1016/j.rcim.2015.06.005>
5. Gillispie G, Prim P, Copus J, et al. Assessment Methodologies for Extrusion-Based Bioink Printability. Biofabrication 2020;12:022003. DOI: 10.1088/1758-5090/ab6f0d.
6. Law N, Doney B, Glover H, et al. Characterisation of Hyaluronic Acid Methylcellulose Hydrogels for 3D Bioprinting. J Mech Behav Biomed Mater 2018;77:389-399. doi: 10.1016/j.jmbbm.2017.09.031
7. Murphy SV, Skardal A, Atala A. Evaluation of hydrogels for bio-printing applications. J Biomed Mater Res A 2013;101:272-84. DOI: 10.1002/jbm.a.34326
8. Duan B, Kapetanovic E, Hockaday LA, et al.3D Printed Trileaflet Valve Conduits Using Biological Hydrogels and Human Valve Interstitial Cells. Acta Biomater 2014;10:1836–1846. doi: 10.1016/j.actbio.2013.12.005.
9. Ahn G, Min KH, Kim C, et al. Precise stacking of decellularized extracellular matrix based 3D cell-laden constructs by a 3D cell printing system equipped with heating modules. Sci Rep 2017;7: 8624. doi: 10.1038/s41598-017-09201-5.
10. Ouyang L, Yao R, Zhao Y, et al. Effect of bioink properties on printability and cell viability for 3D bioplotting of embryonic stem cells. Biofabrication 2016;8:035020. doi: 10.1088/1758-5090/8/3/035020.
11. Klar V, Pere J, Turpeinen T, et al. Shape fidelity and structure of 3D printed high consistency nanocellulose. Sci Rep 2019;9:3822. <https://doi.org/10.1038/s41598-019-40469-x>
12. Hockaday LA, Kang KH, Colangelo NW, et al. Rapid 3D printing of anatomically accurate and mechanically heterogeneous aortic valve hydrogel scaffolds. Biofabrication 2012;4: 035005. doi: 10.1088/1758-5082/4/3/035005.
13. Markl D, Zeitler JA, Rasch C, et al. Analysis of 3D Prints by X-ray Computed Microtomography and Terahertz Pulsed Imaging. Pharm Res 2017;34:1037–1052. <https://doi.org/10.1007/s11095-016-2083-1>
14. George E, Liacouras P, Rybicki FJ, et al. Measuring and Establishing the Accuracy and Reproducibility of 3D Printed Medical Models. RadioGraphics 2017; 37:1424–1450. doi: 10.1148/rg.2017160165.
15. Foteinopoulos P, Esnault V, Komineas G, Papacharalampopoulos A, & Stavropoulos P. Cement-based additive manufacturing: experimental investigation of process quality. Int J Adv Manuf Technol 2020;106(11): 4815-4826.
16. Lim S, Buswell R A, Le T T, Austin S A, Gibb A G, & Thorpe T. Developments in construction-scale additive manufacturing processes. Autom Constr 2012; 21: 262-268.
17. Geomagics Qualify (v9.0; Research Triangle, NC)/ <https://www.3dsystems.com/press-releases/geomagic/qualify-enables-graphical-comparisons-between-cad-maste>.
18. Armenakis C, Gao Y, Sohn G. Co-registration of aerial photogrammetric and LiDAR point clouds in urban environments using automatic plane correspondence. Appl Geomat 2013;5 :155–166. <https://doi.org/10.1007/s12518-013-0105-9>.
19. Stavropoulos P, Foteinopoulos P, & Papapacharalampopoulos A. On the Impact of Additive Manufacturing Processes Complexity on Modelling. Appl Sci 2021; 11(16):7743.
20. Kastner J, Heinzl C. X-ray Computed Tomography for Non-destructive Testing and Materials Characterization, In: Integrated Imaging and Vision Techniques for Industrial Inspection. Advances in Computer Vision and Pattern Recognition. London, Springer, 2015, pp 227-250. <https://doi.org/10.1007/978-1-4471-6741-9_8>
21. Cai X, Malcolm AA, Wong BS, et al. Measurement and characterization of porosity in aluminium selective laser melting parts using Xray CT. Virtual Phys Prototyp2015;195-206. <http://dx.doi.org/10.1080/17452759.2015.1112412>
22. Alginahi Y. Preprocessing Techniques in Character Recognition. In: Character Recognition. Croatia, Janeza Trdine 9, 51000 Rijeka, 2010,pp.1-19.
23. Shapiro LG, Stockman GC. Computer Vision. New Jersey, Prentice-Hall, 2001, pp 279-325,ISGN 0-13-030796-3.
24. Serwata K. Simply Segmentation Technique for Computed Tomography Images, CUER-WS Proceedings, 2018, 2147. http://ceur-ws.org/Vol-2147/p08.pdf.
25. Miah MBA and Yousuf MA. Detection of Lung Cancer from CT Image Using Image Processing and Neural Network, 2nd Int'l Conf on Electrical Engineering and Information & Communication Technology (ICEEICT), 2015. DOI: 10.1109/ICEEICT.2015.7307530.
26. Rogowska J. Overview and Fundamentals of Medical Image Segmentation. In: Handbook of Medical Image Processing and Analysis (Second Edition), Academic Press, 2009, pp. 81-98.
27. Decker [N](https://www.sciencedirect.com/science/article/abs/pii/S0278612520300455?via%3Dihub#!), Wang [Y](https://www.sciencedirect.com/science/article/abs/pii/S0278612520300455?via%3Dihub#!), Huang [Q.](https://www.sciencedirect.com/science/article/abs/pii/S0278612520300455?via%3Dihub#!) Efficiently registering scan point clouds of 3D printed parts for shape accuracy assessment and modeling. J Manuf Syst 2020; [56](https://www.sciencedirect.com/science/journal/02786125/56/supp/C):587-597. <https://doi.org/10.1016/j.jmsy.2020.04.001>.
28. Wu L, Guo X, Zhao L. A Quality Evaluation Scheme to 3D Printing Objects Using Stereovision Measurement. In: Image and Graphics. ICIG 2017. Lecture Notes in Computer Science, Springer, Cham, 2017, 10668,pp 458-467. <https://doi.org/10.1007/978-3-319-71598-8_41>
29. Beer FP, Johnston ER, Cornwell PJ, et al. Vector Mechanics for Engineers: Dynamics, ninth ed., McGraw-Hill, Boston, 2010.
30. Uicker JJ, Pennock GR, Shigley JE. Theory of Machines and Mechanisms fourth ed., Oxford University Press, New York, 2010.
31. Roty/Help Center/MathWorks/ <https://www.mathworks.com/help/phased/ref/roty.html>.

**Supplemental Materials**

The following supplemental materials are provided as a resource explaining some of the more intricate calculations involving the rotation and alignment of virtual objects. In particular, details are given here for the rotation of an analytical hollow cylinder. In the following discourse, the global coordinate system is fixed and is defined by whereas the coordinate system of the object to be rotated is defined by ().

**S.1. Calculate of the Intersection between a Plane and a Hollow Cylinder**

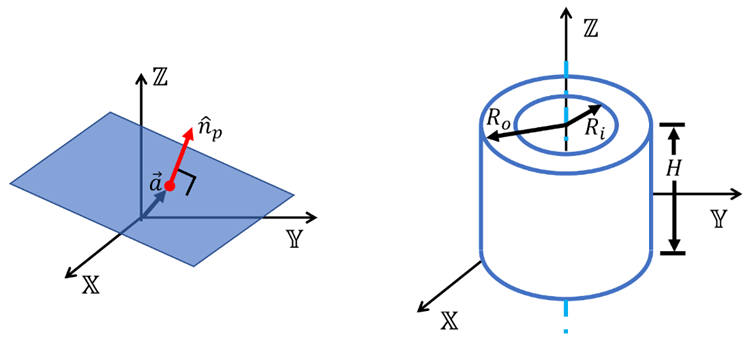
The first challenge is to analytically calculate the intersection contour between an arbitrary plane and a hollow cylinder with finite height. The necessary parameters required to define the plane and the hollow cylinder are given in Figure S.1. Since the hollow cylinder is a solid object, this intersection would result in an area. However, only the intersection between the plane and the outer and inner surfaces are necessary – both of these intersections result in a closed contour line. Since this intersection-operation is effectively the same for both the inner and outer surfaces, the following discussion will focus on just the intersection of a plane and outer cylindrical surface (or shell).

Mathematically speaking, a cylindrical surface aligned with its axis of rotation along the -axis can be written as:

|  |  |  |
| --- | --- | --- |
|  |  | s1 |

Recall that the benchmark cylinder is instantiated with its local coordinate system coincident with the global coordinate system, so the same cylindrical surface can also be easily defined in its local coordinates:

|  |  |  |
| --- | --- | --- |
|  |  | s2 |



**Figure S.1.** A plane can be defined with a coordinate of a point that lies on the plane – defined by position vector – and unit normal vector. The benchmark hollow cylinder is defined by its height , inner radius , and outer radius . The origin of the hollow cylinder is taken be at the benchmark centroid, which is at height and radius .

However, for brevity, the following discussion will proceed using the global coordinate system notation. To simplify the math, it will be convenient to parametrize the equations of the cylinder. In loose terms, the parametrization represents a mapping between the () Cartesian coordinate system and simpler () coordinate system. This mapping can be readily performed using trigonometric functions:

|  |  |  |
| --- | --- | --- |
|  |  | s3 |

Now, the cylinder is defined in the global coordinate system based on just two variable, . Note, this mapping is valid so long as the following bounds are used: , and . Moreover, for now, the cylinder is assumed to be infinitely long in order to guarantee that an intersection exists between the arbitrary plane and cylinder. Later, as a secondary correction, any intersection points that contain -coordinates greater than , or less than , will be thrown out in order to accommodate a hollow cylinder with finite height as depicted in Figure S.1.

Next, an arbitrary (infinite) plane can be defined based on a position vector and a unit normal vector:

|  |  |  |
| --- | --- | --- |
|  |  | s4 |

where is the unit normal vector, is the point on the plane defined by a position vector, and are the Cartesian coordinate variables, . The dot notation represents the vector dot product. Assuming that and , then Equation s4 can be expanded:

|  |  |  |
| --- | --- | --- |
|  |  | s5 |

In order to calculate the intersection contour between the plane and the cylinder, we essentially need to set Equations s3 and s4 equal to one another. Since the cylinder is aligned in the -direction, the definition of the cylinder is conveniently uncoupled in terms of the -variable. Thus, it is easiest to solve for in the equation of a plane, and then substitute this into the parametrized equation of a cylinder. Solving for in Equation s5 yields:

|  |  |  |
| --- | --- | --- |
|  |  | s6 |

and substituting this into the equation of a cylinder results:

|  |  |  |
| --- | --- | --- |
|  |  | s7 |

Note, the parametrized equations for and were substituted into in order to arrive at an analytical formulation dependent on solely and .

Now, using Equation s7, a discrete set of values from 0 to can be substituted for resulting in a series of coordinates representing the intersection between the specified plane and cylindrical surface. In order to account for a finite height of the cylinder, points that have a -coordinate below or above , were excluded from the solution.

**S.2. Rotating the Analytical Cylinder into the Printed Object’s Coordinate System**

The analytically calculated intersections between the plane and the cylinder must now be rotated such that the analytical cylinder’s local coordinate system () is coincident and aligned with the physical part’s (i.e., the printed object’s) local coordinate system (). It is assumed that the analytical cylinder, and its corresponding intersection points, can be treated as a rigid body. In which case, a linear transformation can be derived and applied to all of the calculated intersection points. Recall that the local coordinate systems of the analytical cylinder and the physical cylinder were defined with respect to some global coordinate system (), as shown in Figures 5 and 8.

Referring to Figure 8, the transformation can be comprised of two rotations followed by translation. In theory, only one rotation need be used, but it is easier to formulate the math when this step is broken down into two separate rotations. Moreover, it is far more convenient to perform the rigid-body rotations while the (coordinate system is positioned at the global origin.Let us define this multi-step transformation operation as a “forward” transformation:

|  |  |  |
| --- | --- | --- |
|  |  | s8 |

where is some position vector of a point originally defined in the () coordinate system, and after the forward transformation , a new point is defined based on the () coordinate system. Using a similar thought process, a backward transformation can also be defined:

|  |  |  |
| --- | --- | --- |
|  |  | s9 |

The simplest transformation step in Figure 8 is the translation step. A translation transformation is a straightforward operation that just involves vector addition. If you have point that you wish to translate along the direction defined by unit vector by a distance , then the following vector addition can be used:

|  |  |  |
| --- | --- | --- |
|  |  | s10 |

Next, focus on aligning the -axis with the -axis, which is illustrated as the step between Figure 8c and Figure 8d. To make this step more concrete, the z-axis directions are written in the local coordinate systems in component form:

|  |  |  |
| --- | --- | --- |
|  |  | s11 |

where the vector magnitudes of and are both equal to one. It may look strange to see a z-axis of a coordinate system with components in all three dimensions. However, recall that the -axis and -axis are local coordinate systems that are defined with respect to the global coordinate system. So, Equation s11 expands these unit vectors into components of .

There are a number of methodologies to apply a rotation to a point or rigid body. The most common methods are the use of orthogonal rotation matrices, Euler axis-angle representation, Euler rotations, and quaternions. It is left to the reader to review these topics as needed, however, the method that will be used in this work is the Euler axis-angle representation, which utilizes the Rodrigues' rotation formula to implement the rotation [s1].

The Euler axis-angle representation is a rotation that parameterizes the operation in a three-dimensional Euclidean space by two quantities: a unit vector indicating the direction of an axis of rotation, and an angle describing the magnitude of the rotation about the axis. To rotate the -axis onto the -axis, the desired axis of rotation is the direction that is normal to both the -axis and the -axis. Conveniently, the cross-product between two-unit vectors results in a normal vector with respect to the two input unity vectors,

|  |  |  |
| --- | --- | --- |
|  |  | s12 |

where is the direction of the axis of rotation. Note, has been normalized by its magnitude, , to ensure it is a unit vector in of itself.

The magnitude of the rotation can be determined using the vector dot product. The angle between two vectors is related to the dot product,

|  |  |  |
| --- | --- | --- |
|  |  | s13 |

Recall that that the magnitudes of and are one, and hence, they are not included in Equation s13. Furthermore, the result of will lie between and radians; therefore, will be positive.

Given an axis of rotation and a magnitude of rotation, we can now employ Rodrigues' rotation formula to rotate a point in 3D space. Note, it is assumed that the axis of rotation is rooted at the global origin, which in our case, is true because the () local coordinate system is coincident with the () global origin prior to the translation step. To rotate an arbitrary point about the axis of rotation defined by the unit vector by a magnitude of , the following can be used,

|  |  |  |
| --- | --- | --- |
|  |  | s14 |

where is the newly rotated point.

The last transformation step is to align the -axes to those in the -coordinate system. The procedure will be same as what was done to align the -axis, based on the Euler axis-angle representation and using the Rodrigues' rotation formula. However, what remains to be determined for this second rotation is the definition of a new axis to rotate about, as well as the magnitude of the rotation. In this case, the desired axis of rotation is simply the new -axis. The magnitude, or angle, of the rotation can be found by using the dot product between the -axis and the -axis, similar to before in Equation s13. Now, with the axis and angle defined, the Rodrigues' rotation formula can be used again. After these two rotations, the () coordinate system will contain the same directions as the () coordinate system. All that is left to do is translate the () coordinate system based on Equation s10.

The forward transformation is now complete by applying the aforementioned rotations, followed by a translation, which then aligns the analytical () coordinate system to that of the physical () coordinate system. If the backward transformation is desired, then these translation and rotation steps can be done in reverse using negative values for the direction of translation as well as the directions of each rotation.

**S.3. Extension to Arbitrarily Shaped Objects**

For simplicity, a hollow cylinder was chosen throughout this study as the representative geometry for illustrating our novel methodology related to the geometric comparison of a physical object against an idealized, perfect model. However, this methodology can be applied to arbitrarily shaped objects, which is an important aspect of these techniques since additive manufacturing is known for its capability of producing geometrically complex components. In this section, we discuss more specifically how to use the minimum moment of inertia to geometrically compare an arbitrarily shaped physical object against a perfect model.

The general concept of our novel methodology is to define a unique, local coordinate system within the printed, physical object based on its principal moments of inertia. First, the origin of the local coordinate system () is positioned to be coincident with the centroid of the physical object, as discussed in Section 2.4.4. For this discussion, assume that the physical object is represented by a stack of binarized images (i.e., a voxel model) such that the voids and air-space correspond to black pixels and the solid material corresponds to white pixels. Moreover, the mass is assumed to be homogeneous, and thus, equal to unity for each white pixel. With these assumptions, the center of mass becomes coincident with the centroid of the physical object, as formally defined below,

|  |  |  |
| --- | --- | --- |
|  |  | s15 |

where is the position vector of the center of mass, is the total number of pixels that comprise the physical object, and is the position vector of a white-pixel .

Next, the principal axes and principal moments of inertia must be calculated for the physical object, which is achieved through the eigendecomposition of the moment of inertia tensor as defined in Eq. 2, repeated here for convenience (with ),

|  |  |
| --- | --- |
|  | s16 |

Through eigendecomposition, the moment of inertia tensor is diagonalized by applying a three-dimensional rotation, similar to Eq. 4,

|  |  |
| --- | --- |
|  | s17 |

where is the diagonalized matrix containing the principal moments of inertia (, , ), and is an orthogonal rotation matrix whose -th column is the eigenvector of . For example, eigenvector corresponds to the the principal moment of inertia, . These three eiqenvectors, can (and will) be used to systematically define a coordinate system of the physical object.

Referring to Section 2.4.5, the axis of rotation corresponding to the minimum moment of inertia, which we denote as , is the eigenvector that corresponds to the minimum principal moment of inertia. We will take this direction to be the local -axis, as shown in Figure S.2. The remaining two eigenvectors define the local - and -directions thereby defining a right-handed Cartesian coordinate system on the physical object. A similar eigendecomposition can be performed for the moment of inertia tensor of the perfect model object, where the resultant eigenvectors can then be used to define a global coordinate system (see Figure S.2).

The final step involves the application of rigid-body translations and rotations to the physical object in order to align the local coordinate system () with the global coordinate system (), which Section S.2 discusses the steps to do so. If the perfect object is axisymmetric about its minimum moment of inertia (i.e., about its local -axis), then the final rotation in Figure S.2c is unnecessary. Otherwise, in general, the final rotation shown in Figure S.2c must be peformed in order to ensure that the local coordinate system () of the physical object is aligned and coincident with the global coordinate system () of the perfect object.



**Figure S.2.** (a) An arbitrarily shaped physical object is illustrated with a local () coordinate system. The origin of the local coordinate system is chosen to be coincident with centroid of the physical object, and the () coordinate system is colinear with the principal axes of the physical object as defined by an eigendecomposition of the inertia tensor. Notice that the -axis is defined to be colinear with the eigenvector corresponding to the minimum principal moment of inertia, . The global () coordinate system corresponds to the perfect model object where, once again, the coordinate system is colinear with the principal axes of the perfect object. Similarly, the -axis is defined to be colinear with the axis corresponding to the minimum principal moment of inertia, , of the perfect object. (b) The physical object is rotated and translated such that the local -axis becomes colinear with the global -axis, see Section S.2 for more details. (c) The physical object is rotated about the -axis such that the the principal axes () of the physical object are aligned with the principal axes () of the perfect object.

**S.4. Application Example: 3D Alignment of an Asymmetric Rocker Arm**

The method described above in Section S.3 was thus applied to an automotive rocker arm geometry: an arbitrary object with no axes of symmetry, refer to Figure S.3a. Note, in Figure S.3a, the “ideal” geometry is represented as a surface mesh using the Standard Tessellation Language (STL) file format. As designed, the rocker arm has nominal tangent-to-tangent dimensions of 25 mm by 22 mm by 63 mm. The additively manufactured part was produced from polylactic acid (PLA) using a spool-fed strand printer, see Figure S.3b. The geometry of the printed rocker arm was quantified using X-ray CT, specifically with a North Star Imaging X50 machine. Table S.1 summarizes the imaging conditions that were used.



**Figure S.3.** (a) Mesh-based model of an automotive rocker arm, which represents the “ideal” geometry. The model is shown here via a Standard Tessellation Language (STL) file. (b) A photograph of the rocker arm after it was additively manufactured in polylactic acid (PLA), which represents the “printed” object.

**Table S.1**. X-ray CT imaging conditions for the North Star Imaging X50 machine.

|  |  |
| --- | --- |
| **Source voltage** | 50 kV |
| **Source current** | 280 µA |
| **Source power (voltage amperage)** | 14 W |
| **Exposure time** | 1 s |
| **Voxel size** | 48.6 µm/pixel |
| **Number of radiographs taken over 360** | 1800 radiographs |
| **Number of radiographs averaged at each angle** | 5 radiographs |
| **Detector data resolution** | 14-bit |

The X-ray radiographs were reconstructed using North Star Imaging’s proprietary software, resulting in a stack of 762 grayscale (16-bit) images, each comprising 1359 rows of pixels and 659 columns of pixels. Due to memory constraints related to later steps in post-processing, the images were scaled using bicubic interpolation such that the resultant voxel size was 60.0 µm/pixel. The printed rocker arm was then segmented from the image stack using a similar thresholding methodology as described in Figure 4. In order to be consistent with the print object, the ideal STL model was also voxelized using the same voxel width as the scaled X-ray CT images: a voxel width of 60.0 µm. The Visualization Toolkit (VTK) open-source library was utilized to voxelize the STL model. The two models were then aligned in 3D as described in Section S.3 and Figure S.2. A digitalized rendering of the two models, superimposed onto each other, is given by Figure S.4.

**Diagram

Description automatically generated**

**Figure S.4.** A rendering of the front and back of the ideal geometry and digitized, printed model after they have been aligned as described in Section S.3. More specifically, the centroid of each geometry was made coincident with one another. Then, the principal moments of inertia were calculated for each geometry about their respective centroids. The geometries were rotated in 3D about their respective centroids such that their principal axes (of inertia) were made colinear. Note, for each geometry, the principal axis corresponding to the minimum principal moment of inertia is defined to be the -axis, as was adopted in Figure S.2.

As expected, the printed object contains interior build defects and geometric errors, which in turn, results in the object having a different set of principal axes of inertia compared to the ideal geometry. So, when the principal axes of inertia of the printed object are made to align with those of the ideal geometry, there is some degree of mismatch, which is quantified in the remaining paragraphs of this section.

The volume-based print index was directly calculated using Equation (12). The boundary-based print index was calculated using a modified version of Equation (11):

(s18)

where , the mean distance to the surface orthogonal to the axis of minimum moment of inertia at locations “*i and o* ”, replaces , the distance to the surface orthogonal to the axis of minimum moment of inertia at locations “*i and o .*” The normalizing factor, , i.e. the denominator in the deviation terms, is selected arbitrarily. If the axis of minimum moment of inertia does not pass thorough the surface of the object, then is a logical and convenient choice for the normalizing factor. If, however, the axis of minimum moment of inertia passes through the surface of the object, then deviations will be undefined for those unique points “*i and o* ” for which the axis passes through the object’s surface since is zero for such points. This singularity condition would be true for any axis of reference from which the surface location is measured which passes through the surface of the object and is not unique to the axis of minimum moment of inertia. Table S.2 summarizes the print indexes thus calculated. Note that in this particular case, the surface of the printed object has a reasonably high fidelity while the volume does not due to internal defects of the printed object.

**Table S.2.** Calculated print indexes for the rocker arm object. A total of 13,550 points were sampled for calculating *PIB* .

|  |  |
| --- | --- |
| ***PIV*** (Equation 11) | 0.41 |
| ***PIB*** (Equation S18) | 0.82 |

Using the rocker arm geometry as an illustrative example demonstrates how 3D, arbitrarily shaped objects (i.e., objects with no axes of symmetry) can be readily aligned and compared using their respective principal axes of inertia, as described in Section S.3. Moreover, the novel metrics defined in this work to measure the geometric deviations between the two objects, termed the print indexes, were also calculated for the rocker arm geometry, proving their relevance and effectiveness for arbitrarily shaped geometries.

**Supplemental References**

s1. Olinde Rodrigues, "Des lois géométriques qui régissent les déplacements d'un système solide dans l'espace, et de la variation des coordonnées provenant de ces déplacements considérés indépendants des causes qui peuvent les produire", Journal de Mathématiques Pures et Appliquées 5 (1840), 380–440.

1. Certain commercial equipment, software and/or materials are identified in this paper in order to adequately specify the experimental procedure. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the equipment and/or materials used are necessarily the best available for the purpose. [↑](#footnote-ref-2)