

DATA PROCESSING, MODELING, AND ERROR ANALYSIS IN DISCRETE PART  
GEOMETRIC INSPECTION

by

Prashanth Jaganmohan

A dissertation submitted to the faculty of  
The University of North Carolina at Charlotte  
in partial fulfillment of the requirements  
for the degree of Doctor of Philosophy in  
Mechanical Engineering

Charlotte

2020

Approved by:

---

Dr. Edward Morse

---

Dr. Jimmie Miller

---

Dr. Konstantinos Falaggis

---

Dr. Yuri Godin



## ABSTRACT

PRASHANTH JAGANMOHAN. Data processing, modeling, and error analysis in discrete part geometric inspection. (Under the direction of DR. EDWARD MORSE)

In today's world of modern metrology, it becomes increasingly common to find new technologies which employ non-traditional measuring mechanisms in attempts to provide new advantages to lever over competing measuring instruments. In addition to such new technologies emerging, parts being measured also tend to grow in complexity, presenting new and unique challenges in their measurement. As a result of such advancements in technology and increasing part complexities, it becomes necessary to develop and optimize data processing, data modeling methods and error analysis methods tailored to a given measurement technique or a given part. This dissertation explores these aspects of discrete part metrology and presents solutions for each case considered. A data processing method is developed to allow measurement of any complex part using a precision rotary table and a set of triangulation-based laser line sensors. Machine learning approaches are also explored for a similar measurement problem. The system described is tailored to measure desired discrete parts (gears in this case) and can be considered a custom coordinate measuring system (CMS). An essential step to pushing such custom machines or new measurement technologies into widespread use, is the development of standardized test procedures that enable users to compare different technologies by their performance against traceable standards. Development of such standards often involve designing performance evaluation test procedures that are sensitive to as many known error sources are possible. Therefore, such an approach is

adopted for two measuring technologies, namely X-Ray Computed Tomography (XCT) and Stereo-vision Photogrammetry. In each case, several known error sources are characterized, and recommendations are made on ways to capture them. Further, from an automation perspective, it becomes essential to use a data modeling system that can support descriptions of complex parts and their measurement results, as well as accommodate new measurement technologies and tools. To this end, a promising candidate, the Quality Information Framework (QIF) has been identified.

## DEDICATION

This thesis is dedicated to my parents and my grandmother for their extraordinary encouragement and support.

## ACKNOWLEDGEMENTS

I would first like to thank my dissertation advisor Dr. Edward Morse, who always steered me in the right direction whenever I found myself in a tight spot and needed guidance. I would also like to thank my dissertation committee members, the Center for Precision metrology, its industrial affiliates Vantage corporation, and the National Institute of Standards and Technology (NIST) for their funding in support of my work. I would like to extend my gratitude to Dr. Bala Muralikrishnan, Dr. Daniel Sawyer, Dr. Craig Shakarji, Dr. Meghan Shilling, Dr. Sambit Bhattacharya, Tim White, and other colleagues and friends for all their support.

## TABLE OF CONTENTS

List of Tables	viii
List of Figures	ix
Chapter 1: Introduction	1
Chapter 2: Data Processing: Deterministic Approach	5
2.1. Introduction	5
2.2. Experimental Setup	7
2.3. Measurement Process	10
2.4. Artifact Design	14
2.5. Calibration Methodology	17
2.5.1. Data acquisition	18
2.5.2. Manual fitting using a Graphical User Interface	19
2.5.3. Feature extraction	21
2.5.4. Least squares fitting	22
2.6. Measurement Results	23
2.7. Automation	26
2.8. Outcomes and Future Work	27
Chapter 3: Data Processing: Machine learning approach	28
3.1. Introduction	28
3.2. Simulation Model	30
3.3. Generation of Lookup Table	33
3.4. Dynamic Time Warping as a metric for comparison	34
3.5. Outcomes and Future Work	38

Chapter 4: Data modeling: Quality Information Framework	39
4.1. Introduction	39
4.2. QIF Application Areas	41
4.3. QIF Resources	42
4.4. Objectives and Scope	47
4.5. Application Security	49
4.6. Interfacing and Underlying Processes	50
4.7. QIF Shortcomings Identified	54
4.8. Outcomes and Future Work	55
Chapter 5: Error Analysis: X-Ray Computed Tomography	56
5.1. Introduction	56
5.2. Simulation Setup and Modeling	59
5.3. Simulation Methodology	63
5.4. Results and Discussion	66
5.4.1. Sphere center-to-center distance errors	66
5.4.1. Sphere form errors	71
5.5. Proposed Recommendations	74
5.6. Outcomes	76
Chapter 6: Error Analysis: Stereo-vision photogrammetry	78
6.1. Introduction	78
6.2. Experimental Setup	79
6.3. Model Description	80
6.4. Calibration	85

	ix
6.5. Simulation	88
6.6. Experimental Validation	93
6.7. Outcomes	94
Chapter 7: Conclusions and Future work	95
References	97

## LIST OF TABLES

Table 2.1. Snippet of sensor scan	19
Table 2.2. Measurement results from test part	25
Table 3.1. Nominal parameters and perturbations used for simulation	33
Table 3.2. Perturbation parameters corresponding to the test curve and those of minima identified	37
Table 4.1. Simple database storing authorized user data	49
Table 5.1. Configuration capturing highest center-to-center distance error sensitivity for detector errors and Z location error of stage	68
Table 5.2. Configuration capturing highest center-to-center distance error sensitivity for stage error motions	69
Table 5.3. Configuration capturing highest form error sensitivity for detector errors and Z location error of stage	73
Table 5.4. Configuration capturing highest form error sensitivity for stage error motions	73
Table 6.1. Estimated model parameters for systems A and B	87
Table 6.2. Maximum simulated distance errors- System A	90
Table 6.3. Maximum simulated distance errors- System B	91
Table 6.4. Maximum experimental distance errors- Systems A and B	94

## LIST OF FIGURES

Fig. 1.1. Schematic of typical measurement life cycle of a product	2
Fig. 1.2. Schematic of a typical Instrument performance evaluation process	4
Fig. 2.1. Prototype machine	8
Fig. 2.2. Sensor arrangement	9
Fig. 2.3. (a) Sensor coordinate system and (b) Simulated profile data	10
Fig. 2.4. Schematic of point cloud generation process	11
Fig. 2.5. Artifact design	14
Fig. 2.6. Photograph of (a) Artifact and (b) Gear	16
Fig. 2.7. Example of GUI used	19
Fig. 2.8. Point cloud and nominal artifact (a) before and (b) after manual fitting	21
Fig. 2.9. Examples of feature extraction	22
Fig. 2.10. Sample calibration result file	23
Fig. 2.11. Commercial software package used for gear analysis	23
Fig. 2.12. Point cloud of test part	24
Fig. 2.13. Example of a work order file	26
Fig. 3.1. Schematic of intended machine learning-based approach	29
Fig. 3.2. Examples of redundant cases where different parameter sets are used to describe equivalent systems	31
Fig. 3.3. New nomenclature for system description	31
Fig. 3.4. Shape used as the artifact, and its regions viewed by the sensor in different Configurations	32
Fig. 3.5. Curve from nominal parameter set	33

Fig. 3.6. (a) Sample curve 1, (b) Sample curve 2, and (c) their comparison using DTW	35
Fig. 3.7. (a) DTW distances obtained on comparing test curve with lookup table, and (b) blown up version showing Global minimum and Intuitive solution	36
Fig. 4.1. Quality Measurement Process	40
Fig. 4.2. QIF Application areas	41
Fig. 4.3. QIF resources inheritance diagram	43
Fig. 4.4. Close-up of sensor hierarchy	44
Fig. 4.5. Close-up of Tool with integrated sensor hierarchy	45
Fig. 4.6. Attributes of a Caliper	46
Fig. 4.7. Example of capturing resources in a QIF Resources file	48
Fig. 4.8. Login screen	50
Fig. 4.9. Home screen	50
Fig. 4.10. Computed Tomography Form	51
Fig. 4.11. Relationship between XML files and class objects	52
Fig. 4.12. Snippets showing attributes in (a) Schema and (b) class definitions	53
Fig. 4.13. QIF Resources instance file	54
Fig. 5.1. Overview of simulations for different magnifications	58
Fig. 5.2. Reference object containing 125 spheres chosen for simulation	60
Fig. 5.3. Geometrical error sources associated with the detector	61
Fig. 5.4. Geometrical error sources associated with the stage	61
Fig. 5.5. Example of locus traced on detector	64
Fig. 5.6. Distance error sensitivity to detector displacement in X	66
Fig. 5.7. Distance error sensitivity to stage wobble in X	67

Fig. 5.8. Sensitive lines for detector errors and Z location error of rotation axis	70
Fig. 5.9. Sensitive lines for first order cosine components of stage error motions	70
Fig. 5.10. Sensitive lines for second order cosine components of stage error motions	71
Fig. 5.11. Sensitive lines for third order cosine components of stage error motions	71
Fig. 5.12. Sensitive lines for fourth order cosine components of stage error motions	71
Fig. 5.13. Form error sensitivity to detector Y location error	72
Fig. 5.14. Form error sensitivity to detector Z rotation error	72
Fig. 5.15. Photograph of a typical sphere forest	75
Fig. 5.16. Core minimum set of (a) measurement lines and (b) sphere locations	76
Fig. 6.1. Top view of camera setup showing apex angle	79
Fig. 6.2. Experimental setup for system A	80
Fig. 6.3. Intrinsic and Extrinsic matrix transformations	82
Fig. 6.4. Radial distortion examples	83
Fig. 6.5. Tangential distortion	84
Fig. 6.6. Checkerboard used for calibration	86
Fig. 6.7. Example of checkerboard poses used for calibration	86
Fig. 6.8. Measurement lines for system A (a) Most sensitive lines (b) VDI/VDE lines	89
Fig. 6.9. Measurement lines for system B (a) Most sensitive lines (b) VDI/VDE lines	89
Fig. 6.10. Measurement of grid of points on CMM	93

## CHAPTER 1: INTRODUCTION

The past few decades have witnessed a steady increase in the complexity of manufactured components. In parallel, several new measurement technologies have been developed and commercialized. The evolution of surface metrology is one such example where more complex methods of measurement have been conceived as a result of increasing part complexity, such as parts with structured surfaces [1]. Recent trends in precision metrology, particularly at the nanometer level are described by Manske et al [2]. Developments in large-scale metrology have been discussed in several research efforts [3-6]. Even with traditional parts such as gears, that have been measured by conventional methods for decades, increasing technical specifications translate to increasing number of characteristics or parameters to be measured, calling for new approaches to measurement [7-8].

With such increasing part/measurement complexities and high cycle times commonly associated with conventional contact-based measurement, there arises a need to develop dedicated measurement systems suited for each application, in order to optimize for high volume of measurement and low cycle time. Such discrete part coordinate metrology will be the focus of this dissertation. Fig. 1.1 shows a schematic of a typical measurement life cycle of a product. This dissertation attempts to take each aspect of this measurement cycle and push it forward towards the direction of discrete part coordinate metrology. Specifically, methods in data processing, modeling, and error analysis that provide the flexibility needed for discrete part metrology are discussed.

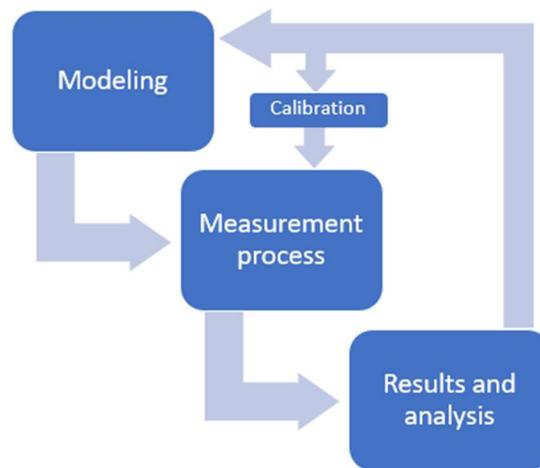


Fig. 1.1. Schematic of typical measurement life cycle of a product.

The organization of this dissertation is as follows. Chapters 2 and 3 dive into a case of data processing, specifically calibration and measurement methodologies developed for a custom Coordinate Measuring System (CMS) that uses a precision rotary table and a set of laser line sensors. These correspond to the ‘Calibration’ and ‘Measurement process’ blocks shown in Fig. 1.1. Chapter 2 describes deterministic methods developed for such a measurement system. The methods presented allow for the traceable measurement of any complex part that can be placed on the table as long as the features to be measured are within the measuring ranges of the sensors. Test parts are measured with this system to compare performance against conventional Coordinate Measuring Machines (CMMs).

Chapter 3 explores alternate solutions to a similar measurement problem simplified to a two-dimensional case. Machine learning approaches are examined, and promising candidate algorithms are identified. The methodology proposed consists of a combination of shape dynamic time warping (a point-to-point matching algorithm) and nearest neighbor search followed by a multi-dimensional interpolation among the chosen neighbors.

Training and test data are generated by simulation methods to test the algorithms and preliminary results are presented.

Chapter 4 focusses on data modeling. While the ‘Modeling’ block in Fig. 1.1 refers to model-based representation in the measurement life cycle, such as CAD models, Product Manufacturing Information (PMI), etc., this chapter considers modeling in a broader sense, for example, the modeling of the measuring equipment (resources) available to perform the measurement task. Efforts made to further the implementation of Quality Information Framework (QIF), a candidate for an enterprise-wide standard solution, are discussed. Such a standard accommodates for all stages of the measurement cycle by modeling the appropriate information in standardized format and making it available and usable at subsequent stages. To promote its implementation in the industry, a database-centric application software is developed to enable users/ industries to catalog their measurement resources in the standard QIF format.

In Chapters 5 and 6, error characterization and analyses are done for two kinds of CMS. In a broad sense, these analyses fit into Fig. 1.1 in the ‘Results and Analysis’ block. Specifically, for the two technologies studied, the effect of several known error sources on the measurand are studied. Such a sensitivity analysis forms the basis of an uncertainty model and can be used to evaluate the task specific uncertainty involved in analyzing a given measurement result. These analyses also contribute to the development of instrument performance standards, as an integral step in developing such performance evaluation standards for any measurement system involves identifying various possible error sources and designing tests to capture them. Further, recommendations are made in each case, on ways to capture all the error sources studied to a reasonable level. Chapter 5 discusses these

analyses performed for X-Ray Computed Tomography (XCT) while chapter 6 discusses similar studies on Stereo vision Photogrammetric systems. Fig. 1.2 shows an extended view of the instrument performance evaluation process.

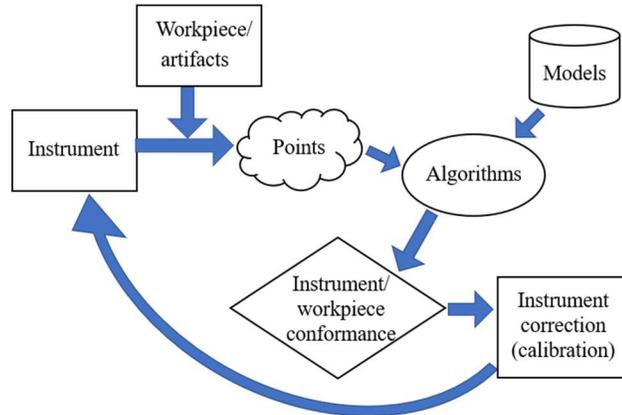


Fig. 1.2. Schematic of a typical Instrument performance evaluation process.

Chapter 7 concludes the dissertation and discusses future work.

## CHAPTER 2: DATA PROCESSING: DETERMINISTIC APPROACH

### 2.1. INTRODUCTION

In case of conventional CMMs which use contact-based methods, the accuracy of measurement relies on precise knowledge of the motion of the probing/ sensing system throughout the measurement. Further, the trajectory of the probing system has to be different for different parts. In other words, new programs need to be written for parts with different geometries or different complexities. Usually, such contact-based methods dramatically increase the measurement time due to the motion requirements of the probing systems. On the other hand, a handful of optical coordinate measuring systems such as laser tracker, laser radar, and some photogrammetric systems are reliant on accurate knowledge of the position and, where applicable, the orientation of a target (or a set of targets mounted on a handheld probe). Even in such cases, the measurement time is still likely to be high due to the need to position the target at several locations in contact with the workpiece, although this increase of measurement time would also depend on the number of features to be measured.

Gear metrology is one such area where classical inspection involves contact-based methods and therefore time constraints pose a challenge to 100% inspection. Consequently, it is common to measure only a handful of teeth (usually 4-6) in each gear. Additionally, contact-based methods are associated with other challenges including a limited compliance with an ever-increasing number and types of gear flank modifications, a need for improved

feedback to the manufacturing process, etc. These are discussed in detail by Goch et al. [8], where a holistic approach is advocated to overcome these challenges.

In gear manufacturing as well as metrology, there have been several advancements over the last few decades, allowing measurement of gears with increasing complexities with a variety of tactile and optical methods [9]. With increasing need to measure larger areas on gears with high complexities and to overcome time limitations associated with mass inspection, there is a steadily growing interest in non-contact inspection methods. One such method has been implemented by Gao et al., where cutting errors on flexspline gears are measured using two laser probes [10]. However, one major limitation associated with the method presented, is that the geometric alignment between the two probes with respect to each other and to the axis of rotation heavily influences the accuracy of the system, except when measuring parameters where the error due to this source can be effectively eliminated by averaging over a complete revolution. A vision-based system was proposed by Pahk et al. [11], where a rotary table was not used, and the measurement relied on image processing algorithms that operated on a top-view snapshot of the gear. Such a methodology, while useful for the application demonstrated, places limitations on the ability to measure helical gears, profile modifications on teeth, etc.

This chapter discusses a particular case of optical gear metrology that uses a custom assembled Coordinate Measuring System (CMS) consisting of triangulation-based laser line scanners [12-13] and a precision rotary table and describes deterministic techniques of data processing essential to measure the part in a traceable manner [14]. Although the methods described here are in the context of measuring a gear, the techniques presented

enable measurement of any complex part, as long as the features to be measured are physically within the measuring ranges of the sensing components.

In the following sections, the system and the measurement process are described. The calibration methodology is presented, and its performance is verified by measuring a simple part. Some distinct advantages of the calibration method developed here include eliminating the need to carefully align the part axis with the rotation axis and the flexibility available in the positioning of the various sensors (since the calibration is sensitive to these changes and can correct the measurement accordingly). The latter enables quick repositioning of the sensors, if necessary, within the machine to enable measurement of parts of different shapes and sizes. Further, the measuring system generates a complete point cloud of the part being measured and is therefore not limited to measuring only specific parameters. These benefits are in addition to those inherent to optical methods, such as the ability to measure large and dense point clouds with a low cycle time, and therefore the opportunity to achieve mass inspection.

## 2.2. EXPERIMENTAL SETUP

The system is a CMS consisting of a precision rotary axis and triangulation-based laser line scanners. The rotary axis is realized by an air bearing spindle and is used to rotate the part, such that the surfaces to be measured are within the field of view of the sensors. The sensors thus ‘see’ the whole surface as the part completes a full rotation. A prototype machine is shown in Fig. 2.1, where the sensors are hidden from view by a red ‘hood’. This serves to protect the sensors from being accidentally knocked, and to some extent, control ambient lighting.



Fig. 2.1. Prototype machine.

The number of sensors used and their placement for a given machine depend on the size of the part, the features to be measured, as well as the measuring range of the sensors along the various axes. For this study, the Gocator 2520 sensors are used. These are commercially available from LMI Technologies. These have a maximum measuring range of 32 mm along the scan line or the X axis (or spanning from -16 mm to +16 mm) with a resolution of 10  $\mu\text{m}$ . However, it is not necessary that the entire measurement range in X must be used. The sensor interface provides options to select region of interest, and often only 25-30 mm of the 32 mm is used. Along the height direction (Z axis), the resolution is 0.5  $\mu\text{m}$  and the range is  $\pm 12$  mm, with the zero point being approximately 60 mm away from the physical surface of the sensor. In cases where the surface is being measured is beyond this range, or there is nothing to reflect the laser line, the sensor simply records a very large value, 999 in this case. Considering these measurement ranges and the fact that gears are the primary focus in this study, sensor placement is chosen accordingly. The arrangement chosen is depicted in Fig. 2.2.

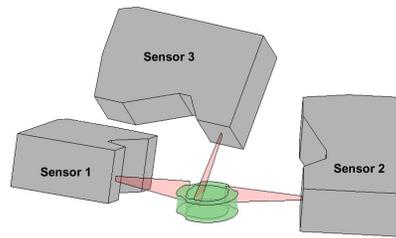


Fig. 2.2. Sensor arrangement.

For this case, it is determined that three sensors are sufficient. Two sensors face each other to measure the same surfaces from opposite sides of a part. These are positioned in such a manner so as to measure the right and left flanks of a gear. As seen from Fig. 2.2, these are slightly tilted from the horizontal plane. Such orientations are chosen to measure helical teeth. Although a sensor's angular orientation is not expected to significantly impact the measured point cloud, it was found that better results were obtained when the laser line was along the slope of the helix. This is likely due to the effect of lag between sensor trigger and recording. The spindle rotates at about 15 rpm, which implies that a single rotation takes four seconds. This in turn implies that even for a delay of 1 millisecond, a point on the outer surface of a part of diameter 40 mm would move 30  $\mu\text{m}$  past the sensor during that time delay. Since the intended parts are gears, during this 30  $\mu\text{m}$  of travel, the measured points on the tooth of the gear can have their coordinates off by a much higher degree as the sensor sees a steep surface pass by. An additional sensor is positioned above the part such that it can measure the top surface and inner hole on a part. This positioning is to enable the measurement of an inner diameter on a gear. In applications where the height of the part (in this case, the face width of the gear) is larger than 32 mm, it may be necessary to have multiple pairs of sensors, with each pair at a given height from the table, until the total height of the part is covered. Similarly, if the tooth height is larger than 24 mm, and

it is desired to measure the top land and bottom land (although these are usually not critical measurements since these surfaces rarely make contact with a mating gear), it may be necessary to have more than two sensors at a given horizontal level.

### 2.3. MEASUREMENT PROCESS

At any given instant when they are triggered, the laser line scanners provide two-dimensional data or an  $(x,z)$  height-map of the surface being measured. This is reported in the sensor's own local coordinate system. This coordinate system and a sample measurement profile are shown in Fig. 2.3 (a) and (b) respectively. Over the course of the measurement process, the workpiece is rotated by a near-constant angular velocity, and for approximately every 0.18 degree of rotation of this axis, the rotation angle is recorded along with the 2D data obtained from the sensors at that instant. This synchronization is achieved by using the same pulse to trigger the encoder as well as the three sensors. As a result, at the end of a full table revolution, about 2000 profiles or lines of data are obtained from each sensor, with each profile containing up to 2500 points.

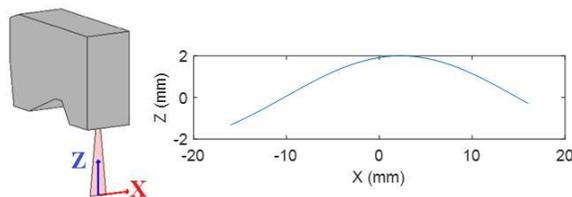


Fig. 2.3. (a) Sensor coordinate system and (b) Simulated profile data.

When the position and orientation (or pose) of a sensor with respect to the rotation axis is known, and the angle of rotation of this axis is known at every instant, the 2D profile data captured by the sensor can be transformed to a point cloud that uses a common coordinate system using a homogeneous transformation matrix (HTM) used for rigid body transformations. Each sensor requires a different HTM to be applied for each angle of rotation of the rotary axis. This allows a complete point cloud of the part to be produced in a common coordinate system, namely the table coordinate system. This process is depicted in Fig. 2.4.

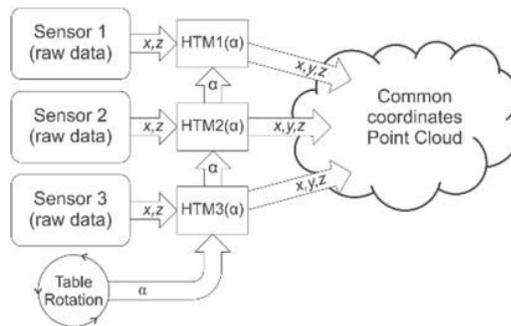


Fig. 2.4. Schematic of point cloud generation process.

The measurement process of generating a point cloud therefore relies on knowledge of the pose of each sensor in space with respect to the rotation axis. This pose information is obtained by a calibration procedure using an artifact of known geometry. This sensor pose is described by the following set of six parameters, where the first three are translations and the last three are rotations about the three axes.

$$\text{Calibration} = [dx, dy, dz, a_x, a_y, a_z]$$

Further, since the sensor's local coordinate system is offset from the physical sensor, a physical corner of the sensor is arbitrarily chosen as reference, and the location of the local coordinate system with respect to this corner, is reported as shown below.

$$Offset = [dx_o, dy_o, dz_o, ax_o, ay_o, az_o]$$

Once the calibration information and offset information is available, The HTMs for each sensor (representing its relationship with the rotation axis) is determined by the following sequence of steps:

1. Accounting for sensor offset (a translation by  $[dx_o, dy_o, dz_o]$ ):
2. Rotation about Y axis by  $a_y$
3. Rotation about X axis by  $a_x$
4. Rotation about Z axis by  $a_z$
5. Translation by  $[dx, dy, dz]$

The order of steps shown above is chosen arbitrarily but is always done in the same order to ensure that a given set of independent parameters always results in the same HTM. The sequence of operations is represented in order, by the following five transformation matrices.

$$H_o = \begin{bmatrix} 1 & 0 & 0 & dx_o \\ 0 & 1 & 0 & dy_o \\ 0 & 0 & 1 & dz_o \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad H_1 = \begin{bmatrix} \cos(a_y) & 0 & \sin(a_y) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(a_y) & 0 & \cos(a_y) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad H_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(a_x) & -\sin(a_x) & 0 \\ 0 & \sin(a_x) & \cos(a_x) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_3 = \begin{bmatrix} \cos(a_z) & -\sin(a_z) & 0 & 0 \\ \sin(a_z) & \cos(a_z) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad H_4 = \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

These are assembled to obtain the HTMs for each sensor as follows.

$$H_{sensor} = [H_4][H_3][H_2][H_1][H_0]$$

$$H_{sensor} = \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(a_z) & -\sin(a_z) & 0 & 0 \\ \sin(a_z) & \cos(a_z) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(a_x) & -\sin(a_x) & 0 \\ 0 & \sin(a_x) & \cos(a_x) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(a_y) & 0 & \sin(a_y) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(a_y) & 0 & \cos(a_y) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & dx_o \\ 0 & 1 & 0 & dy_o \\ 0 & 0 & 1 & dz_o \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

If is the angle of rotation of the rotation axis, or encoder angle at a given instant is  $\theta_1$ , then the profile measured at that instant will have to first be transformed by  $H_{sensor}$  and then rotated about the Z axis by the encoder angle  $\theta_1$ . This is done by the  $H_{encoder}$  transformation matrix as shown below.

$$\text{Profile} = \begin{bmatrix} x_1 & 0 & z_1 & 1 \\ x_2 & 0 & z_2 & 1 \\ x_3 & 0 & z_3 & 1 \\ \dots & \dots & \dots & \dots \\ x_n & 0 & z_n & 1 \end{bmatrix}$$

$$H_{encoder} = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{Point cloud} = [\text{Profile}][H_{sensor}^{-1}][H_{encoder}^{-1}] = \begin{bmatrix} X_1 & Y_1 & Z_1 & 1 \\ X_2 & Y_2 & Z_2 & 1 \\ X_3 & Y_3 & Z_3 & 1 \\ \dots & \dots & \dots & \dots \\ X_n & Y_n & Z_n & 1 \end{bmatrix}$$

It is to be noted that  $H_{encoder}$  changes for every encoder angle, but  $H_{sensor}$  remains constant for each sensor. The combination of  $H_{sensor}$  and  $H_{encoder}$  serves to make the conversion from sensor coordinate frame to table coordinate frame. In this way, the profile captured at every encoder angle adds up to produce a complete point cloud. Since the cloud produced this way is fixed with respect to the rotary axis, the part does not need to be centered or carefully aligned with the axis. However, it is still necessary to hold/clamp the part firmly in place so that it does not move/slip during rotation. Once the point cloud is

generated, it can be readily used by any common analysis software for any necessary subsequent evaluation.

## 2.4. ARTIFACT DESIGN

To determine the pose of each sensor with respect to the rotation axis, a calibration procedure involving measuring an artifact of precisely known geometry becomes essential. The artifact must be designed in such a way that there is reasonably high sensitivity to change in each of the six parameters describing its pose. In other words, if any of the assumed six parameters is incorrect, the point cloud of the artifact resulting from such incorrect assumption of the sensor pose, must be significantly deformed, or the discrepancy must be amplified in a metric used to assess the resulting point cloud.

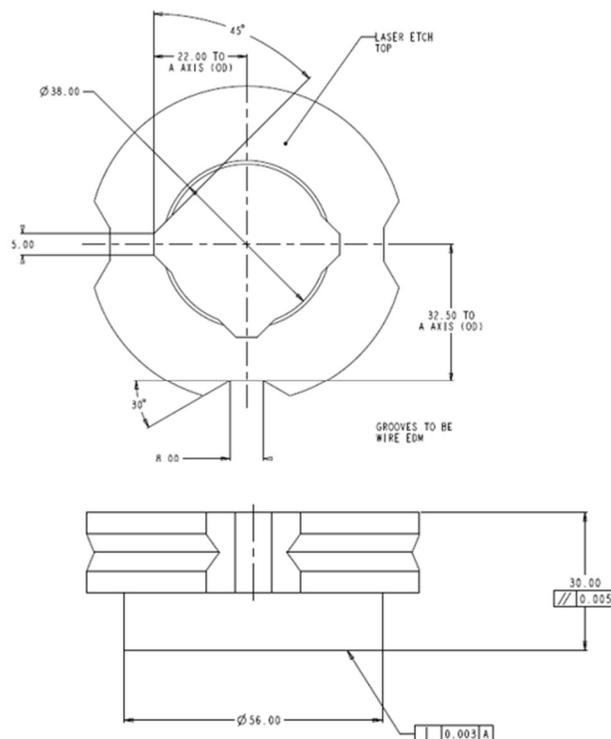


Fig. 2.5. Artifact design

A combination of features that ensures such large sensitivity in each of the six directions would ensure that there is one ‘correct’ sensor pose where the point cloud of artifact measured by each sensor matches correctly with the known geometry of the artifact. Fig. 2.5 shows the design of such an artifact.

The artifact dimensions are chosen such that its volume closely matches that of the gear to be measured. The artifact and the gear are shown in Fig. 2.6 (a) and (b). Since two of the three sensors see only the sides of the artifact and gear, these sides must contain enough information to determine all six parameters for those sensors. Consequently, a combination of an outer diameter, a set of nine planes along the outer diameter, and a V-groove is used to do this task. Any change in orientation of the sensor (or changes in  $a_x$ ,  $a_y$ , and  $a_z$  parameters) will skew these planes considerably, establishing sufficient sensitivity to these parameters. If the sensor were to be translate toward or away from the rotation axis (which would change  $dx$  and  $dy$  parameters), this would be manifested as apparent growing or shrinking in size of the resulting point cloud. In other words, the outer diameter of the cylindrical region of the artifact is sensitive to these parameters. The V-groove establishes sensitivity to the remaining parameter  $dz$  since any change in the height of the sensor will cause a change in position of the measured V-groove in the point cloud. This is required since the planes and outer diameter would otherwise show no variation along the vertical direction, making it impossible to know what region of the artifact is being sensed.

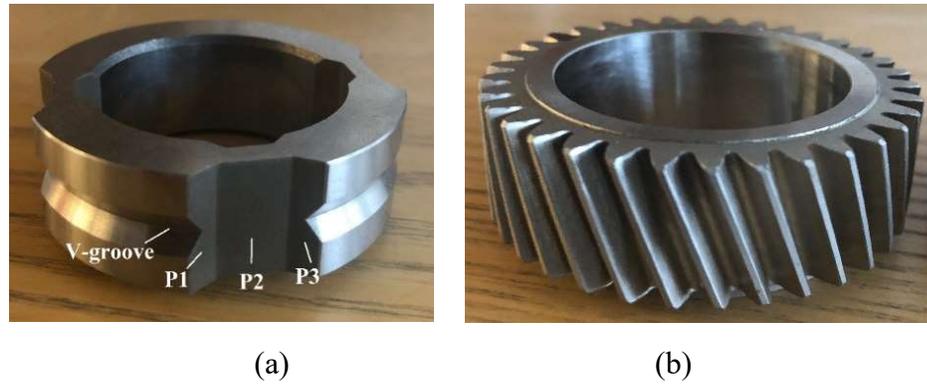


Fig. 2.6. Photograph of (a) Artifact and (b) Gear.

While all nine planes are not necessary to constrain the angular orientations of the sensor as described above (P1, P2, and P3 shown in Fig. 2.6 (a) would suffice), there could be inaccuracies in manufacturing or calibration of these planes (i.e., in their measurement by a CMM) which could significantly impact the estimation of sensor orientation. Therefore, three such triplets of planes are used to reduce the impact of these inaccuracies, and also to provide additional angular relationships between the sets of planes.

Similarly, one of the sensors views only the top surface and the inner diameter of the artifact. This sensor must also be provided with enough information to constrain all its six parameters. Therefore, the inner diameter is also equipped with a similar set of nine planes to constrain/ establish sensitivity to sensor orientation. The inner diameter of the artifact constrains the  $dx$  and  $dy$  parameters. Here, a V-groove is not necessary as the sensor can see the top surface. This is therefore used to constrain  $dz$  of this sensor. All the features of the artifact described here are calibrated via measurement on a traceable CMM.

## 2.5. CALIBRATION METHODOLOGY

As discussed previously, performing a successful measurement or generating a point cloud requires knowledge of sensor pose with respect to the axis of rotation. This knowledge is obtained by calibration, which involves measuring the artifact of known geometry. Similar calibration problems have been studied in the field of robotics to determine the pose of sensors in industrial robots [15-17].

The calibration procedure makes some assumptions. Some of these are listed below.

- Spindle error motions are negligible- In other words, the calibration does not compensate for spindle errors.
- Rotation angles of the rotary axis, or table indexing angles, are known with low uncertainty.
- The latency between sensors is negligible (i.e., the time lag between encoder trigger and actual capture of profile by each sensor). This includes the assumption that all three sensors trigger at the same instant and report the values captured at that same instant.
- Artifact geometry is known with low uncertainty.
- Profile data (specifically the  $z$  values corresponding to a predetermined set of  $x$  values) captured by the sensor has low uncertainty.

It is noteworthy that the calibration procedure does not assume or require that the artifact is centered or aligned with the rotation axis. This is because the optimization procedure determined a set of 6 parameters that represent the artifact pose (or parameters

that make up an HTM representing the transformation from the table coordinate system to the artifact coordinate system) in addition to the parameters that represent the pose of each sensor. This accounts for the unaligned position and tilt of the artifact.

Briefly, the calibration process involves refining the current estimate of sensor pose iteratively until the point cloud of the artifact generated using the current estimate matches well with the known artifact. The various steps involved in the calibration sequence are described below.

#### 2.5.1. Data acquisition.

The first step is to acquire data, by placing the artifact on the table, and rotating it through a full revolution while being scanned by the sensors. The rotary axis begins rotation from an encoder position of  $-45^\circ$ , passes through  $0^\circ$ , makes a complete revolution, passes  $0^\circ$  again, and continues until it reaches  $+45^\circ$ . This  $45^\circ$  band at the beginning and end of motion is to allow sufficient time for acceleration and deceleration so that the angular velocity remains constant during the data acquisition (from the first encoder zero reading to the next). The data collected includes a set of  $x$  values common to all acquired profiles, the encoder angle at which each profile is acquired, and the profile itself, which is given by a row of  $z$  values. The  $x$  values are common to all profiles for a given sensor, as they are predetermined. For example, setting the upper and lower bounds in  $x$  for a given sensor to  $-5$  and  $5$  respectively would mean that the pre-determined  $x$  values are  $[-5.00, -4.99, -4.98, \dots, 5.00]$  and the sensor would then try to determine what the  $z$  values are on the surface at each of the predetermined  $x$  values, thus allowing the data to be organized as shown in Table 2.1 which represents a snippet of such sensor data.

Table 2.1. Snippet of sensor scan.

Profile	Angle	x=-13.97	x=-13.96	.....	x=14.78
1	-0.006	-1.627	-1.624	.....	4.283
2	0.174	-1.626	-1.623	.....	4.279
.....	.....	.....	.....	.....	.....
2000	359.814	-1.629	-1.626	.....	4.312

### 2.5.2. Manual fitting using a Graphical User Interface.

The shape and size of the point cloud created from the acquired data is dependent on the sensor pose assumed in constructing the point cloud. Since the artifact is designed to be sensitive to all the pose-determining parameters, the cloud shows significant variation with small changes in the assumed sensor pose. This property is exploited in developing a method to arrive at a good starting estimate with which to begin the iterative process of sensor pose estimation. This is accomplished by means of a Graphical User Interface (GUI), an example of which is shown in Fig. 2.7.

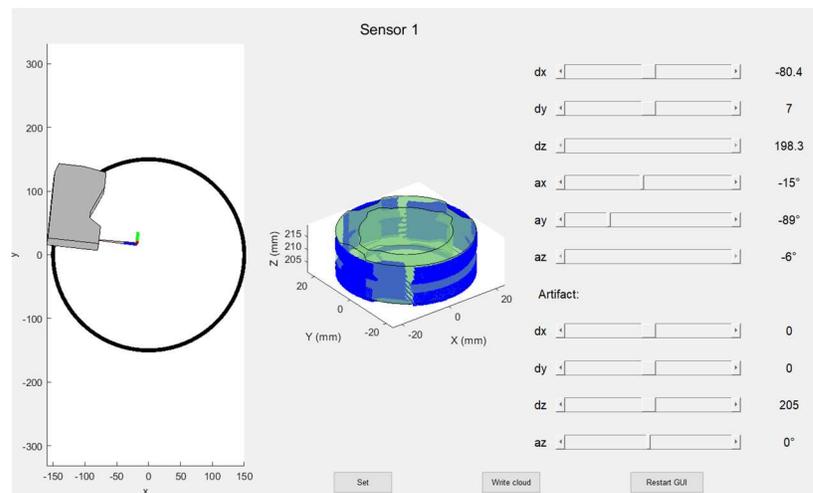


Fig. 2.7. Example of GUI used.

The objective of this GUI is to allow users to modify each of the six pose parameters individually, and to simultaneously display the new point cloud created with these altered estimates. These changes in parameters can be made using the sliders displayed on the right side of the GUI. There are sliders available to change artifact position and orientation as well. This is provided to account for artifact eccentricity and tilt. Further, it may be noted that two of the sliders in Fig 2.7 are disabled. These are  $dz$  and  $a_z$ , which denote the Z-axis translation and rotation. These are disabled because these parameters are fixed for sensor 1. This is done to ensure that the whole point cloud does not endlessly rotate and translate along with the sensors as one rigid unit. Fixing these values anchors the height and rotation about Z-axis for one arbitrarily chosen sensor, and this ensures that the sensor poses are always reported in a common coordinate system that does not change from one calibration to the next. The left section of the GUI shows the real-time positioning of the sensor on the table. In other words, it shows how the sensor's physical location in space with respect to the table changes with the current values set by the sliders. Similarly, the middle section of the GUI displays the real-time point cloud, or the point cloud resulting from using the current values set by the sliders. The point cloud is superimposed on an approximated visual representation of the artifact created in MATLAB for visual aid. The goal is to adjust the sliders until the point cloud matches closely with the artifact, and such a combination of parameters that makes that possible is used as the starting solution for each sensor. For example, the point clouds before and after such fitting are shown in Fig. 2.8 (a) and (b).

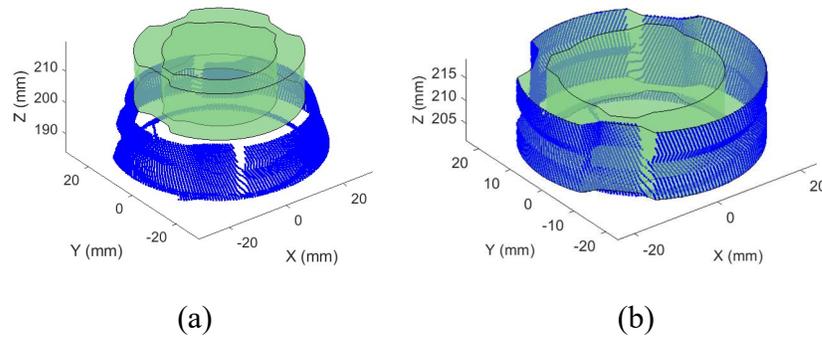


Fig. 2.8. Point cloud and nominal artifact (a) before and (b) after manual fitting.

### 2.5.3. Feature extraction.

Once an approximate starting solution is found using the GUI, a group of points belonging to each feature is selected based on a selection window around the nominal features. A margin is used along the boundary of each feature to avoid selecting points from the regions where one feature transitions into the next. As an example of features extracted from a point cloud, Fig. 2.9 shows the points belonging to the outer nine-pane set, points belonging to the inner diameter of the cylindrical region, and those belonging to the V-groove. The other features are not shown as this particular sensor does not see them. At this stage, another GUI is used as shown in Fig. 2.9.

The objective of this second stage of GUI is to adjust the parameters that define the selection windows for the various features. It also allows selecting/ unselecting any given feature in the optimization process. Unselecting a feature would not include that feature in the function that drives the optimization.

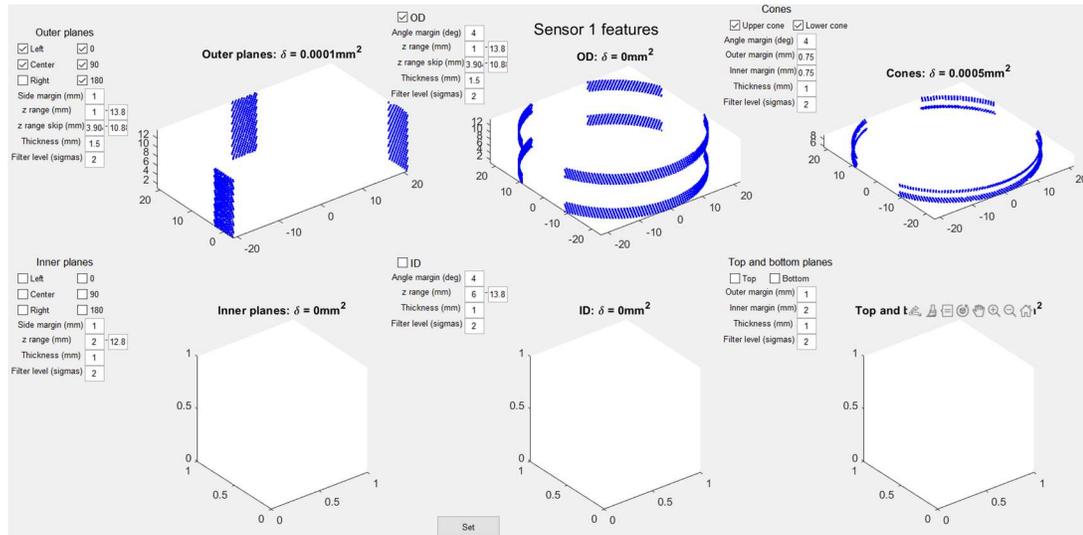


Fig.2.9. Examples of feature extraction.

Further, filters can also be selected here, defining how many standard deviations to use for each group. For instance, a filter value of 2 for the outer planes indicates that for each plane, once the points are fit to the plane and deviations are calculated, all the points whose deviation is greater than twice the standard deviation are discarded. This GUI is used only during the first iteration. After this, these window and filter settings are saved, and the same settings are used to reclassify points into features in subsequent iterations that use updated sensor information.

#### 2.5.4. Least squares fitting.

Following classification into various features, the points in each feature are compared to the CMM measurements of that feature. From the deviations observed, a mean of squares is computed, and the metric that drives the optimization consists of a weighted sum of these mean of squares for all features. Minimizing this metric yields a set of sensor pose parameters that represents the best estimate, or the result of calibration. During the

measurement phase, these parameters are used to build HTMs and generate point clouds as described previously.

```
dx,dy,dz,ax,ay,az
-90.1031 mm,-10.6456 mm,54 mm,6.8874°, -88.8571°, -3°
88.8002 mm,-11.0913 mm,94.7356 mm,-4.939°,88.073°,1.568°
17.4164 mm,17.6099 mm,123.2605 mm,1.571°,52.305°,85.030°
```

Fig. 2.10. Sample calibration result file.

Fig. 2.10 shows an example of a calibration result file. The parameters  $dz$  and  $a_z$  corresponding to sensor 1 (i.e., in the first row) are found to be integers. This is because these were not subject to optimization, for reasons previously discussed. Units are manually added to the file shown, and the number of decimal places is decreased for purposes of clarity.

## 2.6. MEASUREMENT RESULTS

Using the calibration result, a series of measurements are done on a gear using a commercial software package shown in Fig. 2.11.

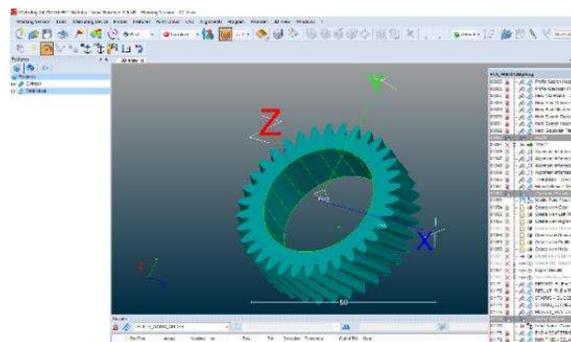


Fig. 2.11. Commercial software package used for gear analysis.

Repeatability within 25  $\mu\text{m}$  was observed when the gear was stationary, i.e., without removing and replacing the gear after each measurement. However, evaluating the accuracy is recognized to be challenging due to the complexity of the gear, and the presence of multiple factors that could create a bias. Therefore, a workpiece with simple geometry is selected, consisting of a cylindrical surface with a flat machined on it, and an inner diameter. Fig. 2.12 shows the measured point cloud of this part. The complete point cloud is not shown, but only the groups of points selected for each feature are shown, each group with a different color. The points in the outer diameter and the planes come from the first two sensors that view the part from the sides. The points in the inner diameter and the top surface come from the third sensor that views the part from the top.

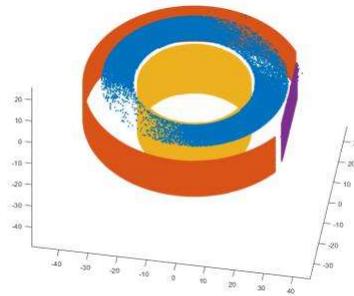


Fig. 2.12. Point cloud of test part.

A series of four independent calibrations are done, and at the end of each calibration, ten consecutive measurements of the test part are done. After each measurement of the test part, its position on the table is slightly moved manually to check for repeatability. In each case, a point cloud of approximately 500,000 points was produced. These large clouds are down-sampled by a factor of 50, and least-squares algorithms [18] are used to apply fits, and the results are shown in Table 2.2.

Table 2.2. Measurement results from test part.

Data set	Characteristic (CMM reference)	Mean (n = 10) mm	St. dev. (n = 10) mm
1	Outer Diameter (74.9577 mm) from Sensors 1&2	74.9539	0.0003
2		74.9541	0.0004
3		74.9542	0.0003
4		74.9531	0.0001
1	Inner Diameter (38.1745 mm) from Sensor 3	38.1877	0.0011
2		38.1865	0.0017
3		38.1867	0.0012
4		38.1861	0.0011

It can be seen that the correlation between the measured inner diameter and the reference value from CMM calibration is slightly less than that of the outer diameter. This bias may be due to slight inaccuracy in the third sensor's calibration result. This could be due to the influence of the physical orientation of the sensor relative to the surface on which the laser line is incident. In case of sensor 1 and 2, similar inaccuracies are observed at first, but these improve significantly on tilting the sensors such that the laser lines are along the direction of the helix, and running the calibration again after such adjustment. While tilting the third sensor slightly in random directions does show variations in these correlation results, the accuracy is not found to change significantly. This may indicate that the optimal orientation for this sensor is not experimentally found yet. Even so, the difference between the measured and reference value is about 12  $\mu\text{m}$ , which is reasonable to expect. It is also clear that there is very little variation from one calibration to the next, indicating that the calibration and subsequent measurement processes are stable.

## 2.7. AUTOMATION

The objective of automation here is to have the calibration and measurement processes automatically triggered and executed every time something is scanned on the machine. Every time the artifact is placed and the scan button is pushed, the calibration should automatically begin, and it should retrieve all the necessary inputs from the operating machine automatically and once the calibration is complete, and the calibration result file and other output files are generated, it is to signal to the machine that it is ready for the next scan. Similarly, when a gear is placed on the table and the scan button is pushed, it should automatically begin the measurement sequence using the latest available calibration file (or sensor pose information).

Such automation is achieved using an infinite loop that continuously monitors a given working directory (chosen by the user) for new work order files. These work order files are generated in this folder once any scan is completed and the scan files are available for use. Fig. 2.13 shows an example of a work order file. It contains the folder paths where all the input data necessary can be found, defines the output directory, and also allows setting parameters that define the extent of sub-sampling the scan files.

```

Mode;1
Part_Type;Artifact_Calibration
SerialNo;DEFAULT
Part_Folder;Y:\V3DCore\PARTS\Artifact_Calibration\
Artifact_File;fca.csv
Artifact_Folder;C:\Users\jpras\Documents\MATLAB\Vantage\Prash work\V3D_Processing\V3DCore\Artifacts\FCA\
Tooling_Set_Folder;C:\Users\jpras\Documents\MATLAB\Vantage\Prash work\V3D_Processing\V3DCore\ToolSets\FCA\
Scans_Folder;C:\Users\jpras\Documents\MATLAB\Vantage\Prash work\V3D_Processing\V3DCore\Artifacts\1-7 Scans\Scan 2\
Output_Folder;C:\Users\jpras\Documents\MATLAB\Vantage\Prash work\V3D_Processing\V3DCore\Artifacts\1-7 Scans\1-7 Scans\Scan 2\
Num_Sensors;3
Sensor1;Sensor1.csv
Sensor2;Sensor2.csv
Sensor3;Sensor3.csv
X_Skip;20
Angle_Skip;10
SensorTravelX; 0.00000
SensorTravelZ; 0.00000

```

Fig. 2.13. Example of a work order file.

Once a work order file is generated, the appropriate function (calibration or measurement) is triggered and begins processing. Once the processing is complete, the results (calibration result files in case of calibration, and point cloud files in case of measurement) are generated and a log file indicating successful completion, or failure, if any, is generated. This log file acts as a trigger to signal to the machine that it is ready for any new scans.

## 2.8. OUTCOMES AND FUTURE WORK

This work has developed a method for calibration of a CMS that measures a point cloud of any part in a short cycle time. The calibration procedure does away with the need for careful alignment of parts as well as sensors since any change in sensor position is identified and compensated for. The calibration procedure also establishes traceability as a calibrated artifact is used. The validity of the calibration results is verified by measurement of a test part. In the absence of test parts, such validation can also be performed by measuring the artifact itself, preferably after moving it to a slightly different location compared to the position at which it was calibrated.

Future work could involve extending this study to other artifact designs, or other approaches to perform calibration. Another avenue worth exploring is the development of methods to compensate for the sensor at the sensor level. For example, it was observed in many cases that when the sensor measures a line on the surface of a high-grade gauge block, the measured profile often shows curves (showing max deviations of about 30-40  $\mu\text{m}$ ) instead of straight lines. Further, these measurements also show that there is room for improvement in noise reduction or filtering methods.

## CHAPTER 3: DATA PROCESSING: MACHINE LEARNING APPROACH

### 3.1. INTRODUCTION

While the approach in Chapter 2 used a deterministic method of iteratively refining a set of parameters (that describe the pose of the calibration artifact and multiple sensors), until the point cloud resulting from those parameters matched closely with the known geometry of the artifact, it is unclear if this is the best approach. Although this procedure is found to work to the expected accuracy, the optimization process itself is a time-consuming process, especially when operating on very dense point clouds of the order of 2 million points. Further, there are various parameters that are set by the user during the process (such as GUI inputs), which have an influence on the calibration result. As a result, the process is not human-error proof. Further, different codes would need to be generated for each application that requires a different type of artifact, for example, artifacts that have two sets of V-grooves to accommodate measurement of wider gears. With such limitations, it becomes desirable to explore approaches that could provide better solutions to these issues. Literature provides quite a few examples of such inverse problems existing in various fields of research where machine learning models and deep learning techniques have been applied [19-21].

Given that a change in each of the parameters always influences the shape of the point cloud in a certain way that is geometry-dependent and not random, it may be possible to use learning methods to determine the set of parameters just by looking at the shape of the point cloud, assuming a machine learning. In order to facilitate this, the machine

learning model must first be trained with sufficient data such that it contains information on what the point cloud would look like for various sets of parameters. Subsequently, it may be able to predict what the parameters are for a given unknown point cloud by comparing the point cloud to the various point clouds present in the training data. Fig. 3.1 shows a schematic of such an approach.

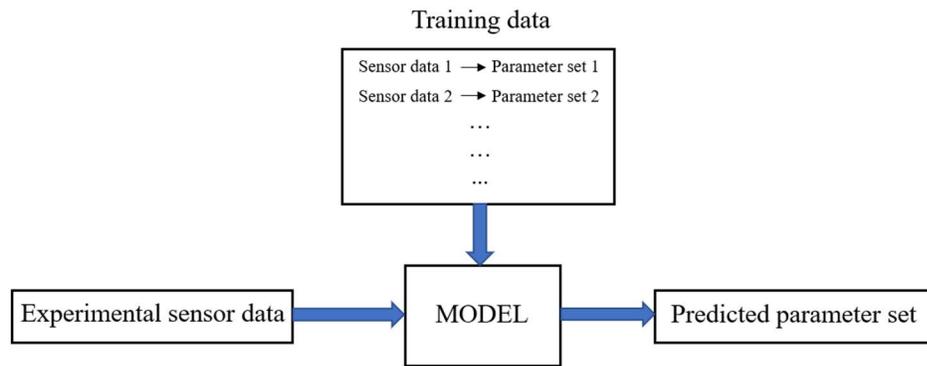


Fig. 3.1. Schematic of intended machine learning-based approach.

This chapter discusses a simpler two-dimensional version of the measurement system described in Chapter 2. The objective of this study is to assess the amenability of this CMS to machine learning methods. Simulation efforts to create training data or sensor data are described here. One approach to building the intended model shown in Fig. 3.1, is discussed here, namely using Dynamic Time Warping, or DTW [22] as a metric for comparing the unknown sensor data (sensor data for which the values of system parameters that generated them is unknown) to existing simulation data to determine the closest match or matches. In other words, the model becomes akin to a large look-up table, with DTW being the metric used to compare the unknown input data with existing entries in the look up table. Based on the closest matches identified, the multi-dimensional parametric space within which the solution is likely to lie, can be determined. Subsequently, a multi-

dimensional interpolation may be employed to determine the best estimate of the parameter set.

### 3.2. SIMULATION MODEL

Machine learning methods require extremely large volumes of sensor data (or training data) that is impractical to obtain by experimental methods due to resource limitations and physical restrictions. Therefore, simulation methods are used to generate this data.

For this study, a much simpler version of the experimental system described in Chapter 2 is modeled. Some of its differences from the experimental system are as follows.

- There is only one sensor instead of three.
- The sensor measures only a point instead of a line. So, at each encoder position (or rotation angle of the rotary axis), there is only one reading sensed by the sensor instead of an  $(x,z)$  height map. This reading, as simulated here, is the distance of the part (or point at which the laser is incident on the part) from the sensor.
- The sensor is constrained to lie in the horizontal plane, i.e.,  $dz$ ,  $a_x$ , and  $a_y$  parameters are zero. The part is also constrained to this plane. Therefore, the whole problem description is essentially reduced to a 2D problem.

For such a 2D case of the 3D problem described in Chapter 2, only three parameters ( $dx$ ,  $dy$ , and  $a_z$ ) are required here to fully describe the sensor's position and orientation (or pose), as shown in Fig. 3.2. The position of the part with respect to the axis is described by  $px$  and  $py$ . However, for such a nomenclature, rotating the sensor purely around the axis would cause the three sensor parameters to change, but the nature of the measurement

system does not really change. In other words, for the three cases shown in Fig. 3.2, the values of  $dx$ ,  $dy$ , and  $a_z$  will be different in each case, but sensed data will be the same, with the only difference being that the first data point is at a different location on the part.

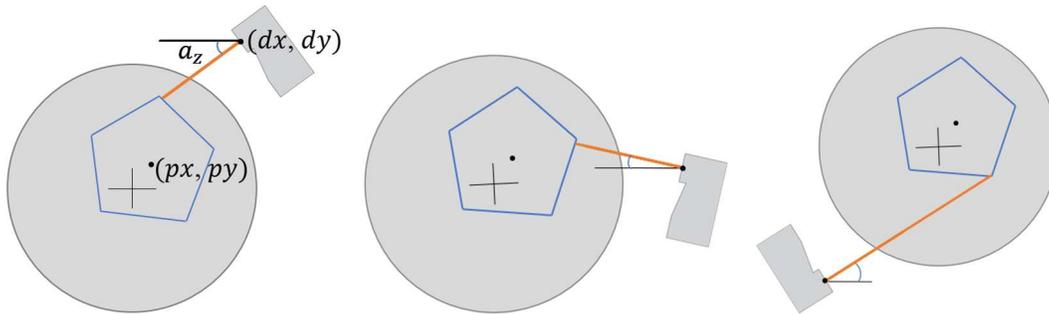


Fig. 3.2. Examples of redundant cases where different parameter sets are used to describe equivalent systems.

Therefore, to remove this type of redundancy in data sets, a slightly different nomenclature is used to describe the sensor parameters, as shown in Fig. 3.3. Here,  $px$  and  $py$  still represent the position of the part, while  $dx$  and  $dy$  of the sensor are replaced with  $d$  representing the perpendicular distance of the sensor from the rotation axis, and  $o$  representing the lateral offset by which the laser line is shifted from the axis.  $a_z$  retains its definition but is renamed to  $\theta$  for simplicity since there is only one angle involved.

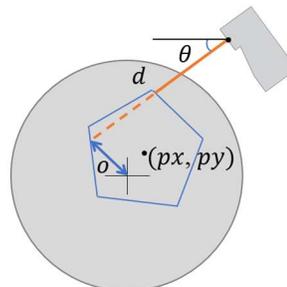


Fig. 3.3. New nomenclature for system description.

In this new nomenclature, the equivalent cases in Fig. 3.2 are represented by the same pair of  $d$  and  $o$  parameters.  $\theta$  is the only parameter that varies in these three cases, and therefore, redundancy is removed. The state of the system, including the part and the sensor, can be completely described by a set of 5 parameters ( $px$ ,  $py$ ,  $d$ ,  $o$ , and  $\theta$ ).

The artifact chosen for this simulation is an oddly shaped closed figure shown in Fig. 3.4 (a). Such an odd shape is chosen so that some parts of its surface are not visible to the sensor in certain configurations. Unlike the pentagon part shown in previous examples, this artifact is a better representation of reality due to possible missing data in recesses of real-world parts. Fig. 3.4 (b) and (c) show the regions of the artifact visible to the sensor in two different sensor positions.

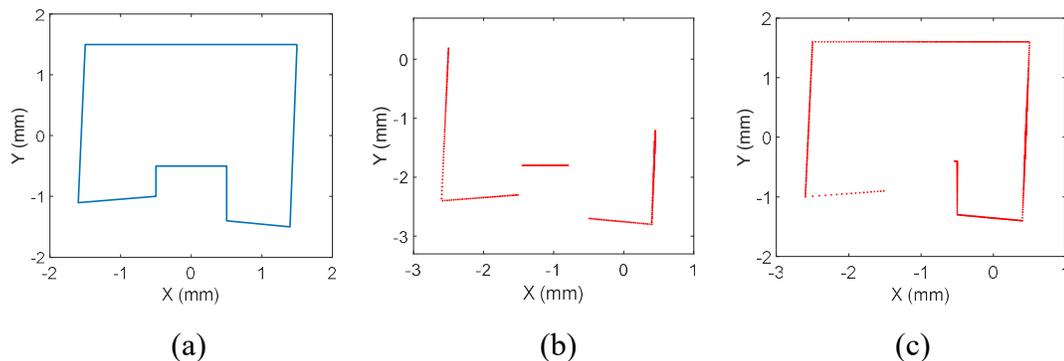


Fig. 3.4. Shape used as the artifact, and its regions viewed by the sensor in different configurations.

For each simulation, 720 table positions are considered, i.e., the table makes a full revolution in steps of half a degree. At each table position, the distance of the part surface from the sensor, or the length of the solid red line shown in Fig. 3.3 is recorded. Each combination of 5 parameters therefore produces a series of 720 data points or distance

values. Fig. 3.5 shows the curve obtained from one such one arbitrarily chosen combination of 5 parameters. This is taken to be the nominal parameter set.

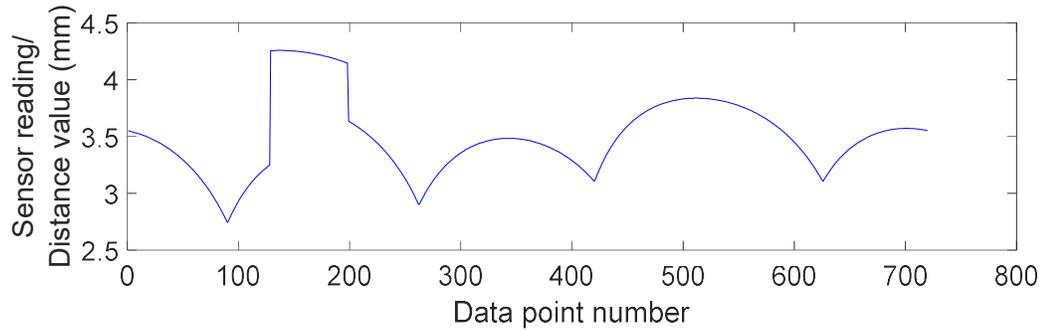


Fig. 3.5. Curve from nominal parameter set.

### 3.3. GENERATION OF LOOKUP TABLE

To generate a lookup table containing various combinations of these parameters within chosen limits, various levels of perturbations are made to each parameter in this nominal set. The nominal values in this set are shown in Table 3.1, and their perturbations are represented as  $\delta px$ ,  $\delta py$ ,  $\delta d$ ,  $\delta o$ , and  $\delta \theta$ . Each of these perturbation parameters assume 11 levels of perturbation (for example,  $\delta px$  takes values (-1, -0.8, ... , 1)).

Table 3.1. Nominal parameters and perturbations used for simulation.

Parameter	Nominal	Perturbation parameter	Min. perturbation	Max. perturbation	Perturbation step size
$px$	0 mm	$\delta px$	-1 mm	1 mm	0.2 mm
$py$	-0.3 mm	$\delta py$	-1 mm	1 mm	0.2 mm
$d$	5 mm	$\delta d$	-1 mm	1 mm	0.2 mm
$o$	-0.3 mm	$\delta o$	-1 mm	1 mm	0.2 mm
$\theta$	0°	$\delta \theta$	-2.5°	2.5°	0.5°

Each entry in the look up table has a set of these perturbation parameters and a curve similar to that shown in Fig. 3.5. For example, the first entry of the table contains the perturbation parameters  $(-1, -1, -1, -1, -2.5)$ . These perturbances are added to the nominal set  $(0, -0.3, 5, -0.3, 0)$ , and the resulting set of 5 parameters are used to generate the first curve in the lookup table. Since each perturbation parameter takes 11 values, the lookup table generated this way contains  $11^5$ , or 161,051 entries.

### 3.4. DYNAMIC TIME WARPING AS A METRIC FOR COMPARISON

Upon creation of a look-up table with various possible curves that could be generated within a given 5-dimensional domain space, a tool capable of comparing an unknown curve to the curves in the look up table, is necessary. Dynamic Time Warping (DTW) has been researched by Zhao [22], who has also proposed a variation of this algorithm, namely shape Dynamic Time Warping, or shape DTW. The DTW algorithm is traditionally used in aligning temporal sequences which may have distortions. This has been used in several fields of research including time series classification [23], speech recognition [24], human activity recognition [25], etc. It is essentially a point-to-point matching algorithm that compares two signals and estimates a distance measure between them, or the DTW distance, indicating the degree to which the signals match, making it appear as a promising candidate for this application.

Traditionally, a Z-normalization is performed on the curves before the DTW distance between them is estimated. This is done to ensure a constant offset between the two curves does not influence the result. However, for this application, such normalization would render the model insensitive to changes in the parameter  $d$ , since an increasing

distance of the sensor from the rotation axis causes the sensed values to increase accordingly, but the shape of the curve remains the same, provided the other parameters do not change. Such insensitivity to this parameter was initially observed when DTW was applied. Subsequently, with the removal of the normalization step, the model gained sensitivity to this parameter.

As an example, Fig. 3.6 (a) and (b) show two different curves generated using different parameter sets, and Fig. 3.6 (c) shows the point-to-point matching performed by the DTW algorithm. The blue lines connecting points on the red curve to those on the black curve indicate that the end points of these lines are determined to be matching. Once these matches are determined, a DTW distance is calculated as the output score or distance metric.

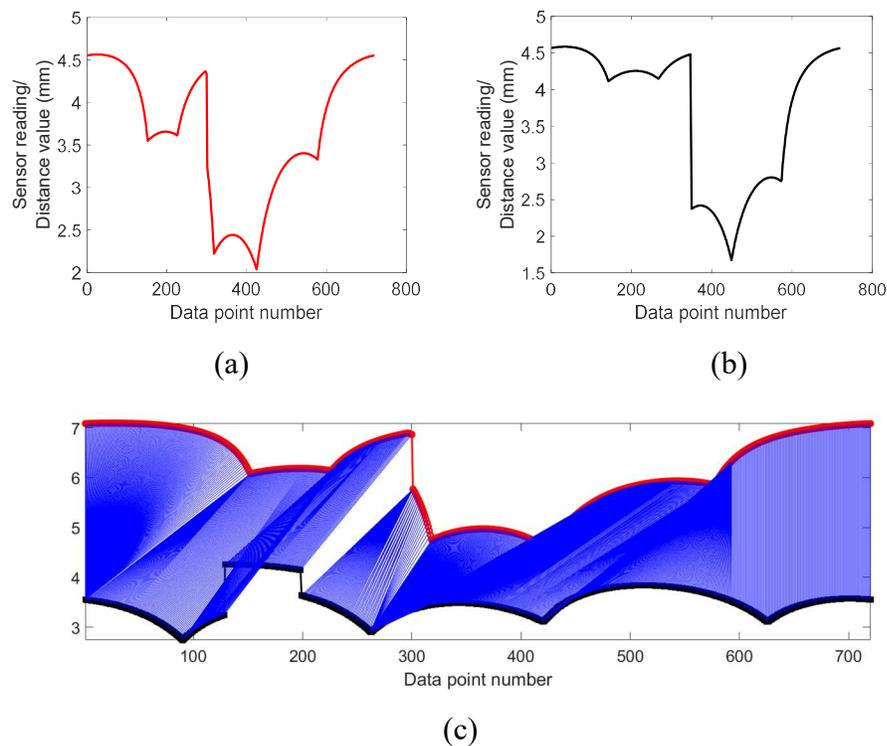


Fig. 3.6. (a) Sample curve 1, (b) Sample curve 2, and (c) their comparison using DTW.

A random set of 5 parameters is used to create a test curve that is compared to all the entries of the generated lookup table using DTW as the metric for comparison. The true values of these 5 parameters is listed in Table 3.2. The DTW distances obtained for each of the comparisons are shown in Fig. 3.7 (a), but the Y axis extends much higher than shown, to the order of 250,000 mm, and is therefore truncated. The minima of such a plot would indicate a very close match between the test curve and the curve in the lookup table corresponding to the minima location. Fig. 3.7 (b) shows a further blown up version and also shows the global minimum found. Also shown is the intuitive solution, which corresponds to the parameter set in the training data that is closest to the ‘true’ values, or the parameters used to create the test curve. In practice, this information is generally unavailable as these true parameters are unknown when the curve is experimentally obtained. This intuitive solution is merely shown to indicate that it is so close to the global minimum.

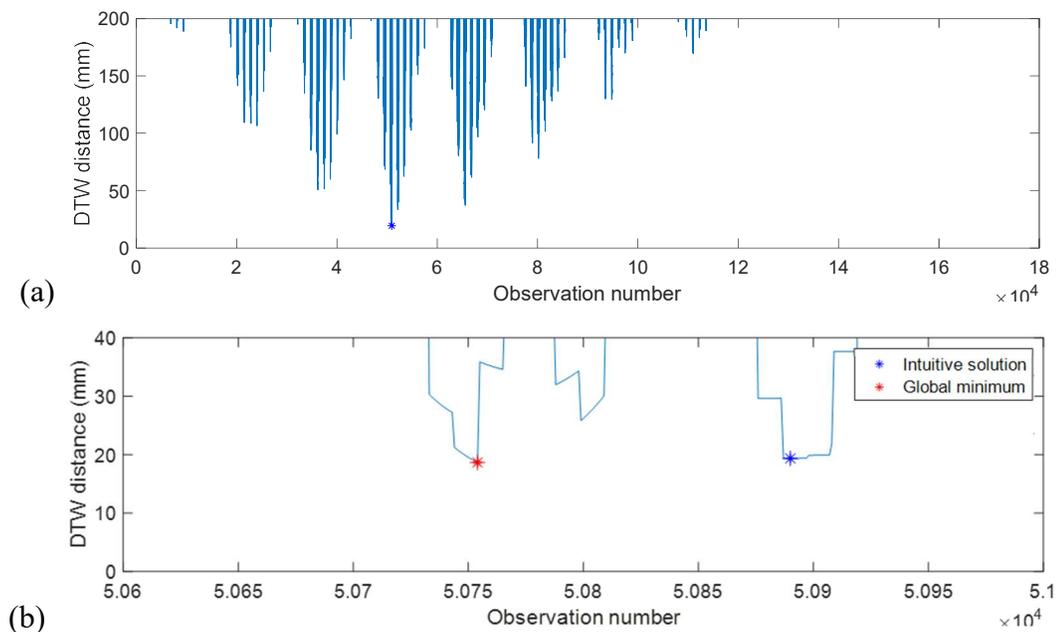


Fig. 3.7. (a) DTW distances obtained on comparing test curve with lookup table, and (b) blown up version showing Global minimum and Intuitive solution.

Localization of minima in the DTW distance curves shown in Fig 3.7 (a) and (b) is dependent on the ordering of the rows in the training data, or the order in which the parameters are perturbed. For example, the perturbation parameter in the inner most loop of data generation will undergo the full range of perturbation before the parameter in the immediately outer loop gets updated. As a result, the global minimum in such a curve is of more significance than local minima.

Consequently, the lowest 0.01% of the 161,051 points in the DTW curve (or the 16 points with the lowest DTW distance values) are identified as the minima of interest, and the 5 perturbation parameters in the lookup table corresponding to these minima are listed in Table 3.2 along with the true values of the parameters that generated the test curve.

Table 3.2. Perturbation parameters corresponding to the test curve and those of minima identified.

Point	$\delta px$	$\delta py$	$\delta d$	$\delta o$	$\delta \theta$
True values	-0.38	0.06	-0.67	0.2	-1.19
Minima 1	-0.4	0	-0.8	-0.2	0.5
Minima 2	-0.4	0	-0.8	-0.2	1
Minima 3	-0.4	0	-0.8	-0.2	1.5
Minima 4	-0.4	0	-0.8	-0.2	2
Minima 5	-0.4	0	-0.8	-0.2	2.5
Minima 6	-0.4	0	-0.6	0.2	-2.5
Minima 7	-0.4	0	-0.6	0.2	-2
Minima 8	-0.4	0	-0.6	0.2	-1.5
Minima 9	-0.4	0	-0.6	0.2	-1
Minima 10	-0.4	0	-0.6	0.2	-0.5
Minima 11	-0.4	0	-0.6	0.2	0
Minima 12	-0.4	0	-0.6	0.2	0.5
Minima 13	-0.4	0	-0.6	0.2	1
Minima 14	-0.4	0	-0.6	0.2	1.5
Minima 15	-0.4	0	-0.6	0.2	2
Minima 16	-0.4	0	-0.6	0.2	2.5

The row with the red outline is the global minimum and the row with the blue outline is the intuitive solution. The presence of the intuitive solution within the 16 lowest points, is encouraging and indicates the effectiveness of the DTW algorithm in narrowing down the possible solutions to within 0.01% of the large volume of solutions available in the training data set. It is to be noted that this intuitive solution is only an approximate solution. To obtain the exact (or close to exact) solution, a multi-dimensional interpolation among these 16 (or higher if needed) nearest neighbors is essential.

### 3.5. OUTCOMES AND FUTURE WORK

This work has shown demonstrated, by means of simulation, the suitability of machine learning for a simple two-dimensional measurement system where a part is rotated by a rotary axis and a 1-D sensor or a laser point scanner is used to scan the part as it makes a complete revolution. For such a system, training data has been generated, and DTW has been successfully applied as a means to perform a table lookup and identify the nearest neighbors in a multi-dimensional parameter space.

Possible future work includes developing a multi-dimensional interpolation method to obtain an exact solution from the available nearest neighbors. Further, the system can be extended to 3D. Also, alternate machine learning methods such as neural networks can be explored.

## CHAPTER 4: DATA MODELING: QUALITY INFORMATION FRAMEWORK

### 4.1. INTRODUCTION

Achieving interoperability in metrology requires use of data modeling techniques that support effective flow of metrology related data throughout the various stages of a product life cycle or manufacturing quality measurement process. With this goal, the Quality Information Framework (QIF) has been initiated by the Digital Metrology Standards Consortium (DMSC, previously known as the Dimensional Metrology Standards Consortium) [26]. This standard has been approved by ANSI (American National Standards Institute) and has also been adopted by the International Organization for Standardization (ISO) [27].

The QIF serves as an integrated, holistic set of information models and supports handling of relevant data throughout the entire measurement life cycle from design to planning to manufacturing to inspection [28]. It can be viewed as an enterprise-wide standard solution for the digital metrology community. There are several other perspectives from which this standard can be viewed. From a philosophy perspective, it is a feature-based ontology of manufacturing quality data. From a manufacturing standpoint, it is an effort to achieve interoperability issues withing quality and metrology [29]. From a software viewpoint, it is a data model defined by a series of XML (Extensible Markup Language) Schema Definitions, or XSD and demonstrated by a variety of tools that support the QIF standard [30].

Fig. 4.1 shows a schematic of the quality measurement process and shows how the several available QIF modules provide data helpful at each of the stages involved.

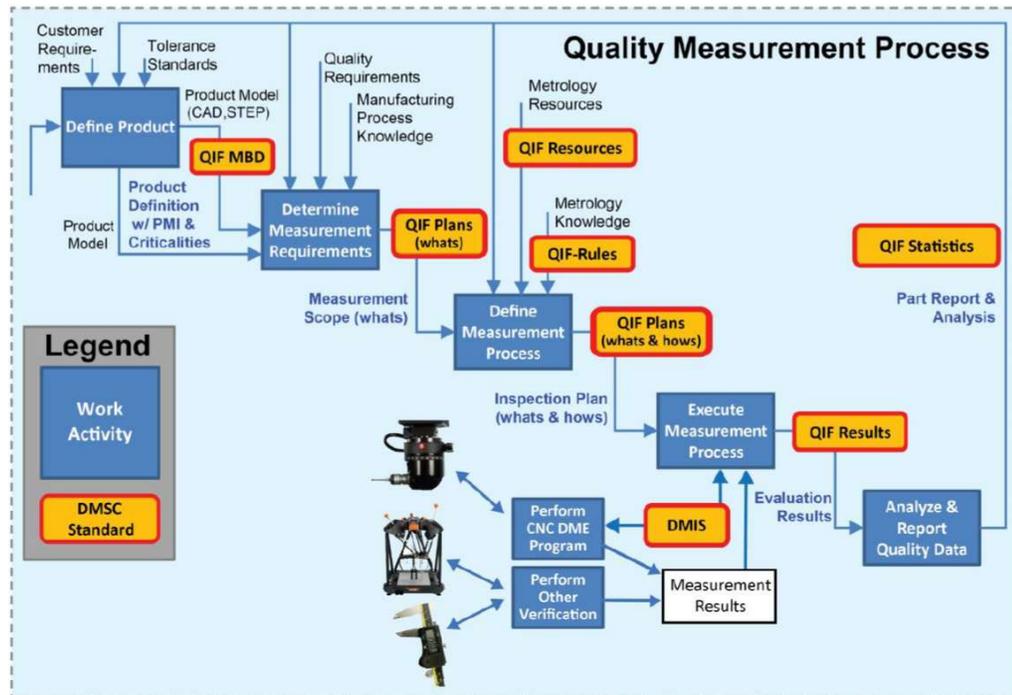


Fig. 4.1. Quality Measurement Process.

QIF provides a solution to digital interoperability, which serves as a foundation for Industry 4.0. One of the distinct advantages of QIF comes from a feature named QIF Persistence Identifier (QPID), which adds a unique tag to each information element. This allows any feature to be tracked throughout the life cycle. For example, a feature tolerance specified in the design can be tracked through manufacturing, inspection, and statistical results, and if necessary, this information can be used to modify the design specifications. All QIF schemas are in XML format, which enables them to be read by machines as well as computers. The information in each schema can thus be readily consumed downstream, which is especially useful for automation. Another inherent advantage of using this XML format is that the resulting QIF files can be automatically validated, ensuring that only valid documents that conform to the rules defined by the schema can be generated.

## 4.2. QIF APPLICATION AREAS

There are six distinct topical areas within the QIF, namely the QIF Model Based Definition (MBD), QIF Plans, QIF Resources, QIF Rules, QIF Results, and QIF Statistics. These are shown in Fig. 4.2 and their roles in the product life cycle/ quality measurement process are briefly described below. As seen from Fig. 4.2, information is exchanged among these schemas, all of which have access to common elements captured in the form of a QIF Library.

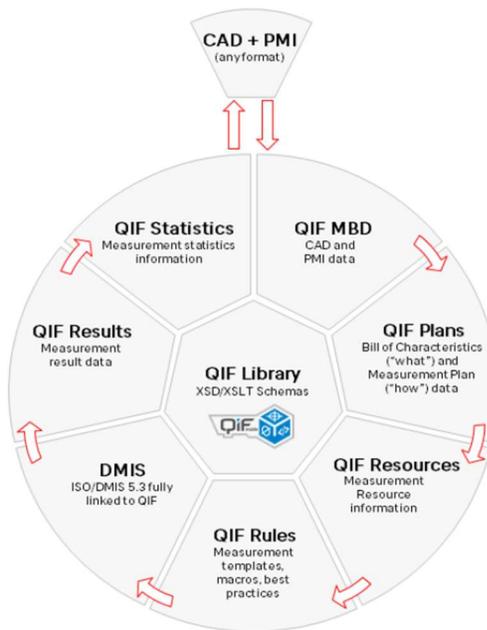


Fig. 4.2. QIF Application areas.

The QIF data flow typically begins with QIF MBD schema, which includes Computer Aided Design (CAD) and Product Manufacturing Information (PMI). This information is then consumed by the Quality planning systems that generate measurement requirements in the form of 'what' to measure (a part of QIF Plans). QIF Resources contain information on available metrology resources. QIF Rules contain a framework for two

kinds of rules, namely Dimensional Measurement Equipment (DME) selection rules and measurement strategy rules. The former set of rules help in selecting a measurement resource for a given metrology task at hand, while the latter set aids in determining how the measurement itself is to be carried out, including information on what point density to use, how many scans to perform, etc. The information from QIF Resources and QIF Rules pour into the ‘whats’ to generate the QIF Plans, containing complete inspection plans that include “how” to measure and verify. Following this, the Plans are imported to generate DME-specific programs, or generic instructions to guide the inspection process. Upon program execution, the characteristics of a single manufactured part or assembly are evaluated, and the measurements are exported as QIF Results. Such single part Results are used to generate analysis of multiple part batch Results as QIF Statistics data. This is done by Analysis systems, typically performing statistical process control.

It is to be noted that none of these schemas contain any prescribed methods or standard operating procedures themselves. Rather, each of these schemas simply contains a template for documenting the relevant information. For example, QIF Rules do not contain any rules themselves, such as prescribed rules on how to select point density, but provides users with the framework necessary to define their own rules on how to select point density based on what their enterprise deems as best practices.

### 4.3. QIF RESOURCES

This work will focus on QIF Resources. The primary objectives of QIF are:

- To capture sufficient information on all metrology resources available in the enterprise.

- To serve as an input to QIF Rules to help in selecting the DME and determining the measurement strategy for a particular task at hand.

A hierarchy has been previously developed to classify and accommodate various metrology resources [31]. The hierarchy, shown in Figs. 4.3, consists of DME, tools, and sensors, along with descriptive attributes for each resource.

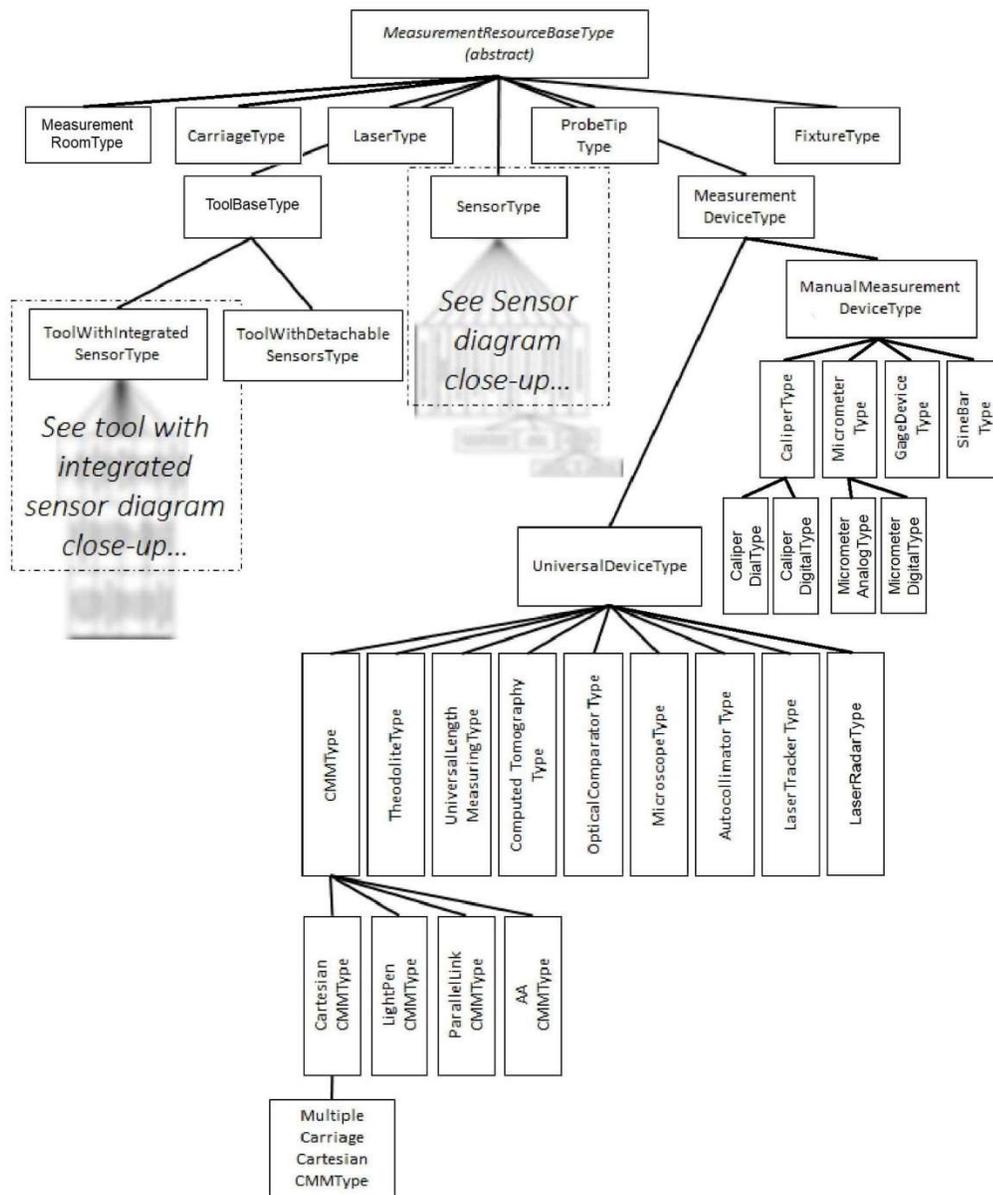


Fig. 4.3. QIF resources inheritance diagram.

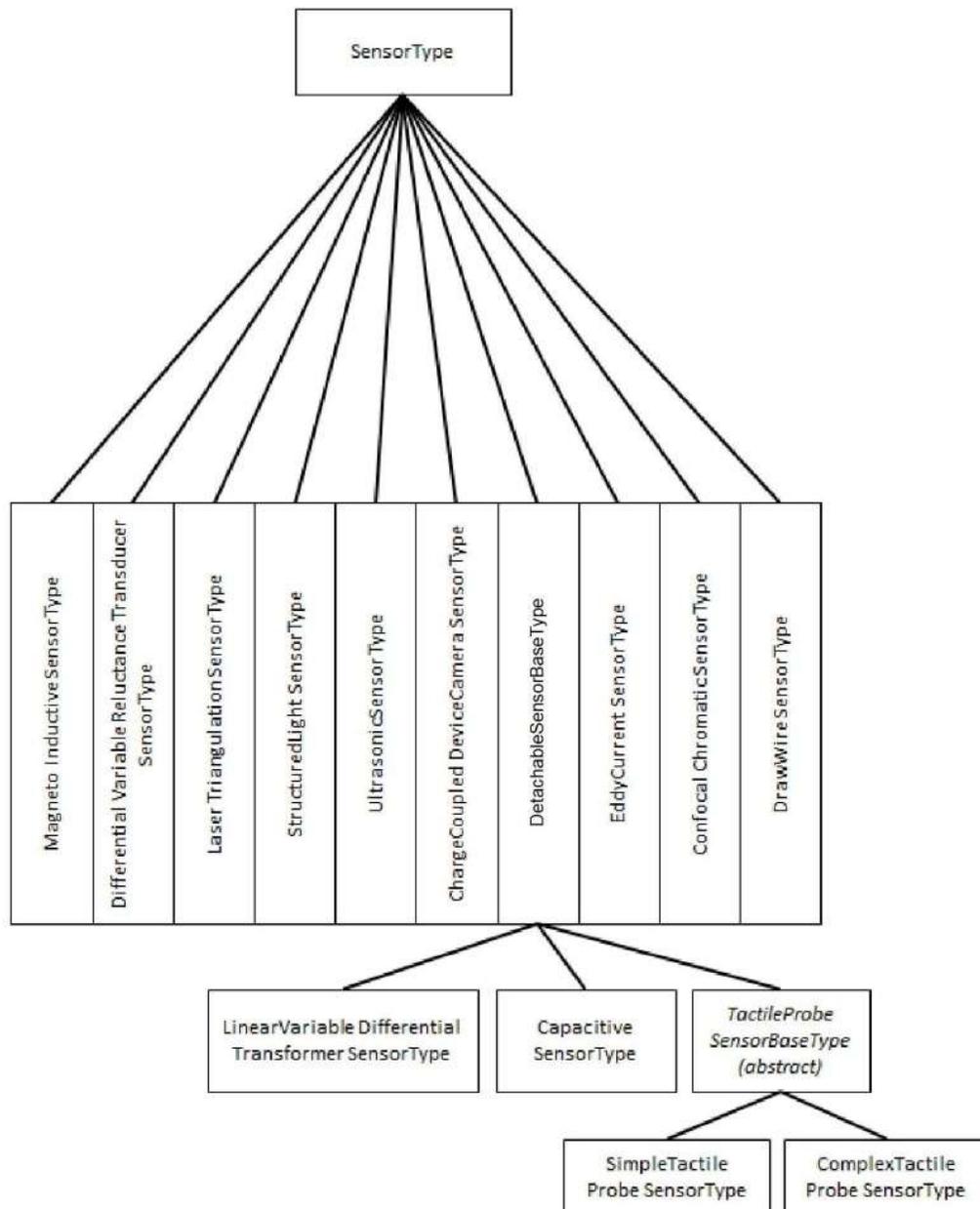


Fig. 4.4. Close-up of sensor hierarchy.

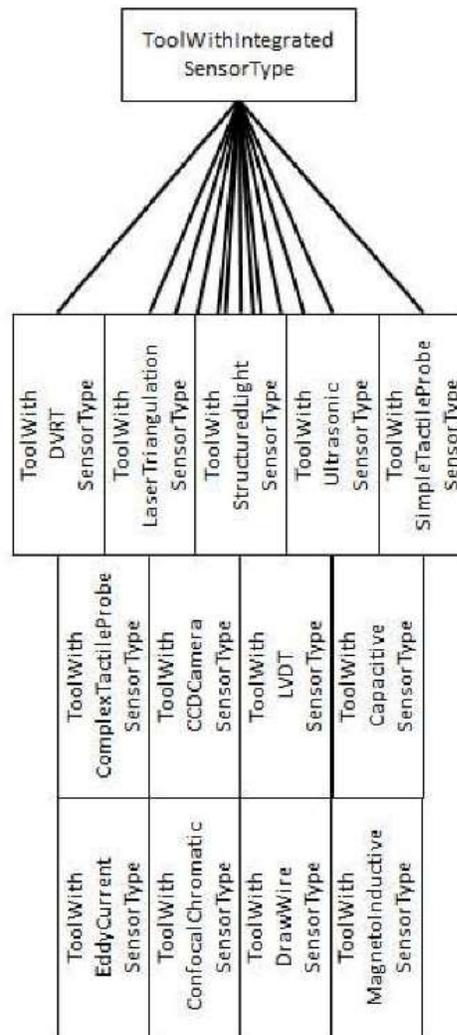


Fig. 4.5. Close-up of Tool with integrated sensor hierarchy.

As depicted in Figs. 4.3 to 4.5, the schema provides the logical space for accommodating various resources, and for important attributes relating to the specific resource. For example, the supported attributes for a simple resource, the caliper, are shown in Fig. 4.6.

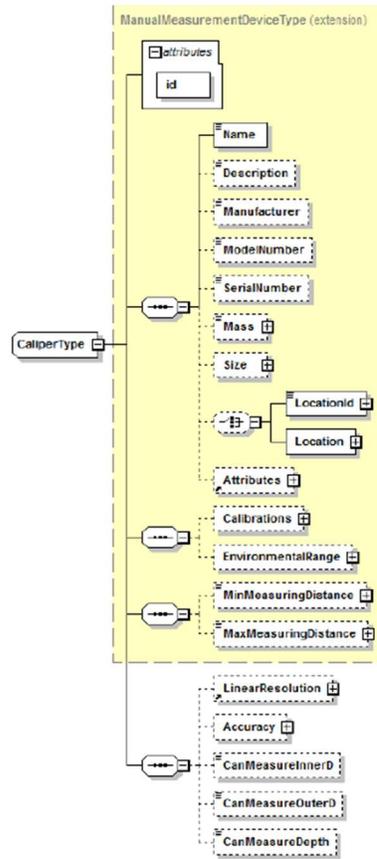


Fig. 4.6. Attributes of a Caliper.

It is to be noted that, in this case, the caliper has some attributes grouped as 'ManualMeasurementDeviceType', and some attributes outside of this grouping. This is because the hierarchy defines each resource as an extension of more general types, wherever possible. Thus, each resource has its own attributes as well as attributes inherent from parent types. In the case of the caliper, the attributes grouped together are those inherited as a result of being an extension of a manual measurement device. The attributes outside this grouping are those unique to the caliper. As another example of the hierarchical structure, a cartesian CMM is defined as an extension of a CMM, which is an extension of a Universal Device, which is an extension of a Measurement Device. At each node of such

a chain, there are special attributes for that node, as well as attributes inherited from the parent node.

#### 4.4. OBJECTIVES AND SCOPE

With the long term objective of the development of QIF Resources schema being to provide the ability to capture a metrology company's entire instrument catalog in the standard format [29], this work attempts to take a step toward that goal by developing an application software that allows users to populate a QIF Resources file via a GUI, i.e., without needing to have a deep knowledge of XML file structure and schema definitions. By providing a user-friendly platform to catalog all the metrology instruments and resources available in an enterprise, it is the hope that such an application would encourage and ease adoption of the QIF standard by prospective industries.

Some features targeted in the development of this application include the ability to

- Add, edit, and update information on several metrology resources.
- Add multiple similar devices at once, when they share similar attributes except for unique identifiers (such as serial number)- For example, if an industry has 100 micrometers, it is undesirable to have to input all the information each time. Rather, the application would allow creating all 100 at once, and provide the ability to edit just the serial number, or other attributes, if necessary.
- Interface with a local database to maintain permanent records of all resources and dynamically updates all underlying information any time a change is made through the app.

- Create instances of classes (objects) from QIF schema documents using XML serializing tools. Specifically, this application uses C# binding tools.
- Create a valid QIF file that captures all the resource information available in the created class instances, thereby creating a QIF Resource instance containing the descriptions and attributes of all metrology resources added via the GUI.

Thus, the overarching goal is to provide the ability to catalog all the resources sufficiently in the standardized format, so as to enable easy and automated consumption downstream the product life cycle. This may include automated DME selection and QIF Plan creation, etc. Fig. 4.7 depicts an example of a very simple QIF Resources instance containing only two micrometers.

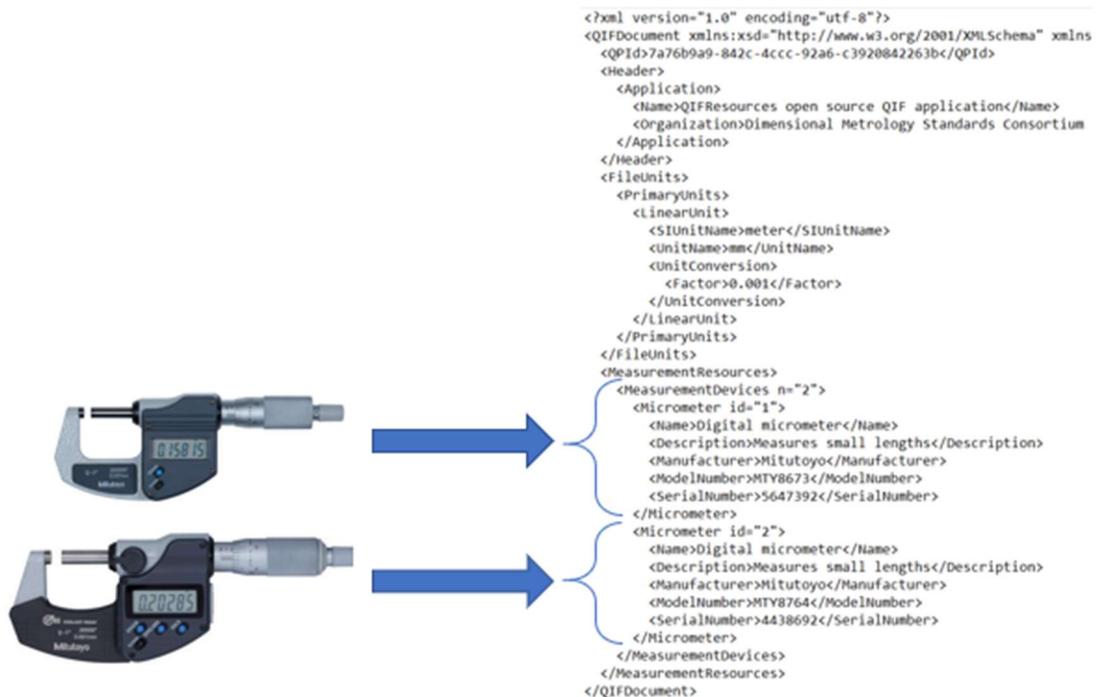


Fig. 4.7. Example of capturing resources in a QIF Resources file.

#### 4.5. APPLICATION SECURITY

As with creating any application, the first step is to ensure safety by allowing access only to authorized users. For testing purposes, a simple database is created containing random information on authorized users as shown in Table 4.1. This database is unavailable for direct viewing through the app by any user for security reasons. While the database may be viewed by accessing it from the file structure, this type of access can be independently restricted by a supervisor in the same way that other sensitive information in the enterprise is hidden from other employees/ users. In other words, the app does not allow a user to bypass the kinds of security that may already be in place and view this database through its interface though other databases including resource lists, etc. can be accessed and viewed directly through the app.

Table 4.1. Simple database storing authorized user data.

ID	First Name	Last Name	Username	Password
1	A	A	1	1
2	B	B	2	2
3	C	C	3	3
4	D	D	4	4

Upon launching the app, the login screen shown in Fig. 4.8 pops up. Username and password combinations entered here are validated against the database shown in Table 4.1. Access is granted when a matching pair is available. To further protect user data, any character entered in the password field will be masked with an asterisk, which is a common method used by several applications and digital platforms to address the need to hide the password from plain view.

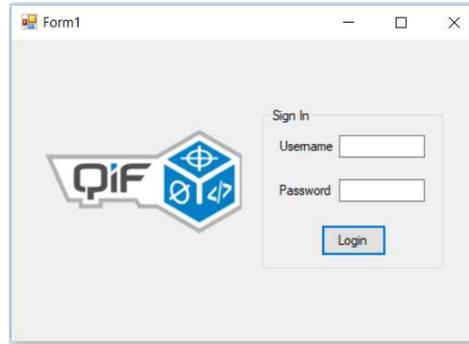


Fig. 4.8. Login screen.

Upon requests from prospective implementers, deeper levels of security such as hack-proofing were briefly explored, but these were soon deemed to be beyond the scope of this work.

#### 4.6. INTERFACING AND UNDERLYING PROCESSES

This section describes the processes involved in creating a QIF Resources file using the app. After a user logs in successfully, the homepage of the app is displayed, and this is shown in Fig. 4.9. This page is designed to give the user the option to add a desired resource to a current session. The drop-down list shows the options currently supported.

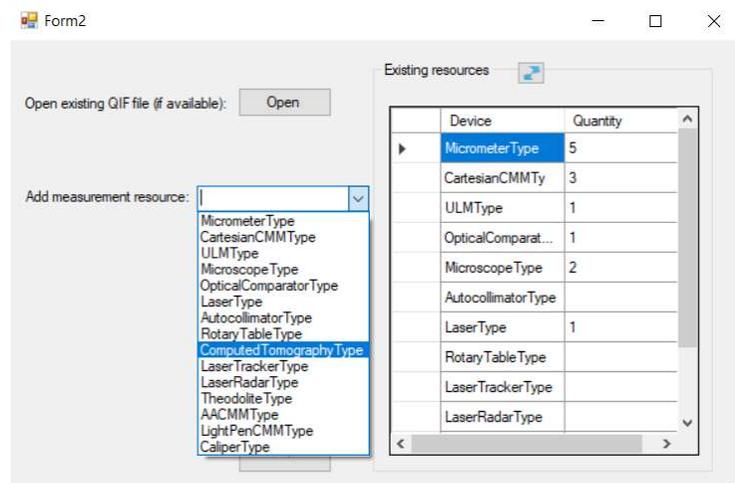


Fig. 4.9. Home screen.

The user is also given the option to open an already existing QIF Resource file and continue editing it. The right half of the homepage displays an overview of the currently available resources or resources added thus far via the app. This inventory is linked to an underlying database that monitors the resources and gets updated in real-time any time a change is made, or a new resource is added, updated, or removed.

When a user selects a resource to be added, a separate form dedicated to that resource type is displayed, containing fields to enter all relevant attributes or checkboxes to make choices when the attributes are logical/ Boolean in nature. For example, selecting Computed Tomography (CT) Type from the drop-down list box displays the form shown in Fig. 4.10, where a CT system can be added to the inventory. Each such form contains a database view of the list of that resource. For example, the database in Fig. 4.10 shows the list of CT systems currently available. These databases also get updated dynamically.

ComputedTomographyForm

\*\* Required fields \* Required fields only if parent type is chosen to be represented  
For parameters under same sub category, please use consistent units where applicable

Basic Info

Id\*\* 1

Name\*\* Zeiss Metrotom

Description

Manufacturer Zeiss

Model number 800/130KV

Serial number

Mass 5700 kg

Size X Axis Length 2190 mm

Y Axis Length 1310

Z Axis Length 1960

LocationId  Location  No Location specification

LocationId

xid

asmPath

id\* Location Description\*

Address Description

Internal Location

Street Number

Street

Postal Box

Town

Country

Add/Next Update Delete Cancel

id	Name	Desc
1	Zeiss Metrotom	

Fig. 4.10. Computed Tomography Form.

In the programming world, classes contain the definitions of what a class object should look like. They contain the definitions of all the member variables and functions. Similarly, in the QIF world, the schema is the entity that dictates what an XML file should look like. It contains the definitions of all the attributes that make up various elements of the file. In order to create a valid XML file that conforms to the schema, the application must be able to decipher the schemas, or serialize them such that all the elements of the schema are equivalently represented as classes. In other words, interconversion between schema and class is essential prior to being able to convert a class object to an XML file (often termed as Marshalling), or an XML file to a class object (Unmarshalling). Fig. 4.11 depicts this relationship.

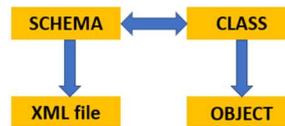


Fig. 4.11. Relationship between XML files and class objects.

Such Marshalling/ Unmarshalling requires the use of XML code binders, which are unique to each programming language, as each language will need a different method of translation. XML code binders for C# are used in this work. Continuing with the example of a CT system, Fig. 4.12 (a) shows a snippet of some attributes defined in the schema, and Fig. 4.12 (b) shows the attributes after serialization, i.e., translation to classes. It is to be noted that although this is the most time-consuming step, it is a one-time process only. These translated classes are already present in the application, and this translation does not need to take place during each attempt at creating a new QIF file.

<pre> &lt;xs:complexType name="ComputedTomographyType"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;       The ComputedTomography element is a computed tomography       measuring machine.     &lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:complexContent&gt;     &lt;xs:extension base="UniversalDeviceType"&gt;       &lt;xs:sequence&gt;         &lt;xs:element name="SourceDescription" type="xs:string" minOccurs="0"&gt;           &lt;xs:annotation&gt;             &lt;xs:documentation&gt;               The optional SourceDescription element describes the source               used for the computed tomography.             &lt;/xs:documentation&gt;           &lt;/xs:annotation&gt;         &lt;/xs:element&gt;         &lt;xs:element name="SamplePenetratingMaterial" type="xs:string" minOccurs="0"&gt;           &lt;xs:annotation&gt;             &lt;xs:documentation&gt;               The optional SamplePenetratingMaterial element is a               material type for a penetrating test with the length               defined in the SamplePenetratingMaterial.Length element.             &lt;/xs:documentation&gt;           &lt;/xs:annotation&gt;         &lt;/xs:element&gt;         &lt;xs:element name="SamplePenetratingMaterial.Length" type="LinearValue" minOccurs="0"&gt;           &lt;xs:annotation&gt;             &lt;xs:documentation&gt;               The optional SamplePenetratingMaterial.Length defines the               penetrating length for the sample used in               SamplePenetratingMaterial element.             &lt;/xs:documentation&gt;           &lt;/xs:annotation&gt;         &lt;/xs:element&gt;         &lt;xs:element name="FocalSpotSize" type="LinearValueType" minOccurs="0"&gt;           &lt;xs:annotation&gt;             &lt;xs:documentation&gt;               The optional FocalSpotSize element defines the               focal spot size of the X-ray tube.             &lt;/xs:documentation&gt;           &lt;/xs:annotation&gt;         &lt;/xs:element&gt;       &lt;/xs:sequence&gt;     &lt;/xs:extension&gt;   &lt;/xs:complexContent&gt; &lt;/xs:complexType&gt; </pre>	<pre> public partial class ComputedTomographyType : UniversalDeviceType {     private string sourceDescriptionField;     private string samplePenetratingMaterialField;     private LinearValueType samplePenetratingMaterialLengthField;     private LinearValueType focalSpotSizeField;     private MassValueType maxWorkpieceMassField;     private string safetyLevelField;     private CartesianWorkingVolumeType cabinetDimensionsField;     private UserDefinedUnitValueType maxPowerField;     private decimal detectorBitNumberField;     private bool detectorBitNumberFieldSpecified;     private LinearValueType detectorPixelSizeField;     private FunctionDiscreteType activePixelsNumberField;     private UserDefinedUnitValueType maxKVField;     private UserDefinedUnitValueType minOperatingCurrentField;     private UserDefinedUnitValueType maxOperatingCurrentField;     private UserDefinedUnitValueType maxFramePerSecondField;     private RotaryTableType rotaryTableField; } </pre>
(a)	(b)

Fig. 4.12. Snippets showing attributes in (a) Schema and (b) class definitions.

Once these class definitions are created, the information stored in the database for each resource can be converted into valid class objects for each of the resources in the application's native language, C# in this case. Subsequently, valid QIF files can be created by the process of Marshalling previously described. Fig. 4.13 shows a simple QIF Resource file generated for the CT example.

Ideally, it is desirable for an industry to have only one QIF Resource instance file that completely represents all their measurement resources. However, since this application is a beta version intended for testing, it supports creation of multiple independent QIF Resource file instances.

```

<MeasurementResources>
  <MeasurementDevices n="1">
    <ComputedTomography id="1">
      <Name>Zeiss_Metrotom</Name>
      <Manufacturer>Zeiss</Manufacturer>
      <ModelNumber>800/130KV</ModelNumber>
      <Mass massUnit="kg">5700</Mass>
      <Size>
        <XAxisLength linearUnit="mm">2190</XAxisLength>
        <YAxisLength linearUnit="mm">1310</YAxisLength>
        <ZAxisLength linearUnit="mm">1960</ZAxisLength>
      </Size>
      <EnvironmentalRange>
        <MaxAmbientTemperature temperatureUnit="C">22</MaxAmbientTemperature>
        <MinAmbientTemperature temperatureUnit="C">18</MinAmbientTemperature>
        <MaxAmbientRelativeHumidity>0.7</MaxAmbientRelativeHumidity>
        <MinAmbientRelativeHumidity>0.4</MinAmbientRelativeHumidity>
      </EnvironmentalRange>
      <CartesianResolution>
        <CombinedCartesianResolution linearUnit="µm">3.5</CombinedCartesianResolution>
      </CartesianResolution>
      <CartesianWorkingVolume>
        <XAxisLength linearUnit="mm">700</XAxisLength>
        <YAxisLength linearUnit="mm">270</YAxisLength>
        <ZAxisLength linearUnit="mm">270</ZAxisLength>
      </CartesianWorkingVolume>
      <FocalSpotSize linearUnit="mm">0.005</FocalSpotSize>
      <MaxWorkpieceMass massUnit="kg">5</MaxWorkpieceMass>
      <DetectorPixelSize linearUnit="µm">127</DetectorPixelSize>
      <MaxKV unitName="KV">130</MaxKV>
      <MaxOperatingCurrent unitName="µA">300</MaxOperatingCurrent>
    </ComputedTomography>
  </MeasurementDevices>
</MeasurementResources>

```

Fig. 4.13. QIF Resources instance file.

#### 4.7. QIF SHORTCOMINGS IDENTIFIED

In the process of designing this application, a few shortcomings in the standard have been identified. Some of these are listed below and could be useful to consider in the next iteration of the standard.

- The QIF standard has limited representation of surface texture characteristics.
- Some important features of sub-components are not associated with the sub-component itself but only with the instrument using it. For example, a LaserTrackerType includes a CalibrationType that describes the calibration details for the laser tracker. An integral part of the laser tracker, that determines how well it is calibrated, is its laser. The laser is described by the attribute LaserType, and does not have a CalibrationType of its own and relies on the tracker's calibration

attributes. However, a laser can be used in a stand-alone fashion independent of other instruments using it. To describe such a stand-alone laser in the QIF format requires recycling the same definition of LaserType which is used as attributes in other larger instruments that have a laser built in. As a result, this stand-alone LaserType will have no CalibrationType.

- Most parameters for any given resource are fixed and do not accept ranges of values (or an expected deviation from a given value)- For example, LaserWavelength is an attribute that accepts only a constant value. However, real world lasers always have a range, and it may be necessary to accommodate such ranges in the standard. If ranges cannot be accommodated, the ability to include an uncertainty associated with the constant quantity (in this example, wavelength) may be included.
- Uncertainty modeling needs to be improved upon, especially in the case of task-specific uncertainty, which would play an important role in DME selection.

#### 4.8. OUTCOMES AND FUTURE WORK

An application software capable of has been created to help users and industries to document their metrology resources in the QIF format, and a beta version of the same has been released. It is hoped that these efforts contribute to decreasing the inertia involved in industrial implementation of this standard. Future work could involve expanding the application and underlying resource class definitions to include more resource types and more subtle aspects of the schemas, as well as updating the schemas themselves. Further, some limitations of the standard have been identified for possible changes to be made in upcoming versions of the standard.

## CHAPTER 5: ERROR ANALYSIS: X-RAY COMPUTED TOMOGRAPHY

### 5.1. INTRODUCTION

The use of X-Ray Computed Tomography (XCT) in industrial metrology has been steadily increasing in recent years as a means for inspection. This is especially true for complex parts where traditional inspection procedures would prove time consuming, or parts with internal features that simply cannot be accessed by contact-based measurement methods. These distinct advantages in XCT, among other benefits have brought about the transition from its predominant use in defect inspection and medical imaging, into dimensional measurements for engineering applications. As a result of such increasing demand and widespread use, there arises a need for development of standardized test procedures to evaluate XCT instrument performance and to support claims of metrological traceability. Along these lines, the International Organization for Standardization (ISO) and the American Society of Mechanical Engineers (ASME) have been working independently to develop XCT performance evaluation standards. Current drafts of these standards advocate testing at different measurement volumes (which can be realized by changing the positions of the rotation stage and detector), but do not provide specific guidance as to how the user should select the locations of these measurement volumes or all of the test positions within these volumes [32-33]. The lack of such specific information in the standards could partly be due to the cumbersome nature of the task of testing various possible scenarios by extensive radiographic testing or by experimental methods. The work

described in this chapter help in overcoming this impractical workload of simulations since much faster simulation methods are adopted.

In the development of performance evaluation standards, one of the primary goals is to design test procedures that are sensitive to as many known error sources as possible. To achieve this, the first step becomes identifying all possible error sources and understanding the effect of each individual error source on the resulting dimensional measurements. For XCT systems, several error sources have been identified and discussed in the VDI/VDE 2630-1.2 [34] and by Ferucci et al [35]. Kumar et al. tried to quantify the effect of XCT geometry errors on dimensional measurements [36]. Ametova et al. developed a novel method to determine the influence of instrument geometry errors by comparing the boundaries of the image after forward projection in the presence of instrument geometry errors against those obtained for a perfect system [37]. Aloisi et al. determined the effect of detector angular errors (tilt) on sphere center-to-center distance errors on a scale bar oriented vertically [38]. In many of these works, the methods presented are effective in determining the influence of the errors for the specific cases chosen, but these works did not identify the most sensitive positions of the scale bar for individual geometry error sources.

The National Institute of Standards and Technology (NIST) has undertaken studies [39-40] on the influence of uncorrected machine geometry errors in cone-beam XCT instruments. In that work, the sphere center-to-center distance error sensitivities and sphere form error sensitivities to various geometrical error sources were studied for a fixed measurement volume, i.e., for chosen fixed positions of the detector and the rotation stage.

The research described in this chapter is an extension of the work done by Muralikrishnan et al. [39-40] by repeating the simulations for several positions of the stage (defined by source-stage distance or source-object distance) and the detector (defined by source-detector distance) [41]. Fig. 5.1 shows an overview of this approach.

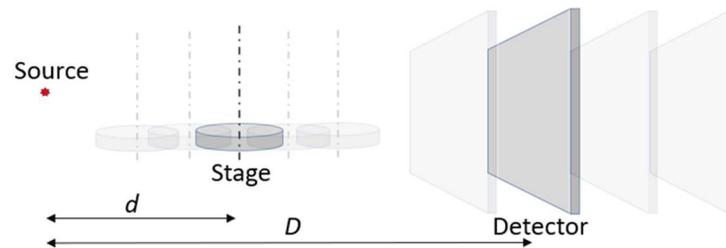


Fig. 5.1. Overview of simulations for different magnifications.

For each error source, the variation of sensitivities with these distances are studied, and the combinations that produce the highest sensitivities are identified. The geometric magnification established by such a combination is given by the ratio of the source-detector distance to the source-stage distance. The dimensions of the measurement volume are also established by this combination of distances. The position and orientation of the measurement line within this measurement volume that produces the highest distance and form error (and therefore, the highest sensitivity) is identified. Based on the results of simulation, a minimum set of measurement locations (defined by stage and detector positions) and a minimum set of test positions (defined by location and orientation of measurement lines within the measurement volume) are proposed.

## 5.2. SIMULATION SETUP AND MODELING

All simulations in the present work are carried out using a simulated reference object consisting of 125 spheres evenly distributed into five horizontal planes, each containing one centrally located sphere, 16 spheres arranged in a circular pattern, and 8 spheres arranged in a smaller circle of half the diameter of the outer circle, as shown in Fig. 5.2. This is the same sphere arrangement described by Muralikrishnan et al [39]. It is to be noted that in Fig. 5.2, the source and detector positions are drawn to scale, but detector size is not. The world coordinate system used in this model has its center at the source, and has its axes directed as shown in Fig. 5.2. Detailed descriptions on establishing the coordinate system by defining each axis, are provided by Muralikrishnan et al [39]. While the previous studies considered the object to be of a single fixed size [39-40], here, the height and diameter of the overall cylindrical shape are functions of the source-stage and source-detector distances. In other words, the simulated object is scaled for each combination of source-stage and source-detector distance combination so that 98% of the area of a 250 mm x 250 mm detector gets filled. The purpose of such scaling is to obtain the largest possible magnitudes of distance errors for each combination of stage and detector positions. The sphere diameters are also scaled accordingly (ranging from 3.40 mm to 18.68 mm at the smallest and largest measurement volume respectively) while ensuring that their projected images still fit on the detector.

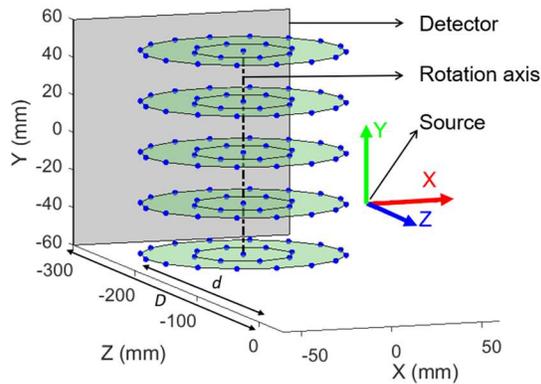


Fig. 5.2. Reference object containing 125 spheres chosen for simulation.

The geometric error sources associated with the detector and stage are studied here. The detector errors include the three location errors along the three mutually perpendicular axes, and three angular positioning errors about the same axes. These are shown in Fig. 5.3. The stage errors include a Z location error (along the axis connecting the source and rotation axis), and the error motions of the stage. These error motions, including encoder scale, axial, radial, and wobble errors, are all assumed to have harmonic components and therefore are represented as sine and cosine functions of the rotation stage indexing angle. This study includes harmonics of orders one through ten. These stage errors are shown in Fig. 5.4. Detailed descriptions of these error sources can be found in Ferrucci et al. [42-43] and Muralikrishnan et al [39-40].

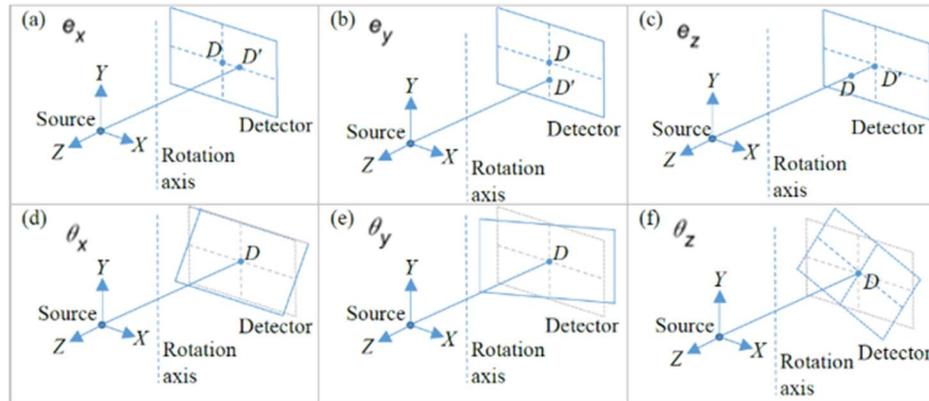


Fig. 5.3. Geometrical error sources associated with the detector. (a) Detector location error parallel to X axis. (b) Detector location error parallel to Y axis. (c) Detector location error along the Z axis. (d) Detector rotation error about an axis parallel to X axis. (e) Detector rotation error about an axis parallel to Y axis. (f) Detector rotation error about the Z axis

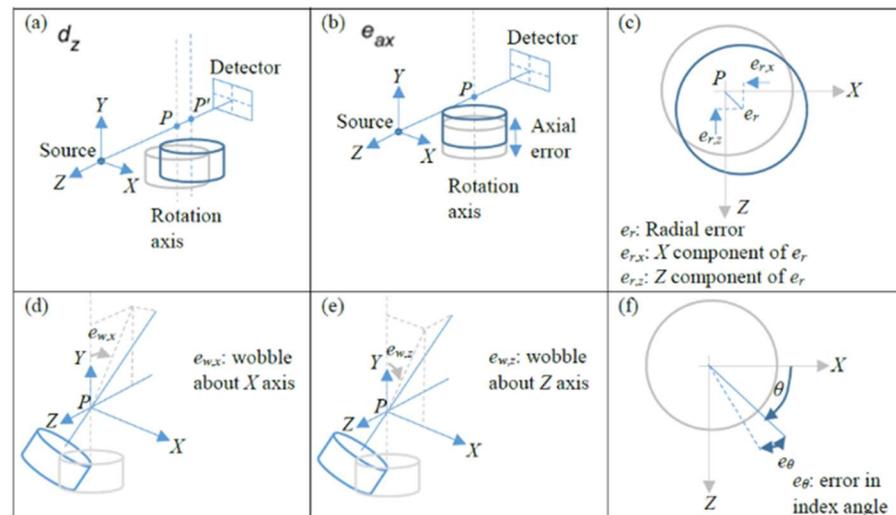


Fig. 5.4. Geometrical error sources associated with the stage. (a) Error in the location of the rotation axis along Z. (b) Axial error motion. (c) Radial error motion components along X and Z. (d) The X component of the wobble error. (e) The Z component of the wobble error. (f) Error in the angular position of the stage.

Positioning the detector in the world coordinate system is accomplished using an HTM for the transformation from the detector coordinate system to the world coordinate system. If the parameter set  $[T_{xd}, T_{yd}, T_{zd}, R_{xd}, R_{yd}, R_{zd}]$  represents the pose of the

detector, the HTM for the detector would then be composed of the following rotation and translation matrices.

$$R_d = \begin{bmatrix} \cos(R_{zd}) & -\sin(R_{zd}) & 0 \\ \sin(R_{zd}) & \cos(R_{zd}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(R_{yd}) & 0 & \sin(R_{yd}) \\ 0 & 1 & 0 \\ -\sin(R_{yd}) & 0 & \cos(R_{yd}) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(R_{xd}) & -\sin(R_{xd}) \\ 0 & \sin(R_{xd}) & \cos(R_{xd}) \end{bmatrix}$$

$$T_d = [T_{xd} \quad T_{yd} \quad T_{zd}]$$

$$HTM_{detector} = \begin{bmatrix} R_{d,11} & R_{d,12} & R_{d,13} & T_{xd} \\ R_{d,21} & R_{d,22} & R_{d,23} & T_{yd} \\ R_{d,31} & R_{d,32} & R_{d,33} & T_{zd} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The nominal values for the parameter set would be [0 mm, 0 mm,  $-D$  mm, 0 rad, 0 rad, 0 rad]. Detector errors are introduced by perturbing these values. For example, perturbing the first parameter  $T_{xd}$  by a constant  $\delta T_{xd}$  would be equivalent to introducing detector X-location error ( $e_x$ ) of magnitude  $\delta T_{xd}$ .

Similarly, the table or rotation stage is positioned in the world coordinate system with an HTM built from a parameter set  $[T_{xt}, T_{yt}, T_{zt}, R_{xt}, R_{yt}, R_{zt}]$  with nominal values [0 mm, 0 mm,  $-d$  mm, 0 rad, 0 rad, 0 rad]. This HTM represents the transformation from the table coordinate system to the world coordinate system and is built as shown below.

$$R_t = \begin{bmatrix} \cos(R_{zt}) & -\sin(R_{zt}) & 0 \\ \sin(R_{zt}) & \cos(R_{zt}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(R_{yt}) & 0 & \sin(R_{yt}) \\ 0 & 1 & 0 \\ -\sin(R_{yt}) & 0 & \cos(R_{yt}) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(R_{xt}) & -\sin(R_{xt}) \\ 0 & \sin(R_{xt}) & \cos(R_{xt}) \end{bmatrix}$$

$$T_t = [T_{xt} \quad T_{yt} \quad T_{zt}]$$

$$HTM_{table} = \begin{bmatrix} R_{t,11} & R_{t,12} & R_{t,13} & T_{xt} \\ R_{t,21} & R_{t,22} & R_{t,23} & T_{yt} \\ R_{t,31} & R_{t,32} & R_{t,33} & T_{zt} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Perturbation to the parameters in the set  $[T_{xt}, T_{yt}, T_{zt}, R_{xt}, R_{yt}, R_{zt}]$  result in radial error in X ( $e_{r,x}$ ), axial error ( $e_{ax}$ ), radial error in Z ( $e_{r,z}$ ), wobble in X ( $e_{w,x}$ ), encoder scale error ( $e_{\theta}$ ), and wobble in Z ( $e_{w,z}$ ) respectively. However, unlike the perturbations for detector errors, the perturbations here are not in the form of constants. Rather they are functions of the table indexing angle as previously discussed. For example, a perturbation  $\delta T_{xt}$  to the parameter  $T_{xt}$  is of the form  $a \sin(n\theta)$  and  $a \cos(n\theta)$  where  $a$  represents the amplitude or magnitude of the geometric error (in mm),  $n$  is the order of the harmonic, and  $\theta$  is the table indexing angle. However, there is an exception in the case of the parameter  $T_{zt}$ , where a constant perturbation is also possible, as opposed to the harmonic perturbation described here. When the perturbation to this parameter is constant, it results in Z-positioning error of the rotation stage ( $d_z$ ).

### 5.3. SIMULATION METHODOLOGY

The single point ray tracing method introduced by Muralikrishnan et al. [39] is the simulation method adopted in this work. This method has been experimentally validated by, and also proven to be a faster and more practical alternative to the full X-ray CT tomographic reconstruction methods for the purposes of estimating the effects of geometric errors using sphere-based artifacts. In this single point ray tracing method, only the sphere centers are projected onto the detector as opposed to the more traditional radiograph-based reconstruction of the whole object. Since this is a purely geometrical approach based on

ray tracing and uses only the centers of the spheres, phenomena such as edge diffraction, scattering and source-related effects are not accounted for in this model. As the stage makes a full rotation, the image of the center of each sphere on the detector traces a locus. For example, Fig. 5.5 shows the locus generated for one sphere as an example.

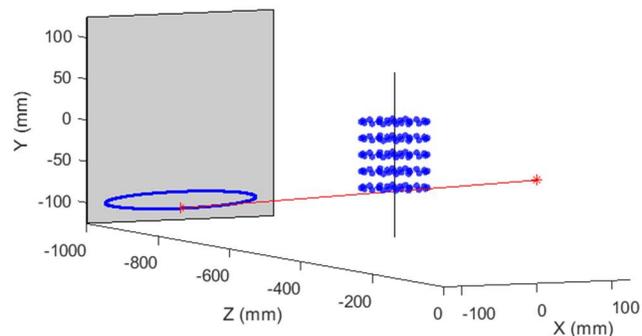


Fig. 5.5. Example of locus traced on the detector.

These loci obtained are used to reconstruct the location of each sphere through a least-squares-minimization based reverse-projection algorithm. Since each sphere produces a separate locus and only this locus is used to reconstruct that sphere location, overlapping spheres do not pose a problem here as they do in conventional tomographic or radiograph-based reconstruction methods. From the spheres thus reconstructed, the center-to-center distances for each pair of the 125 sphere centers are determined. Circles consisting of 120 equally spaced points are constructed normal to each ray connecting the source and the detector, with their centers located on the previously identified least-squares centers. The points inside the resulting point cloud are truncated and only the outer points are used for form error calculation. Here, form error is determined by the range of radii, i.e., the difference between the maximum and minimum radius.

When a particular error source is analyzed, the assumed magnitude of the geometric error is used in the forward projection, for example, the actual yaw angle error of the detector is used to generate the simulated locus for each sphere. However, ideal geometry (or the absence of that geometric error, i.e., zero yaw in this example) is assumed for the reverse-projection or reconstruction algorithm. This discrepancy between actual and assumed geometry parameters represents the magnitude of simulated errors and results in sphere center-to-center distance errors and sphere form errors. In this way, the effect of all geometry errors associated with the detector and stage on the center-to-center distance errors and form errors of the spheres on the reference object are studied. Such simulations are done at several stage and detector positions throughout the working volume to study the effect of magnification.

For each error source under consideration, the pair of spheres that produced the highest center-to-center distance error is identified for the combination of source-stage and source-detector distances selected by Muralikrishnan et al [39]. The line joining this pair of spheres constitutes the line of highest sensitivity, i.e. center-to-center distance error (in mm) per mm or degree of geometric error, and this is tracked across all stage and detector positions. Similarly, the sphere producing the highest form error is identified for one combination of source-stage and source-detector distances and tracked through all combinations of stage and detector positions.

The source-stage distance ( $d$ ) is varied from 200 mm to 1100 mm in steps of 100 mm. For each position of the stage, the source-detector distance ( $D$ ) is varied from  $d+100$  mm to 1200 mm, in steps of 100 mm.

## 5.4. RESULTS AND DISCUSSION

### 5.4.1. Sphere center-to-center distance errors

For all of the error sources considered, the highest distance error sensitivity is observed to occur at one of two configurations. In the first configuration, the detector is positioned as close to the source as possible, and the stage is then positioned as close to the detector as possible. For the purposes of this paper, this configuration shall henceforth be referred to as the ‘near’ configuration. In the simulations conducted, this configuration corresponds to a source-stage distance ( $d$ ) of 200 mm and a source-detector distance ( $D$ ) of 300 mm. As an example, the highest sensitivity to detector displacement along the X direction ( $e_x$ ) occurs in the near configuration. This can be seen in the trend shown in Fig. 5.6, where each curve in the plot represents the sensitivity for a fixed source-stage distance  $d$ , and various source-detector distances  $D$ . Note that the sensitivities shown represent the ratio of the error (varying) in the distance between a pre-determined pair of spheres in the 125-sphere object and a given error (constant) in the detector X position.

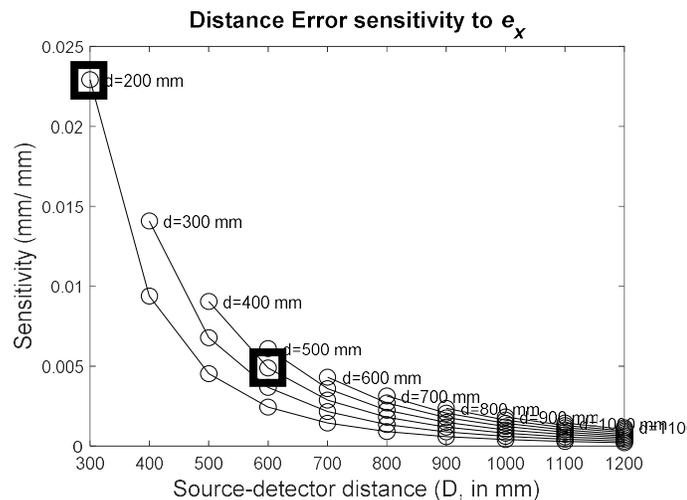


Fig. 5.6. Distance error sensitivity to detector displacement in X.

The second configuration which tends to produce the highest distance sensitivity is where the stage is first positioned as far from the source as possible and the detector is then positioned as close to the stage as possible. This configuration shall be called ‘far’ configuration for simplicity. In this work, this configuration corresponds to  $d=1100$  mm and  $D=1200$  mm. In conventional XCT instruments, this far configuration usually corresponds to the largest possible measurement volume for a given instrument. An example of a case where the far configuration produces the largest distance error sensitivity is the second order cosine component of wobble in X axis ( $e_{w,x}$ ), shown in Fig. 5.7.

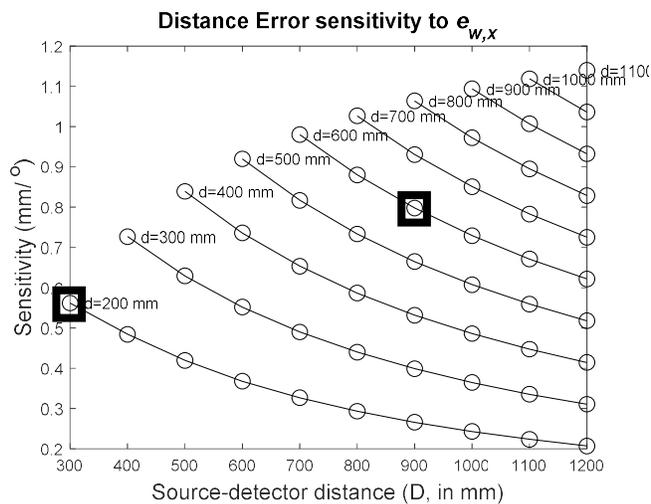


Fig. 5.7. Distance error sensitivity to stage wobble in X.

The magnification for a given XCT measurement setup is determined by the ratio of the source-detector distance to the source-stage distance. Consequently, for typical working volumes, it is possible to achieve the same magnification by various combinations of source-stage and source-detector distances. For example, in Fig. 5.6 and in Fig. 5.7, the highlighted points both have a magnification of 1.5 but the sensitivities in the corresponding cases are significantly different. Therefore, it is clear that even when the

source-stage and source-detector distances are increased proportionately, i.e., maintaining the same magnification, some combinations of stage and detector distances are more sensitive to, and therefore would more clearly reveal, certain error sources.

Trends similar to those shown in Fig. 5.6 and Fig. 5.7 are observed for all error sources, with the highest sensitivity occurring at one of the two ends of the spectrum (near or far configuration). In both these configurations, the magnification is the lowest possible for a given stage position. Also, from the plots shown in Fig. 5.6 and Fig. 5.7, it is clear that, for any given stage position, i.e. for any one of the curves, the highest sensitivity is obtained by minimizing the magnification. This is found to be true of all error sources studied here. The configuration that captures the highest sensitivity for each error source is given in Table 5.1 and Table 5.2. The latter tabulates these configurations for the cosine components of the error motions of the stage, where N represents the order of the harmonic. The sine components of these errors are not shown here but exhibit similar trends.

Table 5.1. Configuration capturing highest center-to-center distance error sensitivity for detector errors and Z location error of stage.

Error source	Configuration showing highest sensitivity
$e_x$	Near
$e_y$	Near
$e_z$	Near
$\theta_x$	Near
$\theta_y$	Near
$\theta_z$	Far
$d_z$	Near

Table 5.2. Configuration capturing highest center-to-center distance error sensitivity for stage error motions.

Error source	$N=1$	$N=2$	$N=3$	$N=4$	$N=5$	$N=6$	$N=7$	$N=8$	$N=9$	$N=10$
$e_{ax}$	Near	Near	Near	Near	NS	NS	NS	NS	NS	NS
$e_{r,x}$	Near	Near	Near	Near	Near	NS	NS	NS	NS	NS
$e_{r,z}$	Near	Near	Near	Near	Near	NS	NS	NS	NS	NS
$e_{w,x}$	Near	Far	Near	Near	Near	Near	NS	NS	NS	NS
$e_{w,z}$	Far	NS								
$e_{\theta}$	Near	Far	Near	Near	Near	Near	NS	NS	NS	NS

$N$ : Order of harmonic

NS: Not Significant – all configurations show similar sensitivity and negligible magnitude

This work acknowledges that not all commercially available XCT systems have the benefit of being able to move the detector as freely as discussed here. In instances where the detector is fixed, the highest sensitivities possible for that case can be realized by simply moving the stage as close to the fixed detector as possible. This case is shown by the points corresponding to a single detector position, or a vertical line, in Fig. 5.6 and Fig. 5.7.

Upon identifying the stage and detector positions that produce highest sensitivities, the position and orientation of the most sensitive line in the corresponding measurement volume was identified for each error source. The solid lines shown in Fig. 5.8 represent the lines that produced that highest distance error sensitivity for the six detector errors and the Z location error of the rotation axis. In cases where multiple lines are shown for a single error source, it indicates that all the lines shown are equivalent in capturing the highest sensitivity shown. The values of  $d$  and  $D$  at which this highest sensitivity is observed are also mentioned in each case.

Figs 5.9 through 5.12 show similar illustrations of sensitive lines for the error motions of the stage. These errors are represented by sine and cosine components of orders 1 to 10. However, the magnitudes of sensitivities corresponding to the first four orders are found to be the most significant and are therefore shown here. Further, only the cosine components of the error sources are shown. The sensitive lines for the sine components are observed to have similar orientations but rotated by 90 degrees about the rotation axis.

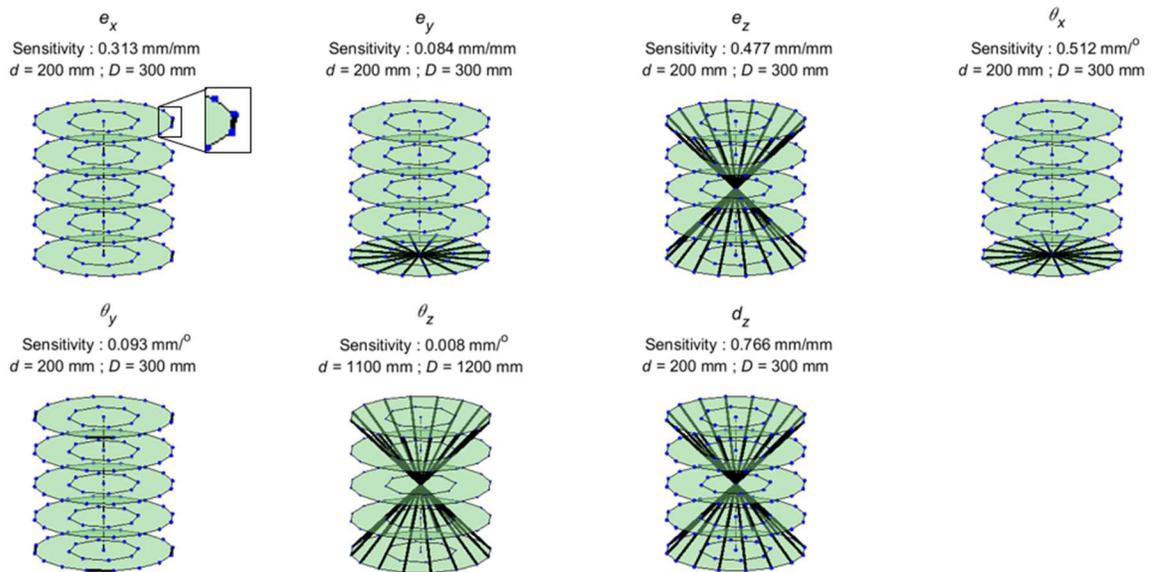


Fig. 5.8. Sensitive lines for detector errors and Z location error of rotation axis.

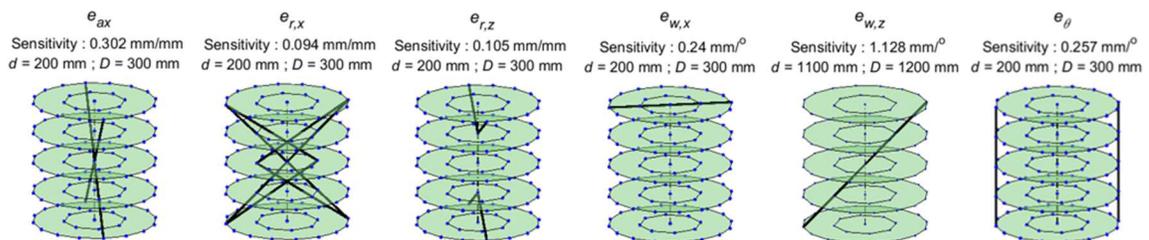


Fig. 5.9. Sensitive lines for first order cosine components of stage error motions.

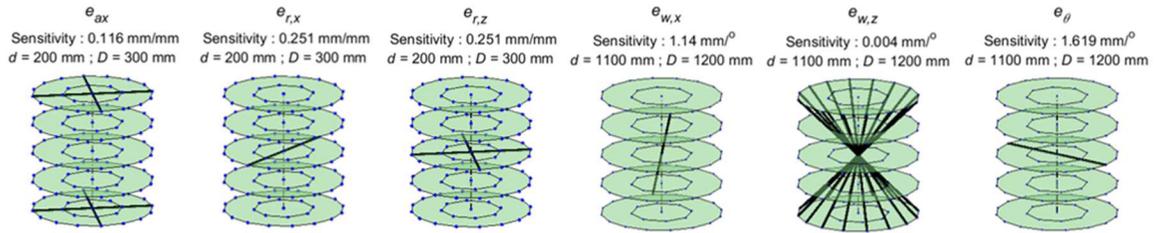


Fig. 5.10. Sensitive lines for second order cosine components of stage error motions.

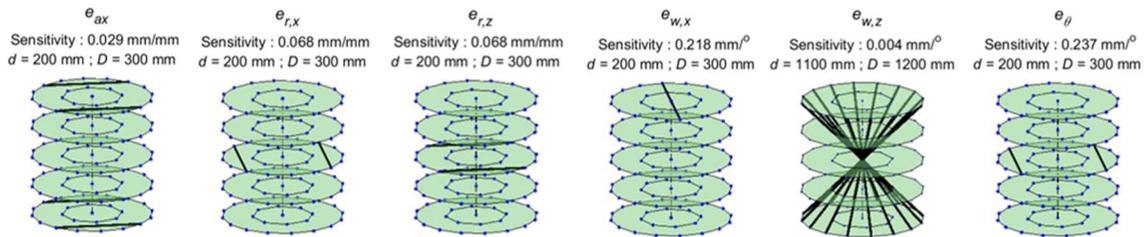


Fig. 5.11. Sensitive lines for third order cosine components of stage error motions.

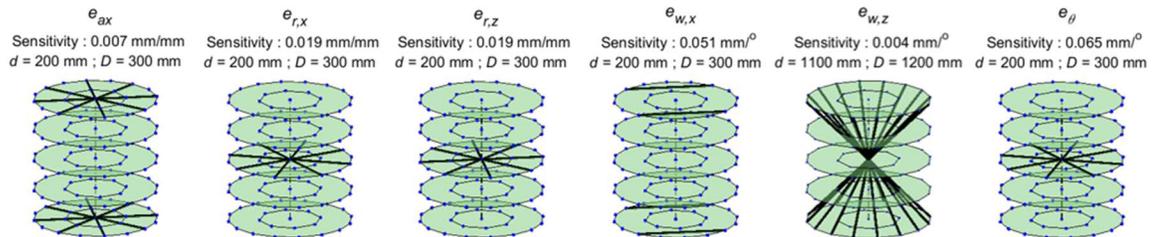


Fig. 5.12. Sensitive lines for fourth order cosine components of stage error motions.

#### 5.4.2. Sphere form errors

For most of the error sources in this study, sensitivities for sphere form errors also show trends similar to those of center-to-center distance errors in that maximum sensitivity is found at the near and far configurations previously described. Examples of these cases are shown in Fig. 5.13 and Fig. 5.14. The trends in form error sensitivity for detector Y displacement error ( $e_y$ ) are shown in Fig. 5.13. Clearly, the largest sensitivity occurs at the near configuration. Fig. 5.14 shows the form error sensitivity to detector rotation error

about the Z axis ( $e_\theta$ ). Here, the largest sensitivity occurs at the far configuration. In some cases, the peak sensitivity occurs at other configurations in the working volume. However, in these rare cases, the sensitivity differs very little across the different configurations.

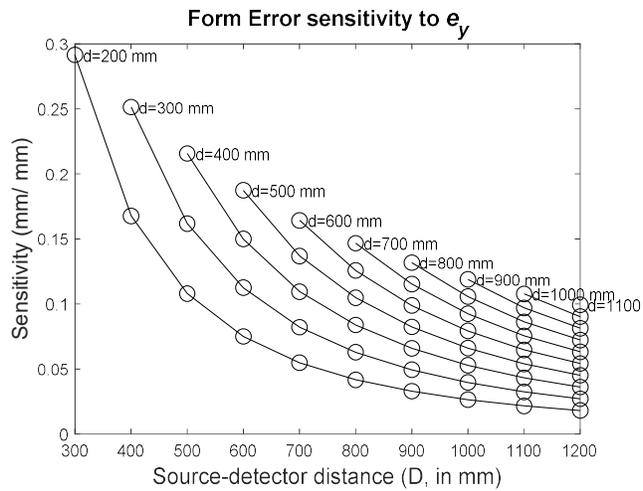


Fig. 5.13. Form error sensitivity to detector Y location error.

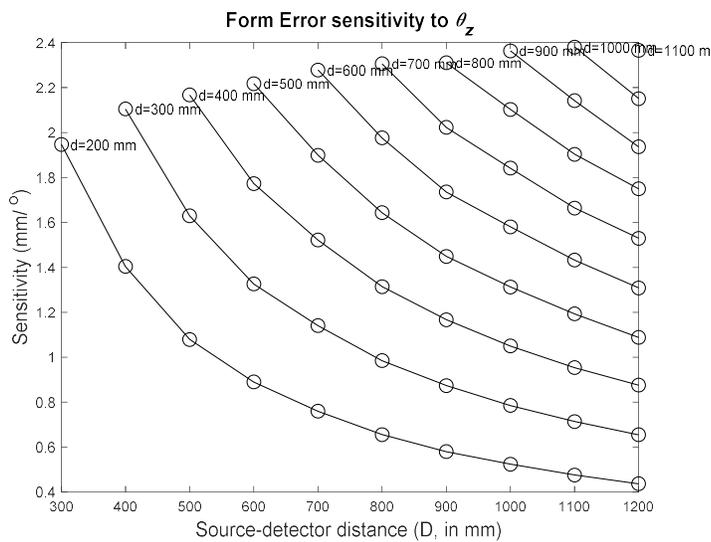


Fig. 5.14. Form error sensitivity to detector Z rotation error.

Table 5.3 and Table 5.4 show the configurations that capture the highest form error sensitivities for each of the error sources. Table 5.4 contains the configurations that best capture the cosine components of the error motions of the stage, where  $N$  denotes the order of the harmonic. The sine components of these errors are not reported here but exhibit similar trends.

Table 5.3. Configuration capturing highest form error sensitivity for detector errors and Z location error of stage.

Error source	Configuration showing highest sensitivity
$e_x$	Far
$e_y$	Near
$e_z$	Near
$\theta_x$	NS
$\theta_y$	Near
$\theta_z$	Far
$d_z$	NS

NS: Not Significant – all configurations show similar sensitivity and negligible magnitude

Table 5.4. Configuration capturing highest form error sensitivity for stage error motions.

Error source	$N=1$	$N=2$	$N=3$	$N=4$	$N=5$	$N=6$	$N=7$	$N=8$	$N=9$	$N=10$
$e_{ax}$	Near	Far	Near	Far	Other	Far	Other	Far	Far	Far
$e_{r,x}$	Near	Near	Far	Near	Near	Near	Far	Other	Far	Other
$e_{r,z}$	Near									
$e_{w,x}$	Far									
$e_{w,z}$	Far									
$e_\theta$	Far									

$N$ : Order of harmonic

For all error sources, the locations of the spheres that produce the highest form error are observed to occur at one of the following three cases:

1. Spheres with highest form error are those that are radially the furthest from the axis of rotation, i.e., independent of their height in the measurement volume.
2. Spheres with the highest form error are those that are radially furthest from the axis of rotation as well as furthest from the  $Y=0$  plane (the horizontal plane located at the middle of the measuring volume).
3. Sphere form errors are negligible for all spheres.

## 5.5. PROPOSED RECOMMENDATIONS

This section describes attempts to condense the results obtained above and encapsulate them into a set of minimum recommended measurement lines and sphere locations that can capture highest possible sphere center-to-center distance error and form error sensitivities for all the geometric errors to a satisfactory degree.

A direct objective of this work was to contribute to the ISO standards by working out a compromise between number and type of artifacts, number of scans and number of measurement lines. Prior to this work, the ISO document appeared artifact specific. It did not describe abstract test positions and gave manufacturers what might be considered by some as too much freedom in selecting test positions. Preference was given to the sphere forest artifact or similar structure [44], which consisted of an array of spheres mounted on a plate. Fig. 5.15 shows an example of a sphere forest. One of the immediately obvious issues of a sphere forest is that though it is theoretically three dimensional, it does not truly

take advantage of this 3D property, i.e., it has a very small aspect ratio and therefore cannot be used to test along the long body diagonal of the measuring volume, or full vertical lines. This work identifies core lines required to capture all geometric errors, makes evaluations as to whether the sphere forest is indeed sufficient, and where this is not the case, makes recommendations on ways to realize the core measurement lines by supplementing the sphere forest with additional artifacts.

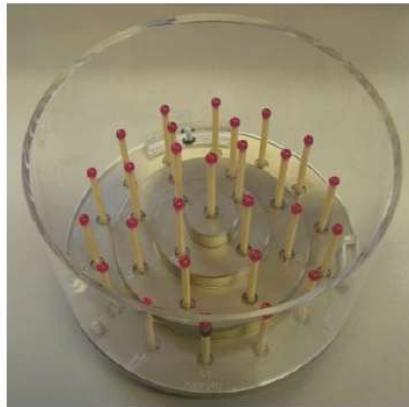


Fig. 5.15. Photograph of a typical sphere forest.

For the purposes of this work, a fixed position of the stage and detector were chosen, with the stage positioned at 900 mm from the source, and the detector at 1000 mm from the source since this represented a common scenario among most commercially available XCT instruments. The 125-sphere artifact with these stage and detector positions was used as the reference. Based on simulation of several what-if situations in trying to minimize the number of measurement lines, the lines shown in Fig. 5.16 (a) were found to be the core minimum set that could sufficiently capture all geometric errors discussed here. It is to be noted that these recommendations are formulated with the current state of the ISO document (favoring the sphere forest) in mind. For example, ease of taking advantage

of the sphere forest is the only reason why the face diagonals in Fig. 5.16 (a) are shown at the bottom plane and not on the top (though they are equivalent), and why the spheres in Fig. 5.16 (b) are shown on the bottom only and not the top.

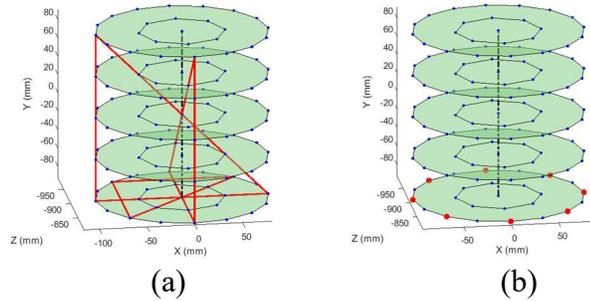


Fig. 5.16. Core minimum set of (a) measurement lines and (b) sphere locations.

From Fig. 5.16 (a) and (b), it can be seen that almost all the measurement lines and spheres can be realized with a sphere forest placed at the bottom of the measuring volume, with the exception of the two body diagonals and two verticals shown in Fig. 5.16 (a). These four lines can be realized using a set of ball bars supplementing the measurement of the sphere forest. It is not necessary that they must be measured at the same time as the sphere forest, or that the ball bar measurements be registered with the spheres on the sphere forest as shown.

## 5.6. OUTCOMES

This work has reported, by means of simulation results, the positions of stage and detector that can result in highest center-to-center distance error and sphere form error sensitivities for all geometric errors associated with the stage and detector in a given XCT instrument and identified the measurement lines and spheres within the corresponding measurement volume that produce these highest distance errors and form errors

respectively. Further, these results have been condensed to provide a core set of minimum recommended lines and spheres that can capture all these errors to a satisfactory degree. Examples of artifact combinations that can be used to realize these measurement lines have also been provided.

## CHAPTER 6: ERROR ANALYSIS: STEREO-VISION PHOTOGRAMMETRY

### 6.1. INTRODUCTION

The use of vision systems in dimensional metrology has long been popular, owing to the distinct advantages it offers over other optical/ non-contact measurement systems. As early as the 1980s, the aircraft and aerospace manufacturing industries began to witness the implementation of photogrammetry for inspecting assemblies, tooling fixtures, etc. [4]. Some of the advantages offered by this technology include the ability to acquire large and dense point clouds in very short cycle times, the ability to easily calibrate with relatively inexpensive equipment, and the lack of a need for an optical cooperative target (such as a retroreflector), although active/ passive targets can be used to improve accuracy.

A crucial component that enables widespread implementation of any measurement technology, including Photogrammetry, is Standardization, or the development of standardized test procedures to evaluate the performance of such technologies. It is imperative for such standard test procedures to be sensitive to all or almost all known error sources associated with the measurement process. The VDI/VDE 2634-1 [45] is one such widely accepted standard that describes test procedures for performance evaluation of optical 3D measurement systems with point-to-point probing. It provides guidelines on test positions and places some restrictions on the length of the longest test length to be used and the orientations of the test lengths. However, it recommends only seven measurement lines, and these appear to be arbitrary since no reasoning is provided, and no references are made to capturing the various possible errors.

The work described in this chapter attempts to investigate this issue by developing stereo-vision photogrammetric systems consisting of two cameras. Simulations are done using a 3D grid of points representing the working volume. For each error source accounted for in the pin-hole camera model [46], the most sensitive line, or the line producing the highest point-to-point distance error is identified. These lines are compared to the seven lines described in the VDI/VDE standards in terms of efficiency in capturing all errors described here. These results are experimentally validated by two contrasting setups.

## 6.2. EXPERIMENTAL SETUP

The experimental setup consists of two identical cameras positioned at approximately at same height and at an angle (about a vertical axis) toward each other such that the angle subtended by the line joining the centers of the two cameras at a point on a desired target, is approximately 30 degrees. This angle, commonly called the apex angle, is illustrated in Fig. 6.1 and is an important determiner of the dimensions of the measurement volume as well as the accuracy of depth perception. Crombrugge at al. have proposed a calibration methodology where such overlap in the fields of view of the cameras is not required and therefore there will not be an apex angle [47].

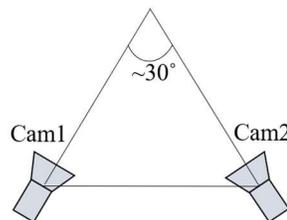


Fig. 6.1. Top view of camera setup showing apex angle.

Two independent systems A and B are used for the purpose of studying the influence of setup variations. In the case of system A, a set of two identical high-quality cameras are used, while a pair of economical webcams are used in system B. In both cases, the cameras are setup on the table of a commercially available CMM. Fig. 6.2 shows the setup of the two cameras used in system A. System B uses a similar setup. The dimensions of the measuring volume in each case is determined by the relative positioning between the two cameras, the field of view of each camera, focal lengths, etc. The measuring volumes for system A and B are  $550 \times 350 \times 250 \text{ mm}^3$  and  $900 \times 600 \times 300 \text{ mm}^3$  respectively.

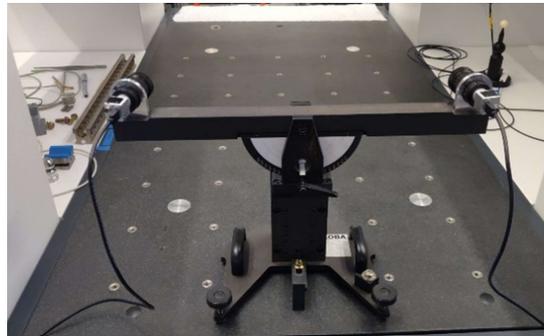


Fig. 6.2. Experimental setup for system A.

Once the measuring volume is determined, a checkerboard is positioned at all the corners of the measurement volume, and the focal lengths of the cameras are adjusted such that the checkerboard is clear and in focus at all extremes in the measurement volume.

### 6.3. MODEL DESCRIPTION

The work described here adopts the pinhole camera model [46] used by a Camera Calibrator application available in the Image Processing and Computer Vision toolbox in

MATLAB. In this model, the following ten parameters or error sources are taken into account for each camera.

- Focal lengths  $F_x$  and  $F_y$
- Pixel coordinates of principal point  $u_0$  and  $v_0$
- Skew factor  $S$
- Radial Distortion coefficients  $\kappa_1, \kappa_2$ , and  $\kappa_3$
- Tangential Distortion coefficients  $t_1$  and  $t_2$

In addition to these parameters, the model includes six parameters representing inter-camera geometry, namely three translations along the X, Y, and Z axes, and three rotations about the same,  $T_x, T_y, T_z, R_x, R_y$ , and  $R_z$ . Hence, a total of 26 parameters are included in this model.

In the pinhole model, each camera is represented by 4x3 camera matrix  $P$  built from an extrinsic matrix (which in turn is composed of a rotation matrix  $R$  and a translation vector  $T$ ) and an intrinsic matrix  $K$  as follows.

$$P = \begin{bmatrix} R \\ T \end{bmatrix} K$$

The camera matrix  $P$  allows transforming a point from the world coordinate system (or object coordinate system) represented as  $[X_w, Y_w, Z_w]$  to a point in the image plane represented as  $[X_i, Y_i, 1]$ .

$$W [X_i, Y_i, 1] = [X_w, Y_w, Z_w] P$$

Where  $W$  is a weight or scale factor, which is simply a factor that normalizes the image plane coordinates. In other words, the right hand side of the above equation results in a 1x3 vector where the third element is not equal to one, and  $W$  is the factor needed to make this last coordinate equal to one. Physically, this is because the image plane is two-dimensional and has no third coordinate.

The extrinsic matrix  $\begin{bmatrix} R \\ T \end{bmatrix}$  represents the rigid transformation to the camera coordinate system. The intrinsic matrix  $K$  for a camera represents the transformation from the camera coordinate system to the image plane. Since this maps a point in a 3D coordinate system to a 2D coordinate system, this is a projective transformation as opposed to a rigid transformation. These relationships are shown in Fig. 6.3.

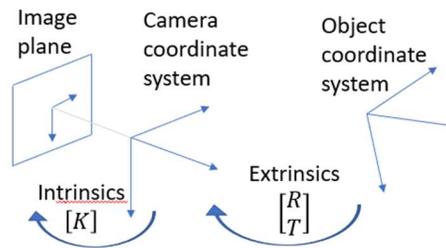


Fig. 6.3. Intrinsic and Extrinsic matrix transformations.

The intrinsic matrix for a given camera is defined as follows:

$$K = \begin{bmatrix} F_x & 0 & 0 \\ S & F_y & 0 \\ u_0 & v_0 & 1 \end{bmatrix}$$

The extrinsic matrix is a 4x3 matrix given by

$$Extrinsics = \begin{bmatrix} R \\ T \end{bmatrix}$$

where  $R$  and  $T$  are defined as follows:

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

$$T = [T_x \quad T_y \quad T_z]$$

In this model, the extrinsic matrix for the second camera (or right camera) conforms to the definition above but is simplified for the first camera (or left camera) as the object/world coordinate system is assumed to be coincident with the camera 1 coordinate system.

As a result, the extrinsic matrix for the first camera becomes

$$Extrinsics_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

This allows the model to be simplified and is the reason why the number of independent parameters is 26 as described above, as opposed to 32.

Although the pinhole model, by itself, does not account for lens distortion [48] since an ideal pinhole camera does not have a lens. However, to accurately represent a real-world camera, this model includes radial and tangential distortion. Examples of radial distortion are shown in Fig. 6.4.

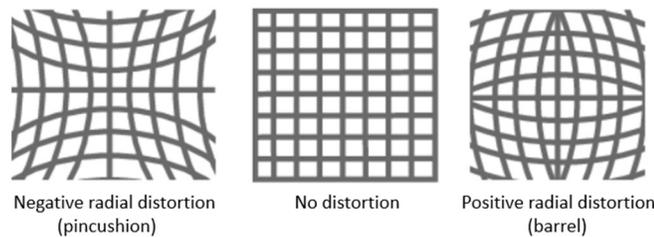


Fig. 6.4. Radial distortion examples.

Radial distortion occurs because light rays further away from the center of the lens experience more bending. As seen in Fig. 6.4, the degree of radial distortion at a given pixel of the image is dependent on the radial distance  $r$  of the pixel from the optic axis. The radially distorted image points  $(x_{rd}, y_{rd})$  are obtained from the undistorted points  $(x_{ud}, y_{ud})$  using the radial distortion coefficients  $\kappa_1, \kappa_2$ , and  $\kappa_3$  as follows.

$$x_{rd} = x_{ud}(1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6)$$

$$y_{rd} = y_{ud}(1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6)$$

where  $r = \sqrt{x_{ud}^2 + y_{ud}^2}$

It is to be noted that  $x_{ud}$  and  $y_{ud}$  are in normalized image coordinates. These are obtained from pixel coordinates by first translating to the optical center and then dividing by the focal length in pixels. Therefore,  $x_{ud}$  and  $y_{ud}$  are dimensionless.

Tangential distortion occurs due to misalignment (or non-parallelism) between the camera sensor (which defines the image plane) and the lens as shown in Fig. 6.5.

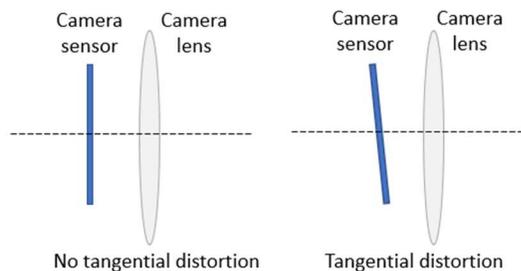


Fig. 6.5. Tangential distortion.

The tangentially distorted image points  $(x_{td}, y_{td})$  are obtained from the undistorted points  $(x_{ud}, y_{ud})$  using the tangential distortion coefficients  $t_1$  and  $t_2$  as follows.

$$x_{td} = x_{ud} + [2t_1x_{ud}y_{ud} + t_2(r^2 + 2x_{ud}^2)]$$

$$y_{td} = y_{ud} + [t_1(r^2 + 2y_{ud}^2) + 2t_2x_{ud}y_{ud}]$$

In this way, both radial and tangential distortions are applied after the undistorted points on the image plane are obtained using the camera matrices  $P_1$  and  $P_2$  for the two cameras.

More detail analyses on applying and compensating for radial and tangential distortions have been done by Heikkila et al [48]. Fish-eye lenses on the other hand produce extreme distortions that do not lend themselves to be easily accommodated by pinhole models. Distortions in fish-eye lenses have been studied extensively by Shah et al [49-51].

#### 6.4. CALIBRATION

The calibration method adopted here is the Zhang method [52-53], a widely used calibration method in computer vision applications. In this method, a checkerboard of unequal number of squares along the length and width is placed in several positions and orientations within the measurement volume. These positions are distributed such that they cover most of the measurement volume and include a wide variety of angular positions of the checkerboard. Such a variety of angular positions is essential to ensure accuracy in depth perception. Fig. 6.6 shows one such position of the checkerboard.

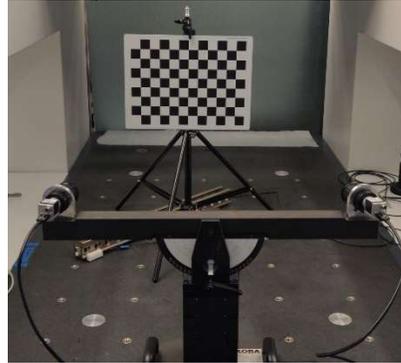


Fig. 6.6. Checkerboard used for calibration.

In this work, at least 20 pairs of images (from right and left camera), or 20 different poses (positions and orientations) of the checkerboard are used for any calibration. Estimates are made for the poses of the checkerboard [54], and these are iteratively used to arrive at an optimal set of the 26 model parameters described above. Fig. 6.7 shows a pictorial representation of this process.

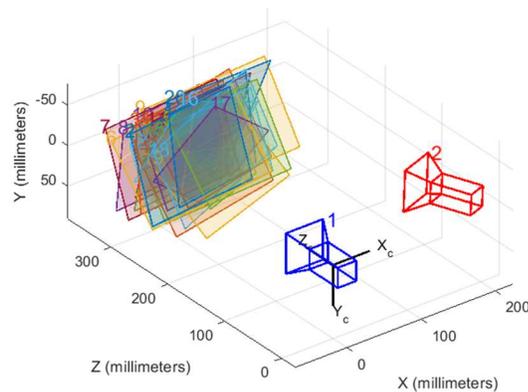


Fig. 6.7. Example of checkerboard poses used for calibration.

At the end of the calibration process, optimal estimates of the 26 parameters are obtained. For the two systems A and B used, these estimates are shown in Table 6.1.

Table 6.1. Estimated model parameters for systems A and B.

Parameter	Nominal value estimated from calibration	
	System A	System B
$F_{x1}$ (pixels)	4975.34	497.33
$F_{y1}$ (pixels)	4971.96	496.72
$u_{01}$ (pixels)	1535.52	329.58
$v_{01}$ (pixels)	1058.32	218.46
$S_1$	0.20	0.10
$k_{11}$	-0.14	0.193
$k_{21}$	0.33	-0.482
$k_{31}$	-1.21	0.178
$t_{11}$	9.5e-5	1.3e-4
$t_{21}$	1.8e-4	5.6e-4
$F_{x2}$ (pixels)	4968.40	499.86
$F_{y2}$ (pixels)	4964.82	499.59
$u_{02}$ (pixels)	1572.99	334.30
$v_{02}$ (pixels)	1106.87	236.03
$S_2$	0.16	0.99
$k_{12}$	-0.13	0.172
$k_{22}$	0.31	-0.464
$k_{32}$	-1.34	0.305
$t_{12}$	-4.1e-4	4.0e-3
$t_{22}$	4.2e-4	-3.2e-3
$T_x$ (mm)	-554.53	-407.77
$T_y$ (mm)	14.38	9.40
$T_z$ (mm)	167.60	74.24
$R_x$ (rad)	-0.02	-0.01
$R_y$ (rad)	-0.47	-0.41
$R_z$ (rad)	-0.01	0.05

## 6.5. SIMULATION

A grid of points representing the measurement volume is chosen. The distances between every pair of points is determined and these are used as reference lengths. For each of the 26 error sources, a perturbation, or an assumed magnitude of error is introduced in the parameter set. This perturbed set of parameters is used to project the 3D points to the image planes of the two cameras, where they become 2D points represented by pixel locations. This includes applying distortion using the radial and tangential distortion coefficients. Subsequently, the matching pixels in the images from the two cameras (the pixels that are the projections of the same point in the real world) are triangulated, or reprojected into the real world, thereby estimating its reprojected coordinates. These coordinates would differ from the initial coordinates due to the perturbation in one parameter that occurred while projecting into the cameras but did not occur during reprojection. In other words, the disparity between the actual parameter used in obtaining the image, and the assumed nominal parameter used in reprojecting the point creates an error in the coordinate of the reprojected point. In this way, all the points are reprojected, and the new distances between every pair of points are determined.

It is to be noted that each error source is treated independently, i.e., when one error is introduced (or a parameter in the model is perturbed), errors due to other sources are assumed to be absent. Therefore, a new set of point-to-point distances are generated for each error source under consideration. In each case, the point-to-point distance errors are determined using the reference lengths. The pair of points that produce the highest point-to-point distance error is identified for each error source. In this way, a set of 26 lines of highest point-to-point distance errors, or the most sensitive lines are obtained. Due to the

variations in the setup of the two systems A and B, these 26 lines are slightly different in each case. Fig. 6.8 (a) shows the most sensitive lines for system A, and Fig. 6.8 (b) shows seven lines that approximate the VDI/VDE lines. Fig. 6.9 (a) and (b) show these lines in system B. Though these are slightly different, these are similar in orientations from the point of view of the cameras. In system A, the optic axes of the cameras are approximately parallel to the XY (horizontal) plane in the world coordinate system. But in system B, the cameras are tilted slightly upwards.

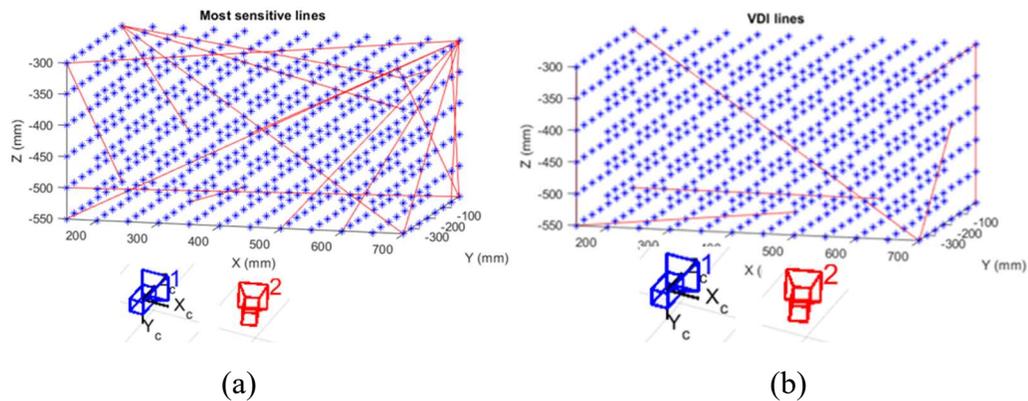


Fig. 6.8. Measurement lines for system A (a) Most sensitive lines (b) VDI/VDE lines.

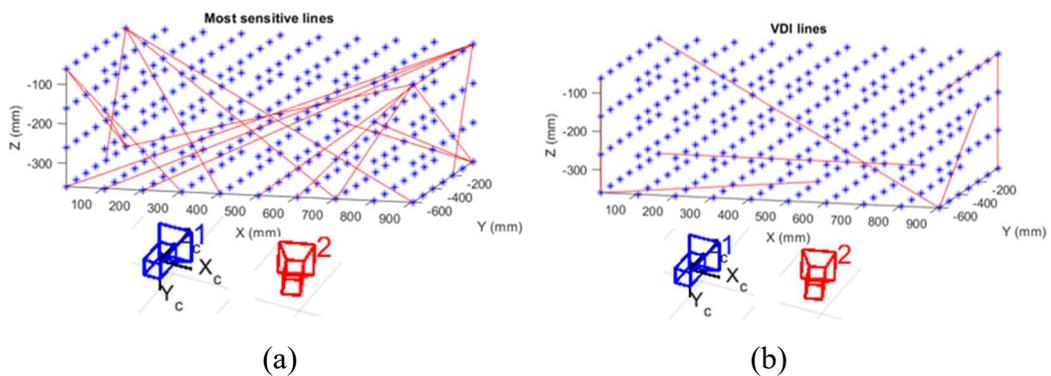


Fig. 6.9. Measurement lines for system B (a) Most sensitive lines (b) VDI/VDE lines.

Tables 6.2 and 6.3 show the distance errors in systems A and B observed along the 26 most sensitive lines compared to those observed along the 7 VDI/VDE lines.

Table 6.2. Maximum simulated distance errors- System A.

Error source	Max distance error (mm)			Max distance error per unit length (mm/m)		
	26 Sensitive lines	7 VDI/VDE lines	% lost	26 Sensitive lines	7 VDI/VDE lines	% lost
$F_{x1}$	0.20	0.19	6	0.31	0.27	12
$F_{y1}$	0.03	0.03	2	0.10	0.10	0
$u_{01}$	0.39	0.36	9	0.56	0.89	-59
$v_{01}$	0.03	0.01	54	0.07	0.04	44
$S_1$	0.04	0.03	21	0.10	0.08	13
$k_{11}$	2e-3	1e-3	42	0.01	0.01	1
$k_{21}$	0.01	2e-3	57	0.01	0.01	49
$k_{31}$	0.21	0.09	58	0.32	0.24	24
$t_{11}$	0.03	0.03	2	0.10	0.10	0
$t_{21}$	0.42	0.42	0	0.60	0.83	-39
$F_{x2}$	0.04	0.03	26	0.09	0.08	6
$F_{y2}$	0.04	0.02	53	0.09	0.08	15
$u_{02}$	3e-3	1e-3	56	0.01	5e-3	22
$v_{02}$	0.01	3e-3	58	0.02	0.01	47
$S_2$	1.31	1.31	0	1.88	1.88	0
$k_{12}$	0.06	0.03	41	0.08	0.08	-2
$k_{22}$	0.78	0.35	55	1.26	0.96	23
$k_{32}$	0.37	0.10	73	0.86	0.38	56
$t_{12}$	3.44	3.12	9	4.93	7.47	-52
$t_{22}$	0.84	0.57	32	1.94	1.65	15
$T_x$	0.07	0.06	10	0.11	0.11	-2
$T_y$	0.01	0.01	12	0.02	0.02	8
$T_z$	6e-4	6e-4	4	1e-3	1e-3	3
$R_x$	0.08	0.03	65	0.12	0.08	35
$R_y$	0.01	1e-3	68	0.01	0.01	36
$R_z$	5e-4	1e-4	71	7e-4	4e-4	37

Table 6.3. Maximum simulated distance errors- System B.

Error source	Max distance error (mm)			Max distance error per unit length (mm/m)		
	26 Sensitive lines	7 VDI/VDE lines	% lost	26 Sensitive lines	7 VDI/VDE lines	% lost
$F_{x1}$	4.51	4.32	4	4.65	3.85	17
$F_{y1}$	0.39	0.29	24	0.58	0.98	-69
$u_{01}$	9.10	7.50	18	8.72	11.12	-28
$v_{01}$	0.37	0.26	30	1.03	0.87	16
$S_1$	0.83	0.41	51	1.30	1.21	8
$k_{11}$	0.01	0.01	31	0.02	0.02	3
$k_{21}$	0.02	0.01	40	0.03	0.02	21
$k_{31}$	4.60	1.87	59	4.74	3.12	34
$t_{11}$	0.39	0.27	32	0.59	0.89	-53
$t_{21}$	9.10	9.10	0	8.11	9.27	-14
$F_{x2}$	0.32	0.23	27	0.88	0.56	37
$F_{y2}$	0.81	0.41	49	1.26	1.37	-9
$u_{02}$	0.01	4e-3	58	0.02	0.01	47
$v_{02}$	0.02	0.01	69	0.03	0.02	47
$S_2$	2.86	2.86	0	2.55	2.55	0
$k_{12}$	0.10	0.06	37	0.09	0.09	1
$k_{22}$	1.78	0.56	68	1.83	1.13	38
$k_{32}$	0.76	0.58	24	1.19	1.93	-62
$t_{12}$	8.47	7.54	11	7.54	9.47	-26
$t_{22}$	1.82	0.84	54	2.75	2.46	10
$T_x$	0.34	0.33	3	0.81	0.57	30
$T_y$	0.11	0.06	46	0.27	0.21	23
$T_z$	0.04	0.02	42	0.09	0.07	17
$R_x$	0.37	0.06	84	0.86	0.20	77
$R_y$	0.13	0.05	64	0.30	0.08	75
$R_z$	0.04	0.02	47	0.12	0.04	68

While some of the distance errors shown in Tables 6.2 and 6.3 could be deemed negligible in magnitude, it is to be noted that these distance errors arise from arbitrarily chosen perturbation values for each of the parameters. These arbitrary magnitudes may not be realistic in the real-world commercial systems. Some of these errors may tend to have much smaller or larger magnitudes than assumed here. Therefore, the quantity that becomes of more significance is the percentage of the distance errors lost when only the 7 VDI/VDE lines are used compared to when the set of 26 most sensitive lines are used. Hence, these ratios are also shown alongside the magnitudes in Tables 6.2 and 6.3.

Many commercial photogrammetry/ vision systems specify their MPE as constant values. However, some systems specify their MPE as a sum of a constant term and a length-dependent term, of the form  $A+BL$ , where  $A$  and  $B$  are constants and  $L$  is the measurement length over which the MPE is claimed to hold. Thus, the distance error per unit length also becomes important, especially when MPEs are often fixed by the results of such point-to-point length measurement tests. The right half of Tables 6.2 and 6.3 show the distance errors per unit length obtained when the most sensitive lines and VDI/VDE lines are used, and shows the percentages lost for these cases.

From the tables, it is clear that for both systems A and B, there are very few cases where the percentage loss in distance errors is less than 10%. To a large extent, this is also true for distance errors per unit length. However, it is noteworthy that the number of low-loss cases is slightly larger in case of distance errors per unit length. This hints that it may be okay to use fewer lines if the length dependence is taken into account in the performance test. Also, in rare cases, the percentage loss is found to be negative. This indicates that for

some error sources, the VDI/VDE lines in fact capture more error per unit length than the lines of highest sensitivity identified here.

## 6.6. EXPERIMENTAL VALIDATION

To experimentally validate the simulation results, a small, printed checkerboard is glued onto the ram of the CMM. This is shown in Fig. 6.10. The ram is then moved to different known positions forming a grid of points. The CMM-reported point-to-point distances within this grid are used as the reference lengths.

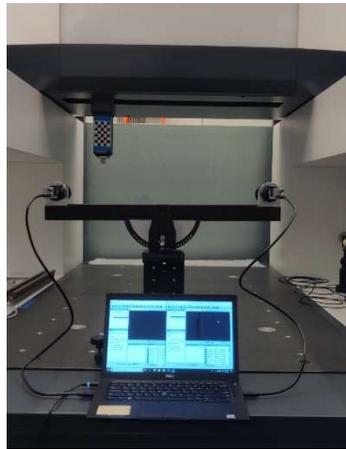


Fig. 6.10. Measurement of grid of points on CMM.

At each ram position, the centroid of the checkerboard is measured by the camera system. This is similar to the work done by Hernández et al. [55], with the difference being that a checkerboard was used instead of a light pen. In this way, a measured grid of points is obtained and the point-to-point distance errors for every pair of points is determined by comparing to the reference lengths. For systems A and B, the observed distance errors

along the lines of highest sensitivity identified from the simulation work and along lines that approximate the VDI/VDE lines are shown in Table 6.4.

Table 6.4. Maximum experimental distance errors- Systems A and B.

System	Max distance error (mm)			Max distance error per unit length (mm/m)		
	26 Sensitive lines	7 VDI/VDE lines	Ratio	26 Sensitive lines	7 VDI/VDE lines	Ratio
A	0.48	0.40	17	0.74	0.57	23
B	6.04	4.90	19	14.24	12.61	11

Unlike the simulation results, here the magnitudes of the distance errors are realistic since they represent real world systems, and therefore bear more significance than the percentages lost. From Table 6.4, it can be seen that the percentages lost when only the VDI/VDE lines are used may not appear too significant. However, there is appreciable difference in magnitudes for the distance errors obtained depending on which lines are chosen, especially in case of system B, where errors tend to be magnified due to nature of setup. This indicates that it is important to measure along sensitive lines, as these are better capable of bringing out the highest errors that could occur in a given measurement volume.

## 6.7. OUTCOMES

In this work, simulation studies have been done to identify the most sensitive lines, and these have been experimentally shown, by means of two contrasting setups, to produce larger distance errors compared to the lines recommended by the VDI/VDE standards. The lines currently in the standard may therefore not always be sufficient in capturing highest sensitivities to all known error sources.

## CHAPTER 7: CONCLUSIONS AND FUTURE WORK

In this dissertation, various individual case studies were conducted with the goal of contributing towards the advancement in discrete part coordinate metrology. In the case of data processing, a custom CMS that creates a point cloud by scanning the surface of a part as it rotates was investigated and a deterministic calibration procedure was developed. Such a procedure was demonstrated to produce traceable results, and its performance was verified by measurement of test parts. Further, machine learning approaches were also explored for a two-dimensional simplification of the same problem statement, and preliminary results were presented. Possible future work in this area includes extension to 3D, improvements in the interpolation algorithms used after the nearest neighbors are found, and development of methods to reduce the large runtime involved in training the system.

A data modeling system capable of supporting complex parts as well as new/ non-traditional measurement technologies, namely the QIF system was studied and an application software was developed to promote its adoption into industry. The beta version developed allows users to catalog their measurement resources in the standardized QIF format. Future work in this regard could include improvements in the app security protocols, graphical user interface, and the inclusion of support for languages other than C#. Some shortcomings of the standard itself have also been identified, and work could be done to overcome these limitations including support for ranges of values, uncertainty modeling, etc.

Finally, error analysis was done for two CMS namely XCT and Stereo vision photogrammetric systems by characterizing several known error sources for each of the systems. In the case of XCT, a series of simulations were performed, and recommendations were made on ways to capture the highest possible sensitivities for sphere center-to-center distance error and sphere form error. A minimum core set of 9 measurement lines were identified. Contributions were made in ASME and ISO on test positions in XCT performance evaluation standards. Future work would involve experimental validation by means of a multi-sphere artifact of various sizes placed in various locations of an uncompensated XCT instrument to verify claims made here on locations of maximum sensitivities.

In the case of Stereo vision photogrammetry, a similar sensitivity analysis was performed to identify the most sensitive measurement lines for each error source. Subsequently, the performance of these lines in capturing all error sources was compared with the lines recommended in the existing VDI/VDE standards. These simulations were experimentally validated, and it has been demonstrated that the VDI/VDE lines are not always sufficient in capturing highest distance error sensitivities. To a large extent, this was also found to be applicable for distance errors per unit length. Recommendations can be made in future on a minimum set of measurement lines that can capture all known error sources to a satisfactory degree, so as to propose a viable alternative to the lines currently in the VDI/VDE standards.

## REFERENCES

- [1] Mathia, T.G., Pawlus, P., Wiczorowski, M., 2011. Recent trends in surface metrology. *Wear* 271, 494–508.
- [2] Manske, E., Füßl, R., Mastlyo, R., Vorbringer-Dorozhovets, N., Birli, O., and Jäger, G., 2015. Ongoing trends in precision metrology, particularly in nanopositioning and nanomeasuring technology, *tm - Technisches Messen*, 82(7-8), 359-366.
- [3] Schmitt, R.H., Peterek, M., Morse, E., Knapp, W., Galetto, M., Härtig, F., Goch, G., Hughes, B., Forbes, A., and Estler, W.T., 2016. Advances in Large-Scale Metrology – Review and future trends. *CIRP Annals* 65, 643–665.
- [4] Estler, W.T., Edmundson, K.L., Peggs, G.N., and Parker, D.H., 2002. Large-Scale Metrology – An Update. *CIRP Annals* 51, 587–609.
- [5] Peggs, G.N., Maropoulos, P.G., Hughes, E.B., Forbes, A.B., Robson, S., Ziebart, M., and Muralikrishnan, B., 2009. Recent developments in large-scale dimensional metrology. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 223, 571–595.
- [6] Puttock, M.J., Large Scale Metrology, 1978. *CIRP Annals* 27(1), 351-356.
- [7] Shi, Z., 2013. Current Status and Trends of Large Gears Metrology. *JME* 49, 35.
- [8] Goch, G., Ni, K., Peng, Y., and Guenther, A., 2017. Future gear metrology based on areal measurements and improved holistic evaluations. *CIRP Annals* 66, 469–474.
- [9] Goch, G., 2003. Gear metrology. *CIRP Annals* 52(2), 659-695.
- [10] Gao, W., Furukawa, M., Kiyono, S., and Yamazaki, H., 2004. Cutting error measurement of flexspline gears of harmonic speed reducers using laser probes. *Precision Engineering* 28, 358-363.

- [11] Pahk, H., Hwang, Y., Lee, I., and Ahn, W., 2010. Vision-based precision inspection system for profile and performance of small and micro gears using intelligent image processing techniques and virtual master gears. *Proceedings of SPIE* 4222, 48-53.
- [12] Xu, C., Wei, X., Zhang, Z., and Zhou, X., 2019. A multi-axis space coordinate system calibration method for composite line laser measuring systems using non-feature planes and multi-angle spheres. *Proc. SPIE* 11171, Sixth European Seminar on Precision Optics Manufacturing 1117103.
- [13] Genta, G., Minetola, P., and Barbato, G., 2016. Calibration procedure for a laser triangulation scanner with uncertainty evaluation. *Optics and Lasers in Engineering* 86, 11-19.
- [14] Morse, E. and Jaganmohan, P., 2020. 6 DOF calibration of profile sensor locations in an inspection station. *CIRP Annals* 69(1), 465-468.
- [15] Müller, R., Vette-Steinkamp, M., and Kanso, A., 2019. Position and orientation calibration of a 2D laser line sensor using closed-form least-squares solution. *IFAC-PapersOnLine* 52 (13), 689-694.
- [16] Chou, J. C. K. and Kamel, M., 1991. Finding the Position and Orientation of a Sensor on a Robot Manipulator Using Quaternions. *The International Journal of Robotics Research* 10(3), 240-254.
- [17] Chekh, B.A., Kortaberria, G., and Gonzalo, O., 2019. Extrinsic calibration and kinematic modelling of a laser line triangulation sensor integrated in an intelligent fixture with 3 degrees of freedom. *Precision Engineering* 60, 235-245.
- [18] EURAMET software library – [www.euramet.org](http://www.euramet.org).
- [19] Lunz, S., Öktem, O., and Schönlieb, C., 2018. Adversarial Regularizers in Inverse Problems. 32nd Conference on Neural Information Processing Systems, Montréal, Canada, 1-10.
- [20] Ongie, G., Jalal, A., Metzler, C., Baraniuk, R., Dimakis, A., and Willett, R., 2020. Deep Learning Techniques for Inverse Problems in Imaging. *IEEE Journal on Selected Areas in Information Theory*, 1 (1), 39-56,

- [21] Adler, J. and Öktem, O., 2017. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems* 33 124007, 1-24.
- [22] Zhao, J. and Itti, L., 2018. ShapeDTW: Shape Dynamic Time Warping. *Pattern Recognition* 74, 171-184.
- [23] Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., and Batista, G., 2015. The ucr time series classification archive.
- [24] Rabiner, L. and Juang, B., 1993. *Fundamentals of Speech Recognition* 14, PTR Prentice Hall Englewood Cliffs.
- [25] Kulkarni, K., Evangelidis, G., Cech, J., and Horaud, R., 2014. Continuous action recognition based on sequence alignment. *IJCV*, 1-25.
- [26] ANSI/QIF version 3.0 – 2018.
- [27] ISO 23952:2020. *Automation systems and integration — Quality information framework (QIF) — An integrated model for manufacturing quality information*. International Organization for Standardization, Geneva, Switzerland.
- [28] Zhao, Y.F., Horst, J.A., Kramer, T.R., Rippey, W. and Brown, R.J., 2012. *Quality Information Framework—Integrating Metrology Processes*. IFAC Proceedings Volumes, 45(6) 1301-1308.
- [29] Morse, E., Heysiattalab, S., Feeney, A., and Hedberg, T., 2016. Interoperability: linking design and tolerancing with metrology, *Procedia CIRP* 43 13–16.
- [30] Huang H.M., Michaloski, J., Campbell, D., Stone, R., Kramer, T., Brown, C., Brown, R., and Tatarliev, G. 2014. NISTIR 8102, End-to-end Demonstration of the Quality Information Framework (QIF) Standard at the International Manufacturing Technology Show (IMTS).
- [31] Heysiattalab, S., 2107. *Data models to support metrology*. Available from Dissertations & Theses at University of North Carolina Charlotte; ProQuest Dissertations & Theses Global; SciTech Premium Collection.

- [32] Draft ISO/CD 10360-11, 2020. - Geometrical product specifications (GPS) — Acceptance and reverification tests for coordinate measuring machines (CMM) — Part 11: CMMs using the principle of computed tomography (CT)
- [33] Draft ASME B89.4.23, 2020. - X-Ray Computed Tomography (CT) Performance Evaluation Standard.
- [34] VDI/VDE 2630 -1.2, 2018. Computed tomography in dimensional measurement; Influencing variables on measurement results and recommendations for computed tomography dimensional measurements.
- [35] Ferrucci, M., Leach, R.K., Giusca, C., Carmignato, S., and Dewulf, W., 2015. Towards geometrical calibration of X-ray computed tomography systems—A review. *Measurement Science and Technology* 26(9):092003.
- [36] Kumar, J., Attridge, A., Wood, P.K.C., and Williams, M.A., 2011. Analysis of the effect of cone-beam geometry and test object configuration on the measurement accuracy of a computed tomography scanner used for dimensional measurement. *Measurement Science and Technology* 22(3):035105.
- [37] Ametova, E., Ferrucci, M., Chilingaryan, S., and Dewulf, W., 2018. A computationally inexpensive model for estimating dimensional measurement uncertainty due to X-ray computed tomography instrument misalignments. *Measurement Science and Technology* 29(6):065007.
- [38] Aloisi, V., Carmignato, S., Schlecht, J., Ferley, E., 2017. Investigation on the effects of X-ray CT system geometrical misalignments on dimensional measurement errors. *Proceedings of the 7th Conference on Industrial Computed Tomography (Leuven, Belgium)*.
- [39] Muralikrishnan, B., Shilling, M., Phillips, S., Ren, W., Lee, V., and Kim, F., 2019. X-ray Computed Tomography Instrument Performance Evaluation, Part I: Sensitivity to Detector Geometry Errors. *Journal of Research of the National Institute of Standards and Technology* 124:124014, 1-16.
- [40] Muralikrishnan, B., Shilling, M., Phillips, S., Ren, W., Lee, V., and Kim, F., 2019. X-ray Computed Tomography Instrument Performance Evaluation, Part II: Sensitivity to Rotation Stage Errors. *Journal of Research of the National Institute of Standards and Technology* 124:124015, 1-13.

- [41] Jaganmohan, P., Muralikrishnan, B., Shilling, M., and Morse, E., 2020. Performance Evaluation of X-Ray Computed Tomography Instruments: Sensitivity to Detector and Stage Errors- Trends for Different Magnifications. *Proceedings of the Thirty-Fifth Annual Meeting of the American Society for Precision Engineering* 73, 122-127.
- [42] Ferrucci, M., Ametova, E., Probst, G., Craeghs, T., and Dewulf, W., 2018. Sensitivity of CT dimensional measurements to rotation stage errors. *Proceedings of the 8th Conference on Industrial Computed Tomography (Wels, Austria)*.
- [43] Ferrucci, M., Ametova, E., Carmignato, S., and Dewulf W., 2016. Evaluating the effects of detector angular misalignments on simulated computed tomography data. *Precision Engineering* 45, 230-241.
- [44] de Aquino Silva, J.B., Hocken, R.J., Miller, J.A., Caskey, G.W., and Ramu, P., 2009. Approach for uncertainty analysis and error evaluation of four-axis co-ordinate measuring machines. *Int J Adv Manuf Technol* 41, 1130.
- [45] VDI/VDE 2634-2, 2002. *Optical 3D measuring systems- Imaging systems with point-by-point probing*.
- [46] Sturm P., 2014. Pinhole Camera Model. In: Ikeuchi K. (eds) *Computer Vision*. Springer, Boston, MA.
- [47] Crombrugge, I.V., Penne, R., and Vanlanduit, S., 2020. Extrinsic camera calibration for non-overlapping cameras with Gray code projection. *Optics and Lasers in Engineering* 134, 106305.
- [48] Heikkila, J. and Silven, O., 1997. A Four-step Camera Calibration Procedure with Implicit Image Correction. *IEEE International Conference on Computer Vision and Pattern Recognition*, 1106-1112.
- [49] Shah, S. and Aggarwal, J. K., 1995. Autonomous mobile robot navigation using fish-eye lenses. *Image Analysis Applications and Computer Graphics* 1024,9-16.
- [50] Shah, S. and Aggarwal, J. K., 1996. Intrinsic parameter calibration procedure for a (high distortion) fish-eye lens camera with distortion model and accuracy estimation. *Pattern Recognition* 29(11), 1775-1778.

- [51] Shah, S. and Aggarwal, J. K., 1997. Mobile robot navigation and scene modeling using stereo fish-eye lens system. *Machine Vision and Applications* 10(4), 159–173.
- [52] Zhang, Z., 2000. A Flexible New Technique for Camera Calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions* 22, 1330-1334.
- [53] Burger, W., 2016. Zhang's Camera Calibration Algorithm: In-Depth Tutorial and Implementation.
- [54] Huang, P., Meng, Z., Guo, J., and Zhang, F., 2018. Pose Measurement Based on Vision Perception. In: *Tethered Space Robot, Dynamics, Measurement, and Control*, 75-119.
- [55] Hernández, O., Barbosa, J., Razo, Y., and Ramos, J., 2017. Performance evaluation of a portable 3D vision coordinate measuring system, *Automatika* 58:3, 253-265.